

**Documentation of Questionnaire/Module
'FRS1004B' on 22-08-2011 at 12:58**

FRS1004B

FAMILY RESOURCES SURVEY 2010-2011

COMPUTE ALWAYS:

NatCen := ONS

COMPUTE ALWAYS:

Edit := No

COMPUTE ALWAYS:

Test := No

COMPUTE ALWAYS:

VerCode := '040_1'

COMPUTE ALWAYS:

TestVer := '01'

COMPUTE IF: NatCen = NI

SuppTxt := ('Please record the reasons why you suppressed ' + 'this warning, then press <SAVE> to save and continue.')

COMPUTE IF: NOT (NatCen = NI)

SuppTxt := ('Please record the reasons why you suppressed ' + 'this warning, then press <Alt> + S to save and continue.')

COMPUTE ALWAYS:

Pd97Txt := ('Please leave a note/remark giving full details ' + 'then press <SAVE> to save and continue.')

COMPUTE ALWAYS:

KeyTxt := ('This is a 'Key Question': It is VERY ' + 'IMPORTANT to get an answer here if possible. If you cannot ' + 'do so (either now, or later) please make a Note about the ' + 'circumstances.')

COMPUTE IF: NatCen = ONS

N := ''

COMPUTE IF: NatCen = ONS

I := ''

COMPUTE IF: NatCen = ONS

B := ''

COMPUTE IF: NatCen = ONS

X := 'H'

COMPUTE IF: NatCen = ONS

O1 := '('

COMPUTE IF: NatCen = ONS

O2 := ')''

COMPUTE IF: NatCen = ONS

IC := 'i '

COMPUTE IF: NatCen = ONS

IS := 'N'

COMPUTE IF: NatCen = ONS

n11 := ''

COMPUTE IF: NatCen = ONS

Anyone_Else := 'Anyone else'

COMPUTE IF: NatCen = ONS

Any_Others := 'Any others'

COMPUTE IF: NatCen = ONS

Any_Other := 'Any other'

COMPUTE IF: NOT (NatCen = ONS)

N := ''

COMPUTE IF: NOT (NatCen = ONS)

I := ''

COMPUTE IF: NOT (NatCen = ONS)

B := ''

COMPUTE IF: NOT (NatCen = ONS)

X := '*'

COMPUTE IF: NOT (NatCen = ONS)

O1 := '('

COMPUTE IF: NOT (NatCen = ONS)

O2 := ')''

```

COMPUTE IF: NOT (NatCen = ONS)
            AND: Edit = Yes

```

```

IC := 'EDITOR: '

```

```

COMPUTE IF: NOT (NatCen = ONS)
            AND: NOT (Edit = Yes)

```

```

IC := 'INTERVIEWER: '

```

```

COMPUTE IF: NOT (NatCen = ONS)

```

```

IS := 'SHOW CARD'

```

```

COMPUTE IF: NOT (NatCen = ONS)

```

```

n11 := '
'

```

```

COMPUTE IF: NOT (NatCen = ONS)

```

```

Anyone_Else := 'Who else'

```

```

COMPUTE IF: NOT (NatCen = ONS)

```

```

Any_Others := 'Which others'

```

```

COMPUTE IF: NOT (NatCen = ONS)

```

```

Any_Other := 'Which other'

```

```

COMPUTE IF: NatCen = Yes

```

```

n12 := '
'

```

```

COMPUTE IF: NOT (NatCen = Yes)

```

```

n12 := ''

```

```

COMPUTE IF: NatCen = NI

```

```

Help_F9 := '<HELP>'

```

```

COMPUTE IF: NOT (NatCen = NI)

```

```

Help_F9 := 'Help <F9>'

```

```

RECORD ALWAYS:

```

IVers

Version code of interview program, eg. I_049_1. I = Interview, 04 = month, 7 = year, 1 = release.

STRING[7]

RECORD ALWAYS:

EVers

Version code of edit program, eg. E_049_1. E = Edit, 04 = month, 7 = year, 1 = release.

STRING[7]

COMPUTE IF: Test = Yes
AND: NatCen = NI

IVers := ('NI_' + TestVer)

COMPUTE IF: Test = Yes
AND: NOT (NatCen = NI)

IVers := ('I_' + VerCode)

COMPUTE IF: Edit = Yes

EVers := ('E_' + VerCode)

COMPUTE IF: NOT (Edit = Yes)

IVers := ('I_' + VerCode)

COMPUTE IF: Edit = Yes

Interviewer := 'Editor'

COMPUTE IF: Edit = Yes

EditVersion := ('Edit version@|@|:@|' + EVers + '
')

COMPUTE IF: NOT (Edit = Yes)

Interviewer := 'Interviewer'

COMPUTE IF: NOT (Edit = Yes)

EditVersion := '
')

COMPUTE ALWAYS:

Days[1] := 'Sunday'

COMPUTE ALWAYS:

Days[2] := 'Monday'

COMPUTE ALWAYS:

Days[3] := 'Tuesday'

COMPUTE ALWAYS:

Days[4] := 'Wednesday'

COMPUTE ALWAYS:

Days[5] := 'Thursday'

COMPUTE ALWAYS:

Days[6] := 'Friday'

COMPUTE ALWAYS:

Days[7] := 'Saturday'

COMPUTE ALWAYS:

Months[1] := 'January'

COMPUTE ALWAYS:

Months[2] := 'February'

COMPUTE ALWAYS:

Months[3] := 'March'

COMPUTE ALWAYS:

Months[4] := 'April'

COMPUTE ALWAYS:

Months[5] := 'May'

COMPUTE ALWAYS:

Months[6] := 'June'

COMPUTE ALWAYS:

Months[7] := 'July'

COMPUTE ALWAYS:

Months[8] := 'August'

COMPUTE ALWAYS:

Months[9] := 'September'

COMPUTE ALWAYS:

Months[10] := 'October'

COMPUTE ALWAYS:

Months[11] := 'November'

COMPUTE ALWAYS:

Months[12] := 'December'

COMPUTE ALWAYS:

AssDo := No

COMPUTE ALWAYS:

BookDo := No

COMPUTE ALWAYS:

NCDVLP := No

COMPUTE ALWAYS:

NCDVIB := 0

COMPUTE ALWAYS:

NCDVOB := 0

COMPUTE ALWAYS:

NCDVDC := No

COMPUTE ALWAYS:

NCDVTC := No

COMPUTE ALWAYS:

NCDVCP := 0

COMPUTE ALWAYS:

NCDVAW := No

COMPUTE ALWAYS:

NCDVRT := No

COMPUTE ALWAYS:

NCDVAA := No

FRS1004B.QSerial

Serial number

```

COMPUTE IF: Test = Yes
    AND: NatCen = Yes

```

```

KeyString := GETENV('KEYVALUE')

```

```

COMPUTE IF: Test = Yes
    AND: NatCen = Yes

```

```

DArea := VAL(SUBSTRING(KeyString,1,5))

```

```

COMPUTE IF: Test = Yes
    AND: NatCen = Yes

```

```

DAddress := VAL(SUBSTRING(KeyString,6,2))

```

```

COMPUTE IF: Test = Yes
    AND: NatCen = Yes

```

```

DHhold := VAL(SUBSTRING(KeyString,8,1))

```

```

COMPUTE IF: Test = Yes
    AND: NOT (NatCen = Yes)
    AND: BIDDData.SEARCH (1)

```

```

DArea := VAL(SUBSTRING(BIDDData.BidField,1,5))

```

```

COMPUTE IF: Test = Yes
    AND: NOT (NatCen = Yes)
    AND: BIDDData.SEARCH (1)

```

```

DAddress := VAL(SUBSTRING(BIDDData.BidField,6,2))

```

```

COMPUTE IF: Test = Yes
    AND: NOT (NatCen = Yes)
    AND: BIDDData.SEARCH (1)

```

```

DHhold := VAL(SUBSTRING(BIDDData.BidField,8,1))

```

```

ASK IF: Test = Yes

```

Area

```

^I AREA NUMBER.

```

```

^B JUST PRESS <Enter>^B.

```

```

1..99997

```

```

ASK IF: Test = Yes

```

Address

```

^I ADDRESS NUMBER.

```

```

^B JUST PRESS <Enter>^B.

```

```

1..97

```

ASK IF: *Test = Yes*

Hhold

^I HOUSEHOLD NUMBER.

^B JUST PRESS <Enter>^B.

1..3

CHECK IF: *Test = Yes*
RESERVECHECK

RESERVECHECK

CHECK IF: *Test = Yes*
RESERVECHECK

RESERVECHECK

CHECK IF: *Test = Yes*
RESERVECHECK

RESERVECHECK

COMPUTE IF: *Test = Yes*
AND: *DArea > 0*

Area := DArea

COMPUTE IF: *Test = Yes*
AND: *DAddress > 0*

Address := DAddress

COMPUTE IF: *Test = Yes*
AND: *DHhold > 0*

Hhold := DHhold

FRS1004B.QSerial

Serial number

```

COMPUTE IF: NOT (Test = Yes)
AND: NatCen = Yes

```

```

KeyString := GETENV('KEYVALUE')

```

```

COMPUTE IF: NOT (Test = Yes)
AND: NatCen = Yes

```

```

DArea := VAL(SUBSTRING(KeyString,1,5))

```

```

COMPUTE IF: NOT (Test = Yes)
AND: NatCen = Yes

```

```

DAddress := VAL(SUBSTRING(KeyString,6,2))

```

```

COMPUTE IF: NOT (Test = Yes)
AND: NatCen = Yes

```

```

DHhold := VAL(SUBSTRING(KeyString,8,1))

```

```

COMPUTE IF: NOT (Test = Yes)
AND: NOT (NatCen = Yes)
AND: BIDDData.SEARCH (1)

```

```

DArea := VAL(SUBSTRING(BIDDData.BidField,1,5))

```

```

COMPUTE IF: NOT (Test = Yes)
AND: NOT (NatCen = Yes)
AND: BIDDData.SEARCH (1)

```

```

DAddress := VAL(SUBSTRING(BIDDData.BidField,6,2))

```

```

COMPUTE IF: NOT (Test = Yes)
AND: NOT (NatCen = Yes)
AND: BIDDData.SEARCH (1)

```

```

DHhold := VAL(SUBSTRING(BIDDData.BidField,8,1))

```

```

ASK IF: NOT (Test = Yes)

```

Area

```

^I AREA NUMBER.

```

```

^B JUST PRESS <Enter>^B.

```

```

1..99997

```

```

ASK IF: NOT (Test = Yes)

```

Address

```

^I ADDRESS NUMBER.

```

```

^B JUST PRESS <Enter>^B.

```

```

1..97

```

ASK IF: NOT (Test = Yes)

Hhold

^I HOUSEHOLD NUMBER.

^B JUST PRESS <Enter>^B.

1..3

CHECK IF: NOT (Test = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: NOT (Test = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: NOT (Test = Yes)
RESERVECHECK

RESERVECHECK

COMPUTE IF: NOT (Test = Yes)
AND: DArea > 0

Area := DArea

COMPUTE IF: NOT (Test = Yes)
AND: DAddress > 0

Address := DAddress

COMPUTE IF: NOT (Test = Yes)
AND: DHhold > 0

Hhold := DHhold

FRS1004B (continued)
FAMILY RESOURCES SURVEY 2010-2011

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: QSerial.Area < 10

StrArea := ('0000' + STR(QSerial.Area,1))

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: QSerial.Area < 100

StrArea := ('000' + STR(QSerial.Area,2))

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: QSerial.Area < 1000

StrArea := ('00' + STR(QSerial.Area,3))

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: QSerial.Area < 10000

StrArea := ('0' + STR(QSerial.Area,4))

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: NOT (QSerial.Area < 10000)

StrArea := STR(QSerial.Area,5)

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: QSerial.Address IN [1 .. 9]

StrAddr := ('0' + STR(QSerial.Address,1))

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: NOT (QSerial.Address IN [1 .. 9])

StrAddr := STR(QSerial.Address,2)

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

StrSerial := (StrArea + StrAddr)

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Serial := VAL(StrSerial)

RECORD IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

NSerial

Natcen Serial Number

STRING[8]

RECORD IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

NPoint

Natcen Point Number

STRING[5]

RECORD IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

ChkLet

Check Lettter

STRING[1]

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

NSerial := (StrSerial + STR(QSerial.Hhold,1))

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

NPoint := SUBSTRING(StrSerial,1,5)

FRS1004B.Checkletter()

Procedure Call

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[1] := 'A'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[2] := 'B'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[3] := 'C'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[4] := 'D'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[5] := 'E'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[6] := 'F'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[7] := 'G'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[8] := 'H'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[9] := 'J'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[10] := 'K'
```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
Letters[11] := 'L'
```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[12] := 'M'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[13] := 'N'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[14] := 'P'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[15] := 'Q'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[16] := 'R'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[17] := 'S'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[18] := 'T'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[19] := 'V'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[20] := 'W'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[21] := 'X'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[22] := 'Y'

```

```

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])

```

```

Letters[23] := 'Z'

```

```
COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
            AND: (LEN (StrSerial) = 7) AND (QSerial.Hhold IN [1 .. 3])
```

```
ChkLet := Letters[ROUND(FRAC(SerNo / 23) * 23) + 1]
```


FRS1004B.QDataBag**Sample information**

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Serial

Natcen Serial number excluding household number.

1..9999997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Hhold

Household number

1..3

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

IntNo

Interviewer Number

0..9999

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SurvId

3-letter acronym for survey.

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SampYear

Year Code

1998..9997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SampMnth

Sample month.

1..12

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SampQtr

Sample quarter

1..4

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Attempt

Issue number.

1..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SSTRTReg

Stratifying region: Survey specific.

1..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

StaRegGB

Statistical region in GB.

1..12

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

GovRegGB

Government office region in GB.

1..12

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Country

1..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

OAC

Output Area Classification.

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SLA

Local Authority Code.

STRING[4]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

LAC

Local Authority Code. GOV version

0..997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

NICoun

Northern Ireland District Council Codes

1..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

NIRate

Northern Ireland District Council Rates

-99.9999..999.9999

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

NINRV

Net rateable value of property (in N. Ireland)

1..9997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

AddType

Address Type

0..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

BRMA

Broad Rental Market Area

0..99997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

URIndEW

Urban and Rural Indicators - England and Wales

0..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

URIndSc

Urban and Rural Indicators - Scotland

0..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

IMD_E

Index of Multiple Deprivation variables for England

0..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

IMD_W

Index of Multiple Deprivation variables for Wales

0..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

IMD_S

Index of Multiple Deprivation variables for Scotland

0..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

ONSPoint

Original Area (Point) - 5 chars with leading 0's

STRING[5]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SMO

Multiple occupancy indicator

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SGridRef

Grid reference

STRING[13]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SInd

STRING[1]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

DNum

Local Authority code DNum4DSS

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Spare1

Spare code frame

0..997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

Spare2

Spare code frame

0..997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SPare3

Spare string

STRING[50]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SAdd1

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SAdd2

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SAdd3

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SAdd4

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SAdd5

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SAdd6

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

SPC

Spare string

STRING[10]

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: DBData.SEARCH (Serial)

QDataBag := DBData.QDataBag

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: DBData.SEARCH (Serial)
AND: SUBSTRING (QDataBag.SLA, 1, 2) =

QDataBag.SLA := (SUBSTRING(QDataBag.SLA,3,2) + ' ')

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: DBData.SEARCH (Serial)
AND: SUBSTRING (QDataBag.SLA, 1, 1) =

QDataBag.SLA := (SUBSTRING(QDataBag.SLA,2,3) + ' ')

FRS1004B.QDataBag

Sample information

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

Serial

Natcen Serial number excluding household number.

1..9999997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

Hhold

Household number

1..3

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

IntNo

Interviewer Number

0..9999

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SurvId

3-letter acronym for survey.

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SampYear

Year Code

1998..9997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SampMnth

Sample month.

1..12

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SampQtr

Sample quarter

1..4

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

Attempt

Issue number.

1..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SSTRTReg

Stratifying region: Survey specific.

1..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

StaRegGB

Statistical region in GB.

1..12

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

GovRegGB

Government office region in GB.

1..12

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

Country

1..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

OAC

Output Area Classification.

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SLA

Local Authority Code.

STRING[4]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

LAC

Local Authority Code. GOV version

0..997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

NICoun

Northern Ireland District Council Codes

1..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

NIRate

Northern Ireland District Council Rates

-99.9999..999.9999

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

NINRV

Net rateable value of property (in N. Ireland)

1..9997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

AddType

Address Type

0..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

BRMA

Broad Rental Market Area

0..99997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

URIndEW

Urban and Rural Indicators - England and Wales

0..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

URIndSc

Urban and Rural Indicators - Scotland

0..97

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

IMD_E

Index of Multiple Deprivation variables for England

0..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

IMD_W

Index of Multiple Deprivation variables for Wales

0..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

IMD_S

Index of Multiple Deprivation variables for Scotland

0..7

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

ONSPoint

Original Area (Point) - 5 chars with leading 0's

STRING[5]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SMO

Multiple occupancy indicator

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SGridRef

Grid reference

STRING[13]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SInd

STRING[1]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

DNum

Local Authority code DNum4DSS

STRING[3]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

Spare1

Spare code frame

0..997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

Spare2

Spare code frame

0..997

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SPare3

Spare string

STRING[50]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SAdd1

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SAdd2

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SAdd3

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SAdd4

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SAdd5

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SAdd6

Spare string

STRING[30]

ASK IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: Test = Yes

SPC

Spare string

STRING[10]

FRS1004B (continued)**FAMILY RESOURCES SURVEY 2010-2011**

RECORD IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

OSerial

Original (ONS) Serial Number

STRING[8]

RECORD IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)

OPoint

Original (ONS) Point

STRING[5]

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: LEN (QDataBag.ONSPoint) = 5

OPoint := QDataBag.ONSPoint

COMPUTE IF: (QSerial.Area = RESPONSE) AND (QSerial.Address = RESPONSE)
AND: LEN (QDataBag.ONSPoint) = 5
AND: (LEN (StrAddr) = 2) AND (QSerial.Hhold IN [1 .. 3])

OSerial := (OPoint + StrAddr + STR(QSerial.Hhold,1))

COMPUTE IF: QDataBag.SampMnth IN [1, 2, 3]

ChkYear := 2011

COMPUTE IF: NOT (QDataBag.SampMnth IN [1, 2, 3])

ChkYear := 2010

COMPUTE ALWAYS:

FYearM3 := 2007

COMPUTE ALWAYS:

FYearM2 := 2008

COMPUTE ALWAYS:

FYearM1 := 2009

COMPUTE ALWAYS:

FYear := 2010

COMPUTE ALWAYS:

LYear := 2011

COMPUTE ALWAYS:

YearNo := '10'

RECORD ALWAYS:

DSerial

DWP style Serial Number

STRING[10]

COMPUTE IF: (LEN (OSerial) = 8) AND (LEN (YearNo) = 2)

**DSerial := (SUBSTRING(OSerial,1,5) + YearNo +
SUBSTRING(OSerial,6,3))**

COMPUTE ALWAYS:

SampInfo := ''

COMPUTE IF: QDataBag.SSTRTReg IN [22 .. 27]

NIreland := No

COMPUTE IF: QDataBag.SSTRTReg IN [22 .. 27]

Scotland := Yes

COMPUTE IF: QDataBag.SSTRTReg IN [22 .. 27]

Wales := No

COMPUTE IF: QDataBag.SSTRTReg IN [22 .. 27]
AND: NatCen = Yes

SampInfo := 'Scotland'

COMPUTE IF: QDataBag.SSTRTReg IN [20 .. 21]

NIreland := No

COMPUTE IF: QDataBag.SSTRTReg IN [20 .. 21]

Scotland := No

COMPUTE IF: QDataBag.SSTRTReg IN [20 .. 21]

Wales := Yes

COMPUTE IF: QDataBag.SSTRTReg IN [20 .. 21]
AND: NatCen = Yes

SampInfo := 'Wales'

COMPUTE IF: QDataBag.SSTRTReg IN [30]

NIreland := Yes

COMPUTE IF: QDataBag.SSTRTReg IN [30]

Scotland := No

COMPUTE IF: QDataBag.SSTRTReg IN [30]

Wales := No

COMPUTE IF: NOT (QDataBag.SSTRTReg IN [30])

NIreland := No

COMPUTE IF: NOT (QDataBag.SSTRTReg IN [30])

Scotland := No

COMPUTE IF: NOT (QDataBag.SSTRTReg IN [30])

Wales := No

COMPUTE IF: NOT (QDataBag.SSTRTReg IN [30])
AND: NatCen = Yes

SampInfo := 'England'

COMPUTE IF: NatCen = Yes

SampInfo := ('
(Sample Info: ' + SampInfo + ' Case, Sample Month=' +
Months[QDataBag.SampMnth] + ', Sample Year=' +
STR(QDataBag.SampYear) + ')')

COMPUTE IF: Test = Yes
AND: NIDCoun = RESPONSE

NIDCoun := ORD(NIDCoun)

COMPUTE IF: QDataBag.NIDCoun IN [1 .. 97]

NIDCoun := QDataBag.NIDCoun

COMPUTE IF: QDataBag.NIRate IN [0.01 .. 9997]

NIRate := QDataBag.NIRate

COMPUTE ALWAYS:

GIP := '/ Guaranteed Income Payment (GIP)'

COMPUTE ALWAYS:

GIP1 := '/ Guaranteed Income Payment (GIP) paid to Service or
ex-service personnel'

COMPUTE ALWAYS:

GIP2 := '/ Guaranteed Income Payment (GIP) paid to surviving
partners of Service personnel'

COMPUTE IF: NatCen = NI

SharOwn := 'co-ownership'

COMPUTE IF: NatCen = NI

SOwners := 'Co-owners'

COMPUTE IF: NatCen = NI

LANIHE := 'NIHE'

COMPUTE IF: NatCen = NI

Councill1 := 'Northern Ireland Housing Executive'

COMPUTE IF: NatCen = NI

Council2 := 'Northern Ireland Housing Executive'

COMPUTE IF: NatCen = NI

GOVSSA := 'Social Security Agency'

COMPUTE IF: NatCen = NI

GOV1 := 'SSA'

COMPUTE IF: NatCen = NI

GOV2 := 'Social Security Agency'

COMPUTE IF: NatCen = NI

JobCen := 'a Social Security Office'

COMPUTE IF: NatCen = NI

RentReb1 := 'rent and/or rates rebate'

COMPUTE IF: NatCen = NI

RentReb2 := 'rent/rates rebate'

COMPUTE IF: NatCen = NI

LAuths := 'Social Services'

COMPUTE IF: NatCen = NI

LAuth1 := 'Social Services'

COMPUTE IF: NatCen = NI

LAuth2 := 'Social Services'

COMPUTE IF: NatCen = NI

IncROI1 := '
Include accounts held in the Republic Of Ireland.'

COMPUTE IF: NatCen = NI

IncROI2 := ('
If account held in the Republic of Ireland open note ' + 'to state
if amount recorded in Euros.')

COMPUTE IF: NatCen = NI

IncROI3 := ('
Include holdings and/or accounts held in the Republic ' + 'Of
Ireland and open note to state if amount recorded in ' + 'Euros.')

COMPUTE IF: NatCen = NI

Dept := 'Department for Social Development'

COMPUTE IF: NatCen = NI

Mid_Pri := '<Code Not Used>'

COMPUTE IF: NatCen = NI

Mid_Sec := '<Code Not Used>'

COMPUTE IF: NatCen = NI

Grammar := '/Grammar'

COMPUTE IF: NatCen = NI

State_run := ''

COMPUTE IF: NatCen = NI

assisted := ''

COMPUTE IF: NatCen = NI

Inland_Revenue := 'HM Revenue and Customs (formerly Inland
Revenue)'

COMPUTE IF: NatCen = NI

rate := ''

COMPUTE IF: NatCen = NI

RatesOrCharges := 'charges'

COMPUTE IF: NatCen = NI

Britain := 'the UK'

COMPUTE IF: NatCen = NI

CSA_Long := 'Child Maintenance and Enforcement Division (C-MED)'

COMPUTE IF: NatCen = NI

CSA_Abbr := 'C-MED'

COMPUTE IF: NatCen = NI

or_CSADWP := 'or the C-MED'

COMPUTE IF: NatCen = NI

ONSNISRA := 'NISRA'

COMPUTE IF: NOT (NatCen = NI)

SharOwn := 'shared ownership'

COMPUTE IF: NOT (NatCen = NI)

SOwners := 'Shared owners'

COMPUTE IF: NOT (NatCen = NI)

LANIHE := 'local authority'

COMPUTE IF: NOT (NatCen = NI)

Council1 := ('The local authority/council/New Town ' +
'development/Scottish Homes')

COMPUTE IF: NOT (NatCen = NI)

Council2 := 'Local Authority or Council (incl. GLC)'

COMPUTE IF: NOT (NatCen = NI)

GOVSSA := 'DWP'

COMPUTE IF: NOT (NatCen = NI)

GOV1 := 'DWP'

COMPUTE IF: NOT (NatCen = NI)

GOV2 := 'DWP'

COMPUTE IF: NOT (NatCen = NI)

JobCen := 'an Employment Service local office or Jobcentre'

COMPUTE IF: NOT (NatCen = NI)

RentReb1 := 'rent rebate'

COMPUTE IF: NOT (NatCen = NI)

RentReb2 := 'rent rebate'

COMPUTE IF: NOT (NatCen = NI)

LAuths := 'Local Authorities'

COMPUTE IF: NOT (NatCen = NI)

LAuth1 := 'Local Authority'

COMPUTE IF: NOT (NatCen = NI)

LAuth2 := 'a Local Authority'

COMPUTE IF: NOT (NatCen = NI)

IncROI1 := ''

COMPUTE IF: NOT (NatCen = NI)

IncROI2 := ''

COMPUTE IF: NOT (NatCen = NI)

IncROI3 := ''

COMPUTE IF: NOT (NatCen = NI)

Dept := 'Department for Work and Pensions'

COMPUTE IF: NOT (NatCen = NI)

Mid_Pri := 'Middle-deemed primary school (state run or assisted)'

COMPUTE IF: NOT (NatCen = NI)

Mid_Sec := 'Middle-deemed secondary school (state run or assisted)'

COMPUTE IF: NOT (NatCen = NI)

Grammar := ''

COMPUTE IF: NOT (NatCen = NI)

State_run := 'State run'

COMPUTE IF: NOT (NatCen = NI)

assisted := '(State run or assisted)'

COMPUTE IF: NOT (NatCen = NI)

Inland_Revenue := 'HM Revenue and Customs (formerly Inland Revenue)(or formerly the DSS)'

COMPUTE IF: NOT (NatCen = NI)

rate := '(rates)'

COMPUTE IF: NOT (NatCen = NI)

RatesOrCharges := 'rates or charges'

COMPUTE IF: NOT (NatCen = NI)

Britain := 'Britain'

COMPUTE IF: NOT (NatCen = NI)

CSA_Long := 'Child Support Agency (CSA)'

COMPUTE IF: NOT (NatCen = NI)

CSA_Abbr := 'CSA'

COMPUTE IF: NOT (NatCen = NI)

or_CSADWP := 'the CSA or DWP'

COMPUTE IF: NOT (NatCen = NI)

ONSNISRA := 'the Office for National Statistics'

ASK ALWAYS:

First

^I^B^IC^B For your information... You are in the ^B Household ^B Schedule for:-
 Area No@|@|:@|^StrArea
 Address No...@|:@|^StrAddr
 Household No@|:@|^QSerial.Hhold

- To go directly to 'Admin', press <Ctrl + Enter>.
- To continue with interview press '1' and <Enter>.

Interview version@|:@|^IVers@|@|@|
 ^EditVersion ^SampInfo

- (1) Continue
-

ASK IF: NatCen = Yes

AdrCheck

^I
 Refer to address label:
 Check that label gives respondent's full current address.

If not, amend address and code 'Address changed'.

- (1) Address confirmed
 - (2) Address changed
-

COMPUTE IF: (QDataBag.SampYear = RESPONSE) AND (QDataBag.SampMnth = RESPONSE)

ThisYear := QDataBag.SampYear

COMPUTE IF: (QDataBag.SampYear = RESPONSE) AND (QDataBag.SampMnth = RESPONSE)

ThisMnth := QDataBag.SampMnth

ASK IF: Test = Yes

ThisYear

FRS Survey Year (eg. 2010 = April 2010 - March 2011).

2010..2097

ASK IF: Test = Yes

ThisMnth

Enter survey month.

1..12

COMPUTE IF: ThisMnth IN [1 .. 12]

MnthOK2 := ThisMnth

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 11

MnthOK1 := 10

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 11

MnthOK3 := 12

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 11

MnthOK4 := 1

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 12

MnthOK1 := 11

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 12

MnthOK3 := 1

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 12

MnthOK4 := 2

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 1

MnthOK1 := 12

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 1

MnthOK3 := 2

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 1

MnthOK4 := 3

COMPUTE IF: ThisMnth IN [1 .. 12]
AND: ThisMnth = 4

MnthOK1 := 4

COMPUTE IF: *ThisMnth* IN [1 .. 12]
AND: *ThisMnth* = 4

MnthOK3 := 5

COMPUTE IF: *ThisMnth* IN [1 .. 12]
AND: *ThisMnth* = 4

MnthOK4 := 6

COMPUTE IF: *ThisMnth* IN [1 .. 12]
AND: NOT (*ThisMnth* = 4)

MnthOK1 := (ThisMnth - 1)

COMPUTE IF: *ThisMnth* IN [1 .. 12]
AND: NOT (*ThisMnth* = 4)

MnthOK3 := (ThisMnth + 1)

COMPUTE IF: *ThisMnth* IN [1 .. 12]
AND: NOT (*ThisMnth* = 4)

MnthOK4 := (ThisMnth + 2)

FRS1004B.QSignIn

RECORD ALWAYS:

StartDat

^I^IC The date on which the interview with this household was started.^I

DATE

RECORD ALWAYS:

DateOK

^I^IC Today's date according to the laptop is ^AStartD.

Is this the correct date?^I

- (1) Yes
 - (2) No
-

COMPUTE IF: StartDat = EMPTY

AStartD := SYSDATE

ASK IF: StartDat = EMPTY

DateOK

^I^IC Today's date according to the laptop is ^AStartD.

Is this the correct date?^I

- (1) Yes
 - (2) No
-

COMPUTE IF: StartDat = EMPTY
AND: DateOK = Yes

StartDat := AStartD

ASK IF: StartDat = EMPTY
AND: DateOK = No

BStartD

^I^IC Enter the date on which the interview with this household was started.^I

DATE

```

COMPUTE IF: StartDat = EMPTY
AND: DateOK = No
AND: (((BStartD.YEAR = ThisYear) OR (BStartD.YEAR = (ThisYear + 1))) AND
  (((BStartD.MONTH = MnthOK1) OR (BStartD.MONTH = MnthOK2)) OR
  (BStartD.MONTH = MnthOK3)) OR (BStartD.MONTH = MnthOK4))) OR (ThisYear <>
  RESPONSE)) OR (ThisMnth <> RESPONSE)

```

StartDat := BStartD

```

CHECK IF: StartDat = EMPTY
AND: DateOK = No
AND: NOT (((BStartD.YEAR = ThisYear) OR (BStartD.YEAR = (ThisYear + 1)))
  AND (((BStartD.MONTH = MnthOK1) OR (BStartD.MONTH = MnthOK2)) OR
  (BStartD.MONTH = MnthOK3)) OR (BStartD.MONTH = MnthOK4))) OR (ThisYear <>
  RESPONSE)) OR (ThisMnth <> RESPONSE)
  ((BStartD.YEAR = ThisYear) OR (BStartD.YEAR = (ThisYear + 1))) AND
  INVOLVING(BStartD)

```

^I The year is wrong for the current FRS survey (^ThisYear)!
Please re-enter the date from the beginning.^I

```

CHECK IF: StartDat = EMPTY
AND: DateOK = No
AND: NOT (((BStartD.YEAR = ThisYear) OR (BStartD.YEAR = (ThisYear + 1)))
  AND (((BStartD.MONTH = MnthOK1) OR (BStartD.MONTH = MnthOK2)) OR
  (BStartD.MONTH = MnthOK3)) OR (BStartD.MONTH = MnthOK4))) OR (ThisYear <>
  RESPONSE)) OR (ThisMnth <> RESPONSE)
  (((BStartD.MONTH = MnthOK1) OR (BStartD.MONTH = MnthOK2)) OR
  (BStartD.MONTH = MnthOK3)) OR (BStartD.MONTH = MnthOK4)) AND
  INVOLVING(BStartD)

```

^I The month is wrong for the current FRS survey (^Months[ThisMnth])!
Please re-enter the date from the beginning.^I

ASK IF: Test = Yes

CStartD

^I EDITOR: Change the interview start date (currently ^StartDat)?^I

- (1) Yes
 - (2) No
-

```

ASK IF: Test = Yes
AND: CStartD = Yes

```

StartDat

^I^IC The date on which the interview with this household was started.^I

DATE

DISPLAY IF: NOT (Test = Yes)

StartDat

^I^IC The date on which the interview with this household was started.^I

DATE

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

RECORD ALWAYS:

IntSTime

^I Interview start time^I

TIME

COMPUTE IF: IntSTime = EMPTY AND StartDat <> EMPTY

IntSTime := STARTTIME

ASK IF: (Edit = Yes) AND (NatCen <> Yes)

Editor

^I Editor at HQ: Enter your identification number.^I

1..97

FRS1004B (continued)
FAMILY RESOURCES SURVEY 2010-2011

COMPUTE ALWAYS:

DateNow := QSignIn.StartDat

COMPUTE ALWAYS:

DatYrAgo := (DateNow + (-1,0,0))

COMPUTE ALWAYS:

DatMnAgo := (DateNow + (0,-1,0))

COMPUTE ALWAYS:

DatWkAgo := (DateNow + (0,0,-7))

COMPUTE ALWAYS:

DatWeek := (Days[DatWkAgo.WEEKDAY] + ' the ' + STR(DatWkAgo.DAY)
+ ' ' + Months[DatWkAgo.MONTH])

COMPUTE IF: DateNow.WEEKDAY = 1

DatLSun := (DateNow + (0,0,-7))

COMPUTE IF: NOT (DateNow.WEEKDAY = 1)

DatLSun := (DateNow + (0,0,-(DateNow.WEEKDAY) + 1))

COMPUTE ALWAYS:

DatSun := ('Sunday the ' + STR(DatLSun.DAY) + ' ' +
Months[DatLSun.MONTH])

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

COMPUTE ALWAYS:

FWDate := TODATE(ChkYear,QDataBag.SampMnth,1)

WARN IF: QDataBag.SampYear <> 0
(QSignIn.StartDat.YEAR = FWDate.YEAR) OR ((QSignIn.StartDat.YEAR =
(FWDate.YEAR + 1)) AND (QSignIn.StartDat <= (FWDate + (0,3,0))))

^I

You have accidentally entered the wrong year. It doesn't agree with the fieldwork period. Please check and amend.

WARN IF: QDataBag.SampMnth <> 0
(QSignIn.StartDat >= FWDate) OR ((FWDate.MONTH = 12) AND
(QSignIn.StartDat.MONTH = 11))

^I

You have accidentally entered the wrong month and/or year. It doesn't agree with the fieldwork period. Please check and amend.

WARN IF: QDataBag.SampMnth <> 0
QSignIn.StartDat <= (FWDate + (0,3,0))

^I

The month of this date is more than 3 months after the fieldwork period, please check and amend.

COMPUTE ALWAYS:

DLYear := (QSignIn.StartDat + (-1,0,0))

FRS1004B.QNames

Names of household members

ASK ALWAYS :

WhoHere

^N

Who normally lives at this address?

(1) Press <Enter> to continue.

FRS1004B.QNames.M[]

ASK IF: In loop FOR Pers := 1 TO 14
AND: (Pers = 1) OR (M[Pers - 1].More = Yes)

Name

^I^IC Enter an identifier for this household member

Please take care to record an accurate name. If the respondent refuses to give his or her name - record something that uniquely identifies this person within the household so you can refer to them later in the interview.

STRING[15]

FRS1004B.QNames.M[.].ProperAdd()

Procedure Call

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
```

OAddLine := AddLine

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
```

PNoChar := NoChar

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
```

NLetter := 0

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
```

NLettID := 0

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
```

AVar[Idx] := LOWERCASE(SUBSTRING(OAddLine,Idx,1))

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
```

ASCIICode[Idx] := SOMETHING(AVar[Idx])

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: ASCIICode[Idx] IN [48 .. 57, 65 .. 90, 97 .. 122]
```

NLetter := (NLetter + 1)

```
COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: ASCIICode[Idx] IN [48 .. 57, 65 .. 90, 97 .. 122]
  AND: NLettID = 0
```

NLettID := Idx

```

CHECK IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: NOT (ASCIICode[1] IN [32, 48 .. 57, 65 .. 90, 97 .. 122])
  ERROR AND INVOLVING(AddLine)

```

You started the text with an invalid character (^AVar[Idx]).

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: ASCIICode[1] IN [97 .. 122]

```

ASCIICode[1] := (ASCIICode[1] - 32)

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: ASCIICode[1] IN [97 .. 122]

```

AVar[1] := UPCASE(AVar[1])

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: Idx > 1
  AND: (ASCIICode[Idx - 1] IN [32, 45, 46]) AND (ASCIICode[Idx] IN [65 .. 90, 97 .. 122])

```

ASCIICode[Idx] := (ASCIICode[Idx] - 32)

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: Idx > 1
  AND: (ASCIICode[Idx - 1] IN [32, 45, 46]) AND (ASCIICode[Idx] IN [65 .. 90, 97 .. 122])

```

AVar[Idx] := UPCASE(AVar[Idx])

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: Idx > 1
  AND: (ASCIICode[Idx - 2] = 77) AND (ASCIICode[Idx - 1] = 99)

```

ASCIICode[Idx] := (ASCIICode[Idx] - 32)

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: Idx > 1
  AND: (ASCIICode[Idx - 2] = 77) AND (ASCIICode[Idx - 1] = 99)

```

```

AVar[Idx] := UPCASE(AVar[Idx])

```

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar

```

```

PAddLine := (PAddLine + AVar[Idx])

```

```

CHECK IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  RESERVECHECK

```

```

  RESERVECHECK

```

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: PAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar

```

```

AVar2[Idx] := SUBSTRING(PAddLine,Idx,1)

```

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: PAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar

```

```

ASCIICode2[Idx] := SOMETHING(AVar2[Idx])

```

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: PAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: (ASCIICode2[Idx] = 32) AND NOT (ASCIICode[Idx + 1] IN [0, 48 .. 57,
    65 .. 90, 97 .. 122])
  AND: In loop FOR Idy := 1 TO PNoChar
  AND: Idy >= Idx

```

```

ASCIICode2[Idx] := ASCIICode2[Idy + 1]

```

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: PAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: (ASCIICode2[Idx] = 32) AND NOT (ASCIICode[Idx + 1] IN [0, 48 .. 57,
    65 .. 90, 97 .. 122])
  AND: In loop FOR Idy := 1 TO PNoChar
  AND: Idy >= Idx

```

AVar2[Idx] := AVar2[Idy + 1]

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: PAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar
  AND: (Idx < NLetID) AND (AVar2[Idx] = )

```

AVar2[Idx] := ''

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE
  AND: PAddLine = RESPONSE
  AND: In loop FOR Idx := 1 TO PNoChar

```

PFAddLine := (PFAddLine + AVar2[Idx])

```

COMPUTE IF: In loop FOR Pers := 1 TO 14
  AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
  AND: Name = RESPONSE
  AND: OAddLine = RESPONSE

```

EFAddLine := PFAddLine

FRS1004B.QNames.M[] (continued)

ASK IF: In loop FOR Pers := 1 TO 14
 AND: (Pers = 1) OR (M[Pers - 1].More = Yes)
 AND: PPers < 14

More

^N

Is there anyone else in this household?

- (1) Yes
- (2) No

FRS1004B.QNames (continued)

Names of household members

COMPUTE IF: In loop FOR Pers := 1 TO 14
 AND: M[Pers].More = No

HSize := Pers

CHECK IF: In loop FOR Pers := 1 TO 14
 RESERVECHECK

RESERVECHECK

CHECK IF: In loop FOR Pers := 1 TO 14
 RESERVECHECK

RESERVECHECK

CHECK IF: In loop FOR Pers := 1 TO 14
 RESERVECHECK

RESERVECHECK

COMPUTE IF: (M[13].More = Yes) AND M[14].Name <> EMPTY

HSize := 14

CHECK ALWAYS:
 RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
 RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
 RESERVECHECK

RESERVECHECK

FRS1004B (continued)
FAMILY RESOURCES SURVEY 2010-2011

RECORD ALWAYS:

HHSize

Household size including any x-ed out

0..14

COMPUTE IF: (QNames.HSize > 0) OR (Edit = No)

HHSize := QNames.HSize

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSize

DMName[Loop1] := QNames.M[Loop1].Name

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSize

DMTypeEd[Loop1] := HHG.P[Loop1].TypeEd

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

FRS1004B.HHG

Data on household members

COMPUTE IF: HHSize > 0

FHHSIZE := PHHSIZE

COMPUTE IF: HHSize > 0
AND: NatCen = Yes

Out772 := '772'

COMPUTE IF: HHSize > 0
AND: NOT (NatCen = Yes)

Out772 := '77'

COMPUTE IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE

P[P1].Person := P1

COMPUTE IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE

P[P1].Name := QNames.M[].Name[P1]

FRS1004B.HHG.P[]

RECORD IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSize*

BenUnit

^I HHG^I
Benefit Unit number.

0..7

RECORD IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSize*

Person

^I HHG^I
^N
Person number in Household Grid.

0..14

RECORD IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSize*

Name

^I HHG^I
^N
First name.

STRING[15]

COMPUTE IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSize*

LName := Name

COMPUTE IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSize*

UName := UPCASE(Name)

ASK IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSize*

Sex

^I HHG^I
^I
Code ^UName's sex.

(1) Male
(2) Female

```

DISPLAY IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

Name

```

^I HHG^I
^N
First name.

STRING[15]

```

```

DISPLAY IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

DVAge

```

^I HHG^I
Derived age variable

0..120

```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: Sex = Male

```

```
heshe := 'he'
```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: Sex = Male

```

```
hisher := 'his'
```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (Sex = Male)

```

```
heshe := 'she'
```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (Sex = Male)

```

```
hisher := 'her'
```

```

ASK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

DoB

```

^I HHG^I
Date of birth is collected so that researchers, using the data from the study, can compare the living
standards and circumstances of people of different ages. This is important as it helps in identifying which
age groups are most in need and which policy changes are needed to help them.

DATE

```

```

WARN IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: DoB = RESPONSE
  DoB <= QSignIn.StartDat

```

```

^I
You've entered a future date!

```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: DoB = RESPONSE

```

```

DVAge := AGE(DoB,QSignIn.StartDat)

```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: DoB = RESPONSE

```

```

AgeOf := DVAge

```

```

ASK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (DoB = RESPONSE)

```

AgeOf

```

^I HHG^I
^N
What was ^LName's age last birthday?^N
^I
If age not given, probe for an estimate.
For later routing, you must know whether:

```

- A) Men are aged 16-64 or 65+
- B) Women are aged 16-59 or 60+

```

0..120

```

```

WARN IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (DoB = RESPONSE)
  AND: (AgeOf = 0) AND (AgeOf = RESPONSE)
  ((QSignIn.StartDat.JULIAN - DoB.JULIAN) <= 365) AND INVOLVING(AgeOf,DoB)

```

```

^I
This date doesn't agree with the age and the date of interview. Please check.

```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (DoB = RESPONSE)

```

```

DVAge := AgeOf

```

```

ASK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: DVAge IN [16 .. 120]

```

MS

^I HHG^I

^I

The aim is to obtain legal marital status, irrespective of any de facto arrangement. The only qualification to this aim is that you should not probe the answer 'separated'. Should a respondent query the term, explain that it covers any person whose spouse is living elsewhere because of estrangement (whether the separation is legal or not). Ignore temporary absences, eg. on oil rig.

A person whose spouse has been working away from home for over 6 months, eg. on a contract overseas or in the armed forces, should still be coded as married and living with husband/wife if the separation is not permanent.

Civil partners may only be of the same sex, and must have obtained legal recognition of their partnership; probe whether partnership was registered under the new provisions that came into force as from December 2005.

- (1) ^N...single, that is, never married
 - (2) ^N...married and living with husband/wife
 - (3) ^N...a civil partner in a legally-recognised Civil partnership
 - (4) ^N...married and separated from husband/wife
 - (5) ^N...divorced
 - (6) ^N...or widowed?
 - (7) ^I Spontaneous only -^I^N In a legally-recognised Civil Partnership and separated from his/her partner
 - (8) ^I Spontaneous only -^I^N Formerly a civil partner, the Civil Partnership now dissolved
 - (9) ^I Spontaneous only -^I^N A surviving civil partner; his/her partner having since died
-

```

ASK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: DVAge IN [16 .. 120]
AND: (FHHSIZE > 1) AND (MS IN [Single, Seperate, Divorced, Widowed, CPSep, CPDiss, CPDied])

```

CupChk

^I HHG^I

^I

Ask or record^I ^N

May I just check, are you / is ^LName living with someone in this household as a couple?^N

^I^IC Only respondents who are living with their partner should be coded as living together as a couple. You may code No without asking the question^B only^B if all members of the household are too closely related for any to be living together in a de facto marital relationship.

- (1) Yes
- (2) No

```

ASK IF: HHSize > 0
      AND: In loop FOR P1 := 1 TO FHHSIZE
      AND: SUBSTRING (Name, 1, 2) <> XX
      AND: MS = Widowed

```

W1

```

^I HHG^I
^N
What was ^LName's age when widowed?

0..120

```

```

WARN IF: HHSize > 0
      AND: In loop FOR P1 := 1 TO FHHSIZE
      AND: SUBSTRING (Name, 1, 2) <> XX
      AND: MS = Widowed
      AND: W1 = RESPONSE
      W1 >= 16

```

```

^I
Are you sure? It is not usual to be married before the age of 16 unless you were married outside the UK.

```

```

CHECK IF: HHSize > 0
      AND: In loop FOR P1 := 1 TO FHHSIZE
      AND: SUBSTRING (Name, 1, 2) <> XX
      AND: MS = Widowed
      AND: W1 = RESPONSE
      (W1 <= DVAge) AND INVOLVING (AgeOf, DoB, W1)

```

```

^I
You've coded that ^LName is ^DVAge years old, but was widowed at the age of ^W1. Please amend the one or the other.

```

```

ASK IF: HHSize > 0
      AND: In loop FOR P1 := 1 TO FHHSIZE
      AND: SUBSTRING (Name, 1, 2) <> XX
      AND: MS = Widowed

```

W2

```

^I HHG^I
^N
Did ^LName have any children aged under 16 when widowed?

(1)  Yes
(2)  No

```

```

COMPUTE IF: HHSize > 0
      AND: In loop FOR P1 := 1 TO FHHSIZE
      AND: SUBSTRING (Name, 1, 2) <> XX
      AND: Sex = Male

```

SonDaughter := 'son'

```

COMPUTE IF: HHSize > 0
      AND: In loop FOR P1 := 1 TO FHHSIZE
      AND: SUBSTRING (Name, 1, 2) <> XX
      AND: Sex = Male

```

BrotherSister := 'brother'

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: Sex = Male

```

FatherMother := 'father'

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (Sex = Male)

```

SonDaughter := 'daughter'

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (Sex = Male)

```

BrotherSister := 'sister'

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: NOT (Sex = Male)

```

FatherMother := 'mother'

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
RESERVECHECK

RESERVECHECK

FRS1004B.HHG.P[.QRel[]

```

ASK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
AND: RPers < PPers

```

R

^I HHG^I

^I

Ask or record ^PName's relationship to ^RName.

^IC Grandparents who are the ^B legal guardian ^B of their grandchild should be recorded as 7 'father/mother (or guardian)'. The grandchild should be recorded as 3 'son/daughter (incl. adopted/legal dependant)'. This ensures that the adult and child are allocated to the correct Benefit Unit.

- (1) spouse,
- (2) cohabitee,
- (3) ^SonDaughter (incl. adopted/legal dependant),
- (4) step-^SonDaughter,
- (5) foster child,
- (6) ^SonDaughter-in-law,
- (7) ^FatherMother (or guardian),
- (8) step-^FatherMother,
- (9) foster parent,
- (10) ^FatherMother-in-law,
- (11) ^BrotherSister (incl. adopted),
- (12) step-^BrotherSister,
- (13) foster ^BrotherSister,
- (14) ^BrotherSister-in-law,
- (15) grand-^SonDaughter,
- (16) grand-^FatherMother,
- (17) other relative,
- (18) other non-relative
- (20) Civil Partner
- (97)

```

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
AND: RPers < PPers
R <> Self

```

^I

Code 97 is not valid for this question.

```

WARN IF: HHSIZE > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
AND: RPers < PPers
NOT (IN (R, [???,???,???]))

```

^I

Warning: This code must only be used if the foster child is covered by a Local Authority Allowance.
Please check that this is the case.

```

RECORD IF: HHSIZE > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
AND: NOT (RPers < PPers)

```

R

^I HHG^I

^I

Ask or record ^PName's relationship to ^RName.

^IC Grandparents who are the ^B legal guardian ^B of their grandchild should be recorded as 7 'father/mother (or guardian)'. The grandchild should be recorded as 3 'son/daughter (incl. adopted/legal dependant)'. This ensures that the adult and child are allocated to the correct Benefit Unit.

- (1) spouse,
 - (2) cohabitee,
 - (3) ^SonDaughter (incl. adopted/legal dependant),
 - (4) step-^SonDaughter,
 - (5) foster child,
 - (6) ^SonDaughter-in-law,
 - (7) ^FatherMother (or guardian),
 - (8) step-^FatherMother,
 - (9) foster parent,
 - (10) ^FatherMother-in-law,
 - (11) ^BrotherSister (incl. adopted),
 - (12) step-^BrotherSister,
 - (13) foster ^BrotherSister,
 - (14) ^BrotherSister-in-law,
 - (15) grand-^SonDaughter,
 - (16) grand-^FatherMother,
 - (17) other relative,
 - (18) other non-relative
 - (20) Civil Partner
 - (97)
-

```

COMPUTE IF: HHSIZE > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
AND: NOT (RPers < PPers)
AND: RPers = PPers

```

R := Self

WARN IF: *HHSize* > 0
AND: In loop FOR *P1* := 1 TO *FHHSIZE*
AND: *SUBSTRING* (*Name*, 1, 2) <> XX
AND: In loop FOR *R1* := 1 TO *FHHSIZE*
RESERVECHECK

RESERVECHECK

WARN IF: *HHSize* > 0
AND: In loop FOR *P1* := 1 TO *FHHSIZE*
AND: *SUBSTRING* (*Name*, 1, 2) <> XX
AND: In loop FOR *R1* := 1 TO *FHHSIZE*
RESERVECHECK

RESERVECHECK

FRS1004B.HHG.P[] (continued)

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: In loop FOR R1 := 1 TO FHHSIZE
RESERVECHECK

```

RESERVECHECK

```

RECORD IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX

```

Spouses

^I HHG^I

0..14

```

RECORD IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX

```

NumParn

^I HHG^I

0..14

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

NumPart

^I HHG^I

0..14

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

NumCohab

^I HHG^I

0..14

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

Parent1

^I HHG^I

Person number of parent 1

0..14

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

Parent2

^I HHG^I

Person number of parent 2

0..14

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

Hholder

^I HHG^I

Is this person coded at QHholder.Hhldr?

- (1) Yes
- (2) No

```

ASK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: DVAge IN [4 .. 74]

```

FTEd

```

^I HHG^I
^N
Is ^LName currently in full-time education?^N

```

^M^IC Include correspondence courses and open learning as well as other forms of full-time courses.

- (1) Yes
- (2) No

```

WARN IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTEd = No)) OR (DVAge IN [19 .. 120])
  AND: DVAge IN [19 .. 23]

```

StillEduc := (' - or is ' + heshe + ' still in full-time education')

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTEd = No)) OR (DVAge IN [19 .. 120])
  AND: NOT (DVAge IN [19 .. 23])

```

StillEduc := ''

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTEd = No)) OR (DVAge IN [19 .. 120])
  AND: DVAge IN [19 .. 25]

```

**continuous := ('Continuous' can include a break, if less than ' + '18 months.
Code '96' if still in continuous F/T Ed.)**

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: DVAge IN [26 .. 120]

```

```

continuous := ('Enter age (or code '96' if still in ' + 'continuous full-time education).')

```

```

COMPUTE IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: NOT (DVAge IN [26 .. 120])

```

```

continuous := ''

```

```

ASK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])

```

TEA

^I HHG^I

^I

Include the following as part of 'continuous education':

- A ^B'gap year'^B between school and college/university, as long as there is/was some clear intention to continue education.
- A^B holiday job^B during a course, provided they intend to continue with the course.
- The working section of a^B sandwich course^B.
- ^B National Service^B, if it occurred between school and college/university.

5..97

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: TEA IN [5 .. 95]
  (TEA <= DVAge) AND INVOLVING (AgeOf, DoB, TEA)

```

^I

You've coded that ^LName is ^DVAge years old, but left full-time education at the age of ^TEA. Please amend the one or the other.

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: (FTed = No) AND (DVAge IN [16 .. 18])
  TEA <> 96

```

^I

At the previous question you say that ^LName is not in full-time education. Please amend your answers.

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])

```

TEAEx

^I HHG^I

^I

This is a 'Key Question': it is very important to get an answer if you possibly can. An ESTIMATE is preferable to Don't know (or refusal).

^SuppTxt

OPEN

```

WARN IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: Edit = No
  AND: TEA = NONRESPONSE
  ERROR

```

^I

This is a 'Key Question': it is very important to get an answer if you possibly can. An ESTIMATE is preferable to Don't know (or refusal).

```

ASK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: Edit = No
  AND: TEA = NONRESPONSE

```

TEAEx

^I HHG^I

^I

This is a 'Key Question': it is very important to get an answer if you possibly can. An ESTIMATE is preferable to Don't know (or refusal).

^SuppTxt

OPEN

```

WARN IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX
  AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
  AND: Edit = No
  AND: TEA = RESPONSE
  AND: DVAge > 25
  TEA <> 96

```

^I

This person is over 25, so is unlikely to still be in CONTINUOUS full-time education (ie. having been OUT of education for less than 18 months). Please check.

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
AND: Edit = No
AND: TEA = RESPONSE
      (TEA >= 14) OR (TEA = 97)

```

^I

This value seems low.

Please check that it is correct.

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: ((DVAge IN [16 .. 18]) AND (FTed = No)) OR (DVAge IN [19 .. 120])
AND: Edit = No
AND: TEA = RESPONSE
AND: TEA < 96
TEA < 28

```

^I

This value seems high.

Please check that it is correct.

```

ASK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: (TEA = 96) OR (FTed = Yes)

```

TypeEd

^I HHG^I

^I

Secondary Schools include: Secondary Modern, Grammar, Comprehensive and Technical Schools.

Age ranges for Middle-deemed Primary:

8-12

9-12

9-13

Age ranges for Middle-deemed Secondary:

9-13

10-13

10-14.

- (1) Nursery School/Nursery Class/Playgroup/Pre-school
- (2) ^State_run Primary (including reception classes)
- (3) Special School ^State_run (e.g. for children with disabilities and special educational needs)
- (4) ^Mid_Pri
- (5) ^Mid_Sec
- (6) Secondary^Grammar school ^assisted
- (7) Non-advanced further education/ 6th form/tertiary/further education college
- (8) Any PRIVATE/Independent school (prep, primary, secondary, City Technology Colleges)
- (9) University/polytechnic/any other higher education
- (10) Home Schooling

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: (TEA = 96) OR (FTed = Yes)
AND: TypeEd = Nursery
DVAge <= 5

```

^I

This doesn't sound right in relation to ^LName's age:
Please check your entry.

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: (TEA = 96) OR (FTed = Yes)
AND: TypeEd = Primry
IN(DVAge, [4..11])

```

^I

This doesn't sound right in relation to ^LName's age:
Please check your entry.

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: (TEA = 96) OR (FTed = Yes)
AND: TypeEd IN [Special .. MidSec, Private, HomeSch]
IN(DVAge, [4..18])

```

^I

This doesn't sound right in relation to ^LName's age:
Please check your entry.

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: (TEA = 96) OR (FTed = Yes)
AND: TypeEd = Sec
IN(DVAge, [11..18])

```

^I

This doesn't sound right in relation to ^LName's age:
Please check your entry.

```

WARN IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: SUBSTRING (Name, 1, 2) <> XX
AND: (TEA = 96) OR (FTed = Yes)
AND: TypeEd = NonAdvFE
DVAge >= 16

```

^I

This doesn't sound right in relation to ^LName's age:
Please check your entry.

WARN IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSIZE*
AND: *SUBSTRING (Name, 1, 2) <> XX*
AND: *(TEA = 96) OR (FTed = Yes)*
AND: *TypeEd = Univ*
DVAge >= 18

^I

This doesn't sound right in relation to ^LName's age:
Please check your entry.

ASK IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSIZE*
AND: *SUBSTRING (Name, 1, 2) <> XX*
AND: *(TEA = 96) OR (FTed = Yes)*
AND: *(TypeEd IN [Special, HomeSch]) AND (DVAge IN [16 .. 19])*

SchChk

^I HHG^I

^I^IC Please check: Is^B Child Benefit^B still received for this person?
(If yes, this confirms they still belong to someone else's benefit unit).

- (1) Yes, child benefit still received
- (2) No

CHECK IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSIZE*
AND: *SUBSTRING (Name, 1, 2) <> XX*
AND: *NatCen = NI*
NOT (IN (TypeEd, [???, ???]))

^I^IC Do not use this code at TypeEd.

ASK IF: *HHSize > 0*
AND: *In loop FOR P1 := 1 TO FHHSIZE*
AND: *SUBSTRING (Name, 1, 2) <> XX*
AND: *((DVAge IN [16 .. 19]) AND (FTed = No)) AND (TEA < 96)*

Trainee

^I HHG^I

^N

Is ^LName currently on a government scheme for employment training?

- (1) Yes
- (2) No

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

Depend

^I HHG^I

^N

Status indicator of whether this adult is treated as dependent.

- (1) Independent adult
 - (2) 16-19 years old AND in F/T education, OR 16-19 in government employment training
 - (3) 0-15 years old
-

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

LiveWith

^I HHG^I

Cohabitee?

- (1) Yes
 - (2) No
-

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

DVMarDF

^I HHG^I

De facto marital status

- (1) Married or in a legally recognised Civil Partnership
 - (2) Cohabiting (including same sex couples)
 - (3) Single
 - (4) Widowed (including surviving civil partner)
 - (5) Divorced or civil partnership dissolved
 - (6) Separated
-

```

RECORD IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: SUBSTRING (Name, 1, 2) <> XX

```

FosterP

^I HHG^I

Whether this adult is treated as a foster parent.

- (1) Yes
 - (2) No
-

```

CHECK IF: HHSize > 0
  AND: In loop FOR P1 := 1 TO FHHSIZE
  RESERVECHECK

```

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK

RESERVECHECK

FRS1004B.HHG (continued)**Data on household members**

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

LegGuard[[P1] := 2

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].NumParn := 0

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].NumPart := 0

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].NumCohab := 0

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].Spouses := 0

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].Parent1 := 1

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].Parent2 := 1

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].LiveWith := No

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE

```

```

P[P1].FosterP := No

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P2 > P1
  AND: P[P2].QRel[P1].R IN [Spouse .. Cohabit, Sib .. ILSib, OthRel, NonRel,
  CivilP]

```

P[P1].QRel[P2].R := P[P2].QRel[P1].R

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P2 > P1
  AND: P[P2].QRel[P1].R IN [Child .. ILChild]

```

P[P1].QRel[P2].R := (ORD(P[P2].QRel[P1].R) + 4)

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P2 > P1
  AND: P[P2].QRel[P1].R IN [Parent .. ILParent]

```

P[P1].QRel[P2].R := (ORD(P[P2].QRel[P1].R) - 4)

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P2 > P1
  AND: P[P2].QRel[P1].R IN [GChild]

```

P[P1].QRel[P2].R := GPARENT

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P2 > P1
  AND: P[P2].QRel[P1].R IN [GPARENT]

```

P[P1].QRel[P2].R := GChild

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R = Cohabit

```

P[P1].LiveWith := Yes

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R = Cohabit

```

P[P1].DVMarDF := Cohab

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P2].QRel[P1].R = FChild

```

P[P1].FosterP := Yes

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].MS IN [Married, CPart]

```

P[P1].DVMarDF := Married

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].CupChk = Yes

```

P[P1].DVMarDF := Cohab

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].DVAge < 16

```

P[P1].DVMarDF := DFSingle

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].MS = Single) AND (P[P1].LiveWith <> Yes)

```

P[P1].DVMarDF := DFSingle

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].MS = Widowed

```

P[P1].DVMarDF := DFWidow

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].MS = Divorced

```

P[P1].DVMarDF := DFDivor

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].MS = Seperate

```

P[P1].DVMarDF := DFSepar

WARN IF: HHSize > 0

AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R IN [Spouse, CivilP]
P[P1].Sex <> P[P2].Sex AND INVOLVING(P[P1].Sex, P[P2].Sex)

^I

Civil partners must have obtained legal recognition of their partnership. Please check whether partnership was registered under the new provisions that came into force as from December 2005.

CHECK IF: HHSize > 0

AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R IN [Spouse, CivilP]
((IN(P[P2].MS, [???, ???])) OR P[P2].MS=EMPTY) AND
INVOLVING(P[P1].MS, P[P2].MS)

^I

You've recorded ^P[P1].Name as the spouse / civil partner of ^P[P2].Name who is NOT 'Married & living with spouse / is NOT in a legally recognised Civil Partnership and living with civil partner'. Please amend one or the other.

CHECK IF: HHSize > 0

AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R = Cohabit
NOT(IN(P[P2].MS, [???, ???])) AND INVOLVING(P[P1].MS, P[P2].MS)

^I

You've recorded ^P[P1].Name as 'cohabiting' with ^P[P2].Name, who is 'MARRIED & living with^B spouse^B / in a legally recognised Civil Partnership and living with^B civil partner^B'. Please amend one or the other

WARN IF: HHSize > 0

AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R = Cohabit
P[P1].Sex <> P[P2].Sex AND INVOLVING(P[P2].QRel[P1].R)

^I

A cohabiting partner has been found in our data to usually be of the opposite sex. Are you sure this is a same sex cohabiting couple?

CHECK IF: HHSize > 0

AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P2].QRel[P1].R = Spouse
((P[P1].MS = Married) OR P[P1].MS=EMPTY) AND INVOLVING(P[P1].MS, P[P2].MS)

^I

You've recorded ^P[P2].Name as the spouse of ^P[P1].Name, who is not coded as being 'Married & living with spouse. Please amend one or the other

```

CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P2].QRel[P1].R IN [CivilP]
((P[P1].MS = CPart) OR P[P1].MS=EMPTY) AND INVOLVING(P[P1].MS,P[P2].MS)

```

^I

You've recorded ^P[P2].Name as the civil partner of ^P[P1].Name, who is not coded as being 'In a legally recognised Civil Partnership and living with civil partner'. Please amend one or the other

```

CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P2].QRel[P1].R = Cohabit
NOT (IN(P[P1].MS, [???, ???])) AND INVOLVING(P[P1].MS,P[P2].MS)

```

^I

You've recorded ^P[P2].Name as 'cohabiting' with ^P[P1].Name, who is coded as 'MARRIED & living with^B spouse^B / in a legally recognised Civil Partnership and living with^B civil partner^B'. Please amend one or the other

```

WARN IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R IN [Parent .. ILParent, GParent]
(P[P1].DVAge > 15) AND INVOLVING(P[P2].QRel[P1].R,P[P1].DVAge)

```

^I

You've coded ^P[P1].Name as a parent (inc. foster/in-law/step) or grandparent, but he/she is less than 16 years old. Please check ^P[P1].Name's age.

```

WARN IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R IN [Child .. ILChild, GChild]
(P[P1].DVAge < P[P2].DVAge) AND INVOLVING(P[P2].QRel[P1].R)

```

^I

Children (inc. foster/in-law/step) and grandchildren should normally be younger than their parents/grandparents/step-parents. Please check the ages you have entered.

```

WARN IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
AND: In loop FOR P2 := 1 TO FHHSIZE
AND: P[P1].QRel[P2].R IN [Parent .. ILParent, GParent]
(P[P1].DVAge > P[P2].DVAge) AND INVOLVING(P[P2].QRel[P1].R)

```

^I

Parents (inc. foster/in-law/step) or grandparents, are normally older than their child/grandchild/step-child. Please check the ages and relationships you've entered.

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Child .. ILChild]
  (P[P1].DVAge < (P[P2].DVAge - 12)) AND INVOLVING(P[P2].QRel[P1].R)

^I

```

Children (inc. foster/in-law/step) are normally at least 12 years younger than parents/step-parents. Please check the ages you have entered.

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Parent .. ILParent]
  (P[P1].DVAge > (P[P2].DVAge + 12)) AND INVOLVING(P[P2].QRel[P1].R)

^I

```

Parents (inc. foster/in-law/step) are normally at least 12 years older than their child/step-child. Please check the ages and relationships you've entered.

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R = GChild
  (P[P1].DVAge < (P[P2].DVAge - 24)) AND INVOLVING(P[P2].QRel[P1].R)

^I

```

Children are normally at least 24 years younger than their grandparents. Please check the ages you have entered.

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R = GPARENT
  (P[P1].DVAge > (P[P2].DVAge + 24)) AND INVOLVING(P[P2].QRel[P1].R)

^I

```

Grandparents are normally at least 24 years older than their grandchildren. Please check the ages and relationships you've entered.

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Spouse, CivilP]

```

```

P[P1].Spouses := (P[P1].Spouses + 1)

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Spouse .. Cohabit, CivilP]

```

```

P[P1].NumPart := (P[P1].NumPart + 1)

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R = Cohabit

```

```

P[P1].NumCohab := (P[P1].NumCohab + 1)

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].DVAge IN [0 .. 15]) AND (P[P1].DVAge = RESPONSE)

```

```

P[P1].Depend := Child

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].NumPart > 0

```

```

P[P1].Depend := Adult

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].DVAge IN [16 .. 18]) AND ((P[P1].TypeEd = Special) AND
    (P[P1].SchChk = No))

```

```

P[P1].Depend := Adult

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].DVAge IN [16 .. 19]) AND (((P[P1].TypeEd IN [Special ..
    Private]) OR (P[P1].Trainee = Yes)) OR ((P[P1].TypeEd = HomeSch) AND
    (P[P1].SchChk = YesCB)))

```

```

P[P1].Depend := DepAd

```

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: NOT ((P[P1].DVAge IN [16 .. 19]) AND (((P[P1].TypeEd IN [Special ..
    Private]) OR (P[P1].Trainee = Yes)) OR ((P[P1].TypeEd = HomeSch) AND
    (P[P1].SchChk = YesCB))))

```

```

P[P1].Depend := Adult

```

```

CHECK IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  (IN(P[P1].NumPart, [0..1])) AND
  INVOLVING(P[P2].QRel[P1].R, P[P1].QRel[P2].R)

```

^I

^P[P1].Name has more than one spouse/cohabitee. Establish who is principal partner, & re-code the other as '17' or '18'.

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Child .. FChild]
  AND: P[P1].Parent1 = EMPTY

```

P[P1].Parent1 := P2

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Child .. FChild]
  AND: P[P1].Parent2 = EMPTY AND (P[P1].Parent1 <> P2)
  P[P[P1].Parent1].Sex <> P[P2].Sex AND INVOLVING(P[P2].QRel[P1].R)

```

^I

The parents of ^P[P1].Name are of the same sex. Please check.

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Child .. FChild]
  AND: P[P1].Parent2 = EMPTY AND (P[P1].Parent1 <> P2)

```

P[P1].Parent2 := P2

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: P[P1].QRel[P2].R IN [Child .. FChild]

```

P[P1].NumParn := (P[P1].NumParn + 1)

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].Depend = Adult) AND INVOLVING (P[P2].QRel[P1].R,
  P[P1].QRel[P2].R)
  P[P1].NumParn <= 2

```

^I

This suggests that ^P[P1].Name has more than two parents. Please check the relationship codes for ^P[P1].Name and select which one to alter.

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].Depend = DepAd) AND INVOLVING (P[P2].QRel[P1].R,
  P[P1].QRel[P2].R)
  P[P1].NumParn <= 2

```

^I

You've given ^P[P1].Name more than two parents (inc step/foster). To calculate Benefit Units properly you must reduce this to a maximum of two. Select which one to alter, and re-code as '17'. (Check who receives Child Benefit for ^P[P1].Name).

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].Depend = Child) AND (P[P1].QRel[P2].R IN [Child .. FChild])

```

LegGuard[[P1] := 1

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: In loop FOR P2 := 1 TO FHHSIZE
  AND: (P[P1].Depend = DepAd) AND (P[P1].QRel[P2].R IN [Spouse .. FChild, CivilP])

```

LegGuard[[P1] := 1

```

WARN IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: (P[P1].Depend = DepAd) AND (P[PHHSIZE].QRel[PHHSIZE - 1].R = RESPONSE)
  (LegGuard[[P1] = 1) AND INVOLVING(P[P1].QRel[1].R, P[P1].Sex)

```

^I

Who in the household is responsible for ^P[P1].Name - is there a legal guardian, or does anyone get Child Benefit for ^P[P1].Name? If so, recode ^P[P1].Name as that person's legal dependent (Code 3) or that person as ^P[P1].Name's parent (code 7). If not, suppress warning and continue.

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: P[P1].Sex = Male

```

himher := 'him'

```

COMPUTE IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: P[P1].Sex = Female

```

himher := 'her'

```

CHECK IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  AND: (P[P1].Depend = Child) AND (P[PHHSIZE].QRel[PHHSIZE - 1].R = RESPONSE)
  (LegGuard[[P1] = 1) AND INVOLVING(P[P1].AgeOf, P[P1].DoB)

```

^I

^P[P1].Name is under 16, so you MUST recode ^himher as the child (relationship code 3) of an adult: in order of priority, the person receiving Child Benefit for ^himher, or the legal guardian, or whoever is responsible for ^himher.

(NB. If no-one in the household is over age 15, the household is ineligible - outcome code ^Out772).

```

CHECK IF: HHSize > 0
  AND: P[FHHSIZE].Sex = RESPONSE
  AND: In loop FOR P1 := 1 TO FHHSIZE
  RESERVECHECK

```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
AND: P[FHHSIZE].Sex = RESPONSE
AND: In loop FOR P1 := 1 TO FHHSIZE
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: HHSize > 0
RESERVECHECK
```

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSize > 0
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

CHECK IF: HHSIZE > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSIZE > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSIZE > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHSIZE > 0
RESERVECHECK

RESERVECHECK

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: (HHG.P[Loop1].MS IN [Married, CPart]) AND (HHG.P[Loop1].Spouses = 0)
AND: SpOut[Loop1] = Amend
HHG.P[Loop1].Sex <> RESPONSE

^I

Press <Enter> to return to the household grid.

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: (HHG.P[Loop1].MS IN [Married, CPart]) AND (HHG.P[Loop1].Spouses = 0)
RESERVECHECK

RESERVECHECK

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: (HHG.P[Loop1].MS IN [Married, CPart]) AND (HHG.P[Loop1].Spouses = 0)
RESERVECHECK

RESERVECHECK

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: (HHG.P[Loop1].MS IN [Married, CPart]) AND (HHG.P[Loop1].Spouses = 0)
RESERVECHECK

RESERVECHECK

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: (HHG.P[Loop1].MS IN [Married, CPart]) AND (HHG.P[Loop1].Spouses = 0)
RESERVECHECK

```

```
RESERVECHECK
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
AllNameNo := ''
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
AdNameNo := ''
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
AllAd := 0
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
AllCh := 0
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
ChUnder1 := 0
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
Over75 := 0
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
Over60 := 0
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
Under19 := 0
```

```
COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
```

```
WorkingAge := 0
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE

```

```

AllNameNo := (AllNameNo + '
' + STR(Loop1,2,0) + '. ' + QNames.M[Loop1].Name)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge IN [16 .. 120]

```

```

AdNameNo := (AdNameNo + '
' + STR(Loop1,2,0) + '. ' + QNames.M[Loop1].Name)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge IN [16 .. 120]

```

```

OneHRP := Loop1

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge IN [16 .. 120]

```

```

AllAd := (AllAd + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: NOT (HHG.P[Loop1].DVAge IN [16 .. 120])

```

```

AllCh := (AllCh + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge IN [0 .. 1]

```

```

ChUnder1 := (ChUnder1 + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge >= 75

```

```

Over75 := (Over75 + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge >= 60

```

```

Over60 := (Over60 + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].DVAge IN [0 .. 18]

```

```

Under19 := (Under19 + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMAge[Loop1] := HHG.P[Loop1].DVAge

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMDob[Loop1] := HHG.P[Loop1].DoB

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMTEA[Loop1] := HHG.P[Loop1].TEA

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMParent1[Loop1] := HHG.P[Loop1].Parent1

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMParent2[Loop1] := HHG.P[Loop1].Parent2

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMNumParn[Loop1] := HHG.P[Loop1].NumParn

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

DMSex[Loop1] := HHG.P[Loop1].Sex

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

PRec[Loop1].Sex := HHG.P[Loop1].Sex

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

PRec[Loop1].MS := HHG.P[Loop1].MS

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

PRec[Loop1].W1 := HHG.P[Loop1].W1

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

PRec[Loop1].W2 := HHG.P[Loop1].W2

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE

```

```

PRec[Loop1].FtEd := HHG.P[Loop1].FTED

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE

```

```

PREC[Loop1].TypeEd := HHG.P[Loop1].TypeEd

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: HHG.P[Loop1].Depend IN [DepAd .. Child]

```

```

NCDVCP := (NCDVCP + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: (HHG.P[Loop1].Depend IN [DepAd .. Child]) AND ((HHG.P[Loop1].NumPart > 0) OR (LegGuard[Loop1] = 2))

```

```

PREC[Loop1].Depend := Adult

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: NOT ((HHG.P[Loop1].Depend IN [DepAd .. Child]) AND ((HHG.P[Loop1].NumPart > 0) OR (LegGuard[Loop1] = 2)))

```

```

PREC[Loop1].Depend := HHG.P[Loop1].Depend

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: DMSex[Loop1] = Male
  AND: DMAge[Loop1] > 64

```

```

PENFLAG[Loop1] := Yes

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: DMSex[Loop1] = Male
  AND: NOT (DMAge[Loop1] > 64)

```

```

PENFLAG[Loop1] := No

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: DMSex[Loop1] = Female
  AND: DMDob[Loop1] < TODATE (1950, 4, 1)

```

```

PENFLAG[Loop1] := Yes

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: DMSex[Loop1] = Female
  AND: (DMDob[Loop1] = TODATE (1950, 4, 1)) AND (QSignIn.StartDat >= TODATE (2010, 4, 1))

```

```

PENFLAG[Loop1] := Yes

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] = TODATE (1950, 4, 2)) AND (QSignIn.StartDat >= TODATE
(2010, 4, 2))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] = TODATE (1950, 4, 3)) AND (QSignIn.StartDat >= TODATE
(2010, 4, 3))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] = TODATE (1950, 4, 4)) AND (QSignIn.StartDat >= TODATE
(2010, 4, 4))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] = TODATE (1950, 4, 5)) AND (QSignIn.StartDat >= TODATE
(2010, 4, 5))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 5, 5)) AND (QSignIn.StartDat >= TODATE
(2010, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 6, 5)) AND (QSignIn.StartDat >= TODATE
(2010, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 7, 5)) AND (QSignIn.StartDat >= TODATE
(2010, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 8, 5)) AND (QSignIn.StartDat >= TODATE
(2010, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 9, 5)) AND (QSignIn.StartDat >= TODATE
(2011, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 10, 5)) AND (QSignIn.StartDat >= TODATE
(2011, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 11, 5)) AND (QSignIn.StartDat >= TODATE
(2011, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1950, 12, 5)) AND (QSignIn.StartDat >= TODATE
(2011, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1951, 1, 5)) AND (QSignIn.StartDat >= TODATE
(2011, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1951, 2, 5)) AND (QSignIn.StartDat >= TODATE
(2011, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 3, 5)) AND (QSignIn.StartDat >= TODATE
(2012, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 4, 5)) AND (QSignIn.StartDat >= TODATE
(2012, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 5, 5)) AND (QSignIn.StartDat >= TODATE
(2012, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 6, 5)) AND (QSignIn.StartDat >= TODATE
(2012, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 7, 5)) AND (QSignIn.StartDat >= TODATE
(2012, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 8, 5)) AND (QSignIn.StartDat >= TODATE
(2012, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDOB[Loop1] <= TODATE (1951, 9, 5)) AND (QSignIn.StartDat >= TODATE
(2013, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1951, 10, 5)) AND (QSignIn.StartDat >= TODATE
(2013, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1951, 11, 5)) AND (QSignIn.StartDat >= TODATE
(2013, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1951, 12, 5)) AND (QSignIn.StartDat >= TODATE
(2013, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1952, 1, 5)) AND (QSignIn.StartDat >= TODATE
(2013, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1952, 2, 5)) AND (QSignIn.StartDat >= TODATE
(2013, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1952, 3, 5)) AND (QSignIn.StartDat >= TODATE
(2014, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1952, 4, 5)) AND (QSignIn.StartDat >= TODATE
(2014, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 5, 5)) AND (QSignIn.StartDat >= TODATE
(2014, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 6, 5)) AND (QSignIn.StartDat >= TODATE
(2014, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 7, 5)) AND (QSignIn.StartDat >= TODATE
(2014, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 8, 5)) AND (QSignIn.StartDat >= TODATE
(2014, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 9, 5)) AND (QSignIn.StartDat >= TODATE
(2015, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 10, 5)) AND (QSignIn.StartDat >= TODATE
(2015, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 11, 5)) AND (QSignIn.StartDat >= TODATE
(2015, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1952, 12, 5)) AND (QSignIn.StartDat >= TODATE
(2015, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1953, 1, 5)) AND (QSignIn.StartDat >= TODATE
(2015, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1953, 2, 5)) AND (QSignIn.StartDat >= TODATE
(2015, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1953, 3, 5)) AND (QSignIn.StartDat >= TODATE
(2016, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1953, 4, 5)) AND (QSignIn.StartDat >= TODATE
(2016, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1953, 5, 5)) AND (QSignIn.StartDat >= TODATE
(2016, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDoB[Loop1] <= TODATE (1953, 6, 5)) AND (QSignIn.StartDat >= TODATE
(2016, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1953, 7, 5)) AND (QSignIn.StartDat >= TODATE
(2016, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1953, 8, 5)) AND (QSignIn.StartDat >= TODATE
(2016, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1953, 9, 5)) AND (QSignIn.StartDat >= TODATE
(2017, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1953, 10, 5)) AND (QSignIn.StartDat >= TODATE
(2017, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1953, 11, 5)) AND (QSignIn.StartDat >= TODATE
(2017, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1953, 12, 5)) AND (QSignIn.StartDat >= TODATE
(2017, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 1, 5)) AND (QSignIn.StartDat >= TODATE
(2017, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 2, 5)) AND (QSignIn.StartDat >= TODATE
(2017, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 3, 5)) AND (QSignIn.StartDat >= TODATE
(2018, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 4, 5)) AND (QSignIn.StartDat >= TODATE
(2018, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 5, 5)) AND (QSignIn.StartDat >= TODATE
(2018, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 6, 5)) AND (QSignIn.StartDat >= TODATE
(2018, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 7, 5)) AND (QSignIn.StartDat >= TODATE
(2018, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 8, 5)) AND (QSignIn.StartDat >= TODATE
(2018, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 9, 5)) AND (QSignIn.StartDat >= TODATE
(2019, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 10, 5)) AND (QSignIn.StartDat >= TODATE
(2019, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 11, 5)) AND (QSignIn.StartDat >= TODATE
(2019, 5, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1954, 12, 5)) AND (QSignIn.StartDat >= TODATE
(2019, 7, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1955, 1, 5)) AND (QSignIn.StartDat >= TODATE
(2019, 9, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1955, 2, 5)) AND (QSignIn.StartDat >= TODATE
(2019, 11, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1955, 3, 5)) AND (QSignIn.StartDat >= TODATE
(2020, 1, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: (DMDob[Loop1] <= TODATE (1955, 4, 5)) AND (QSignIn.StartDat >= TODATE
(2020, 3, 6))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: ((DMDob[Loop1] = DONTKNOW) AND (DMSex[Loop1] = 60)) AND
(QSignIn.StartDat < TODATE (2012, 4, 1))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: ((DMDob[Loop1] = DONTKNOW) AND (DMSex[Loop1] = 61)) AND
(QSignIn.StartDat >= TODATE (2012, 4, 1)) AND (QSignIn.StartDat <= TODATE
(2014, 3, 31))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: ((DMDob[Loop1] = DONTKNOW) AND (DMSex[Loop1] = 62)) AND
(QSignIn.StartDat >= TODATE (2014, 4, 1)) AND (QSignIn.StartDat <= TODATE
(2016, 3, 31))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: ((DMDob[Loop1] = DONTKNOW) AND (DMSex[Loop1] = 63)) AND
(QSignIn.StartDat >= TODATE (2016, 4, 1)) AND (QSignIn.StartDat <= TODATE
(2018, 3, 31))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: ((DMDob[Loop1] = DONTKNOW) AND (DMSex[Loop1] = 64)) AND
(QSignIn.StartDat >= TODATE (2018, 4, 1)) AND (QSignIn.StartDat <= TODATE
(2020, 3, 31))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].Sex = RESPONSE
AND: DMSex[Loop1] = Female
AND: ((DMDob[Loop1] = DONTKNOW) AND (DMSex[Loop1] = 65)) AND
(QSignIn.StartDat >= TODATE (2020, 4, 1))

```

PenFlag[Loop1] := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: DMSex[Loop1] = Female
  AND: (QSignIn.StartDat >= TODATE (2020, 5, 6)) AND (DMAge[Loop1] > 64)

```

```

PenFlag[Loop1] := Yes

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: DMSex[Loop1] = Female
  AND: NOT ((QSignIn.StartDat >= TODATE (2020, 5, 6)) AND (DMAge[Loop1] > 64))

```

```

PenFlag[Loop1] := No

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: (DMAge[Loop1] >= 16) AND (PenFlag[Loop1] <> Yes)

```

```

WorkingAge := (WorkingAge + 1)

```

```

WARN IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: (HHG.P[Loop1].CupChk = Yes) AND HHG.P[HHSIZE].QRel[HHSIZE].R <> EMPTY
  (HHG.P[Loop1].NumCohab > 0) AND
  INVOLVING(HHG.P[HHSIZE].QRel[HHSIZE].R, HHG.P[Loop1].CupChk)

```

^I(^HHG.P[Loop1].Name)^I^N Can I just check, you said earlier that you were living with someone in this household as a couple, is this correct?^N

^I

If Yes: go back and correct the code at Relationship.

If No: go back and correct CupChk to No (2).

```

WARN IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: HHG.P[Loop1].Sex = RESPONSE
  AND: (HHG.P[Loop1].CupChk = No) AND HHG.P[HHSIZE].QRel[HHSIZE].R <> EMPTY
  (HHG.P[Loop1].NumCohab = 0) AND
  INVOLVING(HHG.P[HHSIZE].QRel[HHSIZE].R, HHG.P[Loop1].CupChk)

```

^I(^HHG.P[Loop1].Name)^I^N Can I just check, are you living with anyone in the household as a couple?^N

^I

If Yes: go back and correct CupChk to Yes (1).

If No: go back and correct the code at Relationship.

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: AllAd = 1

```

```

you := 'you'

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: NOT (AllAd = 1)

```

```

you := 'ANY of you'

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
  AND: (AllAd + AllCh) = 1

```

```

any_of_you := 'you'

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
 AND: NOT ((ALLAD + ALLCH) = 1)

any_of_you := 'ANY of you'

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

RentName := ''

FRS1004B.QHholder

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[1] := ' 1. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[2] := ' 2. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[3] := ' 3. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[4] := ' 4. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[5] := ' 5. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[6] := ' 6. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[7] := ' 7. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[8] := ' 8. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[9] := ' 9. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[10] := '10. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[11] := '11. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[12] := '12. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[13] := '13. '

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE

LPAD[14] := '14. '

ASK IF: HHG.P[HHSIZE].DVAge = RESPONSE

HHldr

^N
 In whose name is the accommodation owned or rented?
 ^Anyone_else?^N
 ^I
 Code all that apply.

SET [15] OF

- (1) ^DMName[1]
- (2) ^DMName[2]
- (3) ^DMName[3]
- (4) ^DMName[4]
- (5) ^DMName[5]
- (6) ^DMName[6]
- (7) ^DMName[7]
- (8) ^DMName[8]
- (9) ^DMName[9]
- (10) ^DMName[10]
- (11) ^DMName[11]
- (12) ^DMName[12]
- (13) ^DMName[13]
- (14) ^DMName[14]
- (97) Not a household member

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE

HhldList := ''

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE

HhldNum := 0

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: NotHH IN HHldr

HhldCard := (HHldr.CARDINAL - 1)

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: NOT (NotHH IN HHldr)

HhldCard := HHldr.CARDINAL

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop := 1 TO 14

HhldName[Loop] := ''

FRS1004B.QHholder.PadString()

Procedure Call

```
COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE  
  AND: In loop FOR Loop := 1 TO 14  
  AND: Loop IN HHldr
```

OutString := InString

```
COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE  
  AND: In loop FOR Loop := 1 TO 14  
  AND: Loop IN HHldr  
  AND: ResLength > LEN (OutString)
```

Fin := (ResLength - LENGTH(OutString))

```
COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE  
  AND: In loop FOR Loop := 1 TO 14  
  AND: Loop IN HHldr  
  AND: ResLength > LEN (OutString)  
  AND: In loop FOR PLoop := 1 TO Fin
```

OutString := (OutString + '·')

FRS1004B.QHholder (continued)

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN HHldr

```

```

HhldName[Loop] := ('
' + LPad[Loop] + PadName + '...@| (Age: ' + STR(DMAGE[[Loop], 3, 0)
+ ') ')

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN HHldr

```

```

HhldNum := (HhldNum + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN HHldr
  AND: HhldNum = 1

```

```

HhldList := DMNAME[[Loop]]

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN HHldr
  AND: HhldNum < HhldCard

```

```

HhldList := (HhldList + ', ' + DMNAME[[Loop]])

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN HHldr
  AND: HhldNum = HhldCard

```

```

HhldList := (HhldList + ' and ' + DMNAME[[Loop]])

```

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  RESERVECHECK

```

```

  RESERVECHECK

```

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  RESERVECHECK

```

```

  RESERVECHECK

```

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  RESERVECHECK

```

```

  RESERVECHECK

```

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  RESERVECHECK

```

```

  RESERVECHECK

```

WARN IF: HHG.P[HHSize].DVAge = RESPONSE
RESERVECHECK

RESERVECHECK

ASK IF: HHG.P[HHSize].DVAge = RESPONSE
AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)

WhoResp

^N

Although you have mentioned that the rent or mortgage for this accommodation is paid for by someone outside the household, there needs to be someone within the household who is responsible for the property. Who then in this household is responsible for this accommodation?

Anyone else?^N

^I

Code all that apply.

SET [14] OF

- (1) ^DMName[1]
 - (2) ^DMName[2]
 - (3) ^DMName[3]
 - (4) ^DMName[4]
 - (5) ^DMName[5]
 - (6) ^DMName[6]
 - (7) ^DMName[7]
 - (8) ^DMName[8]
 - (9) ^DMName[9]
 - (10) ^DMName[10]
 - (11) ^DMName[11]
 - (12) ^DMName[12]
 - (13) ^DMName[13]
 - (14) ^DMName[14]
-

COMPUTE IF: HHG.P[HHSize].DVAge = RESPONSE
AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)

HhldList := ''

COMPUTE IF: HHG.P[HHSize].DVAge = RESPONSE
AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)

HhldNum := 0

COMPUTE IF: HHG.P[HHSize].DVAge = RESPONSE
AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)

HhldCard := WhoResp.CARDINAL

COMPUTE IF: HHG.P[HHSize].DVAge = RESPONSE
AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
AND: In loop FOR Loop := 1 TO 14

HhldName [Loop] := ''

FRS1004B.QHholder.PadString()

Procedure Call

```
COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE  
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)  
  AND: In loop FOR Loop := 1 TO 14  
  AND: Loop IN WhoResp
```

OutString := InString

```
COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE  
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)  
  AND: In loop FOR Loop := 1 TO 14  
  AND: Loop IN WhoResp  
  AND: ResLength > LEN (OutString)
```

Fin := (ResLength - LENGTH(OutString))

```
COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE  
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)  
  AND: In loop FOR Loop := 1 TO 14  
  AND: Loop IN WhoResp  
  AND: ResLength > LEN (OutString)  
  AND: In loop FOR PLoop := 1 TO Fin
```

OutString := (OutString + '·')

FRS1004B.QHholder (continued)

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN WhoResp

HhldName[Loop] := (
' + LPad[Loop] + PadName + '...@| (Age: ' + STR(DMAge[[Loop],3,0)
+ ') ')

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN WhoResp

```

```

HhldNum := (HhldNum + 1)

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN WhoResp
  AND: HhldNum = 1

```

```

HhldList := DMName[[Loop]]

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN WhoResp
  AND: HhldNum < HhldCard

```

```

HhldList := (HhldList + ', ' + DMName[[Loop]])

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
  AND: In loop FOR Loop := 1 TO 14
  AND: Loop IN WhoResp
  AND: HhldNum = HhldCard

```

```

HhldList := (HhldList + ' and ' + DMName[[Loop]])

```

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
  AND: (HHldr.CARDINAL = 1) AND (NotHH IN HHldr)
  RESERVECHECK

```

```

  RESERVECHECK

```

```

RECORD IF: HHG.P[HHSIZE].DVAGE = RESPONSE

```

HRPPrtnr

```

  ^N
  Person number of HRP's spouse/partner.

```

```

  1..15

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (HHldr <> EMPTY AND (AllAd = 1)) AND (OneHRP = RESPONSE)

```

HRPPrtnr := 15

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
RESERVECHECK

```

RESERVECHECK

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: WhoResp.CARDINAL > 1

```

own := 'are responsible for'

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: NOT (WhoResp.CARDINAL > 1)

```

own := 'own or rent'

```

ASK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)

```

HiHNum

^I

If respondent asks for period to average over - one year.

Prompt as necessary for joint householders:

Is one of them the sole person with paid work or occupational pension?

1..15

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: WhoResp = EMPTY
AND: HiHNum IN [1 .. 14]
(IN(HiHNum, HHldr)) AND INVOLVING(HiHNum)

^I

This person is not recorded as a householder (at HHldr).

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: NOT (WhoResp = EMPTY)
AND: HiHNum IN [1 .. 14]
(IN(HiHNum, WhoResp)) AND INVOLVING(HiHNum)

^I

This person is not recorded as responsible for the household (at WhoResp).

ASK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = 15

JntEldA

^I

Ask or record.

Enter Person Number of the^B eldest^B joint householder from those with the same highest income.

^HhldName[1]^HhldName[2]^HhldName[3]^HhldName[4]^HhldName[5]
 ^HhldName[6]^HhldName[7]^HhldName[8]^HhldName[9]^HhldName[10]
 ^HhldName[11]^HhldName[12]^HhldName[13]^HhldName[14]

0..14

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = 15
AND: JntEldA = RESPONSE
JntEldA <> 0

^I

Zero (0) is not a valid code.

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL > 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = 15
AND: JntEldA IN [1 .. 14]
AND: WhoResp = EMPTY
(IN(JntEldA, HHldr)) AND INVOLVING(HiHNum)

^I

This person is not recorded as a householder (at HHldr).

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = 15
AND: JntEldA IN [1 .. 14]
AND: NOT (WhoResp = EMPTY)
(IN(JntEldA,WhoResp)) AND INVOLVING(HiHNum)

```

^I

This person is not recorded as responsible for the household (at WhoResp).

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = 15
AND: JntEldA IN [1 .. 14]

```

DVHRPNum := JntEldA

```

ASK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = NONRESPONSE

```

JntEldB

^I

Ask or record.

Enter Person Number of the eldest joint householder.

^HhldName[1]^HhldName[2]^HhldName[3]^HhldName[4]^HhldName[5]
 ^HhldName[6]^HhldName[7]^HhldName[8]^HhldName[9]^HhldName[10]
 ^HhldName[11]^HhldName[12]^HhldName[13]^HhldName[14]

0..14

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = NONRESPONSE
AND: JntEldB = RESPONSE
JntEldB <> 0

```

^I

Zero (0) is not a valid code.

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = NONRESPONSE
AND: JntEldB IN [1 .. 14]
AND: WhoResp = EMPTY
(IN(JntEldB,HHldr)) AND INVOLVING(HiHNum)

```

^I

This person is not recorded as a householder (at HHldr).

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = NONRESPONSE
AND: JntEldB IN [1 .. 14]
AND: NOT (WhoResp = EMPTY)
      (IN(JntEldB,WhoResp)) AND INVOLVING(HiHNum)

```

^I

This person is not recorded as responsible for the household (at WhoResp).

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum = NONRESPONSE
AND: JntEldB IN [1 .. 14]

```

DVHRPNum := JntEldB

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (((HHldr.CARDINAL = 2) AND NOT (NotHH IN HHldr)) OR (HHldr.CARDINAL
> 2)) OR (WhoResp.CARDINAL > 1)
AND: HiHNum IN [1 .. 14]

```

DVHRPNum := HiHNum

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (HHldr.CARDINAL = 1) AND NOT (NotHH IN HHldr)

```

DVHRPNum := ORD(HHldr[1])

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (HHldr.CARDINAL = 2) AND (NotHH IN HHldr)
AND: HHldr[1] = NotHH

```

DVHRPNum := ORD(HHldr[2])

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: (HHldr.CARDINAL = 2) AND (NotHH IN HHldr)
AND: NOT (HHldr[1] = NotHH)

```

DVHRPNum := ORD(HHldr[1])

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: WhoResp.CARDINAL = 1

```

DVHRPNum := ORD(WhoResp[1])

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: NOT (WhoResp.CARDINAL = 1)

```

DVHRPNum := 0

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: DVHRPNum IN [1 .. 14]

```

LName := DMName [DVHRPNum]

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: NOT (DVHRPNum IN [1 .. 14])

```

LName := 'Non-HH Member'

ASK IF: *HHG.P[HHSize].DVAge = RESPONSE*

HRP

^I

The Household Reference Person is:

(^DVHRPNum) ^LName

Press 1 and <Enter> to continue.

(1) Continue

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
AND: Loop1 IN QHholder.HHldr
PRec[Loop1].Sex = RESPONSE

```

^I

Code ^Loop1 is not valid for this question.

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
AND: Loop1 IN QHholder.HHldr
PRec[Loop1].Depend = Adult

```

^I

Person ^Loop1 is a child or a dependent adult. Please amend.

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
AND: Loop1 IN QHholder.HHldr

```

HHG.P[Loop1].Hholder := Yes

```

COMPUTE IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
AND: NOT (Loop1 IN QHholder.HHldr)

```

HHG.P[Loop1].Hholder := No

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
AND: Loop1 IN QHholder.WhoResp
PRec[Loop1].Sex = RESPONSE

```

^I

This figure exceeds the number of household members. Please check and amend your answer.

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
AND: Loop1 IN QHholder.WhoResp
PRec[Loop1].Depend = Adult

```

^I

The person responsible for the property must be an adult household member. Please check and amend your answer.

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
RESERVECHECK

```

RESERVECHECK

```

CHECK IF: HHG.P[HHSIZE].DVAge = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
RESERVECHECK

```

RESERVECHECK

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
RESERVECHECK

```

```
RESERVECHECK
```

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: In loop FOR Loop1 := 1 TO 14
RESERVECHECK

```

```
RESERVECHECK
```

```

CHECK IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
PREC[QHholder.DVHRPNum].DEPEND = Adult

```

```
^I
```

```
Code ^QHholder.DVHRPNum is not valid for this question.
```

```

WARN IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
NOT((PREC[QHholder.DVHRPNum].SEX = Female) AND
(PREC[QHholder.DVHRPNum].MS = Married))

```

```
^I
```

```
For a married couple the man is always Head of household.
```

```
Please amend your coding. (But if he is away for more than 6 months, suppress check and move on.)
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE

```

```
HRPNames := DMName[QHholder.DVHRPNum]
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE

```

```
PREL.PR[Loop1].R := HHG.P[Loop1].QREL[QHholder.DVHRPNum].R
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].QREL[QHholder.DVHRPNum].R IN [Spouse, Cohabit, CivilP]

```

```
HRPNames := (HRPNames + ' and ' + DMName[Loop1])
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: HHG.P[Loop1].QREL[QHholder.DVHRPNum].R IN [Spouse, Cohabit, CivilP]

```

```
QHholder.HRPPrtNr := Loop1
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: DMAge[Loop1] >= 16
AND: Loop1 = QHholder.DVHRPNum

```

```
QHholder.QPerId[Loop1].HOHID := HOH
```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Dmage[Loop1] >= 16
AND: Loop1 = QHholder.DVHRPNum

```

```

QHholder.QPerId[Loop1].HRPID := HRP

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Dmage[Loop1] >= 16
AND: Loop1 = QHholder.DVHRPNum

```

```

QHholder.QPerId[Loop1].CombID := HOHHRP

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Dmage[Loop1] >= 16
AND: NOT (Loop1 = QHholder.DVHRPNum)

```

```

QHholder.QPerId[Loop1].HOHID := NotHOH

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Dmage[Loop1] >= 16
AND: NOT (Loop1 = QHholder.DVHRPNum)

```

```

QHholder.QPerId[Loop1].HRPID := NotHRP

```

```

COMPUTE IF: HHG.P[HHSIZE].DVAGE = RESPONSE
AND: QHholder.DVHRPNum = RESPONSE
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Dmage[Loop1] >= 16
AND: NOT (Loop1 = QHholder.DVHRPNum)

```

```

QHholder.QPerId[Loop1].CombID := HOHonly

```

FRS1004B.QMoveOut

QUESTIONNAIRE MIGRATION - NI ONLY

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI

tdate := SYSDATE

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI

yagodate := (SYSDATE + (-1,-0,-0))

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI

MigrQ1

Is there anyone who usually lived in your household on ^yagodate, who is now living elsewhere?

INTERVIEWER: INCLUDE ALL PERSONS WHO HAVE LEFT THE HOUSEHOLD.
 SPECIFICALLY INCLUDE:

@(i)STUDENTS WHO HAVE MOVED TO UNIVERSITY/COLLEGE HALLS OF RESIDENCE OR
 DIGS

@(ii)PEOPLE WHO HAVE MOVED TO FIND WORK ELSEWHERE

@(iii)FOREIGN MIGRANT WORKERS WHO HAVE RETURNED HOME

@(iv)PEOPLE WHO HAD SUMMER JOBS

@(v)PEOPLE WHO HAVE EMIGRATED

@(vi)PEOPLE ON A GAP-YEAR OVERSEAS

@(vii)YOUNG ADULTS WHO HAVE MOVED OUT OF THE FAMILY HOME TO BE WITH A
 PARTNER

@(viii)ADULTS/CHILDREN AFFECTED BY DIVORCE/RELATIONSHIP BREAK-UP, ETC

@(ix)ELDERLY PEOPLE WHO HAVE MOVED INTO A NURSING HOME, ETC

(1) Yes

(2) No

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 <> RESPONSE

Time1 := SYSTIME

RECORD IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI

Time1

Migration Start

TIME

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
      AND: NatCen = NI
      AND: MigrQ1 = Yes

```

MigrQ2

How many people have left your household in the last year?

INTERVIEWER: ENTER THE NUMBER OF PEOPLE WHO HAVE NOW LEFT THE HOUSEHOLD BUT WERE HOUSEHOLD MEMBERS ON ^yagodate.

1..10

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
      AND: NatCen = NI
      AND: MigrQ1 = Yes
      AND: MigrQ2 = 1

```

MigTxt := 'person who has left your household'

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
      AND: NatCen = NI
      AND: MigrQ1 = Yes
      AND: MigrQ2 > 1

```

MigTxt := 'people who have left your household'

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
      AND: NatCen = NI
      AND: MigrQ1 = Yes

```

Intro2

I would now like to ask a couple of questions about the ^MigTxt in the last year.

This information will be used to help improve population estimates for Northern Ireland in the years that we don't have a census.

(1) Continue

FRS1004B.QMoveOut.QPerson[]

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2

MigrQ3

Is person ^k ...

- (1) Male
- (2) or Female?

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2

MigrQ4

What age was person ^k on their last birthday?

INTERVIEWER: ENTER 0 FOR BABIES UNDER 1 YEAR OLD.

0..99

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2

MigrQ5

Where is person ^k living now?

- (1) Somewhere else in Northern Ireland
- (2) England
- (3) Republic of Ireland
- (4) Scotland
- (5) Wales
- (6) Elsewhere in the world

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2
AND: MigrQ5 = Elsewh

MigrQ6

Which country has person ^k moved to?

INTERVIEWER: ENTER THE NAME OF A COUNTRY ONLY.
IF PERSON IS TRAVELLING AROUND (E.G. ON A GAP YEAR OR A CAREER BREAK) JUST
ENTER

STRING[255]

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2

MigrQ7a

In what month did person ^k leave?

- (1) January
- (2) February
- (3) March
- (4) April
- (5) May
- (6) June
- (7) July
- (8) August
- (9) September
- (10) October
- (11) November
- (12) December

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2

MigrQ7b

In which year did person ^k leave?

- (1) 2009
- (2) 2010
- (3) 2011

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2

MigrQ8

Is person ^k likely to have left your household for more than one year?

INTERVIEWER: ANSWER

- (1) Yes
- (2) No

FRS1004B.QMoveOut (continued)**QUESTIONNAIRE MIGRATION - NI ONLY**

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
AND: MigrQ1 = Yes
AND: In loop FOR k := 1 TO MigrQ2
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: NatCen = NI
RESERVECHECK

RESERVECHECK

FRS1004B.QEthnic

Ethnic data on adults in household

```
COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSize
  AND: PRec[Personnr].Depend = Adult
```

```
P[Personnr].PersId := Personnr
```

```
COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSize
  AND: PRec[Personnr].Depend = Adult
```

```
P[Personnr].BenUnit := DMBU[[Personnr]
```

```
COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSize
  AND: PRec[Personnr].Depend = Adult
```

```
P[Personnr].EName := DMName[[Personnr]
```

FRS1004B.QEthnic.P[]

```

RECORD IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult

```

BenUnit

^I QEthnic^I

Benefit Unit number.

1..7

```

RECORD IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult

```

PersId

^I QEthnic^I

Person identifier.

0..14

```

DISPLAY IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult

```

EName

^I QEthnic^I

STRING[15]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult

```

LName := EName

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult
  AND: PRec[PersId].Sex = Male

```

he_she := 'he'

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult
  AND: NOT (PRec[PersId].Sex = Male)

```

he_she := 'she'

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
    AND: In loop FOR Personnr := 1 TO HHSize
    AND: PRec[Personnr].Depend = Adult
    AND: NatCen = NI

```

```

this_country := 'the UK'

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
    AND: In loop FOR Personnr := 1 TO HHSize
    AND: PRec[Personnr].Depend = Adult
    AND: NOT (NatCen = NI)

```

```

this_country := 'this country'

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
    AND: In loop FOR Personnr := 1 TO HHSize
    AND: PRec[Personnr].Depend = Adult
    AND: NatCen = NI

```

```

the_uk := ' '

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
    AND: In loop FOR Personnr := 1 TO HHSize
    AND: PRec[Personnr].Depend = Adult
    AND: NOT (NatCen = NI)

```

```

the_uk := ' 'This country' refers to the UK.'

```

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
    AND: In loop FOR Personnr := 1 TO HHSize
    AND: PRec[Personnr].Depend = Adult

```

COrign

^I QEthnic^I

^I

(^LName)^I^N In which country were you born?

- (1) England
- (2) Wales
- (3) Scotland
- (4) Northern Ireland
- (5) UK, Britain
- (6) Republic of Ireland
- (7) Hong Kong
- (8) China
- (9) Other

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]

```

CameYr

^I QEthnic^I

^I

(^LName)^I^N In which year did you first arrive in ^this_country?^N

^I^IC Enter in 4 digit format e.g. 2000.

If respondent unsure probe if they can remember their age, the season or similar that may help recall year of arrival.

^the_UK

1900..2097

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: CameYr = RESPONSE
CameYr <= LYear

```

^I

You've entered a future date!

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: QDataBag.SampMnth IN [4 .. 12]
CameYr <> LYear

```

^I

You've entered a future date!

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]

```

ContUK

^I QEthnic^I

^N

Apart from holidays and short visits ^LName have you lived in the UK continuously since then?

- (1) Yes
- (2) No

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSIZE
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: ContUK = No

```

CameYr2

^I QEthnic^I

^N

Which year did ^LName last arrive in ^this_country?

^I^IC If respondent unsure probe if they can remember their age, the season or similar that may help recall year of arrival.

^the_UK

1900..2097

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSIZE
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: ContUK = No
AND: CameYr2 = RESPONSE
CameYr2 <= LYear

```

^I

You've entered a future date!

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSIZE
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: ContUK = No
AND: QDataBag.SampMnth IN [4 .. 12]
CameYr2 <> LYear

```

^I

You've entered a future date!

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: (CameYr = RESPONSE) OR (CameYr2 = RESPONSE)

```

CameMt

^I QEthnic^I

^I

(^LName)^I^N And which month was that?

- (1) January
 - (2) February
 - (3) March
 - (4) April
 - (5) May
 - (6) June
 - (7) July
 - (8) August
 - (9) September
 - (10) October
 - (11) November
 - (12) December
-

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: (CameYr = RESPONSE) OR (CameYr2 = RESPONSE)
AND: CameYr2 = RESPONSE
AND: CameMt = RESPONSE

```

```

BeenHere := (QSignIn.StartDat.JULIAN -
JULIAN(TODATE(CameYr2,ORD(CameMt),1)))

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: (CameYr = RESPONSE) OR (CameYr2 = RESPONSE)
AND: CameYr2 = RESPONSE
AND: NOT (CameMt = RESPONSE)

```

```

BeenHere := (QSignIn.StartDat.JULIAN -
JULIAN(TODATE(CameYr2,1,1)))

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: COrign IN [ROI, HongKong, China, Other]
AND: (CameYr = RESPONSE) OR (CameYr2 = RESPONSE)
AND: CameYr = RESPONSE
AND: CameMt = RESPONSE

```

```

BeenHere := (QSignIn.StartDat.JULIAN -
JULIAN(TODATE(CameYr,ORD(CameMt),1)))

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult
  AND: COrign IN [ROI, HongKong, China, Other]
  AND: (CameYr = RESPONSE) OR (CameYr2 = RESPONSE)
  AND: CameYr = RESPONSE
  AND: NOT (CameMt = RESPONSE)

```

```

BeenHere := (QSignIn.StartDat.JULIAN -
JULIAN(TODATE(CameYr,1,1)))

```

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult
  AND: NatCen <> NI

```

NatID

^I QEthnic^I

^I^IS A2^I^X

^I(^LName)^I^N What do you consider your^B national identity^B to be? Please choose your answer from this card, choose as many or as few as apply.^N

^I

Probe: ^I^N^Any_other?

SET [6] OF

- (1) English
 - (2) Scottish
 - (3) Welsh
 - (4) Irish
 - (5) British
 - (6) Other answer
-

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Personnr := 1 TO HHSIZE
  AND: PRec[Personnr].Depend = Adult
  AND: NatCen <> NI
  AND: Other IN NatID

```

NatOth

^I QEthnic^I

^I

(^LName)^I^N How would you describe your national identity?^N

^I^IC If someone describes themselves as being half English and half Irish or any combination of Welsh, Scottish, Irish or English, code them as 'Mixed British' and then record the mix they specify.

- (1) Mixed British
- (2) ENTER DESCRIPTION OF ETHNIC GROUP

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NatCen <> NI
AND: Other IN NatID

```

XNatOth

^I QEthnic^I

^I

(^LName) Enter description of ethnic group.

STRING[100]

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NatCen <> NI

```

EthGrp

^I QEthnic^I

^I^IS A3^I^X

^N

To which of these ethnic groups does ^LName consider ^he_she belongs?^N

^I^IC This is a question of respondent's (or proxy's) opinion.

- (1) White - British
- (2) White - Irish
- (3) Any other white background (please describe)
- (4) Mixed - White and Black Caribbean
- (5) Mixed - White and Black African
- (6) Mixed - White and Asian
- (7) Any other mixed background (please describe)
- (8) Asian or Asian British - Indian
- (9) Asian or Asian British - Pakistani
- (10) Asian or Asian British - Bangladeshi
- (11) Any other Asian/Asian British background (please describe)
- (12) Black or Black British - Caribbean
- (13) Black or Black British - African
- (14) Any other Black/Black British background (please describe)
- (15) Chinese
- (16) Any other (please describe)

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NatCen <> NI
AND: EthGrp IN [WhtOth, MixedOth, AsianOth, BlackOth, Other]

```

EthOth

```

^I QEthnic^I

^I
(^LName)^I^N Please can you describe your ethnic group?^N
^I
Enter description of ethnic group.

STRING[100]

```

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NOT (NatCen <> NI)

```

NINatID

```

^I QEthnic^I

^I^IS A2^I
^N *^X(^LName) What do you consider your^B national identity^B to be? Please choose your answer
from this card, choose as many or as few as apply.^N

^I Probe: Any other?

SET [8] OF
(1) British
(2) Irish
(3) Ulster
(4) Northern Irish
(5) English
(6) Scottish
(7) Welsh
(8) Other answer

```

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NOT (NatCen <> NI)
AND: Other IN NINatID

```

NINatOth

```

^I QEthnic^I

^I
(^LName)^I^N How would you describe your national identity?^N

^I^IC - If someone describes themselves as being half English and half Irish or any combination of
Welsh, Scottish, Irish or English, code them as 'Mixed British' AND then record the mix they specify.

(1) Mixed British
(2) ENTER DESCRIPTION OF ETHNIC GROUP

```

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NOT (NatCen <> NI)
AND: Other IN NINatID

```

NIXNatOth

^I QEthnic^I

^I

(^LName) Enter description of ethnic GROUP.

STRING[100]

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NOT (NatCen <> NI)

```

NIEthGrp

^I QEthnic^I

^I^IS A3^IS ^N

To which of these ethnic groups does ^LName consider ^he_she belongs?^N

^I^IC This is a question of respondent's (or proxy's) opinion.

- (1) White
- (2) Irish Traveller
- (3) Any other white background (please describe)
- (4) Mixed - White and Black Caribbean
- (5) Mixed - White and Black African
- (6) Mixed - White and Asian
- (7) Any other mixed background (please describe)
- (8) Asian - Indian
- (9) Asian - Pakistani
- (10) Asian - Bangladeshi
- (11) Any other Asian background (please describe)
- (12) Black - Caribbean
- (13) Black - African
- (14) Any other Black background (please describe)
- (15) Chinese
- (16) Any other (please describe)

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
AND: NOT (NatCen <> NI)
AND: NIEthGrp IN [WhtOth, AsianOth, BlackOth, Other]

```

NIEthOth

^I QEthnic^I

^I

(^LName)^I^N Please can you describe your ethnic group?^N

^I Enter description of ethnic group^I.

STRING[100]

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK

```

RESERVECHECK

```
WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK
```

RESERVECHECK

```
WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
AND: PRec[Personnr].Depend = Adult
RESERVECHECK
```

RESERVECHECK

FRS1004B.QEthnic (continued)

Ethnic data on adults in household

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Personnr := 1 TO HHSize
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSize
AND: PRec[Loop1].Depend IN [DepAd .. Child]
AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])

DepParnt := (STR(DMParent1[Loop1],1,0) + ' : ' +
DMName [DMParent1[Loop1]] + '
 ' + STR(DMParent2 [Loop1],1,0) + ' : ' + **DMName** [DMParent2 [Loop1]])

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSize
AND: PRec[Loop1].Depend IN [DepAd .. Child]
AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])
AND: PRec[Loop1].Sex = Male

heshe := 'he'

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSize
AND: PRec[Loop1].Depend IN [DepAd .. Child]
AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])
AND: PRec[Loop1].Sex = Male

hisher := 'his'

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend IN [DepAd .. Child]
  AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
  AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])
  AND: PRec[Loop1].Sex = Female

```

```

heshe := 'she'

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend IN [DepAd .. Child]
  AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
  AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])
  AND: PRec[Loop1].Sex = Female

```

```

hisher := 'her'

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend IN [DepAd .. Child]
  AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
  AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])
  AND: NOT (PRec[Loop1].Sex = Female)

```

```

heshe := 'he/she'

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend IN [DepAd .. Child]
  AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
  AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])
  AND: NOT (PRec[Loop1].Sex = Female)

```

```

hisher := 'his/her'

```

```

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend IN [DepAd .. Child]
  AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
  AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse .. Cohabit, CivilP])

```

LegDep

^M^C

^DMName[Loop1] is classified as a^B dependant adult^B or a^B child^B, ie. ^HeShe will not form a Benefit Unit of ^HisHer own.

To properly assess to which Benefit Unit ^HeShe belongs, please code which of the parents receive Child Benefit for ^DMName[Loop1].

^DepParnt

1..97

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend IN [DepAd .. Child]
  AND: ((DMParent1[Loop1] IN [1 .. 14]) AND (DMParent2[Loop1] IN [1 .. 14]))
  AND NOT (HHG.P[DMParent1[Loop1]].QRel[DMParent2[Loop1]].R IN [Spouse ..
  Cohabit, CivilP])
  AND: LegDep[Loop1] = RESPONSE
  (LegDep[Loop1] = DMParent1[Loop1]) OR (LegDep[Loop1] = DMParent2[Loop1])

^I
Code ^LegDep[Loop1] is not valid for this question.

```

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE

```

ABen[Loop1] := 1

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  RESERVECHECK

```

RESERVECHECK

```

RECORD IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)

```

NewBU

Total number of BUs

0..7

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)

```

ABen[QHholder.DVHRPNum] := 1

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)

```

Last := 1

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [Adult])
  AND: In loop FOR Loop2 := 1 TO HHSIZE
  AND: ABen[Loop2] <> EMPTY AND (HHG.P[Loop1].QRel[Loop2].R IN [Spouse,
    Cohabit, CivilP])

```

ABen[Loop1] := ABen[Loop2]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend = Adult)
  AND: Last < 7

```

Last := (Last + 1)

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend = Adult)
  AND: Last < 7

```

ABen[Loop1] := Last

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend = Adult)
  AND: NOT (Last < 7)

```

ABen[Loop1] := 0

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: LegDep[Loop1] = RESPONSE

```

ABen[Loop1] := ABen[LegDep[Loop1]]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: DMParent1[Loop1] <> 0

```

ABen[Loop1] := ABen[DMParent1[Loop1]]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: DMParent2[Loop1] <> 0

```

ABen[Loop1] := ABen[DMParent2[Loop1]]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: NOT (DMParent2[Loop1] <> 0)
  AND: Last < 7

```

Last := (Last + 1)

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: NOT (DMParent2[Loop1] <> 0)
  AND: Last < 7

```

ABen[Loop1] := Last

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: NOT (DMParent2[Loop1] <> 0)
  AND: NOT (Last < 7)

```

ABen[Loop1] := 0

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: LegDep[Loop1] = RESPONSE

```

ABen[Loop1] := ABen[LegDep[Loop1]]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: DMParent1[Loop1] <> 0

```

ABen[Loop1] := ABen[DMParent1[Loop1]]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: DMParent2[Loop1] <> 0

```

ABen[Loop1] := ABen[DMParent2[Loop1]]

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: NOT (DMParent2[Loop1] <> 0)
  AND: Last < 7

```

Last := (Last + 1)

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: NOT (DMParent2[Loop1] <> 0)
  AND: Last < 7

```

ABen[Loop1] := Last

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: ABen[Loop1] = EMPTY AND (PRec[Loop1].Depend IN [DepAd .. Child])
  AND: NOT (DMParent2[Loop1] <> 0)
  AND: NOT (Last < 7)

```

ABen[Loop1] := 0

```

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)

```

NewBU := Last

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSize

NameInBU[Loop1] := ''

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSize
AND: PRec[Loop1].Depend IN [Adult]

**NameInBU[ABen[Loop1]] := (NameInBU[ABen[Loop1]] +
UPCASE(DMName[Loop1]) + ' ')**

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSize
AND: PRec[Loop1].Depend IN [DepAd .. Child]

**NameInBU[ABen[Loop1]] := (NameInBU[ABen[Loop1]] + DMName[Loop1]
+ ' ')**

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO NewBU

**NameInBU[Loop1] := ('
' + STR(Loop1,1,0) + ': ' + NameInBU[Loop1])**

ASK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)

ShowBen

^I

If you think that the computer has made a mistake in allocating the individuals to Benefit Units, go back to the household grid and check the relationship codes of each person.

1..1

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

CHECK IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)

hhchull := No

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: PRec[Loop1].Sex = RESPONSE

DMBU[Loop1] := ABen[Loop1]

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: PRec[Loop1].Sex = RESPONSE

HHG.P[Loop1].BenUnit := ABen[Loop1]

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: (DMBU[Loop1] = 1) AND (DMAge[Loop1] IN [0 .. 10])

hhchull := Yes

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Loop1 IN QHholder.HHldr

RentName := (RentName + ' ' + DMName[Loop1])

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: Loop1 IN QHholder.HHldr
AND: DMBU[Loop1] <> 1

NotHRPBU := 1

COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
AND: In loop FOR Loop1 := 1 TO HHSIZE
AND: PRec[Loop1].Depend = Adult
AND: BUAdName[DMBU[Loop1]] =

BUAdName[DMBU[Loop1]] := DMName[Loop1]

```
COMPUTE IF: (QHholder.HHldr = RESPONSE) OR (Edit = Yes)
  AND: In loop FOR Loop1 := 1 TO HHSIZE
  AND: PRec[Loop1].Depend = Adult
  AND: NOT (BUAdName[DMBU[Loop1]] =)
```

```
BUAdName[DMBU[Loop1]] := (BUAdName[DMBU[Loop1]] + ' and ' +
DMName[Loop1])
```

FRS1004B.QAccomdat

Questions about accommodation

RECORD ALWAYS :

Ten1Ex

^I QAccomDat^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

RECORD ALWAYS :

Ten2Rs

^I QAccomDat^I

^N

Can I just check do you live rent free because you receive 100% housing benefit?^N

^I^IC If Yes recode tenure to code 4 (rent it).

Please check their rent/mortgage is not paid by^B benefits^B. Only accommodation provided by someone else (employer, relative, etc) is rent-free.

- (1) Passed
 - (2) Hard
 - (3) Soft
 - (4) Suppressed
-

RECORD ALWAYS :

Ten2Ex

^I QAccomDat^I

^I^IC ^SuppTxt

OPEN

ASK ALWAYS :

Tenure

^I QAccomDat^I

^I^IS B1^I

^N

In which of these ways do you occupy this accommodation?

- (1) Own it outright
 - (2) Buying it with the help of a mortgage or loan
 - (3) Pay part rent and part mortgage (^SharOwn)
 - (4) Rent it
 - (5) Live here rent-free (including in a relative's/friend's property; excluding squatting)
 - (6) Squatting
-

WARN IF: *Tenure = NONRESPONSE*
ERROR

^I^KeyTxt

ASK IF: *Tenure = NONRESPONSE*

Ten1Ex

^I QAccomDat^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

WARN ALWAYS:
Tenure <> RentFree

Tenure <> RentFree

ASK IF: *Ten2Rs = Suppressed*

Ten2Ex

^I QAccomDat^I

^I^IC ^SuppTxt

OPEN

ASK IF: *Tenure = Part*

SOBuy

^I QAccomDat^I

^I^IC Ask or record:

^SOwners:^I^N Are you still buying your share in this (house/flat), or have you now paid off that mortgage or loan?

- (1) Still buying
- (2) Mortgage is paid off

ASK ALWAYS:

SubLet

^I QAccomDat^I

^N

Do you have a formal arrangement to let, or sub-let, any part of this accommodation to someone who is^B not^B a member of your household?

- (1) Yes
- (2) No

COMPUTE IF: *SubLet = Yes*

How := ('Thinking just of the accommodation occupied ' + 'by your household,
how')

ASK IF: *SubLet = Yes*

SubLetY

^I QAccomDat^I

^N

Who is that?..^N^I Code first that applies.

^IC Close relatives = Householder's partner,
parent (incl.step-), son or daughter (incl. step-),
brother or sister, or spouse of any of these.

- (1) Close relative
 - (2) Other relative
 - (3) Non-relative
-

COMPUTE IF: *NOT (SubLet = Yes)*

How := 'How'

ASK ALWAYS:

Rooms

^I QAccomDat^I

^I

In general, include any room which is habitable or usable by the household all year round.

If a room is open-plan count it as 2 rooms if it is divided by a fixed sliding or folding partition.

0..20

ASK ALWAYS:

RoomShr

^I QAccomDat^I

^N

Are any of these rooms shared with anyone who is ^B not^B a member of your household?

- (1) Yes
 - (2) No
-

ASK IF: *RoomShr = Yes*

NmRmShar

^I QAccomDat^I

^N

How many rooms are shared with anyone who is not a member of your household?

1..10

ASK ALWAYS:

Bedroom

^I QAccomDat^I

^N

^How many bedrooms do you have in this accommodation?^N

^I^IC Include any room used for sleeping.

1..10

COMPUTE ALWAYS:

**ChkTxt := ('cannot be greater than total number of rooms: ' +
'please check your answers and amend as necessary.')**

CHECK ALWAYS:

NmRmShar <= Rooms

^I^IC Number of shared rooms ^ChkTxt

CHECK ALWAYS:

Bedroom <= Rooms

^I^IC Number of bedrooms ^ChkTxt

ASK IF: NatCen <> Yes

MainAcc

^I QAccomDat^I

^I

Note that this is accommodation occupied by the household. If the household occupies a flat in a converted house, code as a flat.

- (1) ^N a house or bungalow
 - (2) ^N a flat or maisonette
 - (3) ^N a room or rooms
 - (4) ^N or something else?
-

ASK IF: NatCen <> Yes

Shelter

^I QAccomDat^I

^N

Is this sheltered accommodation?^N

^I^IC Housing with a warden and/or alarms.

- (1) Yes
 - (2) No
-

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

Detach := (N + 'detached' + N)

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

SemiDetach := (N + 'semi-detached' + N)

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

Terrace := (N + 'or terraced/end of terrace?' + N)

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

PurposeBuilt := ''

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

ConvertedHouse := ''

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

MobileHome := ''

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

OtherKind := ''

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = House

```

```

Accommodation := 'the house or bungalow'

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = Flat

```

```

Detach := ''

```

```

COMPUTE IF: NatCen <> Yes
  AND: MainAcc IN [House .. Flat, Other]
  AND: MainAcc = Flat

```

```

SemiDetach := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: MainAcc = Flat

```

```

Terrace := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: MainAcc = Flat

```

```

PurposeBuilt := (N + 'a purpose-built block' + N)

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: MainAcc = Flat

```

```

ConvertedHouse := (N + 'or a converted house/some other kind of
building?' + N)

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: MainAcc = Flat

```

```

MobileHome := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: MainAcc = Flat

```

```

OtherKind := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: MainAcc = Flat

```

```

Accommodation := 'the flat/maisonette'

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

Detach := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

SemiDetach := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

Terrace := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

PurposeBuilt := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

ConvertedHouse := ''

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

MobileHome := (N + 'a caravan, mobile home or houseboat' + N)

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

OtherKind := (N + 'or some other kind of accommodation?' + N)

```

```

COMPUTE IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: NOT (MainAcc = Flat)

```

```

Accommodation := 'the accommodation'

```

```

ASK IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]

```

TypeAcc

^I QAccomDat^I

^I

Houses which are joined only by a garage (link-detached) should be coded detached.

- (1) ^Detach
 - (2) ^SemiDetach
 - (3) ^Terrace
 - (4) ^PurposeBuilt
 - (5) ^ConvertedHouse
 - (6) ^MobileHome
 - (7) ^OtherKind
-

```

CHECK IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: TypeAcc = RESPONSE
    ((IN(TypeAcc, [??])) AND (MainAcc = House)) OR ((IN(TypeAcc, [???, ???]))
AND (MainAcc = Flat))) OR ((MainAcc = Other) AND (IN(TypeAcc, [???, ???])))

```

^I^IC This code is not valid for this accommodation.

```

CHECK IF: NatCen <> Yes
    AND: MainAcc IN [House .. Flat, Other]
    AND: TypeAcc IN [Detached .. Purpose]
    (SubLet <> Yes) AND INVOLVING(MainAcc)

```

^I

^IC As part of this accommodation is sub-let, this household cannot be coded as occupying a whole house, flat etc.

Use another code at MainAcc instead.

ASK IF: NatCen <> Yes
AND: (TypeAcc IN [Purpose, Converted]) OR (MainAcc = Rooms)

Floor

^I QAccomDat^I

^N

What is the floor level of this household's accommodation?

- (1) Basement/semi-basement
- (2) Ground floor/street level
- (3) 1st floor (floor above street level)
- (4) 2nd floor
- (5) 3rd floor
- (6) 4th Floor
- (7) 5th to 9th floor
- (8) 10th floor or higher
- (9) Don't Know

ASK IF: NatCen <> Yes

Entry

^I QAccomDat^I

^N

Are there any physical barriers to entry to the house/flat/accommodation?

^I^IC Code all that apply

SET [6] OF

- (1) No physical impediments or barriers
- (2) Locked common entrance
- (3) Locked gates
- (4) Security staff, concierge or other gatekeeper
- (5) Entry phone access, intercom
- (6) Guard dog/patrol animal
- (7) Warden controlled

CHECK IF: NatCen <> Yes
AND: None IN Entry
Entry.CARDINAL = 1

^I

'None' is an exclusive code.

ASK ALWAYS :

YearLive

^I QAccomDat^I

^N

For how many years have you^N^I, (that is ^PHRPName),^I^N lived at this address?^N

^I^IC Probe to classify.

- (1) Less than 12 months
 - (2) 12 months but less than 2 years
 - (3) 2 years but less than 3 years
 - (4) 3 years but less than 5 years
 - (5) 5 years but less than 10 years
 - (6) 10 years but less than 20 years
 - (7) 20 years or longer
-

ASK IF: YearLive = Less12m

MonLive

^I QAccomDat^I

^N

For how many months have you^N^I, (that is ^PHRPName),^I^N lived at this address?^N

^I^IC Enter number of months, to nearest whole month.

0..11

ASK IF: YearLive IN [Less12m, Fr1yr]

YrLvChk

^I QAccomDat^I

^N

Can I just check, did you move to this address on or after 7th April ^FYearM1?

- (1) Yes - on or after 7th April ^FYearM1
 - (2) No - before 7th April ^FYearM1
-

COMPUTE IF: NewBU = 1

HHStat := Conv

ASK IF: NOT (NewBU = 1)

HHStat

^I QAccomDat^I

^I^IC Classify this household as one of the following:

NOTE:^B Conventional Households^B include:

- tenure is owner occupier and a 2nd or 3rd benefit unit is paying rent
- tenure is rent free but 2nd or 3rd BU receives Housing Benefit.
- 2nd or 3rd BU members paying rent to the householder(s) in BU1 are also named as householders (this is similar to part owned/part rented tenure)^I.

- (1) Conventional household: ie. single person or couple - with other family and/or boarder(s) and/or lodger(s)
 - (2) 'Shared' household arrangements: identity of HRP is unclear or arbitrary - eg. students, nurses, unrelated adults etc, sharing ON EQUAL BASIS
-

RECORD ALWAYS:

AnyVeh

^I QAccomDat^I

^N

Do you at present own or have continuous use of any motor vehicles?

- (1) Yes
 - (2) No
-

RECORD ALWAYS:

VehNumb

^I QAccomDat^I

Number of vehicles.

0..8

RECORD ALWAYS:

AdultH

^I QAccomDat^I

Actual number of adults in household.

0..14

RECORD ALWAYS:

DepChldH

^I QAccomDat^I

Actual number of children in household.

0..14

RECORD ALWAYS :

DatYrAgo

^I QAccomDat^I
Date one year ago

DATE

RECORD ALWAYS :

BenUnits

^I QAccomDat^I
Actual number of Benefit Units in household.

0..7

RECORD ALWAYS :

Dentist

^I QAccomDat^I
Anyone having NHS visits to the dentist?

- (1) Yes
 - (2) No
-

RECORD ALWAYS :

EyeTest

^I QAccomDat^I
Anyone having NHS eyetests?

- (1) Yes
 - (2) No
-

RECORD ALWAYS :

Specs

^I QAccomDat^I
Anyone having NHS glasses/lenses?

- (1) Yes
 - (2) No
-

RECORD ALWAYS :

Hospital

^I QAccomDat^I
Anyone having NHS hospital treatment?

- (1) Yes
 - (2) No
-

RECORD ALWAYS :

Pres

^I QAccomDat^I
Anyone having NHS prescriptions?

- (1) Yes
- (2) No

RECORD ALWAYS :

HStart

^I QAccomDat^I
Anyone having Healthy Start Vouchers?

- (1) Yes
- (2) No

RECORD ALWAYS :

SchMeal

^I QAccomDat^I
Anyone having school meals?

- (1) Yes
- (2) No

RECORD ALWAYS :

SchMilk

^I QAccomDat^I
Anyone having school milk?

- (1) Yes
- (2) No

CHECK ALWAYS :

RESERVECHECK

RESERVECHECK

CHECK ALWAYS :

RESERVECHECK

RESERVECHECK

CHECK ALWAYS :

RESERVECHECK

RESERVECHECK

CHECK ALWAYS :

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

CHECK ALWAYS:
RESERVECHECK

RESERVECHECK

FRS1004B (continued)
FAMILY RESOURCES SURVEY 2010-2011

*CHECK ALWAYS:***RESERVECHECK****RESERVECHECK***CHECK ALWAYS:***RESERVECHECK****RESERVECHECK***CHECK ALWAYS:***RESERVECHECK****RESERVECHECK***CHECK ALWAYS:***RESERVECHECK****RESERVECHECK***COMPUTE IF: QAccomdat.HHStat = Shared**AND: In loop FOR Loop1 := 1 TO NewBU***BUHBElig[Loop1] := Yes***COMPUTE ALWAYS:***QAccomdat.AdultH := AllAd***COMPUTE ALWAYS:***QAccomdat.DepChldH := AllCh***COMPUTE ALWAYS:***QAccomdat.DatYrAgo := DLYear***COMPUTE ALWAYS:***QAccomdat.BenUnits := NewBU**

FRS1004B.QRenting

Questions about renters

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]

Landlord

^I QRenting^I

^I

If property is let through a letting agent or estate agent, the question refers to the owner not the agent, so please probe to try to find out who actually owns the property.

If the respondent does not know who the landlord is, use code 7 (other private individual) rather than coding 'Don't know'.

Code 1 (^LANIHE) includes people renting from Housing Action Trusts.

Code 2 (housing association etc.) includes Registered Social Landlords. Nearly all housing associations are now Registered Social Landlords but continue to be known as housing associations.

- (1) ^Council1
 - (2) A housing association, charitable trust or Local Housing Company
 - (3) Employer (organisation) of a household member
 - (4) Another organisation
 - (5) Relative/friend (before you lived here) of household member
 - (6) Employer (individual) of a household member
 - (7) Another individual, private landlord or Letting Agency
-

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]

LLEx

^I QRenting^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]

AND: Edit = No

AND: Landlord = NONRESPONSE

ERROR

^I^IC ^KeyTxt

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: Edit = No
 AND: Landlord = NONRESPONSE

LLEx

^I QRenting^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: Landlord = Council

Allowance := 'Rebate'

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: NOT (Landlord = Council)

Allowance := 'Allowance'

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]

Furnish

^I QRenting^I

^N

Is this accomodation provided...

^N

^I...Running prompt...

(1) ^N furnished,^N

(2) ^N partly furnished (eg. curtains and carpets only),^N

(3) ^N or unfurnished?^N

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: Landlord IN [FrndRel .. OthIndiv]

ResLL

^I QRenting^I

^N

Does the landlord live in the building?

(1) Yes

(2) No

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: Landlord IN [FrndRel .. OthIndiv]
 AND: (ResLL = Yes) AND (QAccomdat.TypeAcc = Purpose)

ResLL2

^I QRenting^I

^N

Does the landlord live in the same flat as you or not

(1) Yes

(2) No

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (ResLL = No) OR (ResLL2 = No)
AND: NatCen <> NI

YStart

^I QRenting^I
 ^I
 Ask or record^I
 ^N
 In which year did you first become a tenant of this accommodation?^N

 ^I^IC 'You'=Person(s) named at 'Hhldr', that is... ^RentName.

- (1) 1988 or earlier
- (2) From 1989 to February 1997
- (3) March 1997 or later

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (ResLL = No) OR (ResLL2 = No)
AND: NOT (NatCen <> NI)

NIYstart

^I QRenting^I

 ^I^IC Ask or record^I
 ^N
 In which year did you first become a tenant of this accommodation?^N

 ^I^IC 'You'=Person(s) named AT 'Hhldr', that is... ^RentName.

- (1) 31 March 2007 or earlier
- (2) 1 April 2007 or later

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (ResLL = No) OR (ResLL2 = No)
AND: ((NatCen <> NI) AND (YStart IN [ToFeb97 .. AftMar97])) OR ((NatCen = NI) AND (NIYstart = Aft1979))

Ctract

^I QRenting^I
 ^N
 When you started to rent this accommodation ^N^I...Running prompt...

- (1) ^N...did you and the landlord sign a written agreement,
- (2) ^N...did you have a written agreement which you didn't sign,
- (3) ^N...or did you just have an unwritten agreement?

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NatCen = NI

various := 'various'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NOT (NatCen = NI)

```

```

various := 'various other'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NatCen <> NI
AND: (YStart IN [ToFeb97 .. AftMar97]) OR ((ResLL = Yes) AND (ResLL2 = EMPTY
OR (ResLL2 = Yes)))
AND: Scotland = Yes

```

```

AssuredSH := 'Short Assured'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NatCen <> NI
AND: (YStart IN [ToFeb97 .. AftMar97]) OR ((ResLL = Yes) AND (ResLL2 = EMPTY
OR (ResLL2 = Yes)))
AND: NOT (Scotland = Yes)

```

```

AssuredSH := 'Assured Shorthold'

```

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NatCen <> NI
AND: (YStart IN [ToFeb97 .. AftMar97]) OR ((ResLL = Yes) AND (ResLL2 = EMPTY
OR (ResLL2 = Yes)))

```

TenType

^I QRenting^I

^I^IS B3^I

^N

Can you tell me what kind of tenancy you have?^N

^I^IC If tenancy type written on contract/notice ask respondent to read out.

- (1) ^AssuredSH
- (2) Assured
- (3) Regulated (tenancy must have started in 1988 or earlier)
- (4) Resident landlord
- (5) Let by educational institution
- (6) Other type of let

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NatCen <> NI
AND: (YStart IN [ToFeb97 .. AftMar97]) OR ((ResLL = Yes) AND (ResLL2 = EMPTY
OR (ResLL2 = Yes)))
AND: TenType = OthLet

```

OthType

^I QRenting^I

^I^IS B4^I

^N

There are ^various ways in which landlords can let accommodation. Will you please look at this card and tell me if the letting is one of these?

- (1) Crown tenancy/licence (includes H.M Forces)
 - (2) Service occupancy (excludes H.M. Forces)
 - (3) Business or agricultural tenancy
 - (4) Assured agricultural occupancy
 - (5) Asylum seeker let (issued by National Asylum Support Service NASS)
 - (6) Holiday let
 - (7) Other type of let
-

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: NOT (NatCen <> NI)

```

OthType

^I QRenting^I

^I^IS B4^I

^N

There are ^various ways in which landlords can let accommodation. Will you please look at this card and tell me if the letting is one of these?

- (1) Crown tenancy/licence (includes H.M Forces)
 - (2) Service occupancy (excludes H.M. Forces)
 - (3) Business or agricultural tenancy
 - (4) Assured agricultural occupancy
 - (5) Asylum seeker let (issued by National Asylum Support Service NASS)
 - (6) Holiday let
 - (7) Other type of let
-

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
(((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
AftMar97]))) OR ((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
Aft1979))
AND: NatCen = NI

```

Assured := 'a Protected'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
(((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
AftMar97]))) OR (((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
Aft1979))
AND: NatCen = NI

```

Tenancy := 'a Protected Shorthold Tenancy'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
(((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
AftMar97]))) OR (((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
Aft1979))
AND: NatCen = NI

```

Shorthold := 'a Protected Shorthold'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
(((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
AftMar97]))) OR (((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
Aft1979))
AND: NOT (NatCen = NI)

```

Assured := 'an Assured'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
(((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
AftMar97]))) OR (((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
Aft1979))
AND: NOT (NatCen = NI)

```

Tenancy := 'for an Assured Shorthold Tenancy'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
(((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
AftMar97]))) OR (((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
Aft1979))
AND: NOT (NatCen = NI)

```

Shorthold := 'an Assured Shorthold'

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: (((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = ToFeb97)) OR
 (((Scotland = Yes) AND (ResLL2 <> Yes)) AND (YStart IN [ToFeb97 ..
 AftMar97]))) OR (((NatCen = NI) AND (ResLL2 <> Yes)) AND (NIYstart =
 Aft1979))
AND: NatCen <> NI
AND: TenType = NONRESPONSE OR OthType = NONRESPONSE

Short1

^I QRenting^I

^N

There is a form of tenancy called ^Assured Shorthold. It had to be initially for a fixed period and you had to be given a special notice in writing by the landlord that told you it was ^Tenancy. Here is an example of a notice to a tenant saying that the tenancy is ^Shorthold.

^I^IC SHOW EXAMPLE OF NOTICE.^I

^N

Does your notice state that it is ^Assured Shorthold or not?

- (1) Yes, an Assured Shorthold
- (2) Other agreement

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: Ctract IN [Signed .. NotSign]
AND: ((NatCen <> NI) AND (Scotland <> Yes)) AND (YStart = AftMar97)
AND: TenType = NONRESPONSE OR OthType = NONRESPONSE

Short2

^I QRenting^I

^N

Most tenancies are Assured Shortholds. There are others, just called 'Assured'. For these you have to be given a notice, in writing by the landlord, that tells you it is NOT an Assured Shorthold agreement.^N

^I^IC SHOW EXAMPLE OF NOTICE.^I

^N

Does your agreement or notice state that it is NOT an Assured Shorthold?^N

^I^IC The law changed in March 1997 to the effect that by default all tenancy agreements are assured shortholds, unless the landlord gave written notice to the contrary.

- (1) Not an Assured Shorthold
- (2) No, other agreement

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (((Scotland <> Yes) AND (YStart = Bef1988)) AND ((ResLL = No) OR
 (ResLL2 = No))) OR ((Scotland = Yes) AND ((ResLL = No) OR (ResLL2 = No)))
 OR ((NatCen = NI) AND ((ResLL = No) OR (ResLL2 = No)))
AND: NatCen = NI

assessed := 'assessed'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (((Scotland <> Yes) AND (YStart = Bef1988)) AND ((ResLL = No) OR
(ResLL2 = No))) OR ((Scotland = Yes) AND ((ResLL = No) OR (ResLL2 = No)))
OR ((NatCen = NI) AND ((ResLL = No) OR (ResLL2 = No)))
AND: NatCen = NI

```

```

rent_officer := ('rent officer for Northern Ireland who will set
up an independent ' + 'committee to assess and fix an appropriate
rent')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (((Scotland <> Yes) AND (YStart = Bef1988)) AND ((ResLL = No) OR
(ResLL2 = No))) OR ((Scotland = Yes) AND ((ResLL = No) OR (ResLL2 = No)))
OR ((NatCen = NI) AND ((ResLL = No) OR (ResLL2 = No)))
AND: NOT (NatCen = NI)

```

```

assessed := 'registered'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (((Scotland <> Yes) AND (YStart = Bef1988)) AND ((ResLL = No) OR
(ResLL2 = No))) OR ((Scotland = Yes) AND ((ResLL = No) OR (ResLL2 = No)))
OR ((NatCen = NI) AND ((ResLL = No) OR (ResLL2 = No)))
AND: NOT (NatCen = NI)

```

```

rent_officer := ('local rent officer or rent assessment committee
to decide on a ' + 'fair rent which is then registered')

```

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Landlord IN [FrndRel .. OthIndiv]
AND: (((Scotland <> Yes) AND (YStart = Bef1988)) AND ((ResLL = No) OR
(ResLL2 = No))) OR ((Scotland = Yes) AND ((ResLL = No) OR (ResLL2 = No)))
OR ((NatCen = NI) AND ((ResLL = No) OR (ResLL2 = No)))

```

FairRent

^I QRenting^I

^N

Has the rent been registered by the local rent officer or rent committee?

- (1) Yes
 - (2) No
-

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]

```

AccJob

^I QRenting^I

^N

Does this accommodation go with the job (past or present) of anyone in your household?

- (1) Yes
 - (2) No
-

```
ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccJob = Yes
```

AccJbPer

```
^I QRenting^I
^N
Who is that?^N
```

```
^I^IC Code all that apply.
```

```
SET [14] OF
(1) ^DMName[1]
(2) ^DMName[2]
(3) ^DMName[3]
(4) ^DMName[4]
(5) ^DMName[5]
(6) ^DMName[6]
(7) ^DMName[7]
(8) ^DMName[8]
(9) ^DMName[9]
(10) ^DMName[10]
(11) ^DMName[11]
(12) ^DMName[12]
(13) ^DMName[13]
(14) ^DMName[14]
```

```
CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccJob = Yes
AND: In loop FOR Index := 1 TO 14
AND: Index IN AccJbPer
PRec[].Depend[Index] = Adult
```

```
^I
Code ^Index is not valid for this question.
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: QAccomdat.HHStat = Shared
```

```
es_household := (' you, that is, just ' + B + HRPNames + B + ',')
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: QAccomdat.HHStat = Shared
```

```
IsAre := 'Are'
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: NOT (QAccomdat.HHStat = Shared)
```

```
es_household := 'es your household'
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: NOT (QAccomdat.HHStat = Shared)
```

```
IsAre := 'Is'
```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]

RentDoc

^I QRenting^I

^N

Do you have a rent book, rent card, Housing Benefit (or Local Housing Allowance) statement or some other rent document that you could consult?^N

^I^IC If HB statement available please consult this.

- (1) Housing Benefit (or Local Housing Allowance) Statement
- (2) Some other document
- (3) None

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]
AND: RentDoc IN [HBStmt, Oth]

Consult_the_document := 'Please consult the document.'

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]
AND: NOT (RentDoc IN [HBStmt, Oth])

Consult_the_document := ''

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]

Rent

^I QRenting^I

^I

If in arrears, enter amount last paid but open a Note and give the date of payment (and say if an extra amount is included to pay towards the arrears).

If 100% rent rebate/HB is received and water/sewerage rates and other services etc are^B not^B included in the rent, then the amount at 'Rent' should be zero. But if water, etc rates ARE included in the rent, then the amount paid for these rates should be entered at the question 'Rent'.

If rent includes an element for the business part of the property (eg a shop beneath a flat), and the amount for the residential part cannot be determined, enter 'DK' at 'Rent'.

0.00..999997.00

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]

RentEx

^I QRenting^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]
AND: Edit = No
AND: Rent = NONRESPONSE
ERROR

^I^IC ^KeyTxt

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]
AND: Edit = No
AND: Rent = NONRESPONSE

RentEx

^I QRenting^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]
AND: Rent > 0

RentPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: PTenure IN [Rents, Part]
AND: Rent > 0

RentPd

^I QRenting^I

^N

How long does this cover?

- (1) One week
 - (2) Two weeks
 - (3) Three weeks
 - (4) Four weeks
 - (5) Calendar month
 - (7) Two Calendar months
 - (8) Eight times a year
 - (9) Nine times a year
 - (10) Ten times a year
 - (13) Three months/13 weeks
 - (26) Six months/26 weeks
 - (52) One Year/12 months/52 weeks
 - (90) Less than one week
 - (95) One off/lump sum
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: PTenure IN [Rents, Part]
 AND: Rent > 0
 AND: RentPd = Note

RentPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: PTenure IN [Rents, Part]
 AND: Rent > 0
 AND: Edit = Yes
 RentPd <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QRenting.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[8] := 6.5

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[9] := 5.78

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
      AND: PTenure IN [Rents, Part]
      AND: Rent > 0
```

PdConW[10] := 5.2

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: PTenure IN [Rents, Part]
    AND: Rent > 0
```

PdConW[13] := 13

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: PTenure IN [Rents, Part]
    AND: Rent > 0
```

PdConW[26] := 26

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: PTenure IN [Rents, Part]
    AND: Rent > 0
```

PdConW[52] := 52

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: PTenure IN [Rents, Part]
    AND: Rent > 0
    AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])
```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: PTenure IN [Rents, Part]
    AND: Rent > 0
    AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))
```

PWeekly := 0

FRS1004B.QRenting (continued)

Questions about renters

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: PTenure IN [Rents, Part]
  AND: Rent > 0
  AND: RentPd IN [OneWeek .. Year]
  AND: LWeekly1 >= 0.01
```

RentWkly := LWeekly1

```
WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: PTenure IN [Rents, Part]
  AND: Rent > 0
  AND: RentPd IN [OneWeek .. Year]
  AND: LWeekly1 >= 0.01
  AND: Landlord = Council
  (RentWkly < 101) AND INVOLVING(RentPd,Rent)
```

^I

This comes to £^RentWkly a week.

Rents for Council tenants are normally below £100 a week.

```
WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: PTenure IN [Rents, Part]
  AND: Rent > 0
  AND: RentPd IN [OneWeek .. Year]
  AND: LWeekly1 >= 0.01
  AND: Edit = No
  ((RentWkly < 151) OR (Landlord = Council)) AND INVOLVING(RentPd,Rent)
```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: PTenure IN [Rents, Part]
  AND: Rent = REFUSAL
```

HMissVar := (HMissVar + 1)

```
ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: PTenure IN [Rents, Part]
  AND: Rent = DONTKNOW
```

RentDK

^I QRenting^I

^I^IC@|Is this 'don't know' because rent is partly for^B business^B,
@|@|@|and you cannot establish a separate amount for the^B domestic^B accommodation?

- (1) Yes (Please give full details in a Note)
- (2) No

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: PTenure IN [Rents, Part]
    AND: Rent = DONTKNOW
    AND: RentDK <> Yes

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: Rent <> EMPTY

```

RentHol

^I QRenting^I

^N

Do you have a rent holiday?^N

^I^IC Some people know this as 'Rent free week(s)'.

(1) Yes

(2) No

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: Rent <> EMPTY
    AND: RentHol = Yes

```

WeekHol

^I QRenting^I

^N

For how many weeks of the year do you have a rent holiday?

1..52

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: Rent <> EMPTY
    AND: RentHol = Yes
    WeekHol <= 8

```

^I

Rent holidays do not normally exceed 8 weeks per year.

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: Landlord = Council

```

```

allowed := 'allowed'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: Landlord = Council

```

```

directly := ''

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: NOT (Landlord = Council)

```

```

allowed := 'receiving'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: NOT (Landlord = Council)

```

```

directly := ', either directly or by having it paid to your landlord'

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]

HBenefit

^I QRenting^I

^N

Are you ^allowed Housing Benefit (or Local Housing Allowance) or Rent ^Allowance, to help with paying your rent^directly?

(1) Yes

(2) No

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]

HBenEx

^I QRenting^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]

AND: Edit = No

AND: HBenefit = NONRESPONSE

ERROR

^I^IC ^KeyTxt

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]

AND: Edit = No

AND: HBenefit = NONRESPONSE

HBenEx

^I QRenting^I

^I^IC ^KeyTxt

^SuppTxt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes

HBRecp

^I QRenting^I

^I

A split payment might occur when an LHA customer is entitled to an excess above their contractual rent but they are classed as vulnerable, the rent will be paid to the landlord but the excess will still be paid to the customer.

- (1) Directly to recipient of Housing Benefit (or Local Housing Allowance) or Rent ^Allowance (i.e. the respondent or an appointee)
- (2) Directly to landlord / property agent
- (3) Split payment (i.e. payment directly to landlord and just the excess of rent and housing benefit (or Local Housing Allowance) to respondent)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: ((Rent = 0) AND (Rent = RESPONSE)) AND (HBenefit = Yes)

Rebate

^I QRenting^I

^N

You said that you paid no rent last time, is that because you get 100% Housing Benefit?

- (1) Yes
- (2) No

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: ((Rent = 0) AND (Rent = RESPONSE)) AND ((HBenefit = No) OR (Rebate = No))

RebateO

^I QRenting^I

^N

Can I just check, what is the reason for your paying no rent last time?

STRING[60]

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HHSize > 1

you_all := 'you and the other members of your household'

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: NOT (HHSize > 1)

you_all := 'you'

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes

NCDVIB := (NCDVIB + 1)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes

HBenAmt

^I QRenting^I

^N

How much Housing Benefit (or Local Housing Allowance)/ rent rebate/ allowance are ^you_all allowed?^N

^I^IC Some respondents may receive more housing benefit than the amount of their rent.

0.01..997.00

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBenAmt = NONRESPONSE

HMissVar := (HMissVar + 1)

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBenAmt > 0

HBenPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBenAmt > 0

HBenPd

^I QRenting^I

^N

How long does this cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: HBenAmt > 0
 AND: HBenPd = Note

HBenPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: HBenAmt > 0
 AND: Edit = Yes
 HBenPd <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QRenting.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[8] := 6.5

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[9] := 5.78

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
```

PdConW[10] := 5.2

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: HBenAmt > 0
```

```
PdConW[13] := 13
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: HBenAmt > 0
```

```
PdConW[26] := 26
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: HBenAmt > 0
```

```
PdConW[52] := 52
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: HBenAmt > 0
    AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])
```

```
PWeekly := (PAmount / PdConW[ORD(PPeriod)])
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: HBenAmt > 0
    AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))
```

```
PWeekly := 0
```

FRS1004B.QRenting (continued)

Questions about renters

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
  AND: HBenPd IN [OneWeek .. Year]
  AND: LWeekly1 >= 0.01

```

HBenWkly := LWeekly1

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
  AND: HBenPd IN [OneWeek .. Year]
  AND: LWeekly1 >= 0.01
  AND: Edit = No
  (HBenWkly < 200) AND INVOLVING (HBenPd,HBenAmt)

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
  AND: HBenPd IN [OneWeek .. Year]
  AND: LWeekly1 >= 0.01
  AND: Edit = Yes
  AND: (HBenWkly = RentWkly) OR (Rent = HBenAmt)
  ERROR AND INVOLVING (Rent,HBenAmt)

```

^I

Editor: The respondent has given exactly the same figure for rent and housing benefit. Please check that there is no double counting.

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: HBenAmt > 0
  AND: Rent > 0

```

HBenChk

^I QRenting^I

^N

Can I just check, is the amount of £^rent for rent that you mentioned earlier, BEFORE or AFTER taking off the Housing Benefit (or Local Housing Allowance)?

- (1) Before
 - (2) After
-

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: (HBenWkly >= 0) AND (RentWkly >= 0)
  AND: HBenChk = Aftr

```

RentHBWk := (RentWkly + HBenWkly)

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: (HBenWkly >= 0) AND (RentWkly >= 0)
    AND: HBenChk = Aftr

```

RentHB := (Rent + HBenAmt)

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: (HBenWkly >= 0) AND (RentWkly >= 0)
    AND: NOT (HBenChk = Aftr)

```

RentHBWk := RentWkly

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: (HBenWkly >= 0) AND (RentWkly >= 0)
    AND: NOT (HBenChk = Aftr)

```

RentHB := Rent

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: (HBenAmt = DONTKNOW) AND ((Rent = DONTKNOW) OR (Rent > 0))

```

RentFull

^I QRenting^I

^N

How much is your FULL rent - that is, BEFORE Housing Benefit or Rent ^Allowance?

0.00..999997.00

```

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentFull > 0

```

RentPx1

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: RentFull > 0

RentPd1

^I QRenting^I

^N

How long does this cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: RentFull > 0
 AND: RentPd1 = Note

RentPx1

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: RentFull > 0
 AND: Edit = Yes
 RentPd1 <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes

HBWeeks

^I QRenting^I

^N

For how long have you been on Housing Benefit (or Local Housing Allowance) or Rent ^Allowance this time?^N

^I^IC Please note that a number of changes of circumstances, not just a change of address could have resulted in the break of a claim (for example a partnership forming/dissolving, change in number of dependants etc).

- (1) Up to 2 years
- (2) 2 years but less than 3
- (3) 3 years but less than 4
- (4) 4 years but less than 5
- (5) 5 or more years

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y

HBWeeks2

^I QRenting^I

^N

Please tell me how many weeks you have been on Housing Benefit (or Local Housing Allowance) or Rent ^Allowance this time?

0..997

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y
AND: HBWeeks2 = RESPONSE
IN (HBWeeks2, [1..104])

^I^IC Enter a value between 1 and 104

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y

HBYear

^I QRenting^I

^N

Can I just check, in which year did you begin your^B current^B Housing Benefit (or Local Housing Allowance) claim?

2008..2011

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y
AND: QDataBag.SampMnth IN [4 .. 12]
HBYear <> LYear

^I
You've entered a future date!

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y
AND: (QSignIn.StartDat = RESPONSE) AND (HBYear = RESPONSE)
HBYear <= QSignIn.StartDat.YEAR

^I
You've entered a future date!

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y

HBMnth

^I QRenting^I
^N
And which month was that?

- (1) January
- (2) February
- (3) March
- (4) April
- (5) May
- (6) June
- (7) July
- (8) August
- (9) September
- (10) October
- (11) November
- (12) December

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: HBWeeks = Less2Y
AND: (QSignIn.StartDat = RESPONSE) AND (HBYear = RESPONSE)
AND: HBMnth IN [January .. December]
TODATE (HBYear, ORD (HBMnth), 1) <= QSignIn.StartDat

^I
You've entered a future date!

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: RentDoc = HBStmt

EligAmt

^I QRenting^I

^N

On the (rent book/ card/ statement), what is the amount shown for eligible rent?^N

^I^IC This must be the ^B eligible rent ^B (may not be same as the amount of benefit)
 Eligible rent = after deductions.

1.00..1000.00

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: RentDoc = HBStmt
AND: EligAmt = NONRESPONSE

HMissVar := (HMissVar + 1)

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: RentDoc = HBStmt
AND: EligAmt > 0

EligPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: RentDoc = HBStmt
AND: EligAmt > 0

EligPd

^I QRenting^I

^N

What period does that cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: RentDoc = HBStmt
 AND: EligAmt > 0
 AND: EligPd = Note

EligPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: HBenefit = Yes
 AND: RentDoc = HBStmt
 AND: EligAmt > 0
 AND: Edit = Yes
 EligPd <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QRenting.Weekly()**Procedure Call**

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[1] := 1

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[2] := 2

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[3] := 3

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[4] := 4

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[5] := 4.333

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[7] := 8.67

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[8] := 6.5

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: HBenefit = Yes
  AND: RentDoc = HBStmt
  AND: EligAmt > 0

```

```

PdConW[9] := 5.78

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentDoc = HBStmt
    AND: EligAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentDoc = HBStmt
    AND: EligAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentDoc = HBStmt
    AND: EligAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentDoc = HBStmt
    AND: EligAmt > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentDoc = HBStmt
    AND: EligAmt > 0
    AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
    AND: HBenefit = Yes
    AND: RentDoc = HBStmt
    AND: EligAmt > 0
    AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QRenting (continued)

Questions about renters

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: RentDoc = HBStmt
AND: EligAmt > 0
AND: EligPd IN [OneWeek .. Year]
AND: LWeekly1 >= 0.01

EligWkly := LWeekly1

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = Yes
AND: RentDoc = HBStmt
AND: EligAmt > 0
AND: EligPd IN [OneWeek .. Year]
AND: LWeekly1 >= 0.01
AND: Edit = No
(EligWkly < 200) AND INVOLVING(EligPd,EligAmt)

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: HBenefit = No

HBenWait

^I QRenting^I

^N

Are you awaiting the outcome of a claim for Housing Benefit (or Local Housing Allowance) - that is, either Rent Rebate or Rent Allowance?

- (1) Yes
- (2) No

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)

WSInc

^I QRenting^I

^N

Were water or sewerage charges ^rate included in the rent which you mentioned?^N

^I^Consult_the_document

- (1) Both water & sewerage
- (2) Water only
- (3) Sewerage only
- (4) Neither

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: WSInc = Both
```

```
COMBINED_AMOUNT := 'Enter combined amount.'
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: WSInc = Water
```

```
water_sewerage := 'water'
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: WSInc = Sewer
```

```
water_sewerage := 'sewerage'
```

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (WSInc = Sewer)
```

```
water_sewerage := 'water/sewerage'
```

FRS1004B.QRenting.PdTxt1()**Procedure Call**

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: RentPd IN [OneWeek .. LessWeek]
AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]

```

LInThat := ' in that '

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: RentPd IN [OneWeek .. LessWeek]
AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
AND: PPeriod = OneWeek

```

PPdTxt := 'one week period'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: RentPd IN [OneWeek .. LessWeek]
AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
AND: PPeriod = TwoWeek

```

PPdTxt := 'two week period'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: RentPd IN [OneWeek .. LessWeek]
AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
AND: PPeriod = ThrWeek

```

PPdTxt := 'three week period'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: RentPd IN [OneWeek .. LessWeek]
AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
AND: PPeriod = Fourweek

```

PPdTxt := 'four week period'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: RentPd IN [OneWeek .. LessWeek]
AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
AND: PPeriod = Month

```

PPdTxt := 'calendar month'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
  AND: PPeriod = ThrMonth

```

PPdTxt := 'three month period'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
  AND: PPeriod = SixMonth

```

PPdTxt := 'six month period'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [OneWeek .. Month, ThrMonth .. Year]
  AND: PPeriod = Year

```

PPdTxt := 'year'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [TwoMonth .. TenYear]

```

LInThat := ' in those '

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [TwoMonth .. TenYear]
  AND: PPeriod = TwoMonth

```

PPdTxt := 'two calendar months'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [TwoMonth .. TenYear]
  AND: PPeriod = EighYear

```

PPdTxt := 'eight payments'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [TwoMonth .. TenYear]
  AND: PPeriod = NineYear

```

PPdTxt := 'nine payments'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod IN [TwoMonth .. TenYear]
  AND: PPeriod = TenYear

```

PPdTxt := 'ten payments'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod = LessWeek

```

LInThat := ' in that '

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: PPeriod = LessWeek

```

PPdTxt := 'week'

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: NOT (PPeriod = LessWeek)

```

LInThat := ''

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]
  AND: NOT (PPeriod = LessWeek)

```

PPdTxt := ''

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd IN [OneWeek .. LessWeek]

```

PPdTxt := (LInThat + PPdTxt)

FRS1004B.QRenting (continued)

Questions about renters

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd IN [OneWeek .. LessWeek])

```

```

in_that_period := ''

```

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]

```

WSIncAmt

^I QRenting^I

^N

How much was included for ^water_sewerage ^in_that_period?^N

^I^IC ^COMBINED_AMOUNT ^Consult_the_document

1.00..100.00

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: Rent >= 0
  NOT (WSIncAmt > Rent)

```

^I

The amount included in rent for water/sewerage is greater than the rent!

```

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: WSIncAmt > 0

```

RentPx2

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: WSIncAmt > 0

RentPd2

^I QRenting^I

^N

How long did this cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: WSIncAmt > 0
AND: RentPd2 = Note

RentPx2

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QRenting.Weekly()**Procedure Call**

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[1] := 1

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[2] := 2

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[3] := 3

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[4] := 4

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[5] := 4.333

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[7] := 8.67

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: RentPd2 = RESPONSE
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QRenting.Weekly()**Procedure Call**

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[1] := 1

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[2] := 2

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[3] := 3

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[4] := 4

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[5] := 4.333

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[7] := 8.67

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: NOT (RentPd2 = RESPONSE)

```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
  AND: (Scotland <> Yes) AND (NatCen <> NI)
  AND: WSInc IN [Both, Water, Sewer]
  AND: NOT (RentPd2 = RESPONSE)
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QRenting (continued)

Questions about renters

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: (WSIncAmt > 0) AND ((RentPd IN [OneWeek .. Year]) OR (RentPd2 IN
[OneWeek .. Year]))
AND: LWeekly1 >= 1

```

WSIWkly := LWeekly1

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: (WSIncAmt > 0) AND ((RentPd IN [OneWeek .. Year]) OR (RentPd2 IN
[OneWeek .. Year]))
AND: LWeekly1 >= 1
AND: Edit = No
(WSIWkly < 20) AND INVOLVING(WSIncAmt)

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: (Scotland <> Yes) AND (NatCen <> NI)
AND: WSInc IN [Both, Water, Sewer]
AND: WSIncAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: Rent <> 0

```

SerInc

^I QRenting^I

^I^IS B10^I

^N

Does the rent which you mentioned include any of the services shown on this card?^N

^I^Consult_the_document^I

^I^IC Code all that apply.

SET [5] OF

- (1) Heating
- (2) Lighting
- (3) Hot water
- (4) Fuel for cooking
- (5) TV licence fees
- (6) None of these services

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) AND Rent <> EMPTY
AND: Rent <> 0
AND: None IN SerInc
SerInc.CARDINAL = 1

^I

'None of these' is an exclusive code for this question.

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: (PTenure IN [Rents, Part]) OR ((PTenure IN [RentFree, Squatting]) AND (AccJob <> Yes))

AccNonHH

^I QRenting^I

^N

(Apart from Housing Benefit (or Local Housing Allowance)) does anyone outside your household pay any rent on this accommodation on your behalf?^N

^I^IC Exclude Housing Benefit (or Local Housing Allowance) - ie. Rent Rebate or Rent Allowance.

(1) Yes

(2) No

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccNonHH = Yes

AccPay

^I QRenting^I

^N

Who is that?^N

^I^IC Code all that apply.^I

SET [5] OF

(1) ^GOVSSA

(2) Employer

(3) Other organisation

(4) Friend or relative

(5) Other

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccNonHH = Yes
NOT (IN (GOV, AccPay))

^I

Are you sure? ^GOV1 only ever pay^B arrears^B of rent. Double-check, that respondent is not thinking of Housing Benefit (or Local Housing Allowance).

If genuine arrears, suppress this warning.

FRS1004B.QRenting.QAccPay[]

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

Payer[1] := GOV1

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

Payer[2] := 'employer'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

Payer[3] := 'other organisation'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

Payer[4] := 'relative or friend'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

Payer[5] := '

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

LRent := PRent

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

RentSeq := PSeq

```

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay

```

```

AccPay := PSeq

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: AccNonHH = Yes
 AND: In loop FOR Index := 1 TO 5
 AND: Index IN AccPay

AccAmt

^I QRenting^I

^N

How much rent did the ^Payer[AccPay] pay for you last time?

0.01..999997.00

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: AccNonHH = Yes
 AND: In loop FOR Index := 1 TO 5
 AND: Index IN AccPay
 AND: AccAmt = NONRESPONSE

HMissVar := (HMissVar + 1)

RECORD IF: QAccomdat.Tenure IN [Part .. Squatting]
 AND: AccNonHH = Yes
 AND: In loop FOR Index := 1 TO 5
 AND: Index IN AccPay
 AND: AccAmt > 0

AccPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccNonHH = Yes
AND: In loop FOR Index := 1 TO 5
AND: Index IN AccPay
AND: AccAmt > 0

AccPd

^I QRenting^I

^N

How long did that cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccNonHH = Yes
AND: In loop FOR Index := 1 TO 5
AND: Index IN AccPay
AND: AccAmt > 0
AND: AccPd = Note

AccPx

^I QRenting^I

^I^IC ^Pd97Txt

OPEN

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: AccNonHH = Yes
AND: In loop FOR Index := 1 TO 5
AND: Index IN AccPay
AND: AccAmt > 0
AND: Edit = Yes
AccPd <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QRenting.QAccPay[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QRenting.QAccPay[] (continued)

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
  AND: AccPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01

```

AccWkly := LWeekly

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: AccAmt > 0
  AND: AccPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01
  AND: Edit = No
  (AccWkly < 151) AND INVOLVING(AccPd,AccAmt)

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

ASK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: PRent > 0

```

AccChk

^I QRenting^I

^N

Can I just check, is the amount of £^LRent for rent, that you mentioned earlier, BEFORE or AFTER deducting this payment?

- (1) Before
- (2) After

FRS1004B.QRenting (continued)

Questions about renters

```

COMPUTE IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: ((QAccPay[Index].AccWkly = RESPONSE) AND (RentWkly = RESPONSE)) AND
    (QAccPay[Index].AccChk = Befor)

```

```

AccC := (AccC + QAccPay[Index].AccWkly)

```

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  AND: Index IN AccPay
  AND: ((QAccPay[Index].AccWkly = RESPONSE) AND (RentWkly = RESPONSE)) AND
    (QAccPay[Index].AccChk = Befor)
  (AccC <= RentWkly) AND
  INVOLVING (QAccPay[Index].AccPd, QAccPay[Index].AccAmt)

```

^I

The amount recorded for help with your rent is greater than the rent recorded.

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  AND: AccNonHH = Yes
  AND: In loop FOR Index := 1 TO 5
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
  RESERVECHECK

```

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Edit = Yes
AND: Rent = NONRESPONSE OR RentPd = NONRESPONSE
NOT(IN(Landlord, [???])) **AND**
INVOLVING(QAccomdat.Rooms, QAccomdat.TypeAcc)

^I

Missing information for rent amount or period.
Follow edit instructions for 'Rent'

CHECK IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: Edit = Yes
AND: HBenAmt = NONRESPONSE OR (HBenPd = NONRESPONSE **AND** (HBenefit = Yes))
NOT(IN(Landlord, [???]))

^I

Missing information FOR Housing Benefit amount and/or period.
Follow edit instructions for 'Housing Benefit'

FRS1004B (continued)**FAMILY RESOURCES SURVEY 2010-2011**

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
AND: QAccomdat.Tenure = RentFree
IN(QRenting.Landlord, [??])

```

^I

It is very unusual for Local Authority or Housing Association tenants to be living rent-free. Please check with respondent. Change 'Tenure' to renting if 100% Housing Benefit received, or somebody else pays the rent.

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Part .. Squatting]
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: In loop FOR Loop1 := 1 TO HHSize
AND: Loop1 IN QHholder.HHldr

```

```

Oldest := (MAX(DMAge[Loop1], Oldest))

```

FRS1004B.QOwner1

Questions about mortgages

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy = StillM))

BuyYear

^I QOwner1^I

^I

This should be the year^B this^B property was bought. Even if the respondent states that the current mortgage was 'carried over' from a previous property, enter the purchase date for^B this^B property - not the previous one.

1900..2097

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy = StillM))
AND: (BuyYear = RESPONSE) AND (POldest > 0)
BuyYear >= (DLYear.YEAR - POldest)

^I

This is before the date of birth of the oldest householder. Please check your figures.

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy = StillM))
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy = StillM))
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy = StillM))
RESERVECHECK

RESERVECHECK

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy = StillM))
RESERVECHECK

RESERVECHECK

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  RESERVECHECK

```

```

RESERVECHECK

```

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  AND: BuyYear = RESPONSE
  BuyYear <= LYear

```

```

^I
You've entered a future date!

```

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  AND: QDataBag.SampMnth IN [4 .. 12]
  BuyYear <> LYear

```

```

^I
You've entered a future date!

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))

```

```

YearLive := ORD(QAccomdat.YearLive)

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])
  AND: YearLive = 4

```

```

YearLive := 5

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])
  AND: YearLive = 5

```

```

YearLive := 10

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])
  AND: YearLive = 6

```

```

YearLive := 20

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
  StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

```

Time[1] := '12 months'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

Time[2] := '2 years'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

Time[3] := '3 years'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

Time[5] := '5 years'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

Time[10] := '10 years'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

Time[20] := '20 years'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])

```

MorgYear := (QSignIn.StartDat.YEAR - BuyYear)

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
    StillM))
  AND: (BuyYear = RESPONSE) AND (YearLive IN [1 .. 6])
  (YearLive >= MorgYear) AND INVOLVING(QAccomdat.YearLive,BuyYear)

```

^I

The respondent has lived here for less than ^Time[YearLive], but the^B mortgage^B started in ^BuyYear - ^MorgYear years ago. Please check that BuyYear is when the mortgage on THIS PROPERTY was taken out. (If so, suppress & continue)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
      AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
      StillM))

```

PurcLoan

^I QOwner1^I

^N

Can I just check, did you take out one loan to purchase this accommodation, or more than one?

(1) One

(2) Two (or more) loans for purchase

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
      AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
      StillM))
      AND: PTenure = Part

```

your_share_in := ' your share in'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
      AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
      StillM))

```

PurcAmt

^I QOwner1^I

^N

What was the purchase price of^your_share_in your house/flat?

-99999999.99..99999999.99

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
      AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
      StillM))
      AND: PurcAmt = RESPONSE
      PurcAmt >= 0

```

^I^IC Please enter a positive amount (>=0)

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
      AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
      StillM))
      AND: Edit = No
      PurcAmt < 500000

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
      AND: (PTenure = Mortgage) OR ((PTenure = Part) AND (QAccomdat.SOBuy =
      StillM))
      AND: PurcAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

ASK IF: *QAccomdat.Tenure IN [Outright .. Part]*
AND: *(PTenure = Outright) OR ((PTenure = Part) AND (QAccomdat.SOBuy = Paid))*

OthMort3

^I QOwner1^I

^N

May I just check, are you currently using ^B this house/flat ^B as security for a mortgage or loan of any kind?

- (1) Yes
- (2) No

FRS1004B.QOwner1.QMortgage.M[]

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq IN [1 .. 2]
  AND: PPTenure = Part

```

```

to_buy_this_house := ' to buy your share in this house/flat'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq IN [1 .. 2]
  AND: NOT (PPTenure = Part)

```

```

to_buy_this_house := ' to buy this house/flat'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq IN [1 .. 2]

```

```

fill := (' , in ' + STR(PBuyYear))

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq = 3

```

```

to_buy_this_house := ' for essential repairs'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq = 1

```

```

MORTGAGE := (B + ' main ' + B + ' mortgage
@|for ' + 'the purchase of this accommodation.')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq = 1

```

```

INSTRUC := ('

```

```

@| (Questions about any other, ' + 'separate mortgage will
follow.)')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq = 2

```

```

MORTGAGE := (B + ' second ' + B + ' mortgage
@|for ' + 'the purchase of this accommodation.')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq = 2

```

```

INSTRUC := ('

```

```

@| (Questions about any other, ' + 'separate mortgage will follow.)
')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: PSeq = 3

```

```

MORTGAGE := 'loan for essential repairs'

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))

```

IntroM

^I QOwner1^I

^I

@|The next questions are about the ^MORTGAGE ^INSTRUC

(1) Press <Enter> to continue.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))

```

MortSeq := PSeq

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: PSeq = 2

```

Loan2Y

^I QOwner1^I

^I^IC Check: Do they still have this other mortgage for purchase? (If now repaid, use code 2)

- (1) Yes, still have this mortgage
 - (2) No, mortgage has been repaid
-

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PSeq = 3

```

LoanYrRs

^I QOwner1^I

^I^IC You have entered that the respondent took out a second loan on this property before they purchased it. Please check your answer. You MUST make a note if you suppress this check.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PSeq = 3

```

LoanYrEx

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PSeq = 3

```

LoanYear

^I QOwner1^I

^N

In which year did you take out this mortgage or loan?

1900..2097

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PSeq = 3
AND: LoanYear = RESPONSE
LoanYear <= LYear

```

^I

You've entered a future date!

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PSeq = 3
AND: QDataBag.SampMnth IN [4 .. 12]
LoanYear <> LYear

```

^I

You've entered a future date!

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PSeq = 3
  AND: (Edit <> Yes) AND ((LoanYear > 0) AND (BuyYear > 0))
  BuyYear <= LoanYear

  BuyYear <= LoanYear

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PSeq = 3
  AND: (Edit <> Yes) AND ((LoanYear > 0) AND (BuyYear > 0))
  AND: (LoanYrRs = Suppressed) OR LoanYrEx <> EMPTY

```

LoanYrEx

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid

LPurcAmt := PPurcAmt

```

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

BorAmtRs

^I QOwner1^I

^I^IC The amount borrowed is more than the purchase price - this is very unusual. Please check your figures and, if necessary, explain in a Note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

BorAmtEx

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

BorrAmt

^I QOwner1^I

^I

This should be the^B original^B amount of this mortgage, as taken out when the property was purchased (in 'BuyYear').

^B

Properties/mortgages partly for business^B: If the amount borrowed includes the purchase of non-domestic accommodation or land, eg. a farm, a shop with flat above, try to obtain purchase and mortgage details for the domestic element only.

-99999999.99..99999999.99

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: BorrAmt = RESPONSE
BorrAmt >= 0

```

^I^IC Please enter a positive amount. Negative amounts (eg. -10) are not allowed.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: ((Edit <> Yes) AND (BorrAmt > 0)) AND (LPurcAmt > 0)
BorrAmt <= LPurcAmt

```

BorrAmt <= LPurcAmt

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: ((Edit <> Yes) AND (BorrAmt > 0)) AND (LPurcAmt > 0)
AND: (BorAmtRs = Suppressed) OR BorAmtEx <> EMPTY

```

BorAmtEx

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: Edit = No
AND: BorrAmt = RESPONSE
BorrAmt < 250000

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: BorrAmt = RESPONSE
  BorrAmt >= 500

```

^I

That seems very low - please check your figures.

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (PSeq = 1) AND (BorrAmt = DONTKNOW)

```

BorAmtDK

^I QOwner1^I

^I^IC Is this 'Don't know' because the^B original^B mortgage was to buy domestic accommodation^B and^B for business purposes, and you cannot get a separate figure for the^B domestic^B part?

- (1) Yes (Please give full details in a Note)
- (2) No

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: BorrAmt = REFUSAL

```

HMissVar := (HMissVar + 1)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((PSeq = 1) AND (BorrAmt = DONTKNOW)) AND (BorAmtDK <> Yes)) OR ((PSeq
  <> 1) AND (BorrAmt = DONTKNOW))

```

HMissVar := (HMissVar + 1)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

RMort

^I QOwner1^I

^I

'Re-mortgage': a new mortgage is taken out and is used to REPAY an existing mortgage on a property. Typically this happens when a new lender offers better terms, eg. a lower rate of interest. The new loan can be bigger than the old one.

'Further advance or top-up': the loan is EXTENDED (eg. from £30,000 to £40,000). But, if there are SEPARATE payments to cover the new sum borrowed, this counts as a separate loan - NOT a further advance/top-up. Separate loans are dealt with later, at 'OthMort1'.

- (1) Yes
- (2) No

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes

```

RMortYr

^I QOwner1^I

^N

In which year did you take out the most recent re-mortgage/further advance?

1900..2097

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes
AND: RMortYr = RESPONSE
RMortYr <= LYear

```

^I

You've entered a future date!

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes
AND: QDataBag.SampMnth IN [4 .. 12]
RMortYr <> LYear

^I

You've entered a future date!

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes
AND: (RMortYr = RESPONSE) AND (PBuyYear = RESPONSE)
RMortYr >= PBuyYear

^I

The first mortgage was taken out in ^PBuyYear, so the re-mortgage can't have been taken out before that.
 Please amend your answers.

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes

RMAMt

^I QOwner1^I

^N

What was the total amount of the mortgage, after re-mortgaging/taking out the further advance?^N

^I^IC Total should be after all re-mortgages and further advances.

-99999999.99..99999999.99

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes
AND: RMAMt = RESPONSE
RMAMt >= 0

^I^IC Please enter a positive amount. Negative amounts (eg. -10) are not allowed.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes
AND: Edit = No
RMAmt < 250000

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes
AND: RMAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes

```

RMPur

^I QOwner1^I

^I^IS B5^I

^N

Which of these items best describe the reasons why you took out a re-mortgage/ further advance?

^Any_others?^N

^I^IC Code all that apply.

SET [8] OF

- (1) To make improvements or extensions to this property
- (2) To help purchase a major item like a car, boat, caravan or second home
- (3) To get a better, or fixed, interest rate
- (4) In connection with a business
- (5) To buy out another person's share in the property
- (6) For essential repairs to make the property fit for occupation
- (7) To move to a more flexible mortgage
- (8) Some other purpose (SPECIFY IN A NOTE.)

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid

MortTyEx

^I QOwner1^I

^I^MortTTxt

^SuppTxt

OPEN

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid

MortType

^I QOwner1^I

^I^IS B6^I

^N

Looking at this card, please tell me which of these options describe your mortgage or loan?^N

^I^IC If necessary add 'With a repayment mortgage, by repaying the original loan we mean the original capital sum borrowed.'

From April 2008, all PEPs automatically become stocks and shares ISAs. Please record any reported PEPs as ISAs.

- (1) an ENDOWMENT mortgage
(where your mortgage payments cover interest only)
- (2) a REPAYMENT mortgage
(where your mortgage payments cover interest and part of the original loan)
- (3) a PENSION mortgage
(where your mortgage payments cover interest only)
- (4) a Unit Trust or ISA mortgage
- (5) both an endowment (or other interest only) AND a repayment mortgage
- (6) an interest only mortgage with more than one linked investment
(e.g. pension and unit trust, endowment and ISA)
- (7) an interest only mortgage with NO linked investment
(e.g. NO endowment, pension or ISA)
- (8) or another type (not listed above)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: MortType = NONRESPONSE

```

MortTTxt := KeyTxt

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: MortType = NONRESPONSE
  ERROR

```

^I
^MortTTxt

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: MortType = NONRESPONSE

```

MortTyEx

^I QOwner1^I

^I^MortTTxt

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: MortType = Other

```

**MortTTxt := ('Interviewer: This is a rather unusual sort of ' +
'mortgage. Are you sure that it is not covered ' + 'by one of the
codes above? If it is not, please ' + 'record the title in a note
and answer follow-up ' + 'questions as fully as possible.')**

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: MortType = Other
  ERROR

^I
^MortTTxt

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: Edit = No
  AND: MortType = Other

```

MortTyEx

^I QOwner1^I

^I^MortTTxt

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = Endow

```

this_kind_of := 'an endowment'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = Repay

```

this_kind_of := 'a repayment'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = Pension

```

```

this_kind_of := 'a pension'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = UTISA

```

```

this_kind_of := 'a Unit Trust or ISA'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = EndRep

```

```

this_kind_of := 'an endowment & repayment'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [IntLink, IntNoLnk]

```

```

this_kind_of := 'an interest only'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT (MortType IN [IntLink, IntNoLnk])

```

```

this_kind_of := 'this kind of'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: MortType IN [Endow, EndRep]

```

```

Is_the := 'Apart from the endowment I mentioned earlier is the'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: NOT (MortType IN [Endow, EndRep])

```

Is_the := 'Is the'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]

```

EndwPrin

^I QOwner1^I

^I^IS B7^I

^N

^Is_the repayment of the original loan covered by any of the things on this card?^N

^I^IC Code all that apply.

From April 2008, all PEPs automatically become stocks and shares ISAs. Please record any reported PEPs as ISAs.

SET [4] OF

- (1) Current payments into a Pension Plan (pension mortgage)
- (2) Current payments into an ISA
- (3) Current payments into a Unit Trust or Investment Trust scheme
- (4) Current payments into any^B other^B savings/investment scheme
- (5) Proceeds of sale from existing house only
- (6) None of the above.

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: None IN EndwPrin
  EndwPrin.CARDINAL = 1

```

^I

None is an exclusive code for this question.

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]
RESERVECHECK

RESERVECHECK

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]

EndwP1Ex

^I QOwner1^I

^I

Please leave a note to describe how the respondent will be repaying their mortgage.

^SuppTxt

OPEN

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]

EndwP2Ex

^I QOwner1^I

^I^EndP2Txt

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]

```

```

EndwPTxt := ('This method of capital repayment does not match the
' + 'type of mortgage recorded earlier at MortType. ' + '
Please resolve, or make a note.')

```

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: None IN EndwPrin
  AND: NOT (MortType IN [Endow, EndRep])
  ERROR AND INVOLVING(MortType,EndwPrin)

```

^I

Please leave a note to describe how the respondent will be repaying their mortgage.

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: None IN EndwPrin
  AND: NOT (MortType IN [Endow, EndRep])

```

EndwP1Ex

^I QOwner1^I

^I

Please leave a note to describe how the respondent will be repaying their mortgage.

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Endow
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

EndP2Txt := EndwPTxt

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Endow
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)
  ERROR AND INVOLVING(MortType,EndwPrin)

```

^I
^EndP2Txt

```

Ask IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Endow
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

EndwP2Ex

^I QOwner1^I

^I^EndP2Txt

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Pension
  AND: (((ISA IN EndwPrin) OR (UnitT IN EndwPrin)) OR (OthSch IN EndwPrin))
  OR (None IN EndwPrin)

```

EndP2Txt := EndwPTxt

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Pension
  AND: (((ISA IN EndwPrin) OR (UnitT IN EndwPrin)) OR (OthSch IN EndwPrin))
  OR (None IN EndwPrin)
  ERROR AND INVOLVING(MortType,EndwPrin)

```

^I
^EndP2Txt

```

Ask IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Pension
  AND: (((ISA IN EndwPrin) OR (UnitT IN EndwPrin)) OR (OthSch IN EndwPrin))
  OR (None IN EndwPrin)

```

EndwP2Ex

^I QOwner1^I

^I^EndP2Txt

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = UTISA
  AND: ((Pension IN EndwPrin) OR (OthSch IN EndwPrin)) OR (None IN EndwPrin)

```

EndP2Txt := EndwP2Txt

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = UTISA
  AND: ((Pension IN EndwPrin) OR (OthSch IN EndwPrin)) OR (None IN EndwPrin)
ERROR AND INVOLVING(MortType,EndwPrin)

```

^I
^EndP2Txt

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = UTISA
  AND: ((Pension IN EndwPrin) OR (OthSch IN EndwPrin)) OR (None IN EndwPrin)

```

EndwP2Ex

^I QOwner1^I

^I^EndP2Txt

^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = IntNoLnk
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

```

EndP2Txt := ('You described your mortgage as an interest ' + 'only
with NO linked investments, can I just ' + 'check is this
savings/investment scheme linked ' + 'to your mortgage? Please
amend the answer at ' + 'MortType as appropriate.')

```

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = IntNoLnk
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)
  ERROR AND INVOLVING(EndwPrin,MortType)

```

```

^I
^IEndP2Txt

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = IntNoLnk
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

EndwP2Ex

```

^I QOwner1^I

```

```

^I^IEndP2Txt

```

```

^ISuppTxt

```

```

OPEN

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Other
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

```

EndP2Txt := ('If there is a pension, ISA or other ' +
'savings/investment scheme to cover the ' + 'repayment of the
original loan, please ' + 'amend the answer at MortType as
appropriate.')

```

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Other
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)
  ERROR AND INVOLVING(EndwPrin,MortType)

```

```

^I
^IEndP2Txt

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
  AND: MortType = Other
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

EndwP2Ex

```

^I QOwner1^I

```

```

^I^IEndP2Txt

```

```

^ISuppTxt

```

```

OPEN

```

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Edit = No
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: Pension IN EndwPrin

```

DMAEndwPrin := Pension

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

payment1 := 'contribution to the'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

payment2 := '(pension plan/ISA/Unit Trust)'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = IntLink

```

payment1 := 'premium/payment on the'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = IntLink

```

**payment2 := ('endowment policy / pension / unit trust /' + 'ISA
/ investment trust / other policy')**

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT (MortType = IntLink)

```

payment1 := 'premium on the'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT (MortType = IntLink)

```

payment2 := 'endowment policy'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType = IntLink

```

**policy := ('policy / pension / unit trust / ISA / investment ' +
'trust / other policy')**

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT (MortType = IntLink)

```

policy := 'policy'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: MortType IN [Endow, EndRep]

```

Are := '(Can I just check), are'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]
  AND: NOT (MortType IN [Endow, EndRep])

```

Are := 'Are'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortType IN [Endow, Pension .. Other]

```

MenPol

^I QOwner1^I

^N

^Are there any endowment policies covering the repayment of this mortgage or loan?

- (1) Yes
- (2) No

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]
AND: Edit = No
AND: MortType IN [Endow, EndRep]
((MenPol = Yes) OR (MortType = Other)) AND INVOLVING(MortType,MenPol)

```

^I

Normally there^B would^B be an endowment policy, with an endowment mortgage: please check.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]
AND: MenPol = Yes
NOT(IN(MortType,[???,???])) AND INVOLVING(MortType,MenPol)

```

^I

You described your mortgage as an interest only with NO linked investments or another type of mortgage, can I just check is this savings/investment scheme linked to your mortgage? Please amend the answer at MortType as appropriate.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]
AND: MenPol = Yes
(MortType <> Other) AND INVOLVING(MortType,MenPol)

```

^I

If there is an endowment, pension, ISA or other arrangement to cover the repayment of the original loan, please amend the answer at MortType as appropriate.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MortType IN [Endow, Pension .. Other]
AND: MenPol = No
(MortType <> Other) AND INVOLVING(MortType,MenPol)

```

^I

It is unusual for there to be no policies to cover the repayment of the loan. Please check. If original loan is included in monthly payments, please amend your answer at MortType to a Repayment (code 2) OR Endowment and Repayment (code 5) mortgage.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = No

```

```

How_Long := ('How long is the term of your mortgage. By this we
mean ' + 'the agreed term?')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT (RMort = No)

```

**How_Long := ('How long is the term of your mortgage. By this we
' + 'mean the agreed term since you have remortgaged or ' + 'extended
the original loan?')**

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid

```

MorFlc

^I QOwner1^I

^N

All-in-one accounts are a new type of flexible mortgage which allow a person to link together accounts - for example, a current account, a savings account and a mortgage (as well as other types of loans). There are two types of all-in-one account: current account mortgages and offset mortgages.^N

^I^IC Examples include the One account (RBoS), a Woolwich Open Plan or some other all-in-one account.^I ^N

Is your mortgage an all-in-one account?

- (1) Yes
 - (2) No
-

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MorFlc = Yes

```

MorAll

^I QOwner1^I

^I^IS B8 AND B9^I

^N

Is your all-in-one account mortgage, a current account mortgage or an offset mortgage?

- (1) Current account mortgage
 - (2) Offset mortgage
-

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

MortEnd

^I QOwner1^I

^N

^How_Long^N

^I^IC If remortgaged agreed term is from the point of remortgaging.

1..60

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = No

```

remortgage := 'mortgage was taken out'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT (RMort = No)

```

remortgage := 'last re-mortgage'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
MortEnd <= 40

```

^I

Are you sure? The end-date would not normally be more than 40 years after the ^remortgage. Please check your figures.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: MorAll = Current

```

```

What_amount := ('What is the amount of the negative balance or '
+ 'overdraft on your current account mortgage')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT (MorAll = Current)

```

```

What_amount := ('What is the amount still outstanding on your '
+ 'mortgage/loan from this source - that is, how ' + 'much do you
still have to pay off')

```

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

MortL1Rs

^I QOwner1^I

^N

For ^this_kind_of mortgage, the amount outstanding should equal the ^amount ^borrowed. Please check and amend, else explain in a Note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

MortL1Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid

```

MortL2Rs

^I QOwner1^I

^N

For ^this_kind_of mortgage, the amount outstanding should be less than the ^amount ^borrowed. Please check and amend, else explain in a Note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid

```

MortL2Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid

```

MortLeft

^I QOwner1^I

^N

^What_amount?

-99999999.99..99999999.99

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortLeft = RESPONSE
  MortLeft >= 0

```

^I^IC Please enter a positive amount. Negative amounts (eg. -10) are not allowed.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: MortLeft = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: RMort = Yes

```

borrowed := 'of the re-mortgage'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: RMort = Yes

```

amount := 'total amount'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: RMort = Yes
  (ABS(MortLeft - RMAmt) <= 50) AND INVOLVING(MortLeft)

  (ABS(MortLeft - RMAmt) <= 50) AND INVOLVING(MortLeft)

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: RMort = Yes
  AND: (MortL1Rs = Suppressed) OR MortL1Ex <> EMPTY

```

MortL1Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: NOT (RMort = Yes)

```

borrowed := 'originally borrowed'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: NOT (RMort = Yes)

```

amount := 'amount'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: NOT (RMort = Yes)
  (ABS (MortLeft - BorrAmt) <= 50) AND INVOLVING (MortLeft)

```

(ABS (MortLeft - BorrAmt) <= 50) AND INVOLVING (MortLeft)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Endow, Pension, UTISA, Other]
  AND: NOT (RMort = Yes)
  AND: (MortL1Rs = Suppressed) OR MortL1Ex <> EMPTY

```

MortL1Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: RMort = Yes

```

borrowed := 'of re-mortgage'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: RMort = Yes

```

amount := 'amount'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: RMort = Yes
  (MortLeft < RMamt) AND INVOLVING(MortLeft)

(MortLeft < RMamt) AND INVOLVING(MortLeft)

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: RMort = Yes
  AND: (MortL2Rs = Suppressed) OR MortL2Ex <> EMPTY

```

MortL2Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: NOT (RMort = Yes)

```

borrowed := 'originally borrowed'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: NOT (RMort = Yes)

```

amount := 'amount'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: NOT (RMort = Yes)
  (MortLeft < BorrAmt) AND INVOLVING(MortLeft)

```

(MortLeft < BorrAmt) AND INVOLVING (MortLeft)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [Repay, EndRep]
  AND: NOT (RMort = Yes)
  AND: (MortL2Rs = Suppressed) OR MortL2Ex <> EMPTY

```

MortL2Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: RMort = Yes

```

borrowed := 'of the re-mortgage'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: RMort = Yes

```

amount := 'amount'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: RMort = Yes
  (MortLeft = RMAmt) AND INVOLVING(MortLeft)

(MortLeft = RMAmt) AND INVOLVING(MortLeft)

```

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: RMort = Yes
  AND: (MortL1Rs = Suppressed) OR MortL1Ex <> EMPTY

```

MortL1Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: NOT (RMort = Yes)

```

borrowed := 'originally borrowed'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: NOT (RMort = Yes)

```

amount := 'amount'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: NOT (RMort = Yes)
  (MortLeft = BorrAmt) AND INVOLVING(MortLeft)

```

(MortLeft = BorrAmt) AND INVOLVING (MortLeft)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MortLeft = RESPONSE) AND (BorrAmt = RESPONSE)
  AND: MortType IN [IntLink, IntNoLnk]
  AND: NOT (RMort = Yes)
  AND: (MortL1Rs = Suppressed) OR MortL1Ex <> EMPTY

```

MortL1Ex

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: RMort = Yes

```

taking_out_the_loan := 'you re-mortgaged'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT (RMort = Yes)

```

taking_out_the_loan := 'taking out the original loan'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))

```

MorInPay

^I QOwner1^I

^N

How much was your last payment on this mortgage or loan?

0.00..9999.97

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPay = RESPONSE

```

LastPay := STR(MorInPay,6,2)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPay = DONTKNOW

```

LastPay := '??????'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPay = DONTKNOW

```

HMissVar := (HMissVar + 1)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPay = REFUSAL

```

LastPay := '!!!!!!'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPay = REFUSAL

```

HMissVar := (HMissVar + 1)

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))

```

MorInPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))

MorInPd

^I QOwner1^I

^N

How long did this cover?^N

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: MorInPd = Note

MorInPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: Edit = Yes
MorInPd <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))

MorInUs

^I QOwner1^I

^N

Is this the amount you usually pay each time?

- (1) Yes
- (2) No

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: MorInUs = No

MorUs

^I QOwner1^I

^I

If the last interest payment includes arrears accept the actual amount given but if it was a long time ago make a note using <Ctrl+M> to give the date, or if the amount was greater than normal to cover past arrears. Payments by people outside the household should be included.

0.00..9999.97

```
RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: MorInUs = No
```

MorUPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```
ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: MorInUs = No
```

MorUPd

^I QOwner1^I

^N

How long does this cover?^N

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: MorInUs = No
AND: MorUPd = Note

```

MorUPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: NOT ((MorAll = Current) OR (MortType = Repay))
AND: MorInUs = No
AND: Edit = Yes
MorUPd <> Note

```

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QOwner1.QMortgage.M[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
```

PdConW[5] := 4.333

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[7] := 8.67

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorUs > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QOwner1.QMortgage.M[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
```

PdConW[5] := 4.333

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[7] := 8.67

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: NOT (MorUs > 0)
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QOwner1.QMortgage.M[] (continued)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97

```

MorIWkly := LWeekly

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: MortLeft = RESPONSE

```

EPIntC := ((MorIWkly * 52) / MortLeft) * 100

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: MortLeft = RESPONSE

```

IntFill := ROUND(EPIntC)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: MortLeft = RESPONSE
  AND: EPIntC <= 2

```

higher := 'lower'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: MortLeft = RESPONSE
  AND: EPIntC >= 11

```

higher := 'higher'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: NOT ((MorAll = Current) OR (MortType = Repay))
  AND: MorInPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: MortLeft = RESPONSE
  ((EPIntC > 2) AND (EPIntC < 11)) AND INVOLVING(MorInPd,MorInPay)

```

^[]

The interest payments work out roughly at ^IntFill per cent which is ^higher than most current interest rates available for a mortgage of this size.

If no particular reason for this, please check your answers.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  RESERVECHECK

```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK
```

RESERVECHECK

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
```

MenPolAm0 := Yes

FRS1004B.QOwner1.QMortgage.M[.QEndow[]

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)

```

MortSeq := PPSeq

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)

```

EndowSeq := PCount

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: PMenPol = Yes
  AND: PCount > 1

```

next := 'next'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: PMenPol = Yes
  AND: NOT (PCount > 1)

```

```

next := 'first'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: PMenPol = Yes

```

```

premium_payment := 'premium'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: PMenPol = Yes

```

```

policies_plans := 'endowment policies'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: NOT (PMenPol = Yes)

```

```

premium_payment := 'payment'

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: NOT (PMenPol = Yes)

```

policies_plans := 'savings or investment plans'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)

```

MenPolAm

^I QOwner1^I

^N

How much was your last ^payment1 ^next ^payment2?^N

^I^IC For interest only mortgages include combined interest and endowment payment.

0.00..9999.97

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```
RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: MenPolAm > 0
```

MenPolPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```
ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: MenPolAm > 0
```

MenPolPd

^I QOwner1^I

^N

How long did this cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```
ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: MenPolAm > 0
AND: MenPolPd = Note
```

MenPolPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: MenPolAm > 0
AND: Edit = Yes
MenPolPd <> Note
```

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QOwner1.QMortgage.M[.QEndow[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
```

PdConW[3] := 3

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[4] := 4

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[5] := 4.333

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[7] := 8.67

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QOwner1.QMortgage.M[.QEndow[] (continued)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
  AND: MenPolPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 999.97

```

MenPWkly := LWeekly

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
  (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
  AND: MorAll <> Current
  AND: In loop FOR Count := 1 TO 4
  AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
  AND: MenPolAm > 0
  AND: MenPolPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 999.97
  AND: Edit = No
  (MenPWkly < 100) AND INVOLVING(MenPolPd, MenPolAm)

```

^]

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: (SUBSTRING (PLastPay, 1, 1) <> 0) AND (MenPolAm > 0)

IncInInt

^I QOwner1^I

^N

Is this ^premium_payment included in the amount you mentioned earlier (£^PLastPay)?

- (1) Yes
- (2) No

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: PCount < 4

MPMore

^I QOwner1^I

^N

Can I just check, are there any more savings or investment plans covering the repayment of the mortgage or loan?

- (1) Yes
- (2) No

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(Unitt IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(Unitt IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(Unitt IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(Unitt IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
RESERVECHECK

```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
RESERVECHECK
```

RESERVECHECK

FRS1004B.QOwner1.QMortgage.M[] (continued)

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
(QEndow[Count].MenPWkly <= MorIWkly) AND
INVOLVING (QEndow[Count].MenPolAm)

```

^I

The payment/endowment premium is more than the last mortgage payment at MorInPay. This is very unusual - please check your figures.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: (MorIWkly > 0) AND (QEndow[Count].IncInInt = Yes)
(QEndow[Count].MenPWkly < MorIWkly) AND
INVOLVING (QEndow[Count].MenPolAm, QEndow[Count].MenPolPd)

```

^I

The payment/endowment premium is included in the interest payment of £^LastPay, so it can't exceed this amount. Please check your figures.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
AND: (Count = 1) OR (QEndow[Count - 1].MPMore = Yes)
AND: QEndow[Count].MenPolAm <> 0

```

MenPolAm0 := No

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: MorAll <> Current
AND: In loop FOR Count := 1 TO 4
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: (MortSeq = 1) AND (QEndow[1].MPMore = No)
NOT (QMortgage.M[].MortType[MortSeq] = IntLink) AND
INVOLVING(QEndow[1].MPMore)

```

^I

Are you sure? Earlier the respondent said they had more than 1 savings / investment policy for this mortgage. Check if the respondent has another policy to tell you about. If not you MUST make a note of the circumstances.

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)

```

MPMore

^I QOwner1^I

^N

Are there any more policies/plans covering the repayment of the mortgage or loan?

- (1) Yes
- (2) No

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (((MenPol = Yes) OR (Pension IN EndwPrin)) OR (ISA IN EndwPrin)) OR
(UnitT IN EndwPrin)) OR (OthSch IN EndwPrin)
AND: QEndow[1].MPMore = Yes

```

MPMore := Yes

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (MorAll <> Current) AND (MortType = Repay)

```

IntPrPay

^I QOwner1^I

^N

How much was your last payment on this mortgage or loan?

0.00..9999.97

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: (Edit = No) AND (IntPrPay = RESPONSE)
  NOT(IntPrPay = 0)

```

^I

You have entered that the respondent's last instalment on the mortgage/loan was £0. Please do not enter zero even if they paid nothing last time. Please collect the amount they usually pay or if there is no usual, the contractual or notional amount they would need to pay in order for the mortgage/loan to be paid off in the agreed period.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: IntPrPay = RESPONSE

```

LastPay := STR(IntPrPay,6,2)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: IntPrPay = DONTKNOW

```

LastPay := '??????'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: IntPrPay = DONTKNOW

```

HMissVar := (HMissVar + 1)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: IntPrPay = REFUSAL

```

LastPay := '!!!!!!'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: IntPrPay = REFUSAL

```

HMissVar := (HMissVar + 1)

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)

```

IntPrPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)

```

IntPrPd

^I QOwner1^I

^N

How long did this cover?^N

- (1) One week
 - (2) Two weeks
 - (3) Three weeks
 - (4) Four weeks
 - (5) Calendar month
 - (7) Two Calendar months
 - (8) Eight times a year
 - (9) Nine times a year
 - (10) Ten times a year
 - (13) Three months/13 weeks
 - (26) Six months/26 weeks
 - (52) One Year/12 months/52 weeks
 - (90) Less than one week
 - (95) One off/lump sum
 - (97) None of these ^I(Explain in a note)
-

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (MorAll <> Current) AND (MortType = Repay)
AND: IntPrPd = Note

```

IntPrPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (MorAll <> Current) AND (MortType = Repay)
AND: Edit = Yes
IntPrPd <> Note

```

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QOwner1.QMortgage.M[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
```

PdConW[4] := 4

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[5] := 4.333

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[7] := 8.67

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QOwner1.QMortgage.M[] (continued)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97

```

IntPWkly := LWeekly

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  (IntPWkly < 650) AND INVOLVING(IntPrPd,IntPrPay)

```

^I

Are you sure? That is higher than the amount usually entered here. Confirm that the last payment was this amount and if Yes suppress check.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: (RMAmt = RESPONSE) OR (BorrAmt = RESPONSE)
  AND: RMAmt > 0

```

```

PrintC := (((IntPWkly - (RMAmt / (25 * 52))) * 52) / (RMAmt * 0.62))
* 100)

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: (RMamt = RESPONSE) OR (BorrAmt = RESPONSE)
  AND: BorrAmt > 0

```

```

PrIntC := (((IntPWkly - (BorrAmt / (25 * 52))) * 52) / (BorrAmt
* 0.62)) * 100)

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: (RMamt = RESPONSE) OR (BorrAmt = RESPONSE)
  AND: PrIntC < 3

```

higher := 'lower'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: (MorAll <> Current) AND (MortType = Repay)
  AND: NOT (Edit = Yes)
  AND: IntPrPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 9999.97
  AND: (RMamt = RESPONSE) OR (BorrAmt = RESPONSE)
  AND: PrIntC > 10

```

higher := 'higher'

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: (MorAll <> Current) AND (MortType = Repay)
AND: NOT (Edit = Yes)
AND: IntPrPd IN [OneWeek .. Year]
AND: LWeekly > 0
AND: LWeekly <= 9999.97
AND: (RMAmt = RESPONSE) OR (BorrAmt = RESPONSE)
((PrIntC >= 3) AND (PrIntC <= 10)) AND INVOLVING(IntPrPay)

```

^[]

You have entered an amount that is ^higher than that usually paid for a mortgage of this size. Please check that you have entered the correct payment.

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MenPol = Yes

```

**Apart do := ('Apart from any endowment policies already ' +
'mentioned, do')**

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MenPol = Yes

```

redundancy := ' or redundancy'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MenPol = Yes

```

death := '(NOT USED)'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: NOT (MenPol = Yes)

```

Apart_do := 'Do'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: NOT (MenPol = Yes)

```

redundancy := ', redundancy or death'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: NOT (MenPol = Yes)

```

death := 'Death'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]

```

MortProt

^I QOwner1^I

^I

Only include policies which specifically pay the^B mortgage^B. Do not confuse these with policies that simply pay out money in the event of redundancy or sickness (and could be used to pay for anything).

Information about mortgage protection policies is needed to calculate housing costs. Comparisons of different households' income before and after housing costs are used to monitor Government child and pensioner poverty targets. It is therefore important that mortgage protection policies are recorded on the FRS.

- (1) Yes
- (2) No

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: (MortProt = RESPONSE) AND (MortType <> Endow)
MortProt = Yes

```

^I

Interviewer: for this type of mortgage there is normally a protection policy. Please check - is it included in the last mortgage payment? (If no policy, suppress warning and continue.)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes

```

MPCover

^I QOwner1^I

^N

What is covered by the mortgage protection policy?^N

^I^IC Probe to classify.

Code all that apply.

SET [3] OF

- (1) Sickness/accident
 - (2) Redundancy/loss of employment
 - (3) ^death
-

```

CHECK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: MenPol = Yes
NOT (IN (Dead, MPCover) )

```

^I

This code is not valid for this question.

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: MPCover.CARDINAL > 1

```

MPolNo

^I QOwner1^I

^N

Can I check, is there^B one^B mortgage protection policy, or^B more than^B one?^N

^I^IC Count as separate policy if separate^B payments (premiums)^B are made.

Enter number of policies.

1..3

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: ((IntPrPay > 0) OR (MorInPay > 0)) OR (MenPolAm0 = No)
AND: IntPrPay <> EMPTY OR (MorInPay > 0)

```

```

PCP := ('your last payment on the mortgage/loan (£' + LastPay +
')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: ((IntPrPay > 0) OR (MorInPay > 0)) OR (MenPolAm0 = No)
AND: IntPrPay <> EMPTY OR (MorInPay > 0)
AND: MorInPay > 0
AND: MenPolAm0 = No
AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
OR (OthSch IN EndwPrin)

```

```

PCP := (PCP + ' or in the (pension/ISA/Unit' + ' Trust)
contribution')

```

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: ((IntPrPay > 0) OR (MorInPay > 0)) OR (MenPolAm0 = No)
  AND: IntPrPay <> EMPTY OR (MorInPay > 0)
  AND: MorInPay > 0
  AND: MenPolAm0 = No
  AND: NOT (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN
  EndwPrin)) OR (OthSch IN EndwPrin)

```

PCP := (PCP + ' or in the endowment premium')

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: ((IntPrPay > 0) OR (MorInPay > 0)) OR (MenPolAm0 = No)
  AND: MorInPay <> EMPTY AND (MenPolAm0 = No)
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

PCP := 'the (pension/ISA/Unit Trust) contribution'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: ((IntPrPay > 0) OR (MorInPay > 0)) OR (MenPolAm0 = No)
  AND: MorInPay <> EMPTY AND (MenPolAm0 = No)
  AND: NOT (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN
  EndwPrin)) OR (OthSch IN EndwPrin)

```

PCP := 'the endowment premium'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN EndwPrin))
  OR (OthSch IN EndwPrin)

```

PC := 'pension/ISA/Unit Trust contribution'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: NOT (((Pension IN EndwPrin) OR (ISA IN EndwPrin)) OR (UnitT IN
EndwPrin)) OR (OthSch IN EndwPrin)

```

PC := 'endowment premium'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes

```

Order[1] := 'FIRST'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes

```

Order[2] := 'SECOND'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes

```

Order[3] := 'THIRD'

FRS1004B.QOwner1.QMortgage.M[].QMortProt[]

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
```

LMortSeq := PPSeq

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
```

LPayment_etc := ppayment

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: LPayment_etc =
```

LPayment_etc := 'the mortgage payment you mentioned earlier'

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)

```

IncMPAmt

^I QOwner1^I

^B^I *** ^Order[Count] MORTGAGE PROTECTION POLICY ***^B^I

^I

If the precise amount for the mortgage protection policy cannot be given, please ask the respondent to given an estimate rather than accept DK.

0.00..9997.99

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: IncMPAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: IncMPAmt > 0

```

IncMPPx

^I QOwner1^I

^B^I *** ^Order[Count] MORTGAGE PROTECTION POLICY ***^B^I

I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: IncMPAmt > 0

IncMPPd

^I QOwner1^I

^B^I *** ^Order[Count] MORTGAGE PROTECTION POLICY ***^B^I
^N
How long did this cover?^N

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: IncMPAmt > 0
AND: IncMPPd = Note

IncMPPx

^I QOwner1^I

^B^I *** ^Order[Count] MORTGAGE PROTECTION POLICY ***^B^I

I^IC ^Pd97Ttxt

OPEN

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: IncMPAmt > 0
AND: Edit = Yes
IncMPPd <> Note
```

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QOwner1.QMortgage.M[.QMortProt[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[8] := 6.5

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0

```

PdConW[52] := 52

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])
```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))
```

PWeekly := 0

FRS1004B.QOwner1.QMortgage.M[.QMortProt[] (continued)

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
  AND: IncMPPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 997.99
```

IncMWkly := LWeekly

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: PPTenure IN [Mortgage, Part]
  AND: MortProt = Yes
  AND: In loop FOR Count := 1 TO 3
  AND: (Count = 1) OR (Count <= MPolNo)
  AND: IncMPAmt > 0
  AND: IncMPPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: LWeekly <= 997.99
  (IncMWkly < 30) AND INVOLVING(IncMPPd, IncMPAmt)
```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: (IncMPAmt > 0) OR IncMPAmt = NONRESPONSE
AND: PMorAll <> Current

IncMP

^I QOwner1^I

^B^I *** ^Order[Count] MORTGAGE PROTECTION POLICY ***^B^I

^N

Was this mortgage protection payment included in ^LPayment_etc?

- (1) Yes
- (2) No

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: (IncMPAmt > 0) OR IncMPAmt = NONRESPONSE
AND: PMorAll <> Current
AND: ((IncMP = Yes) AND (SUBSTRING (PLastPay, 1, 1) <> 0)) AND (PMenPolAm0 = No)

IncMIncl

^I QOwner1^I

^B^I *** ^Order[Count] MORTGAGE PROTECTION POLICY ***^B^I

^I

Ask or record.^I

^N

Was it included in the mortgage payment or the ^ppremium?

- (1) mortgage payment
- (2) ^ppremium

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
RESERVECHECK
```

RESERVECHECK

FRS1004B.QOwner1.QMortgage.M[] (continued)

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: IntPWkly = RESPONSE
(QMortProt[Count].IncMWkly <= IntPWkly) AND
INVOLVING(QMortProt[Count].IncMPAmt,QMortProt[Count].IncMPPd,IntPrPay)

```

^I

The mortgage protection premium is more than the last mortgage payment at IntPrPay (£^LastPay). This is very unusual - please check your figures.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: MorAll = Current
QMortProt[Count].IncMPAmt <> DONTKNOW

```

^I^IC Please try not to collect Don't Know at this question. Ask your respondent to be more specific and provide you with their best estimate as to their monthly payment.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
AND: (Count = 1) OR (Count <= MPolNo)
AND: Edit = Yes
QMortProt[Count].IncMPAmt<>NONRESPONSE

```

^I

Missing amount for Mortgage Protection Policy. Note the size of last mortgage payment (£^LastPay), then follow Edit Instructions to fill in IncMPAmt.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: PPTenure IN [Mortgage, Part]
AND: MortProt = Yes
AND: In loop FOR Count := 1 TO 3
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

```

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
RESERVECHECK

RESERVECHECK

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid

OutsMort

^I QOwner1^I

^N

Does anyone from outside the household pay anything towards THIS mortgage/loan on your behalf, on a regular basis?

- (1) Yes
- (2) No

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes

```

QOutsPay

^I QOwner1^I

^N

Who is that?

SET [6] OF

- (1) ^GOV2
- (2) Employer
- (3) Other organisation
- (4) Friend or relative
- (5) Mortgage protection/insurance policy
- (6) Other

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes

```

Payer[1] := GOV1

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes

```

Payer[2] := 'employer'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes

```

Payer[3] := 'other organisation'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes

```

Payer[4] := 'relative or friend'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes

```

Payer[5] := 'policy'

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes

```

Payer[6] := '

FRS1004B.QOwner1.QMortgage.M[.QOutside[

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay

```

MortSeq := PPSeq

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay

```

ContSeq := POutsPay

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay

```

OutsPay := POutsPay

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay

```

OutsAmt

^I QOwner1^I

^N

How much did the ^PPayer pay last time?

0.01..999997.00

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt = NONRESPONSE
```

HMissVar := (HMissVar + 1)

```
RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

OutsPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
AND: OutsAmt > 0

OutsPd

^I QOwner1^I

^N

How long did that cover?^N

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two) AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
AND: OutsAmt > 0
AND: OutsPd = Note

OutsPx

^I QOwner1^I

^I^IC ^Pd97Txt

OPEN

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR (PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
AND: OutsAmt > 0
AND: Edit = Yes
OutsPd <> Note
```

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QOwner1.QMortgage.M[.QOutside[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[8] := 6.5

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[9] := 5.78

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[10] := 5.2

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[13] := 13

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[26] := 26

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
```

PdConW[52] := 52

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])
```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))
```

PWeekly := 0

FRS1004B.QOwner1.QMortgage.M[.QOutside[(continued)

```

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
  AND: OutsPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01
  AND: LWeekly <= 999997

```

OutWkly := LWeekly

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
  AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
  M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  AND: OutsAmt > 0
  AND: OutsPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01
  AND: LWeekly <= 999997
  AND: Edit = No
  (OutWkly < 159) AND INVOLVING(OutsPd,OutsAmt)

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
AND: (OutsAmt <> 0) AND (SUBSTRING (PLastPay, 1, 1) <> 0)
```

OutsIncl

^I QOwner1^I

^N

Was this included in the mortgage payment that you mentioned earlier?

- (1) Yes
- (2) No

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
  AND: PurcAmt <> EMPTY
  AND: In loop FOR ii := 1 TO 3
  AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
  AND: Loan2Y <> Repaid
  AND: OutsMort = Yes
  AND: In loop FOR Count := 1 TO 6
  AND: Count IN QOutsPay
  RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
AND: Count IN QOutsPay
RESERVECHECK
```

RESERVECHECK

FRS1004B.QOwner1.QMortgage.M[] (continued)

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Loan2Y <> Repaid
AND: OutsMort = Yes
AND: In loop FOR Count := 1 TO 6
RESERVECHECK

```

RESERVECHECK

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (PSeq = 1) AND (PPurcLoan = One)

```

OthMort1

^I QOwner1^I

^N

I have already asked you about the loan you had to purchase this house/flat. Apart from that, do you have any other mortgage or loan secured on^B this^B property?

- (1) Yes
- (2) No

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (PSeq = 2) AND (PPurcLoan = Two)

```

OthMort2

^I QOwner1^I

^N

May I just check, are you currently using^B this house/flat^B as security for a mortgage or loan of any other kind?

- (1) Yes
- (2) No

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (OthMort1 = Yes) OR (OthMort2 = Yes)

```

OthPurRs

^I QOwner1^I

^I

This should only apply to loans for purchase. Please resolve, or make a Note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

```

RECORD IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (OthMort1 = Yes) OR (OthMort2 = Yes)

```

OthPurEx

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (OthMort1 = Yes) OR (OthMort2 = Yes)

```

OthPur

^I QOwner1^I

^I^IS B11^I ^N

Which of these items best describe the reasons why you took out the other loan or loans?

^Any_others?^N

^I^IC Code all that apply.

SET [7] OF

- (1) To make improvements or extensions to this property
- (2) To help purchase a major item like a car, boat, caravan or second home
- (3) To get a better, or fixed, interest rate
- (4) In connection with a business
- (5) To buy out another person's share in the property
- (6) For essential repairs to make the property fit for occupation
- (7) Some other purpose (INTERVIEWER: SPECIFY IN A NOTE.)

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (OthMort1 = Yes) OR (OthMort2 = Yes)
NOT (IN (IntrRate, OthPur))

```

NOT (IntrRate IN OthPur)

```

ASK IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (OthMort1 = Yes) OR (OthMort2 = Yes)
AND: (OthPurRs = Suppressed) OR OthPurEx <> EMPTY

```

OthPurEx

^I QOwner1^I

^I^IC ^SuppTxt

OPEN

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Edit = Yes
AND: NOT (MortType IN [Endow, EndRep])
NOT (IN (None, EndwPrin))

```

^I

Editor: Mortgage Capital repaid by 'unknown' method: There should be a note attached. Please re-code into 1-4, IF possible.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Edit = Yes
IntPrPay<>NONRESPONSE AND IntPrPd<>NONRESPONSE

```

^I

Missing amount and/or period for Mortgage Instalment.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Edit = Yes
MorInPay<>NONRESPONSE AND MorInPd<>NONRESPONSE

```

^I

Missing amount and/or period for Mortgage Instalment.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: Edit = Yes
MenPol <> No

```

^I

There are no endowment policies covering the repayment of this mortgage OR loan.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
AND: (RMamt = RESPONSE) AND (BorrAmt = RESPONSE)
BorrAmt <= RMamt

```

^I

The re-mortgage amount would normally be at least as large as the original mortgage. Please check your figures.

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK

```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK
```

RESERVECHECK

```
WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
AND: (((PPPurcLoan IN [One .. Two]) AND (ii = 1)) OR ((PPPurcLoan = Two)
AND (ii = 2))) OR ((ii = 3) AND ((Repairs IN M[1].OthPur) OR (Repairs IN
M[2].OthPur)))
RESERVECHECK
```

RESERVECHECK

FRS1004B.QOwner1.QMortgage

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
AND: In loop FOR ii := 1 TO 3
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
AND: PurcAmt <> EMPTY
RESERVECHECK

RESERVECHECK

FRS1004B.QOwner1 (continued)

Questions about mortgages

WARN IF: *QAccomdat.Tenure IN [Outright .. Part]*
RESERVECHECK

RESERVECHECK

WARN IF: *QAccomdat.Tenure IN [Outright .. Part]*
RESERVECHECK

RESERVECHECK

WARN IF: *QAccomdat.Tenure IN [Outright .. Part]*
RESERVECHECK

RESERVECHECK

WARN IF: *QAccomdat.Tenure IN [Outright .. Part]*
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part]
RESERVECHECK

RESERVECHECK

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: (((QAccomdat.Tenure IN [Mortgage .. Part]) OR (QOwner1.OthMort3 = Yes)) AND (QOwner1.QMortgage.M[1].MorInPay = EMPTY OR (QOwner1.QMortgage.M[1].MorInPay > 0))) AND (QOwner1.QMortgage.M[2].MorInPay = EMPTY OR (QOwner1.QMortgage.M[2].MorInPay > 0))

AskStruc := 1

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: (((QAccomdat.Tenure IN [Outright .. Part]) OR QAccomdat.Tenure = NONRESPONSE) OR (QRenting.Landlord IN [Assocn .. OthIndiv])) OR QRenting.Landlord = NONRESPONSE
AND: AskStruc = 1

AskStruc := 3

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: (((QAccomdat.Tenure IN [Outright .. Part]) OR QAccomdat.Tenure = NONRESPONSE) OR (QRenting.Landlord IN [Assocn .. OthIndiv])) OR QRenting.Landlord = NONRESPONSE
AND: NOT (AskStruc = 1)

AskStruc := 2

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: (QAccomdat.Tenure = Part) AND (QAccomdat.SOBuy = Paid)

AskStruc := 2

FRS1004B.QInsur

Questions about structure insurance.

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]

StrMort

^I QInsur^I

Structural insurance is treated as a housing cost. Comparisons of different households' income before and after housing costs are used to monitor Government child and pensioner poverty targets. It is therefore important that insurance is accurately recorded on the FRS.

- (1) Yes
 - (2) No
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes

StrCov

^I QInsur^I

Structural insurance is treated as a housing cost. Comparisons of different households' income before and after housing costs are used to monitor Government child and pensioner poverty targets. It is therefore important that insurance is accurately recorded on the FRS.

- (1) ^N...buildings insurance^B only^B
- (2) ^N...contents insurance,^B only^B
- (3) ^N...or buildings^B and^B contents insurance?

FRS1004B.QInsur.QStructure[]

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: (PStrCov = Struct) OR (PCovOths = Struct)

```

```

combined := ' structure'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: PStrCov = Furn

```

```

combined := ' contents'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: (PStrCov = Combine) OR (PCovOths = Combine)

```

```

combined := ' combined'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: PSeq = 1

```

```

included := ' included in your last mortgage payment'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: NOT (PSeq = 1)

```

```

last := ' last'

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes

```

StrAmt

^I QInsur^I

^N

How much was the^last premium^included for this^combined policy?

0.01..99997.00

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: StrAmt = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: StrAmt > 0

StrPx

^I QInsur^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: StrAmt > 0

StrPd

^I QInsur^I

^N

How long did this cover?^N

- (1) One week
 - (2) Two weeks
 - (3) Three weeks
 - (4) Four weeks
 - (5) Calendar month
 - (7) Two Calendar months
 - (8) Eight times a year
 - (9) Nine times a year
 - (10) Ten times a year
 - (13) Three months/13 weeks
 - (26) Six months/26 weeks
 - (52) One Year/12 months/52 weeks
 - (90) Less than one week
 - (95) One off/lump sum
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
AND: StrAmt > 0
AND: StrPd = Note

StrPx

^I QInsur^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QInsur.QStructure[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[8] := 6.5

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QInsur.QStructure[] (continued)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
  AND: StrPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01

```

StrWkly := LWeekly

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  AND: StrAmt > 0
  AND: StrPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01
  AND: Edit = No
  (StrWkly < 50) AND INVOLVING(StrPd,StrAmt)

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  StrPd <> Note

```

^I

Editor: Code 97 must be re-coded into existing list.
If you temporarily suppress this check you must come back to resolve it.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [1, 3]
  AND: StrMort = Yes
  RESERVECHECK

```

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [1, 3]
AND: StrMort = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QInsur (continued)

Questions about structure insurance.

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: (StrMort = EMPTY OR (StrMort = No)) OR (StrCov = Furn)

StrOths

^I QInsur^I

Structural insurance is treated as a housing cost. Comparisons of different households' income before and after housing costs are used to monitor Government child and pensioner poverty targets. It is therefore important that insurance is accurately recorded on the FRS.

- (1) Yes
 - (2) No
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes

CovOths

^I QInsur^I

Structural insurance is treated as a housing cost. Comparisons of different households' income before and after housing costs are used to monitor Government child and pensioner poverty targets. It is therefore important that insurance is accurately recorded on the FRS.

- (1) ^N...buildings insurance^B only^B,
- (2) ^N...or buildings and contents insurance?

FRS1004B.QInsur.QStructure[]

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: (PStrCov = Struct) OR (PCovOths = Struct)

```

```

combined := ' structure'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: PStrCov = Furn

```

```

combined := ' contents'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: (PStrCov = Combine) OR (PCovOths = Combine)

```

```

combined := ' combined'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: PSeq = 1

```

```

included := ' included in your last mortgage payment'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: NOT (PSeq = 1)

```

```

last := ' last'

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes

```

StrAmt

^I QInsur^I

^N

How much was the^last premium^included for this^combined policy?

0.01..99997.00

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: StrAmt = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: StrAmt > 0

StrPx

^I QInsur^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: StrAmt > 0

StrPd

^I QInsur^I

^N

How long did this cover?^N

- (1) One week
 - (2) Two weeks
 - (3) Three weeks
 - (4) Four weeks
 - (5) Calendar month
 - (7) Two Calendar months
 - (8) Eight times a year
 - (9) Nine times a year
 - (10) Ten times a year
 - (13) Three months/13 weeks
 - (26) Six months/26 weeks
 - (52) One Year/12 months/52 weeks
 - (90) Less than one week
 - (95) One off/lump sum
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
AND: StrAmt > 0
AND: StrPd = Note

StrPx

^I QInsur^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QInsur.QStructure[.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[8] := 6.5

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: PAskStruc IN [2 .. 3]
    AND: StrOths = Yes
    AND: StrAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: PAskStruc IN [2 .. 3]
    AND: StrOths = Yes
    AND: StrAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: PAskStruc IN [2 .. 3]
    AND: StrOths = Yes
    AND: StrAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: PAskStruc IN [2 .. 3]
    AND: StrOths = Yes
    AND: StrAmt > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: PAskStruc IN [2 .. 3]
    AND: StrOths = Yes
    AND: StrAmt > 0
    AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: PAskStruc IN [2 .. 3]
    AND: StrOths = Yes
    AND: StrAmt > 0
    AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QInsur.QStructure[] (continued)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
  AND: StrPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01

```

StrWkly := LWeekly

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  AND: StrAmt > 0
  AND: StrPd IN [OneWeek .. Year]
  AND: LWeekly >= 0.01
  AND: Edit = No
  (StrWkly < 50) AND INVOLVING(StrPd,StrAmt)

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  StrPd <> Note

```

^I

Editor: Code 97 must be re-coded into existing list.
If you temporarily suppress this check you must come back to resolve it.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: PAskStruc IN [2 .. 3]
  AND: StrOths = Yes
  RESERVECHECK

```

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: PAskStruc IN [2 .. 3]
AND: StrOths = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QInsur (continued)

Questions about structure insurance.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: QInsur.QStructure[1].StrWkly = RESPONSE
AND: QOwner1.QMortgage.M[1].MorIWkly = RESPONSE
 (QInsur.QStructure[1].StrWkly < QOwner1.QMortgage.M[1].MorIWkly) AND
 INVOLVING(QOwner1.QMortgage.M[1].MorInPay, QInsur.QStructure[1].StrAmt,
 QInsur.QStructure[1].StrPd)

^I

The amount you recorded for the premium on the insurance on the structure is greater than the amount recorded for the last mortgage payment.
 Please check whether this is correct.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: QInsur.QStructure[1].StrWkly = RESPONSE
AND: QOwner1.QMortgage.M[1].IntPWkly = RESPONSE
 (QInsur.QStructure[1].StrWkly < QOwner1.QMortgage.M[1].IntPWkly) AND
 INVOLVING(QOwner1.QMortgage.M[1].IntPrPay, QInsur.QStructure[1].StrAmt,
 QInsur.QStructure[1].StrPd)

^I

The amount you recorded for the premium on the insurance on the structure is greater than the amount recorded for the last mortgage payment.
 Please check whether this is correct.

FRS1004B.QCounTax

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandAMax := 1100

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandBMax := 1295

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandCMax := 1480

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandDMax := 1660

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandEMax := 2030

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandFMax := 2400

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandGMax := 2770

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandHMax := 3320

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandIMax := 3690

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

BandAMin := 455

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NatCen <> NI

```

```

CTIntro := (B + '
Now there are some questions about Council Tax
' + B)

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

CTConDoc

^I QCounTax^I

^N^CTIntro

For your Council Tax, do you have a bill, or a payment book that you could consult?^N

^I^IC Accept a statement/bill from the year ^FYearM1-^FYear if no payment for ^FYear-^LYear yet made.

(1) Yes - consulted now

(2) No - no document (or will not consult)

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

CTBand

^I QCounTax^I

^N

Could you please tell me which Council Tax band this accommodation is in?

^I^IC This must be the band given by the council - do not accept respondent's own estimate of value of property.

Council Tax band I exists in Wales for properties over £400,000.

If this household's accommodation is not valued separately (eg. because it's a rented part of larger premises), then use code 10.

If respondent's initial band allocation was later changed because they are disabled enter^B original^B band here.

(1) Band A

(2) Band B

(3) Band C

(4) Band D

(5) Band E

(6) Band F

(7) Band G

(8) Band H

(9) Band I

(10) Household accommodation not valued separately

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Wales <> Yes
CTBand <> **BandI**

^I^IC This is not a Welsh property - that code is invalid.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

```

```

RESERVECHECK

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

```

```

CTXAmt := 'Missing'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

```

```

CTSxAmt := 'Missing'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

```

```

CTSWAmt := 'Missing'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

```

```

CTSSAmt := 'Missing'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: CTData.SEARCH (QDataBag.SLA)

```

```

RCTXAmt := CTData.BandAmt [ORD (CTBand) ]

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: CTData.SEARCH (QDataBag.SLA)

```

```

CTXAmt := STR (RCTXAmt,7,2)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: (Scotland = Yes) AND CTScot.SEARCH (QDataBag.SLA)

```

```

RCTXAmt := CTScot.BandAmt [ORD (CTBand) ]

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: (Scotland = Yes) AND CTScot.SEARCH (QDataBag.SLA)

```

```

CTSxAmt := STR (RCTXAmt,7,2)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: NOT ((Scotland = Yes) AND CTScot.SEARCH (QDataBag.SLA))

```

```

CTXAmt := 'N/A'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: (Scotland = Yes) AND ScotWat.SEARCH (QDataBag.SLA)

```

```

RCTSWAmt := ScotWat.BandAmt [ORD (CTBand)]

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: (Scotland = Yes) AND ScotWat.SEARCH (QDataBag.SLA)

```

```

CTSWAmt := STR (RCTSWAmt, 7, 2)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: NOT ((Scotland = Yes) AND ScotWat.SEARCH (QDataBag.SLA))

```

```

CTSWAmt := 'N/A'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: (Scotland = Yes) AND ScotSew.SEARCH (QDataBag.SLA)

```

```

RCTSSAmt := ScotSew.BandAmt [ORD (CTBand)]

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: (Scotland = Yes) AND ScotSew.SEARCH (QDataBag.SLA)

```

```

CTSSAmt := STR (RCTSSAmt, 7, 2)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) AND (Edit = Yes)
AND: NOT ((Scotland = Yes) AND ScotSew.SEARCH (QDataBag.SLA))

```

```

CTSSAmt := 'N/A'

```

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: CTConDoc <> EMPTY
CTBand = RESPONSE

```

^I

Editor: The Council Tax Band is missing. The Local Authority will need to be telephoned. Consult the fact sheet & telephone them.

```

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI

```

CTValid

```

^I QCounTax^I

```

```

STRING[2]

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[1] := 'A'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[2] := 'B'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[3] := 'C'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[4] := 'D'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[5] := 'E'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[6] := 'F'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[7] := 'G'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[8] := 'H'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

Letters[9] := 'I'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTBand IN [BandA .. BandI]

```

Letter := Letters[ORD(CTBand)]

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTBand = NotApp

```

Letter := 'Not valued separately'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTBand = DONTKNOW

```

Letter := 'Don't know'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTBand = REFUSAL

```

Letter := 'Missing'

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

CTLVBand

^I QCounTax^I

^N

Was your Council Tax bill reduced to a lower band because there is a disabled person in the household?^N

^I^IC Households must make a special application in order to obtain this reduction.

- (1) Yes
 - (2) No
-

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: (CTLVBand = Yes) AND (CTBand IN [BandA .. BandI])

```

CTLVChk

^I QCounTax^I

^N

You said you were in Band ^Letter; is that the Band^B after^B this lower valuation, or^B before^B?

- (1) After lower valuation
- (2) Before

```

CHECK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: (CTLVBand = Yes) AND (CTBand IN [BandA .. BandI])
  AND: CTBand = BandI
  CTLVChk <> Aftr

```

^I^IC Band I is the HIGHEST band, so it cannot be the band AFTER the lower valuation. Please change one or the other.

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: Scotland = Yes

```

```

ScotFill := (' including' + B + ' Domestic water & sewerage charges '
+ B)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: Scotland = Yes

```

```

ScotInst := ('

' + IC + 'If the respondent has a private water supply or a septic
tank and does not pay for water ' + 'and sewerage with their council
tax,' + B + ' make a note' + B + ' of whether water or sewerage
are paid for ' + 'separately and the amount currently paid for
each.')
```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: Scotland = Yes

```

```

ScotHelp := ('

' + B + 'Sources of private water supply' + B + ' include surface
water, such as streams or rivers as well as ' + 'private impoundment
reservoirs, and ground water such as wells and boreholes or
springs.')
```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: NOT (Scotland = Yes)

```

```

ScotFill := ''

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: NOT (Scotland = Yes)

```

```

ScotInst := ''

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
    AND: NOT (Scotland = Yes)

```

ScotHelp := ' '

'

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

CTAmt

^I QCounTax^I

^I

If the respondent has not paid any tax for any reason then enter 0 and the later questions will probe the reasons.

^ScotHelp

0.00..9999.97

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
    AND: Scotland = Yes
    CTAmt <> 0

```

^I^IC No Council Tax paid, but you should record here the last payment of domestic water charge and domestic sewerage charge, which are not paid for by C. Tax benefit. If they have not been paid, suppress warning and continue.

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
    AND: CTAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
    AND: (CTAmt > 0) OR CTAmt = NONRESPONSE

```

CTInstal

^I QCounTax^I

^N^O1 Can I just check, ^O2 Was that the full payment for the year, or was it an instalment? ^N

^I^IC 'Year' = April to March (12 months).

- (1) Full annual payment
- (2) An instalment

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (CTAmt > 0) OR CTAmt = NONRESPONSE
AND: CTInstal = Instal

```

CTTime

```

^I QCounTax^I
^N
How many instalments are there, over the whole year?^N

^I^IC 'Whole year' = April to March (12 months).
If payment given is from^B last^B year, enter number of instalments made last year.

2..52

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (CTAmt > 0) OR CTAmt = NONRESPONSE
AND: CTInstal = Instal
AND: (CTConDoc = Yes) AND (CTTime = RESPONSE)

```

CTAnnual

```

^I QCounTax^I

^I^IC Refer to document being consulted:^I
^N
On the statement/bill, what is the^B total^B amount payable for the year,^ScotFill after deducting any
discounts or benefit?^N

^I^IC 'Year' = April to March (12 months)

0.00..9999.97

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (CTAmt = RESPONSE) AND (CTInstal = RESPONSE)
AND: CTInstal = Full

```

CTAmtYr := CTAmt

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (CTAmt = RESPONSE) AND (CTInstal = RESPONSE)
AND: (CTInstal = Instal) AND (CTTime = RESPONSE)

```

CTAmtYr := (CTAmt * CTTime)

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CWat1Rs

^I QCounTax^I

^I

In Scotland, Domestic Water Charge should be^B included^B in the total Council Tax bill for the year - if not, please explain in a note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CWat1Ex

^I QCounTax^I

^I^IC^SuppTxt

OPEN

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CSew1Rs

^I QCounTax^I

^I

In Scotland, Domestic Sewerage Charge should be^B included^B in the total Council Tax bill for the year - if not, please explain in a note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CSew1Ex

^I QCounTax^I

^I^IC ^SuppTxt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((Edit = No) AND (Scotland = Yes)) AND (CTConDoc = Yes)) AND ((CTInstal = Full) OR (CTAnnual > 0))

CWatAmt1

^I QCounTax^I

^N

How much is the annual Domestic Water Charge, as shown on the bill?^N

^I^IC Enter the full charge, before any status discount.

0.00..9999.97

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((Edit = No) AND (Scotland = Yes)) AND (CTConDoc = Yes)) AND ((CTInstal = Full) OR (CTAnnual > 0))
AND: CWatAmt1 = RESPONSE
CWatAmt1 > 0

CWatAmt1 > 0

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((Edit = No) AND (Scotland = Yes)) AND (CTConDoc = Yes)) AND ((CTInstal = Full) OR (CTAnnual > 0))
AND: CWatAmt1 = RESPONSE
AND: (CWat1Rs = Suppressed) OR CWat1Ex <> EMPTY

CWat1Ex

^I QCounTax^I

^I^IC^SuppTxt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((Edit = No) AND (Scotland = Yes)) AND (CTConDoc = Yes)) AND ((CTInstal = Full) OR (CTAnnual > 0))
AND: CWatAmt1 = RESPONSE

CSewAmt1

^I QCounTax^I

^N

How much is the annual Domestic Sewerage Charge, as shown on the bill?^N

^I^IC Enter the full charge, before any status discount or transitional relief.

0.00..9999.97

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((Edit = No) AND (Scotland = Yes)) AND (CTConDoc = Yes)) AND ((CTInstal = Full) OR (CTAnnual > 0))
AND: CWatAmt1 = RESPONSE
AND: CSewAmt1 = RESPONSE
CSewAmt1 > 0

CSewAmt1 > 0

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((Edit = No) AND (Scotland = Yes)) AND (CTConDoc = Yes)) AND ((CTInstal = Full) OR (CTAnnual > 0))
AND: CWatAmt1 = RESPONSE
AND: CSewAmt1 = RESPONSE
AND: (CSew1Rs = Suppressed) OR CSew1Ex <> EMPTY

CSew1Ex

^I QCounTax^I
 ^I^IC ^SuppTxt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: ((Edit = Yes) AND (Scotland = Yes)) AND (CTConDoc = Yes)

CWatAmt1

^I QCounTax^I
 ^N

How much is the annual Domestic Water Charge, as shown on the bill?^N

^I^IC Enter the full charge, before any status discount.

0.00..9999.97

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: ((Edit = Yes) AND (Scotland = Yes)) AND (CTConDoc = Yes)

CSewAmt1

^I QCounTax^I
 ^N

How much is the annual Domestic Sewerage Charge, as shown on the bill?^N

^I^IC Enter the full charge, before any status discount or transitional relief.

0.00..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CTRebPx

^I QCounTax^I

^I^IC ^Pd97Txt

OPEN

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CTRebRs

^I QCounTax^I

^I

That seems rather high. Please check the amount and frequency of payment. If correct, suppress warning and explain circumstances in a note.

- (1) Passed
- (2) Hard
- (3) Soft
- (4) Suppressed

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CTRebEx

^I QCounTax^I

^I

^IC ^SuppTxt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

CTReb

^I QCounTax^I

^N

Are you allowed Council Tax Benefit or rebate, to help pay your Council Tax?

- (1) Yes
- (2) No

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = No
AND: (CTAmt = 0) OR CTAmt = NONRESPONSE

```

WhyNoCT

^I QCounTax^I

^I^IC No council tax is paid, but no benefit received.

Ask or code: what was the reason for paying no council tax?

- (1) Bill not yet^B received^B and household not previously liable for C.Tax
- (2) Bill not yet^B paid^B and household not previously liable for C.Tax
- (3) Deliberate non-payment, in dispute, appeal, etc.
- (4) Household only recently moved into accommodation
- (5) Household has a 'formal exemption' from the Tax
(all students; MoD property; severely mentally impaired.)
- (6) Paid by absent partner
- (7) Paid by employer
- (8) Other reason (DESCRIBE IN A NOTE)

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = No

```

CTBWait

^I QCounTax^I

^N

Are you awaiting the outcome of a claim for Council Tax benefit or rebate?

- (1) Yes
- (2) No

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = Yes

```

CTRebAmt

^I QCounTax^I

^N

How much was allowed?

0.00..9999.97

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = Yes
AND: CTRebAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = Yes
AND: CTRebAmt > 0

CTRebPd

^I QCounTax^I

^N

How long did this cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = Yes
AND: CTRebAmt > 0
AND: CTRebPd = Note

CTRebPx

^I QCounTax^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QCounTax.Weekly()

Procedure Call

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[7] := 8.67

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
```

PdConW[8] := 6.5

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0

```

PdConW[9] := 5.78

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0

```

PdConW[52] := 52

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QCounTax (continued)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
  AND: CTRebPd IN [OneWeek .. Year]
  AND: LWeekly > 0

```

CTRWkly := LWeekly

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
  AND: CTRebPd IN [OneWeek .. Year]
  AND: LWeekly > 0

```

CTRebYr := (CTRWkly * 52)

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
  AND: CTRebPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: (CTRebYr > 0) AND (CTBand = RESPONSE)
  (((((((((CTBand = BandA) AND (CTRebYr <= BandAMax)) OR ((CTBand = BandB)
AND (CTRebYr <= BandBMax)))) OR ((CTBand = BandC) AND (CTRebYr <= BandCMax))))
OR ((CTBand = BandD) AND (CTRebYr <= BandDMax))) OR ((CTBand = BandE) AND
(CTRebYr <= BandEMax))) OR ((C

((((((((CTBand = BandA) AND (CTRebYr <= BandAMax)) OR ((CTBand = BandB) AND (CTRebYr
<= BandBMax)))) OR ((CTBand = BandC) AND (CTRebYr <= BandCMax)))) OR ((CTBand = BandD)
AND (CTRebYr <= BandDMax)))) OR ((CTBand = BandE) AND (CTRebYr <= BandEMax))) OR
((CTBand = BandF) AND (CTRebYr <= BandFMax))) OR ((CTBand = BandG) AND (CTRebYr <=
BandGMax))) OR ((CTBand = BandH) AND (CTRebYr <= BandHMax))) OR ((CTBand = BandI) AND
(CTRebYr <= BandIMax))) AND INVOLVING (CTBand, CTRebPd, CTRebAmt)

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
  AND: CTReb = Yes
  AND: CTRebAmt > 0
  AND: CTRebPd IN [OneWeek .. Year]
  AND: LWeekly > 0
  AND: (CTRebYr > 0) AND (CTBand = RESPONSE)
  AND: (CTRebRs = Suppressed) OR CTRebEx <> EMPTY

```

CTRebEx

```

^I QCounTax^I
^I
^IC ^SuppTxt

```

OPEN

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((NewBU >= 2) AND (CTConDoc = Yes)) AND (CTReb = Yes)) AND (NotHRPBU
= 1)

```

WhoseCTB

^I QCounTax^I

^N

According to the statement, who is the Council Tax Benefit for?^N

^I^IC Code all that apply.

SET [7] OF

- (1) ^BUAdName[1]
- (2) ^BUAdName[2]
- (3) ^BUAdName[3]
- (4) ^BUAdName[4]
- (5) ^BUAdName[5]
- (6) ^BUAdName[6]
- (7) ^BUAdName[7]
- (8) Someone else (SPECIFY IN A NOTE)
- (9) Not on statement

```

CHECK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((NewBU >= 2) AND (CTConDoc = Yes)) AND (CTReb = Yes)) AND (NotHRPBU
= 1)
AND: In loop FOR Index := 1 TO 7
AND: Index IN WhoseCTB
BUAdName[[Index] <> ''

```

^I

Code ^Index is not valid for this question.^I

```

CHECK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: (((NewBU >= 2) AND (CTConDoc = Yes)) AND (CTReb = Yes)) AND (NotHRPBU
= 1)
AND: NS IN WhoseCTB
WhoseCTB.CARDINAL = 1

```

^I

'Not known/not on statement' is an exclusive code!^I

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTReb = Yes

```

are := 'In addition to your rebate/ benefit, are'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: NOT (CTReb = Yes)

```

are := 'Are'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: NatCen <> NI

```

```

SHOWCARD := (IS + ' B12')

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: NOT (NatCen <> NI)

```

```

SHOWCARD := ''

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE

```

CTDisc

^I QCounTax^I

^I

The council tax assumes two adults per household. If only one adult lives there, a 25% status discount will usually apply. If a student/student nurse/apprentice etc. lives with^B one^B other adult then a discount of 25% will apply. However no discount will apply if that person lives with two or more other adults.

- (1) Yes
 - (2) No
-

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTDisc = Yes

```

CT25D50D

^I QCounTax^I

^I

Some households get a discount on their Council Tax because of the^B type^B of person living there. Usually this is a 25% discount. The rules are very detailed, but in general:

25% for Single adult households

25% for one adult, plus: a student/student nurse/person under 18/apprentice/YT Trainee/care worker/severely mentally impaired

50% when a household is made up of a mixture of those who are exempt (eg. student/student nurse/person under 18/apprentice/YT Trainee/care worker/severely mentally impaired)

Note: Households wholly occupied by students or under 18 year olds are entirely^B exempt^B from Council Tax.

- (1) 25%
- (2) 50%

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: CTDisc = Yes
CT25D50D <> D50

^I

Are you sure? Discount is usually 25%. The 50% discount applies only if ALL household members belong to the groups shown on ^SHOWCARD. Please check with respondent. If discount IS DEFINITELY 50%, suppress warning and continue.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTBand IN [BandA .. BandI]) OR CTBand = NONRESPONSE
AND: AllAd = 1
(CTDisc = Yes) AND (CT25D50D = D25)

^I

Are you sure? Households with only one adult would normally have a status discount (25% reduction of the bill).

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (CTAmtYr > 0) AND (CTBand = RESPONSE)
(((((CTBand = BandA) AND (CTAmtYr <= BandAMax)) OR ((CTBand = BandB) AND (CTAmtYr <= BandBMax))) OR ((CTBand = BandC) AND (CTAmtYr <= BandCMax))) OR ((CTBand = BandD) AND (CTAmtYr <= BandDMax))) OR ((CTBand = BandE) AND (CTAmtYr <= BandEMax))) OR ((C

^I

That's £^CTAmtYr a year which seems rather high for a property in this Band. Please check the amount and frequency of payment. If correct, suppress warning and explain circumstances in a Note.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: ((CTAmt > 0) AND (CTDisc <> Yes)) AND (CTReb <> Yes)
      ((CTInstal = Instal) AND ((CTAmt * CTTime) >= BandAMin)) OR ((CTInstal =
      Full) AND (CTAmt > BandAMin))

```

^I

The annual Council Tax comes to less than the cheapest Council Tax. No discount or rebate is received, so please check for a typing error. If correct, please give explanation in a Note.

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: ((CTAmt = RESPONSE) AND (CTTime = RESPONSE)) AND (CTAnnual = RESPONSE)

```

CTReal := (CTAmt * CTTime)

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: ((CTAmt = RESPONSE) AND (CTTime = RESPONSE)) AND (CTAnnual = RESPONSE)
      CTAnnual < (1.2 * CTReal)

```

^I

£CTAnnual is more than would be expected, given the instalments mentioned earlier. Please check, from the document consulted, that it's for the^B same year^B as the instalments.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: ((CTAmt = RESPONSE) AND (CTTime = RESPONSE)) AND (CTAnnual = RESPONSE)
      CTAnnual > (0.8 * CTReal)

```

^I

£CTAnnual is less than would be expected, given the instalments mentioned earlier. Please check, from the document consulted, that it's for the^B same year^B as the instalments.

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
      (CTBand <> NotApp) AND (CTBand <> DONTKNOW)

```

^I^IC If necessary check which is correct; this accom:

- is NOT valued separately for C.Tax (code 9),
 - or it IS valued for C.Tax, but respondent DOESN'T KNOW the Tax Band
 (enter Don't know).

If correct, suppress check & continue.

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (((CTAmt = RESPONSE) AND (CTTime = RESPONSE)) AND (CTInstal = Instal))
      AND (CTAnnual = RESPONSE) AND (CWatAmt1 = RESPONSE)

```

CTReal := (CTAmt * CTTime)

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: (((CTAmt = RESPONSE) AND (CTTime = RESPONSE)) AND (CTInstal = Instal))
AND (CTAnnual = RESPONSE)) AND (CWatAmt1 = RESPONSE)
CWatAmt1 <= CTReal

^I

The Domestic^B Water^B charge is more than the total amount of Council Tax paid for the year. Make sure that the last payment of Domestic Water charge (and domestic sewerage charge) was included at CTAmt.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: ((CWatAmt1 = RESPONSE) AND (CTAmt = RESPONSE)) AND (CTInstal = Full)
CWatAmt1 <= CTAmt

^I

The Domestic^B Water^B Charge is more than the total amount of Council Tax paid for the year. Make sure that the last payment of Domestic Water charge (and domestic sewerage charge) was included at CTAmt.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: CTReb <> EMPTY AND (CTRebAmt = RESPONSE)
CTRebAmt <> 0

^I

Zero amount of Council Tax benefit: this contradicts previous answer (at 'CTReb') that benefit WAS received. Please resolve if possible.

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

OrgWatAmt

^I QCounTax^I

Domestic Water Charge, original entry before discount.

0.00..9999.97

```

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

```

OrgSewAmt

^I QCounTax^I
Domestic Sewerage Charge, original entry before discount.

0.00..9999.97

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes
AND: (CTBand IN [BandA .. BandI]) AND ScotWat.SEARCH (QDataBag.SLA)

```

OrgWatAmt := ScotWat.BandAmt [ORD (CTBand)]

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes
AND: CWatAmt1 <> EMPTY

```

OrgWatAmt := CWatAmt1

```

DISPLAY IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes

```

OrgWatAmt

^I QCounTax^I
Domestic Water Charge, original entry before discount.

0.00..9999.97

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes

```

CTDiscR := 1

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes
AND: CT25D50D = D25

```

CTDiscR := 0.75

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes
AND: CT25D50D = D50

```

CTDiscR := 0.5

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: Edit = Yes
    AND: Scotland = Yes

```

```

CWatAmt := (OrgWatAmt * CTDiscR)

```

```

DISPLAY IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: Edit = Yes
    AND: Scotland = Yes

```

CWatAmt

^I QCounTax^I

Water charge: Final value (after discount)

0.00..9999.97

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: Edit = Yes
    AND: Scotland = Yes
    AND: (CTBand IN [BandA .. BandI]) AND ScotSew.SEARCH (QDataBag.SLA)

```

```

OrgSewAmt := ScotSew.BandAmt [ORD (CTBand) ]

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: Edit = Yes
    AND: Scotland = Yes
    AND: CSewAmt1 <> EMPTY

```

```

OrgSewAmt := CSewAmt1

```

```

DISPLAY IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: Edit = Yes
    AND: Scotland = Yes

```

OrgSewAmt

^I QCounTax^I

Domestic Sewerage Charge, original entry before discount.

0.00..9999.97

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
    AND: NatCen <> NI
    AND: Edit = Yes
    AND: Scotland = Yes

```

```

CSewAmt := (OrgSewAmt * CTDiscR)

```

```

DISPLAY IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes

```

CSewAmt

^I QCounTax^I
 Sewerage charge: Final value (after discount)
 0.00..9999.97

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

```

CTChkB := ''

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

```

CTChkC := 'Not known'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

```

CTChkD := ''

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

```

CTChkE := 'N/A'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes

```

CTChkF := 'Not calculated'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: CTAnnual = RESPONSE

```

CTChkCR := CTAnnual

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: CTAnnual = RESPONSE

```

CTChkC := STR(CTAnnual,7,2)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: ((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTTime <>
NONRESPONSE
AND: (CTInstal = Full) OR (CTAmt = 0)

```

CTChkCR := CTAmt

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: ((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTTime <>
  NONRESPONSE
  AND: (CTInstal = Full) OR (CTAmt = 0)

```

CTChkC := STR(CTAmt,7,2)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: ((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTTime <>
  NONRESPONSE
  AND: CTInstal = Instal

```

CTChkCR := (CTAmt * CTTime)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: ((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTTime <>
  NONRESPONSE
  AND: CTInstal = Instal

```

CTChkC := STR(CTAmt * CTTime,7,2)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: CTDisc = No

```

CTChkD := 'None'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: CT25D50D = D25

```

CTChkD := '25%'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: CT25D50D = D50

```

CTChkD := '50%'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: CT25D50D = NONRESPONSE

```

CTChkD := 'Amount not known'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: (CTRebAmt = RESPONSE) AND (CTRebPd = RESPONSE)

```

CTChkE := STR(CTRWkly * 52,7,2)

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: CTRebAmt = NONRESPONSE OR CTRebPd = NONRESPONSE

```

```

CTChkE := 'Annual amount not known'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: (((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTime <>
NONRESPONSE) AND CTRebAmt <> NONRESPONSE) AND CTRebPd <> NONRESPONSE

```

```

CTChkFR := (CTChkCR + (CTRWkly * 52))

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: (((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTime <>
NONRESPONSE) AND CTRebAmt <> NONRESPONSE) AND CTRebPd <> NONRESPONSE
  AND: CT25D50D = D25

```

```

CTChkFR := ((CTChkFR * 4) / 3)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: (((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTime <>
NONRESPONSE) AND CTRebAmt <> NONRESPONSE) AND CTRebPd <> NONRESPONSE
  AND: CT25D50D = D50

```

```

CTChkFR := (CTChkFR * 2)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: (((CTAmt = RESPONSE) AND CTInstal <> NONRESPONSE) AND CTime <>
NONRESPONSE) AND CTRebAmt <> NONRESPONSE) AND CTRebPd <> NONRESPONSE

```

```

CTChkF := STR(CTChkFR,7,2)

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = Yes

```

```

Tab1 := '@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = Yes

```

```

Tab2 := '@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = Yes

```

```

Tab3 := '@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = Yes

```

```

Tab4 := '@|@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = Yes

```

```

Tab5 := '@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = ONS

```

```

Tab1 := '@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = ONS

```

```

Tab2 := ''

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = ONS

```

```

Tab3 := '@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = ONS

```

```

Tab4 := '@|@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = ONS

```

```

Tab5 := ''

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = NI

```

```

Tab1 := '@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = NI

```

```

Tab2 := ''

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = NI

```

```

Tab3 := '@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = NI

```

```

Tab4 := '@|@|@|@|@|'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  AND: NatCen = NI

```

```

Tab5 := '@|'

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes

```

CTChk

^I QCounTax^I

^I

Editor: The following calculations are based on the council tax details.^I^N

Local Authority@|@|@|: ^N^I@|^QDataBag.SLA^I^N

CTBand@|@|@|@|: ^N^I@|^Letter^I^N

Annual set charge^Tab1: ^N^I@|^£^CTXAmt^I^N
(from lookup)

SCOTLAND ONLY (from lookup):

Taking off water/sewerage^Tab2@|: ^N^I@|^£^CTSXAmt^I^N

charges^Tab3(Water@|: ^N^I@|^£^CTSWAmt^I^N,@|Sewerage: ^N^I £^CTSSAmt^I^N)

Respondent's annual payment@|: ^N^I@|^£^CTChkc^I^N

Discount^Tab4: ^N^I@|^CTChkd^I^N

Annual benefit received@|@|: ^N^I@|^£^CTChke^I^N

EXPECTED annual charge^Tab5@|: ^N^I@|^£^CTChkf

PRESS <Enter> To continue.

STRING[1]

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NatCen <> NI
  AND: Edit = Yes
  (CTConDoc <> No) AND INVOLVING(CTChk)

```

^I

Editor: No Council Tax document consulted

Examine display at 'CTChk' for discrepancies.

Compare set charge with expected charge (Should be v. similar) and refer to supervisor if necessary.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
CTAmt<>NONRESPONSE

^I

Missing amount of council tax. Refer to display at 'ctchk' and enter annual payment, after taking off discount/rebate (Also check for notes). If in Scotland, and discount applies, then see edit instructions.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: CTAmt = RESPONSE
CTAmt <> 0

^I

EDITOR: Zero Council Tax recorded. Please check the details as necessary

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
CTTime<>NONRESPONSE

^I

Missing Period for Council Tax.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
CTRebAmt<>NONRESPONSE

^I

Missing amount for Council Tax rebate.^I

Editor: For new claims made from April 1999, the maximum rebate for bands F, G & H is the band^B E^B total.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
CTRebPd<>NONRESPONSE

^I

Missing period for Council Tax Rebate.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: (CTBand = RESPONSE) AND CTLVChk <> EMPTY
CTLVChk <> Afttr

^I

EDITOR: The CT band is the band^B after^B Disablement re-valuation. But it should be the band^B before^B. Please change '^B CTBand^B' to the^B next band up^B (Eg. from 'C' to 'D') and then change the answer at '^B CTLVChk^B' to code 2, 'before'.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Edit = Yes
CTRebPd <> Note

^I

EDITOR: Code 97 must be re-coded into existing list.
 If you temporarily suppress this check you must come back to resolve it.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: (CTRebYr > 0) AND (CTBand = RESPONSE)
 (((CTBand = BandA) AND (CTRebYr <= BandAMax)) OR ((CTBand = BandB) AND
 (CTRebYr <= BandBMax))) OR ((CTBand = BandC) AND (CTRebYr <= BandCMax)))
 OR ((CTBand = BandD) AND (CTRebYr <= BandDMax))) OR ((IN(CTBand, [??]))
 AND (CTRebYr <= BandEMax))

^I

EDITOR: CT rebate is £^CTRebYr a year. For new claims made from April 1998, the maximum rebate
 for bands F, G & H is the band^B E^B total. For claims before April 1998 this capping does not apply.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: (CTRebYr > 0) AND (CTBand <> RESPONSE)
CTRebYr <= BandHMax

^I

EDITOR: Council Tax rebate is £^CTRebYr a year which is greater than even the highest council tax
 rebate allowed. Please check the amount and period of payment.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
WhyNoCT <> Other

^I

EDITOR: Other reason for CT non-payment. Please check for a note; and see
 (a) If the reason can be re-coded, at '^B WHYNOC'^B' {or at '^B CTEXRE'^B}';
 Or
 (b) If the hhold is in fact not liable for CT - Eg. note states 'paid to landlord' or 'included in rent' (If so,
 change '^B CTBand'^B' to '9').

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes
AND: (CTInstal = Full) OR (CTAnnual > 0)
CWatAmt1 = RESPONSE

^I

EDITOR: Missing amount for Domestic Water Charge (Scotland): Please impute using figures provided.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NatCen <> NI
AND: Edit = Yes
AND: Scotland = Yes
AND: (CTInstal = Full) OR (CTAnnual > 0)
CSewAmt1 = RESPONSE

^I

EDITOR: Missing amount for domestic sewerage charge (Scotland): Please impute using figures provided.

FRS1004B (continued)**FAMILY RESOURCES SURVEY 2010-2011**

WARN IF: *QAccomdat.HHStat* <> *EMPTY* OR (*Edit* = *Yes*)

AND: *NatCen* <> *NI*

QCounTax.WhyNoCT <> **Other**

^I

If Tax is 'paid by landlord' or 'included in rent', check whether the accommodation is valued separately and if not, recode CTBand;

if a deduction is made from pay, enter it as a payment at CTAmt.

WARN IF: *QAccomdat.HHStat* <> *EMPTY* OR (*Edit* = *Yes*)

AND: *NatCen* <> *NI*

AND: *QCounTax.CTReb* <> *EMPTY* AND ((*QRenting.HBenefit* = *Yes*) OR (*GOV IN QOwner1.QMortgage.M[1].QOutsPay*))

QCounTax.CTReb = **Yes**

^I

Earlier, the respondent said they get Housing Benefit (or Local Housing Allowance) or help from ^GOV1 with mortgage. They should usually also get Council Tax Benefit. Please check: is their C.Tax bill 'reduced' - does the Council take anything off it? (apart from Discount). If so, change answer to 'Yes'.

FRS1004B.QNIRates

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 1
```

NIRate := 6.767

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 2
```

NIRate := 6.101

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 3
```

NIRate := 7.001

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 4
```

NIRate := 6.765

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 5
```

NIRate := 6.486

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 6
```

NIRate := 6.729

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 7
```

NIRate := 6.148

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 8
```

NIRate := 6.803

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 9
```

NIRate := 5.51

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 10
```

NIRate := 6.262

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 11
```

NIRate := 6.172

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 12
```

NIRate := 7.082

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 13
```

NIRate := 6.673

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 14
```

NIRate := 6.042

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 15
```

NIRate := 5.78

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 16
```

NIRate := 6.926

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 17
```

NIRate := 7.163

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 18
```

NIRate := 5.922

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 19
```

NIRate := 7.141

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 20
```

NIRate := 5.886

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 21
```

NIRate := 7.158

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 22
```

NIRate := 6.411

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 23
```

NIRate := 6.607

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 24
```

NIRate := 6.01

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 25
```

NIRate := 7.008

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
      AND: NIDCoun = 26
```

NIRate := 6.735

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
```

PDCode[1] := 52

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
```

PDCode[2] := 26

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
```

PDCode[3] := 17.33

```
COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)
```

PDCode[4] := 13

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[5] := 12

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[7] := 6

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[8] := 8

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[9] := 9

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[10] := 10

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[13] := 4

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[26] := 2

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[52] := 1

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[90] := 1

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

PDCODE[95] := 1

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
      AND: NOT (NatCen <> NI)

```

```

RTIntro := 'Now there are some questions about Rates.'

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)

BillRate

^I QNIRates^I

^N

^RTIntro

Do you get a bill for rates on this accommodation?

^N

(1) Yes

(2) No

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: BillRate = No

NoRate

^I QNIRates^I

^N

Why do you not get a rates bill?^N

^I^IC Note: the household can only receive full rate relief if occupied solely by full-time students/trainees, under 18's or those leaving care.

(1) Rented accommodation with rates included in rent

(2) Rent/rates free

(3) Receive full rate rebate (Housing Benefit)

(4) Receive full rate relief

(5) Receive both rate relief and rate rebate

(6) Other reason (specify)

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: BillRate = No
AND: NoRate = Other

OthReas

^I QNIRates^I

^N

Please specify this other reason

STRING[100]

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)

PayRate

^I QNIRates^I

^N

Do you, or someone in this household, pay the rates bill?

(1) Yes

(2) No

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = No

```

NoPay

```

^I QNIRates^I
^N
Why don't you pay your rates bill?

STRING[100]

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes

```

RTConDoc

```

^I QNIRates^I
^N
For your Rates, do you have a bill, statement, or payment book that you could consult?^N

^I^IC Accept a statement/bill from the most recent year if none available for 2010-2011.

(1)   Yes - consulted now

(2)   No - no document (or will not consult)

```

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTConDoc = Yes

```

RTStatYr

```

^I QNIRates^I
^I
Refer to document being consulted:^I
^N
Which year does the statement/bill refer to?

(1)   2007-08
(2)   2008-09
(3)   2009-10
(4)   2010-11

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTConDoc = Yes

```

```

Document := '
Refer to document being consulted:
'

```

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NOT (NatCen <> NI)
  AND: NOT (BillRate = No)
  AND: PayRate = Yes
  AND: RTConDoc = Yes

```

can_you := 'On the statement/bill'

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NOT (NatCen <> NI)
  AND: NOT (BillRate = No)
  AND: PayRate = Yes
  AND: RTConDoc = No

```

Document := ''

```

COMPUTE IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NOT (NatCen <> NI)
  AND: NOT (BillRate = No)
  AND: PayRate = Yes
  AND: RTConDoc = No

```

can_you := 'Can you tell me'

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NOT (NatCen <> NI)
  AND: NOT (BillRate = No)
  AND: PayRate = Yes

```

RTAnnual

^I QNIRates^I

^I^Document^I ^N

^can_you, what is the total amount of rates payable, after deducting any discounts or benefit?^N

^I^IC If not consulting a statement/bill - probe to ensure amount given is as accurate as possible and excludes any discounts or benefits. We need to know the amount they actually pay.

'Year' = April to March (12 months)

Note : No rates are payable in February and March each year

0.00..9999.97

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
  AND: NOT (NatCen <> NI)
  AND: NOT (BillRate = No)
  AND: PayRate = Yes

```

RTArrear

^I QNIRates^I

^I^Document^I ^N

Are there any arrears included in the total amount payable?

(1) Yes

(2) No

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTArrear = Yes

RTArrAmt

^I QNIRates^I

^N

What is the total amount of arrears payable?

0.00..9999.97

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes

RTInstal

^I QNIRates^I

^N

(Can I just check,) is this total amount payable the full payment for the year, or was it an instalment?^N

^I^IC 'Year' = April to March (12 months).

- (1) Full annual payment
 - (2) An instalment
-

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTInstal = Instal

RTTimePx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTInstal = Instal

```

RTTimePd

^I QNIRates^I

^N

How often do you pay instalments?^N

^I^IC 'Whole year' = April to March (12 months).

Note : If respondent paying by Direct Debit there are usually 10 monthly instalments per year as no rates are payable in February and March.

If payment given is from last year, enter number of instalments made last year.

- (1) One week
- (5) Calendar month
- (10) Ten times a year
- (52) One Year/12 months/52 weeks
- (97) None of these ^I(Explain in a note)

```

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTInstal = Instal
AND: RTTimePd = Note

```

RTTimePx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTInstal = Instal
AND: RTTimePd = Year
(RTInstal <> Instal) AND INVOLVING(RTInstal,RTTimePd)

```

The respondent said they pay one instalment per year - therefore go to RTInstal and change code to 'Full annual payment'

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes

RTDpa

^I QNIRates^I

^N

Does your household receive a discount on your rates bill through receipt of Disabled Persons Allowance?

^I^IC Disabled Persons Allowance : If the head of household, or someone else in the household is, a person with a disability, and meet the conditions of this scheme, they may be entitled to a reduction of 25% in their domestic rates bill. The property must be adapted to meet the needs of a person with a disability who lives within the home.

(1) Yes

(2) No

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTDpa = Yes

RTDpaAmt

^I QNIRates^I

^N

How much was allowed?

0.01..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTDpa = Yes

RTDpaPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTDpa = Yes

RTDpaPd

^I QNIRates^I

^N

How long did this cover?

- (1) One week
 - (5) Calendar month
 - (10) Ten times a year
 - (52) One Year/12 months/52 weeks
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTDpa = Yes
AND: RTDpaPd = Note

RTDpaPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes

RTReb

^I QNIRates^I

^N

Are you allowed a Rates Rebate?

- (1) Yes
 - (2) No
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTReb = Yes

RTRebAmt

^I QNIRates^I

^N

How much was allowed?

0.01..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTReb = Yes

RTRebPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTReb = Yes

RTRebPd

^I QNIRates^I

^N

How long did this cover?

- (1) One week
 - (5) Calendar month
 - (10) Ten times a year
 - (52) One Year/12 months/52 weeks
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTReb = Yes
AND: RTRebPd = Note

RTRebPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes

RTRtR

^I QNIRates^I

^N

Does your household receive a discount on your rates bill through receipt of Rate Relief?

^I^IC If household in receipt of Rate Relief please probe whether it is 'general Rate Relief' or Rate Relief for 'full-time students/trainees, under 18's or those leaving care'.

Rate Relief (general): The new general Rate Relief scheme helps if the ratepayer is a pensioner with savings of less than £50,000, getting Housing Benefit for only part of their rate bill or just outside the income limit for receiving Housing Benefit.

Rate Relief (full-time students, trainees, under 18's and those leaving care): Full-time students, those in full-time training; those aged under 18; or a young person leaving care may not have to pay any rates on their property (whether they own it or rent it) so long as they are either living alone, or in a property that is occupied only by other people in these groups.

- (1) Rate Relief (general)
- (2) Rate Relief (full time students/trainees, under 18's, leaving care)
- (3) Does not receive Rate Relief

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: (RTRtR = RRGEn) OR (RTRtR = RRStu)

RTRtRAmt

^I QNIRates^I

^N

How much was allowed?

0.01..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: (RTRtR = RRGEn) OR (RTRtR = RRStu)

RTRtRPx

^I QNIRates^I

^I^IC ^Pd97Ttxt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: (RTRtR = RRGEn) OR (RTRtR = RRStu)

RTRtRPd

^I QNIRates^I

^N

How long did this cover?

- (1) One week
 - (5) Calendar month
 - (10) Ten times a year
 - (52) One Year/12 months/52 weeks
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: (RTRtR = RRGEn) OR (RTRtR = RRStu)
AND: RTRtRPd = Note

RTRtRPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: ((HHG.P[1].DVAge >= 70) AND (QNames.HSize = 1)) AND (PayRate = Yes)

RTLpa

^I QNIRates^I

^N

Are you in receipt of Lone Pensioner Allowance?^N

^I^IC Lone Pensioners Allowance has been introduced in NI for pensioners living alone. A new 20% discount is available for pensioners aged 70 or over; living on their own; AND paying rates for their home.

This household may be eligible for this discount so please check if it is received.

- (1) Yes
- (2) No

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: ((HHG.P[1].DVAge >= 70) AND (QNames.HSize = 1)) AND (PayRate = Yes)
AND: RTLpa = Yes

RTLpaAmt

^I QNIRates^I
 ^N
 How much was allowed?
 0.01..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: ((HHG.P[1].DVAge >= 70) AND (QNames.HSize = 1)) AND (PayRate = Yes)
AND: RTLpa = Yes

RTLpaPx

^I QNIRates^I
 ^I^IC ^Pd97Txt
 OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: ((HHG.P[1].DVAge >= 70) AND (QNames.HSize = 1)) AND (PayRate = Yes)
AND: RTLpa = Yes

RTLpaPd

^I QNIRates^I
 ^N
 How long did this cover?
 (1) One week
 (5) Calendar month
 (10) Ten times a year
 (52) One Year/12 months/52 weeks
 (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: ((HHG.P[1].DVAge >= 70) AND (QNames.HSize = 1)) AND (PayRate = Yes)
AND: RTLpa = Yes
AND: RTLpaPd = Note

RTLpaPx

^I QNIRates^I
 ^I^IC ^Pd97Txt
 OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes

RTOther

^I QNIRates^I

^N

Are you in receipt of any other allowance, not yet mentioned, that has reduced the amount you pay for rates?^N

^I^IC If 'yes' please specify name of allowance.

(1) Yes

(2) No

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTOther = Yes

RTOthAll

^I QNIRates^I

^N

Please specify this other allowance.

STRING[100]

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTOther = Yes

RTOthAmt

^I QNIRates^I

^N

How much was allowed?

0.01..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTOther = Yes

RTOthPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTOther = Yes

RTOthPd

^I QNIRates^I

^N

How long did this cover?

- (1) One week
 - (5) Calendar month
 - (10) Ten times a year
 - (52) One Year/12 months/52 weeks
 - (97) None of these ^I(Explain in a note)
-

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: NOT (BillRate = No)
AND: PayRate = Yes
AND: RTOther = Yes
AND: RTOthPd = Note

RTOthPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: ((NoRate = RateRbt) OR (NoRate = RateRel)) OR (NoRate = RateBth)

RT2RebAmt

^I QNIRates^I

^N

How much rates would be payable on this property if you did not ^NoRate?^N

^I^IC If respondent does not know how much rates would be payable, code Don't Know <?>.

0.01..9999.97

RECORD IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: ((NoRate = RateRbt) OR (NoRate = RateRel)) OR (NoRate = RateBth)
AND: RT2RebAmt <> DONTKNOW

RT2RebPx

^I QNIRates^I

^I^IC ^Pd97Txt

OPEN

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: ((NoRate = RateRbt) OR (NoRate = RateRel)) OR (NoRate = RateBth)
AND: RT2RebAmt <> DONTKNOW

RT2RebPd

^I QNIRates^I
 ^N
 How long did this cover?

- (1) One week
- (5) Calendar month
- (10) Ten times a year
- (52) One Year/12 months/52 weeks
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: ((NoRate = RateRbt) OR (NoRate = RateRel)) OR (NoRate = RateBth)
AND: RT2RebAmt <> DONTKNOW
AND: RT2RebPd = Note

RT2RebPx

^I QNIRates^I

 ^I^IC ^Pd97Txt

 OPEN

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: QNIRates.RTreb <> EMPTY AND ((QRenting.HBenefit = Yes) OR (GOV IN QOwner1.QMortgage.M[1].QOutsPay))
QNIRates.RTreb = Yes

^I

Earlier, the respondent said they get Housing Benefit or help from ^GOV1 with mortgage. They should usually also get a Rates Rebate. Please check: is their Rates bill 'reduced' - does the Rate Collection Agency take anything off it? If so, change answer to 'Yes'.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
AND: QNIRates.BillRate <> EMPTY AND (QNIRates.NoRate = RateInc)
((QAccomdat.Tenure = Part) OR (QAccomdat.Tenure = Rents)) AND INVOLVING(QAccomdat.Tenure)

^I

Earlier, the respondent said they did NOT rent/part rent this accommodation - the rates could NOT be included in the rent!!

Please check that Tenure is correct.

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
AND: NOT (NatCen <> NI)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.HHStat <> EMPTY OR (Edit = Yes)
RESERVECHECK

RESERVECHECK

COMPUTE IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (QRenting.WSInc = EMPTY OR QRenting.WSInc = NONRESPONSE) OR
 (QRenting.WSInc IN [Sewer, Neith])

AskWater := Yes

COMPUTE IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (QRenting.WSInc = EMPTY OR QRenting.WSInc = NONRESPONSE) OR
 (QRenting.WSInc IN [Water, Neith])

AskSewer := Yes

FRS1004B.QWaterSew

Questions about sewerage and water rates

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *(NatCen = NI) AND ((PAskWater = Yes) OR (PAskSewer = Yes))*

NIWIntro

^I QWaterSew^I

^N

From April 2007, the government has introduced water and sewerage charges to Northern Ireland.

These charges are based on the 'capital value' of your home and are being 'phased in' for the first two years - where you pay one-third of the yearly charge in the first year and two-thirds in the second year. In the third year, from April 2009, you will pay the full amount.

There are two charges - water & sewerage - but you only pay for the services you receive.

By now you should have received a letter/bill indicating how much you will be eligible to pay for the period 1st April 2007 to 31 March 2008.

I will now ask you some questions about these charges.

PRESS <ENTER> TO CONTINUE

STRING[1]

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *PAskWater = Yes*

WaterMet

^I QWaterSew^I

^I^B Northern Ireland only^B - Meters will only be installed for pensioner households, all new properties and those properties connecting to the water supply for the first time.

- (1) Yes
 - (2) No
-

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *PAskWater = Yes*

WaterPay

^I QWaterSew^I

^N

Do you pay water ^RatesOrCharges?

- (1) Yes
 - (2) No
-

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *PAskSewer = Yes*

SewerPay

^I QWaterSew^I
^N
Do you pay sewerage ^RatesOrCharges?

- (1) Yes
- (2) No

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *((WaterPay = Yes) AND (SewerPay = Yes)) AND (NatCen <> NI)*

SewSep

^I QWaterSew^I
^N
Do you pay separate or combined water and sewerage rates or charges?

- (1) Separate
- (2) Combined

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *(SewSep = Separate) OR ((WaterPay = Yes) AND (SewerPay <> Yes))*

WatTime

^I QWaterSew^I
^N
How many times a year do you pay water ^RatesOrCharges?^N

^I^IC Enter times a year.

1..52

ASK IF: *QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *(SewSep = Separate) OR ((WaterPay = Yes) AND (SewerPay <> Yes))*

WatAmt

^I QWaterSew^I
^N
How much did you actually pay last time?

0.01..9997.00

ASK IF: *QCountax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *(SewSep = Separate) OR ((WaterPay = Yes) AND (SewerPay <> Yes))*
AND: *WaterMet = No*

WatAnul

^I QWaterSew^I
 ^N
 How much is your^B annual^B bill?

 0.01..9997.00

COMPUTE IF: *QCountax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *(WatAmt = RESPONSE) AND (WatTime = RESPONSE)*

WatWkly := ((WatAmt * WatTime) / 52)

WARN IF: *QCountax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *Edit = No*
(WatWkly <= 12) AND INVOLVING(WatTime,WatAmt)

^I
 Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

WARN IF: *QCountax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *Edit = No*
AND: *((WatTime = RESPONSE) AND (WatAnul = RESPONSE)) AND (WatAmt = RESPONSE)*
(ABS((WatTime * WatAmt) - WatAnul) <= 25) AND
INVOLVING(WatTime,WatAnul,WatAmt)

^I
 Interviewer: The Annual payment for water rates/charges (WatAnul) is very different from the total for individual payments (WatTime x WatAmt).
 Please check these figures.

ASK IF: *QCountax.CTBand <> EMPTY AND (Scotland <> Yes)*
AND: *(AskWater = Yes) OR (AskSewer = Yes)*
AND: *(SewSep = Separate) OR ((SewerPay = Yes) AND (WaterPay <> Yes))*

SewTime

^I QWaterSew^I
 ^N
 How many times a year do you pay sewerage ^RatesOrCharges?^N

 ^I^IC Enter times a year.

 1..52

ASK IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Separate) OR ((SewerPay = Yes) AND (WaterPay <> Yes))

SewAmt

^I QWaterSew^I
 ^N
 How much did you actually pay last time?
 0.01..9997.00

ASK IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Separate) OR ((SewerPay = Yes) AND (WaterPay <> Yes))

SewAnul

^I QWaterSew^I
 ^N
 How much is your^B annual^B bill?^N

 ^I^IC Code as don't know if respondent has not yet received their annual bill because they have a septic tank.
 0.01..9997.00

COMPUTE IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Separate) OR ((SewerPay = Yes) AND (WaterPay <> Yes))
AND: (SewAmt = RESPONSE) AND (SewTime = RESPONSE)

SewWkly := ((SewAmt * SewTime) / 52)

WARN IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Separate) OR ((SewerPay = Yes) AND (WaterPay <> Yes))
AND: Edit = No
(SewWkly <= 8) AND INVOLVING(SewTime,SewAmt)

^I
 Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

WARN IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Separate) OR ((SewerPay = Yes) AND (WaterPay <> Yes))
AND: Edit = No
AND: ((SewTime = RESPONSE) AND (SewAnul = RESPONSE)) AND (SewAmt = RESPONSE)
(ABS((SewTime * SewAmt) - SewAnul) <= 25) AND
INVOLVING(SewTime,SewAnul,SewAmt)

^I
 Interviewer: The Annual payment for sewerage rates/charges (SewAnul) is very different from the total for individual payments (SewTime x SewAmt).
 Please check these figures.

ASK IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Combined) OR (((NatCen = NI) AND (SewerPay = Yes)) AND (WaterPay = Yes))

WSewTime

^I QWaterSew^I
 ^N
 How many times a year do you pay?^N

 ^I^IC Enter times a year.

 1..52

ASK IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Combined) OR (((NatCen = NI) AND (SewerPay = Yes)) AND (WaterPay = Yes))

WSewAmt

^I QWaterSew^I

 ^I^B Metered Water^B - Charges made via a water meter should be treated as water rate payments and the last amount actually paid entered.

 0.01..9997.00

ASK IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Combined) OR (((NatCen = NI) AND (SewerPay = Yes)) AND (WaterPay = Yes))

WSewAnul

^I QWaterSew^I
 ^N
 How much is your^B annual^B bill?^N

 ^I^IC Code as don't know if respondent has not yet received their annual bill.

 0.01..9997.00

COMPUTE IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Combined) OR (((NatCen = NI) AND (SewerPay = Yes)) AND (WaterPay = Yes))
AND: (WSewAmt = RESPONSE) AND (WSewTime = RESPONSE)

WSewWkly := ((WSewAmt * WSewTime) / 52)

WARN IF: QCountax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Combined) OR (((NatCen = NI) AND (SewerPay = Yes)) AND (WaterPay = Yes))
AND: Edit = No
(WSewWkly < 20) AND INVOLVING(WSewTime,WSewAmt)

^I
 Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

WARN IF: QCountTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (SewSep = Combined) OR (((NatCen = NI) AND (SewerPay = Yes)) AND (WaterPay = Yes))
AND: Edit = No
AND: ((WSewTime = RESPONSE) AND (WSewAnul = RESPONSE)) AND (WSewAmt = RESPONSE)
(ABS((WSewTime * WSewAmt) - WSewAnul) <= 25) AND
INVOLVING(WSewTime,WSewAnul,WSewAmt)

^I

Interviewer: The Annual payment for water/sewerage rates/charges (WSewAnul) is very different from the total for individual payments (WSewTime x WSewAmt).
Please check these figures.

ASK IF: QCountTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)

NIWRedT

^I QWaterSew^I

^I

Any entitlement to Housing Benefit will automatically mean the householder qualifies for the Reduced Tariff.

Those on the Reduced Tariff will pay no more than £60 in 2007-08 for their water/sewerage charges.

- (1) Yes
- (2) No

COMPUTE IF: QCountTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)
AND: (WatAmt = RESPONSE) AND (WatTime = RESPONSE)

WatWkly := ((WatAmt * WatTime) / 52)

WARN IF: QCountTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)
AND: NIWRedT = Yes
(WatWkly < 1.16) AND INVOLVING(WatAmt,WatTime,NIWRedT)

^I

The respondent said they qualified for the Reduced Tariff, therefore the total annual charge cannot be more than £60. Please check your answers

COMPUTE IF: QCountTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)
AND: (SewAmt = RESPONSE) AND (SewTime = RESPONSE)

SewWkly := ((SewAmt * SewTime) / 52)

WARN IF: QCountTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)
AND: NIWRedT = Yes
(SewWkly < 1.16) AND INVOLVING(SewAmt,SewTime,NIWRedT)

^I

The respondent said they qualified for the Reduced Tariff, therefore the total annual charge cannot be more than £60. Please check your answers

COMPUTE IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)
AND: (WSewAmt = RESPONSE) AND (WSewTime = RESPONSE)

WSewWkly := ((WSewAmt * WSewTime) / 52)

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (NatCen = NI) AND (PAskWater = Yes)
AND: NIWRedT = Yes
 (WSewWkly < 1.16) AND INVOLVING(WSewAmt,WSewTime,NIWRedT)

^I

The respondent said they qualified for the Reduced Tariff, therefore the total annual charge cannot be more than £60. Please check your answers

ASK IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
AND: (WaterMet = Yes) AND (NatCen <> NI)

WatRb

^I QWaterSew^I

^I

The vast majority of people have to pay the full water and sewage charges but there are also a few metered who are eligible for assistance under the Vulnerable Groups Scheme (Water Industry Act 1999). Under this scheme people who qualify receive a bill capped at the average charge for their region and do not have to pay the measured charge reflecting their genuine water consumption.

- (1) Yes
 - (2) No
-

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)
AND: (AskWater = Yes) OR (AskSewer = Yes)
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)

AND: (AskWater = Yes) OR (AskSewer = Yes)

RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)

AND: (AskWater = Yes) OR (AskSewer = Yes)

RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)

AND: (AskWater = Yes) OR (AskSewer = Yes)

RESERVECHECK

RESERVECHECK

WARN IF: QCounTax.CTBand <> EMPTY AND (Scotland <> Yes)

AND: (AskWater = Yes) OR (AskSewer = Yes)

RESERVECHECK

RESERVECHECK

FRS1004B.QAccomCharge

Questions on charges with accommodation.

ASK IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*

Charge

^I QAccomCharge^I

^I

Use the combined charge category when it is not possible for respondents to split out separate amounts for ground rent, service charge and maintenance charge. If you record a combined amount, do not record the separate amounts as well.

SET [8] OF

- (1) Ground Rent
 - (2) Chief Rent
 - (3) Service charge
 - (4) Compulsory or regular maintenance charges
 - (5) Site rent (caravans)
 - (6) Factoring (Payments to a land steward)
 - (7) Any other regular payments
 - (8) Combined charges (eg. ground rent, service charge, maintenance charge, factoring etc.)
 - (9) None of these
-

WARN IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*

AND: *((Ground IN Charge) AND (Service IN Charge)) AND (Maint IN Charge))
AND (Combined IN Charge)*

ERROR

^I

Only code combined charges instead of the separate ground rent, service charge, maintenance charge, factoring etc.

ASK IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*

AND: *Other IN Charge*

ChargeO

^I QAccomCharge^I

^N

Please specify the other type of payments.

STRING[50]

ASK IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*

AND: *(Service IN Charge) OR (Combined IN Charge)*

ChIns

^I QAccomCharge^I

^N

Does this service charge include insurance?

- (1) Yes
- (2) No

FRS1004B.QAccomCharge.QChargeAmtPd[]

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[1] := 'Ground Rent'

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[2] := 'Chief Rent'

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[3] := 'Service charge'

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[4] := 'Maintenance charges'

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[5] := 'Site rent'

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[6] := 'Factoring'

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[7] := PCharge0

```
COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
  AND: In loop FOR Idx := 1 TO 8
  AND: Idx IN Charge
```

LCharges[8] := ('Combined charges (eg. ground rent, service charge, ' + 'maintenance charge, factoring etc.)')

ASK IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
 AND: In loop FOR Idx := 1 TO 8
 AND: Idx IN Charge

ChrgAmt

^I QAccomCharge^I

^N

I would now like to ask about the charges you pay for ^LCharges[PSeq].
How much did you pay last time?

0.01..9997.00

COMPUTE IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
 AND: In loop FOR Idx := 1 TO 8
 AND: Idx IN Charge
 AND: ChrgAmt = NONRESPONSE

HMissVar := (HMissVar + 1)

RECORD IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
 AND: In loop FOR Idx := 1 TO 8
 AND: Idx IN Charge
 AND: ChrgAmt > 0

ChrgPx

^I QAccomCharge^I

^I^IC ^Pd97Txt^I

OPEN

ASK IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
AND: Idx IN Charge
AND: ChrgAmt > 0

ChrgPd

^I QAccomCharge^I

^N

How long did this cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
AND: Idx IN Charge
AND: ChrgAmt > 0
AND: ChrgPd = Note

ChrgPx

^I QAccomCharge^I

^I^IC ^Pd97Txt^I

OPEN

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
AND: Idx IN Charge
AND: Edit = Yes
ChrgPd <> Note

^I

Editor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QAccomCharge (continued)

Questions on charges with accommodation.

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: In loop FOR Idx := 1 TO 8
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
AND: None IN Charge
Charge.CARDINAL = 1

^I
'None of these' is an exclusive code for this question.

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
RESERVECHECK

RESERVECHECK

WARN IF: QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*
RESERVECHECK

RESERVECHECK

WARN IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*
RESERVECHECK

RESERVECHECK

WARN IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*
RESERVECHECK

RESERVECHECK

WARN IF: *QAccomdat.Tenure IN [Outright .. Part, RentFree, Squatting]*
RESERVECHECK

RESERVECHECK

FRS1004B.QLodger

```

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
  AND: QAccomdat.HHStat = Conv
  AND: In loop FOR count := 1 TO HHSize
  AND: ((PRec[count].Depend IN [Adult .. DepAd]) AND (PRel.PR[count].R IN
[Child .. NonRel])) AND (ECount < 5)

```

```

ECount := (ECount + 1)

```

```

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
  AND: QAccomdat.HHStat = Conv
  AND: In loop FOR count := 1 TO HHSize
  AND: ((PRec[count].Depend IN [Adult .. DepAd]) AND (PRel.PR[count].R IN
[Child .. NonRel])) AND (ECount < 5)

```

```

ELodger[ECount] := count

```

```

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0

```

```

BordLodg[count].BenUnit := DMBU[[ELodger[count]]]

```

```

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0

```

```

BordLodg[count].PersId := ELodger[count]

```

```

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRel.PR[ELodger[count]].R IN [FChild, FParent, FSib, GChild .. NonRel]

```

```

Relation := Distant

```

```

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRec[ELodger[count]].Depend = DepAd

```

```

Relation := Skip

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRel.PR[ELodger[count]].R IN [Child .. StChild, ILChild .. StParent,
  ILParent .. StSib, ILSib]

```

Relation := Close

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRec[ELodger[count]].Sex = Male

```

heshe := 'he'

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: NOT (PRC[ELodger[count]].Sex = Male)

```

heshe := 'she'

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0

```

LName := DMName[ELodger[count]]

FRS1004B.QLodger.BordLodg[]

```

RECORD IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0

```

BenUnit

^I QLodger^I

Benefit Unit of respondent.

0..7

```

RECORD IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0

```

PersId

^I QLodger^I

Person number of respondent.

0..14

```

ASK IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant

```

ConvBL

^I QLodger^I

^N

(Can I just check), is ^LName^N ^I ...Running Prompt...

- (1) ^N...a^B boarder:^B that is, someone who pays you a^B rent^B for board AND lodging
- (2) ^N...a^B lodger:^B that is, someone who pays you a^B rent for lodging, but not food
- (3) ^N...or neither of these?

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL = Board

```

pay := ' pay for board and lodging'

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL = Lodg

```

pay := ' pay'

```

ASK IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]

```

CvPay

^I QLodger^I

^N

How much rent did ^LName^pay last time it was due, after deducting any Housing Benefit (or Local Housing Allowance)?

0.00..997.00

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

RECORD IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay > 0

```

CvPx

^I QLodger^I

^I^IC ^Pd97Txt

OPEN

```

ASK IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL IN [Board .. Lodg]
AND: CvPay > 0

```

CvPd

^I QLodger^I

^N

How long does that cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL IN [Board .. Lodg]
AND: CvPay > 0
AND: CvPd = Note

```

CvPx

^I QLodger^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QLodger.BordLodg[.Weekly()

Procedure Call

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: QAccomdat.HHStat = Conv  
AND: ECount > 0  
AND: In loop FOR count := 1 TO 5  
AND: ELodger[count] > 0  
AND: PRelation = Distant  
AND: ConvBL IN [Board .. Lodg]  
AND: CvPay > 0
```

PdConW[1] := 1

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: QAccomdat.HHStat = Conv  
AND: ECount > 0  
AND: In loop FOR count := 1 TO 5  
AND: ELodger[count] > 0  
AND: PRelation = Distant  
AND: ConvBL IN [Board .. Lodg]  
AND: CvPay > 0
```

PdConW[2] := 2

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: QAccomdat.HHStat = Conv  
AND: ECount > 0  
AND: In loop FOR count := 1 TO 5  
AND: ELodger[count] > 0  
AND: PRelation = Distant  
AND: ConvBL IN [Board .. Lodg]  
AND: CvPay > 0
```

PdConW[3] := 3

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: QAccomdat.HHStat = Conv  
AND: ECount > 0  
AND: In loop FOR count := 1 TO 5  
AND: ELodger[count] > 0  
AND: PRelation = Distant  
AND: ConvBL IN [Board .. Lodg]  
AND: CvPay > 0
```

PdConW[4] := 4

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: QAccomdat.HHStat = Conv  
AND: ECount > 0  
AND: In loop FOR count := 1 TO 5  
AND: ELodger[count] > 0  
AND: PRelation = Distant  
AND: ConvBL IN [Board .. Lodg]  
AND: CvPay > 0
```

PdConW[5] := 4.333

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay > 0

```

PdConW[7] := 8.67

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay > 0

```

PdConW[8] := 6.5

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay > 0

```

PdConW[9] := 5.78

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: QAccomdat.HHStat = Conv
  AND: ECount > 0
  AND: In loop FOR count := 1 TO 5
  AND: ELodger[count] > 0
  AND: PRelation = Distant
  AND: ConvBL IN [Board .. Lodg]
  AND: CvPay > 0

```

PdConW[13] := 13

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL IN [Board .. Lodg]
AND: CvPay > 0

```

PdConW[26] := 26

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL IN [Board .. Lodg]
AND: CvPay > 0

```

PdConW[52] := 52

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL IN [Board .. Lodg]
AND: CvPay > 0
AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: PRelation = Distant
AND: ConvBL IN [Board .. Lodg]
AND: CvPay > 0
AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QLodger.BordLodg[] (continued)

COMPUTE IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 AND: ECount > 0
 AND: In loop FOR count := 1 TO 5
 AND: ELodger[count] > 0
 AND: PRelation = Distant
 AND: ConvBL IN [Board .. Lodg]
 AND: CvPay > 0
 AND: CvPd IN [OneWeek .. Year]
 AND: LWeekly > 0

CvWkly := LWeekly

ASK IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 AND: ECount > 0
 AND: In loop FOR count := 1 TO 5
 AND: ELodger[count] > 0
 AND: PRelation = Distant
 AND: ConvBL IN [Board .. Lodg]
 AND: (ConvBL = Lodg) AND (CvPay > 0)

CvHt

^I QLodger^I

^N

Is HEATING included in that, or is it paid for separately?

- (1) Included
- (2) Paid for separately

WARN IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 AND: ECount > 0
 AND: In loop FOR count := 1 TO 5
 AND: ELodger[count] > 0
 CvPd <> Note

^I

EDITOR: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

FRS1004B.QLodger (continued)

```
WARN IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: QAccomdat.HHStat = Conv
AND: ECount > 0
AND: In loop FOR count := 1 TO 5
AND: ELodger[count] > 0
AND: Edit = No
AND: BordLodg[count].CvWkly = RESPONSE
      (BordLodg[count].CvWkly < 119) AND
INVOLVING (BordLodg[count].CvPd, BordLodg[count].CvPay)
```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

FRS1004B (continued)**FAMILY RESOURCES SURVEY 2010-2011**

COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 AND: In loop FOR Loop1 := 1 TO 5
 AND: QLodger.BordLodg[Loop1].CvPay > 0

BUHBElig[QLodger.BordLodg[Loop1].BenUnit] := Yes

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 RESERVECHECK

RESERVECHECK

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 RESERVECHECK

RESERVECHECK

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 RESERVECHECK

RESERVECHECK

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
 AND: QAccomdat.HHStat = Conv
 RESERVECHECK

RESERVECHECK

FRS1004B.QSharer

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO HHSIZE
  AND: (PRec[count].Depend = Adult) AND (ECount < 8)

```

```

ECount := (ECount + 1)

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO HHSIZE
  AND: (PRec[count].Depend = Adult) AND (ECount < 8)

```

```

ESharer[ECount] := count

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0

```

```

Sharer[count].BenUnit := DMBU[[ESharer[count]]]

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0

```

```

Sharer[count].PersId := ESharer[count]

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0

```

```

LName := DMName[[ESharer[count]]]

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0
  AND: PRec[ESharer[count]].Sex = Male

```

```

heshe := 'he'

```

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0
  AND: NOT (PRec[ESharer[count]].Sex = Male)

```

```

heshe := 'she'

```

FRS1004B.QSharer.Sharer[]

```
RECORD IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
```

BenUnit

^I QSharer^I

BU number of person

0..7

```
RECORD IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
```

PersId

^I QSharer^I

Person identifier.

0..14

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: PersId = 1
```

**Preamb := ('Now I'd like to ask how much each of you' + ' pays towards
certain things.')**

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: NOT (PersId = 1)
```

Preamb := ''

```

ASK IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1

```

SRentAmt

^I QSharer^I

^N

How much rent did ^LName pay last time it was due, after deducting any Housing Benefit (or Local Housing Allowance)?

0.00..997.00

```

COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

RECORD IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

SRentPx

^I QSharer^I

^I^IC ^Pd97Txt

OPEN

ASK IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

SRentPd

^I QSharer^I

^N

How long does that cover?

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: (QCountTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0
AND: SRentPd = Note

SRentPx

^I QSharer^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QSharer.Sharer[.Weekly()

Procedure Call

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
AND: In loop FOR count := 1 TO 8  
AND: ESharer[count] > 0  
AND: BenUnit > 1  
AND: SRentAmt > 0
```

PdConW[1] := 1

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
AND: In loop FOR count := 1 TO 8  
AND: ESharer[count] > 0  
AND: BenUnit > 1  
AND: SRentAmt > 0
```

PdConW[2] := 2

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
AND: In loop FOR count := 1 TO 8  
AND: ESharer[count] > 0  
AND: BenUnit > 1  
AND: SRentAmt > 0
```

PdConW[3] := 3

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
AND: In loop FOR count := 1 TO 8  
AND: ESharer[count] > 0  
AND: BenUnit > 1  
AND: SRentAmt > 0
```

PdConW[4] := 4

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
AND: In loop FOR count := 1 TO 8  
AND: ESharer[count] > 0  
AND: BenUnit > 1  
AND: SRentAmt > 0
```

PdConW[5] := 4.333

```
COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
Yes)  
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
AND: In loop FOR count := 1 TO 8  
AND: ESharer[count] > 0  
AND: BenUnit > 1  
AND: SRentAmt > 0
```

PdConW[7] := 8.67

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

PdConW[8] := 6.5

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

PdConW[9] := 5.78

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

PdConW[10] := 5.2

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

PdConW[13] := 13

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

PdConW[26] := 26

```

COMPUTE IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: BenUnit > 1
AND: SRentAmt > 0

```

PdConW[52] := 52

```
COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
    Yes)  
    AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
    AND: In loop FOR count := 1 TO 8  
    AND: ESharer[count] > 0  
    AND: BenUnit > 1  
    AND: SRentAmt > 0  
    AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])
```

```
PWeekly := (PAmount / PdConW[ORD(PPeriod)])
```

```
COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =  
    Yes)  
    AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)  
    AND: In loop FOR count := 1 TO 8  
    AND: ESharer[count] > 0  
    AND: BenUnit > 1  
    AND: SRentAmt > 0  
    AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))
```

```
PWeekly := 0
```

FRS1004B.QSharer.Sharer[] (continued)

```

COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0
  AND: BenUnit > 1
  AND: SRentAmt > 0
  AND: SRentPd IN [OneWeek .. Year]
  AND: LWeekly >= 0

```

SRntWkly := LWeekly

```

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0
  SRentPd <> Note

```

^I

EDITOR: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

```

COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0
  AND: BenUnit = 1

```

How := 'Apart from any rent, how'

```

COMPUTE IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit =
  Yes)
  AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
  AND: In loop FOR count := 1 TO 8
  AND: ESharer[count] > 0
  AND: NOT (BenUnit = 1)

```

How := 'How'

FRS1004B.QSharer (continued)

WARN IF: (QCountax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
AND: In loop FOR count := 1 TO 8
AND: ESharer[count] > 0
AND: Edit = No
AND: Sharer[count].SRntWkly = RESPONSE
(Sharer[count].SRntWkly < 103) AND
INVOLVING(Sharer [count] .SRentPd,Sharer [count] .SRentAmt)

^I

Are you sure? Enter here only the RESPONDENT'S SHARE of the household rent.

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
RESERVECHECK

RESERVECHECK

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
RESERVECHECK

RESERVECHECK

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
RESERVECHECK

RESERVECHECK

WARN IF: (QCounTax.CTBand <> EMPTY OR QNIRates.BillRate <> EMPTY) OR (Edit = Yes)
AND: (QAccomdat.HHStat = Shared) AND (NewBU > 1)
RESERVECHECK

RESERVECHECK

FRS1004B.QProperty

Questions about other property

ASK IF: *QAccomdat.SubLet = Yes*

SubRent

^I QProperty^I

^N

You mentioned earlier that you let, or sub-let, part of this accommodation to someone outside your household.

How much rent have you received from this in the last 12 months, ie. since ^DLYear : that's BEFORE deducting any income tax that might be due on it?

0.00..99999.97

COMPUTE IF: *QAccomdat.SubLet = Yes*

AND: *SubRent = RESPONSE*

SubWkly := (SubRent / 52)

WARN IF: *QAccomdat.SubLet = Yes*

AND: *Edit = No*

(SubWkly < 180) AND INVOLVING (SubRent)

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

COMPUTE IF: *QAccomdat.SubLet = Yes*

AND: *SubRent = NONRESPONSE*

HMissVar := (HMissVar + 1)

ASK IF: *QAccomdat.SubLet = Yes*

SubAllow

^I QProperty^I

^N

And is that BEFORE or AFTER deducting allowable expenses?

(1) Before

(2) After

COMPUTE IF: *QAccomdat.SubLet = Yes*

Im := 'Apart from that, in'

COMPUTE IF: *NOT (QAccomdat.SubLet = Yes)*

Im := 'In'

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

*COMPUTE ALWAYS:*PersList[1] := ''

*COMPUTE ALWAYS:*PersList[2] := ''

*COMPUTE ALWAYS:*PersList[3] := ''

*COMPUTE ALWAYS:*Elig[1] := 0

*COMPUTE ALWAYS:*Elig[2] := 0

*COMPUTE ALWAYS:*Elig[3] := 0

*COMPUTE IF: In loop FOR Loop1 := 1 TO HHSIZE**AND: ((DMAge[Loop1] IN [16 .. 50]) AND (PRec[Loop1].Sex = Female)) OR
(DMAge[Loop1] IN [0 .. 15])*

```
PersList[1] := (PersList[1] + STR(Loop1,2) + ' : ' + DMName[Loop1]
+ '
')
```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSIZE
    AND: ((DMAge[Loop1] IN [16 .. 50]) AND (PRec[Loop1].Sex = Female)) OR
    (DMAge[Loop1] IN [0 .. 15])

```

```

Elig[1] := (Elig[1] + 1)

```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSIZE
    AND: PRec[Loop1].TypeEd IN [Nursery .. NonAdvFE]

```

```

PersList[2] := (PersList[2] + STR(Loop1,2,0) + ' : ' +
DMName[Loop1] + '
')

```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSIZE
    AND: PRec[Loop1].TypeEd IN [Nursery .. NonAdvFE]

```

```

Elig[2] := (Elig[2] + 1)

```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSIZE
    AND: (DMAge[Loop1] IN [2 .. 18]) AND (PRec[Loop1].TypeEd IN [Nursery ..
    NonAdvFE])

```

```

PersList[3] := (PersList[3] + STR(Loop1,2,0) + ' : ' +
DMName[Loop1] + '
')

```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO HHSIZE
    AND: (DMAge[Loop1] IN [2 .. 18]) AND (PRec[Loop1].TypeEd IN [Nursery ..
    NonAdvFE])

```

```

Elig[3] := (Elig[3] + 1)

```

FRS1004B.QWelfare

Questions about free meals etc

```
COMPUTE IF: PAllCh > 0
```

```
incl_child := ' (including any of your children under 16) '
```

```
COMPUTE IF: PAllCh = 1
```

```
incl_child := ' (including your child under 16) '
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 1
```

```
READ_OUT := 'Individual prompt...'
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: Elig[1] >= 1
```

```
HStart := (N + '...any Healthy Start Vouchers?' + N)
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: NOT (Elig[1] >= 1)
```

```
HStart := ''
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: Elig[2] >= 1
```

```
SchMilk := (N + '...any free school milk?' + N)
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: NOT (Elig[2] >= 1)
```

```
SchMilk := ''
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: Elig[3] >= 1
```

```
SchMeal := (N + '...any free school meals?' + N)
```

```
COMPUTE IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: NOT (Elig[3] >= 1)
```

```
SchMeal := ''
```

ASK IF: $((Elig[1] + Elig[2]) + Elig[3]) > 0$

FreeItem

^I QWelfare^I

^I^B

Healthy Start Vouchers^B

Healthy Start vouchers are available to pregnant women and to families with children aged under four, who are also in receipt of Income Support, income-based jobseeker's Allowance, Child Tax Credit or Working Tax Credit run-on. They are also available to any pregnant woman aged under 18. Vouchers are claimed via a health professional (e.g. midwife or health visitor etc.). Forms have to be completed and the application countersigned by the health professional.

Healthy Start vouchers are worth:

£3.10 per week if you are pregnant (one voucher).

£6.20 per week for each baby aged under one (two vouchers).

£3.10 per week for each child aged over one and under four (one voucher).

^B

Health in Pregnancy Grant^B

Healthy Start vouchers should not be confused with the Health in Pregnancy Grant which is a lump sum payment of £190. Health in Pregnancy grant is claimed during contact with a health professional such as a mid-wife. Health in Pregnancy Grant receipt should be recorded at the relevant benefits question.

^B

Free School Meals in Scotland^B

The Scottish Government has an agreement with local government to extend entitlement to free school lunches.

Entitlement to free school lunches was extended to pupils whose parents or carers are in receipt of both maximum working tax credit and maximum child tax credit from August 2009.

Legislation was passed in November 2008 to enable local authorities to provide free school lunches to all pupils in the first three years of primary school from August 2010.

SET [3] OF

(1) ^HStart

(2) ^SchMilk

(3) ^SchMeal

(4) None of these

CHECK IF: $((Elig[1] + Elig[2]) + Elig[3]) > 0$
AND: None IN FreeItem
FreeItem.CARDINAL = 1

^I

'None of these' is an exclusive code for this question.

CHECK IF: $((Elig[1] + Elig[2]) + Elig[3]) > 0$
AND: HStrt IN FreeItem
Elig[1] > 0

^I

Code 1 is not valid for this question.

CHECK IF: $((Elig[1] + Elig[2]) + Elig[3]) > 0$
AND: SMilk IN FreeItem
Elig[2] > 0

^I

Code 2 is not valid for this question.

```
CHECK IF: ((Elig[1] + Elig[2]) + Elig[3]) > 0  
AND: SMeal IN FreeItem  
Elig[3] > 0
```

^I

Code 3 is not valid for this question.

FRS1004B.QWelfare.HStQ[]

```
RECORD IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
```

Person

^I QWelfare^I

Person identifier.

0..14

```
RECORD IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
```

BenUnit

^I QWelfare^I

BU number of recipient.

0..7

```
COMPUTE IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: PHHSize = 1
```

HSVPer := 1

```
COMPUTE IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: PHHSize = 1
```

NameOf := 'you'

```

ASK IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (PHHSize = 1)

```

HSVPer

^I QWelfare^I

^I

The full allowance of Healthy Start Vouchers are sent together in one letter every four weeks.

Healthy Start Vouchers are received as follows:

One voucher per week for a pregnant woman

Two vouchers per week for^B each^B baby aged under one

One voucher per week for^B each^B child aged one or over and under four.

^B

Each voucher^B is worth £3.10

Healthy Start vouchers should not be confused with the Health in Pregnancy Grant which is a lump sum payment of £190.

0..14

```

CHECK IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (PHHSize = 1)
  AND: HSVPer = RESPONSE
  (HSVPer > 0) AND (HSVPer <= PHHSize)

```

^I

This code is not valid for this question.

```

WARN IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (PHHSize = 1)
  AND: HSVPer = RESPONSE
  AND: PRec[HSVPer].Sex = Female
  (DMAge[HSVPer] <= 50) AND INVOLVING(HSVPer)

```

^I

Only pregnant women aged between 12 and 50 are eligible to healthy start vouchers on their own behalf.

Please check whether the healthy start voucher is received for the child. The child should be recorded as the recipient.

```

WARN IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (PHHSize = 1)
  AND: HSVPer = RESPONSE
  AND: (PRec[HSVPer].Sex = Female) AND (DMAge[HSVPer] >= 16)
  ERROR AND INVOLVING(HSVPer)

```

^I

Women will receive the voucher for pregnancy. Are you sure that applies in this case. If the voucher is received for the child then the child should be recorded as the recipient.

```

WARN IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (PHHSize = 1)
  AND: HSVPer = RESPONSE
  AND: DMAge[HSVPer] IN [4 .. 15]
  ERROR AND INVOLVING(HSVPer)

```

^I

Healthy Start vouchers are only given to pregnant women and children aged under 4. Please check whether the child receives free school meals.

```

COMPUTE IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (PHHSize = 1)

```

NameOf := DMName [HSVPer]

```

COMPUTE IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)

```

Person := HSVPer

```

WARN IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  RESERVECHECK

```

RESERVECHECK

```

ASK IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: Elig[1] > 1

```

HSVIntro

^I QWelfare ^I

^I^IC PROMPT -^I^N Has anyone else received Healthy Start Vouchers?

- (1) Yes
 - (2) No
-

```

COMPUTE IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  AND: NOT (Elig[1] > 1)

```

HSVIntro := No

```

WARN IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
  RESERVECHECK

```

RESERVECHECK

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: HStrt IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QWelfare (continued)

Questions about free meals etc

```
COMPUTE IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (HStQ[ii - 1].HSVIntro = Yes)
```

```
HStQ[ii].BenUnit := DMBU[[HStQ[ii].HSVPer]
```

```
CHECK IF: HStrt IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  AND: jj < ii
  HStQ[jj].HSVPer <> HStQ[ii].HSVPer
```

^I

You have already entered this person number.

FRS1004B.QWelfare.SMkQ[]

```

RECORD IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)

```

Person

^I QWelfare^I

Person identifier.

0..14

```

RECORD IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)

```

BenUnit

^I QWelfare^I

BU number of recipient.

0..7

```

ASK IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)

```

SMkPer

^I QWelfare^I

^I

Free School Milk may be available if the child is in full-time education at a state school. The rules for offering Free School Milk differ by Local Education Authority.
The age of children for whom it can be claimed also varies by LEA.

0..14

```

COMPUTE IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)

```

Person := SMkPer

```

ASK IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)

```

SMkIt

^I QWelfare^I

^N

Thinking just of the past seven days ending yesterday - how many cartons or bottles did

^DMName[SmkPer] receive?

0..97

```

WARN IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
  AND: Edit = No
  SMkIt <= 6

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

ASK IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
  AND: Elig[2] > 1

```

SMIntro

^I QWelfare^I

^I^IC PROMPT ^I^N Has any other child had any free school milk during the past seven days ending yesterday?^N

^I

Only applicable to children at state schools.

- (1) Yes
 - (2) No
-

```

COMPUTE IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
  AND: NOT (Elig[2] > 1)

```

SMIntro := No

```

WARN IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
  RESERVECHECK

```

RESERVECHECK

```
WARN IF: SMilk IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: SMilk IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: SMilk IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: SMilk IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: SMilk IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: SMilk IN FreeItem
      AND: In loop FOR ii := 1 TO 14
      AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QWelfare (continued)

Questions about free meals etc

```
COMPUTE IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMkQ[ii - 1].SMIntro = Yes)

SMkQ[ii].BenUnit := DMBU[[SMkQ[ii].SMkPer]
```

```
CHECK IF: SMilk IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  AND: jj < ii
  SMkQ[jj].SMkPer <> SMkQ[ii].SMkPer
```

^I

You have already entered this person number.

FRS1004B.QWelfare.SMIQ[]

```

RECORD IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)

```

BenUnit

^I QWelfare^I

BU number of recipient.

0..7

```

RECORD IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)

```

Person

^I QWelfare^I

Person identifier.

0..14

```

COMPUTE IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: PHHSize = 1

```

MLPer := 1

```

COMPUTE IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: PHHSize = 1

```

have_you := 'have you'

```

ASK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: NOT (PHHSize = 1)

```

MLPer

^I QWelfare^I

^N

Who received the^B free school meals^B?

^N ^I

Only applicable to children at state schools. Can include 16-18 year olds.

^IC Type in person number.^I^N

^PersList[3]

0..14

```

COMPUTE IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: NOT (PHHSize = 1)

```

```

have_you := ('has ' + DMName[ [MLPer] )

```

```

COMPUTE IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)

```

```

Person := MLPer

```

```

ASK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: MLPer = RESPONSE

```

SMIIt

^I QWelfare^I

^N

Thinking just of the PAST seven days ending yesterday, how many free school meals ^have_you had?

0..97

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: MLPer = RESPONSE
  AND: Edit = No
  SMIIt <= 21

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

ASK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: Elig[3] > 1

```

MLIntro

^I QWelfare^I

^I^IC PROMPT -^I^N Has any other child had any free school meals during the past seven days ending yesterday?

^N ^I

Only applicable to children at state schools. Can include 16-18 year olds.

- (1) Yes
 - (2) No
-

```

COMPUTE IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMIQ[ii - 1].MLIntro = Yes)
  AND: NOT (Elig[3] > 1)

```

```

MLIntro := No

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

```

WARN IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMQ[ii - 1].MLIntro = Yes)
  RESERVECHECK

RESERVECHECK

```

FRS1004B.QWelfare (continued)

Questions about free meals etc

```

COMPUTE IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: (ii = 1) OR (SMLQ[ii - 1].MLIntro = Yes)

```

```

SMLQ[ii].BenUnit := DMBU[[SMLQ[ii].MLPer]

```

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  AND: jj < ii
  SMLQ[jj].MLPer <> SMLQ[ii].MLPer

```

^I

You have already entered this person number.

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  RESERVECHECK

```

RESERVECHECK

```

CHECK IF: SMeal IN FreeItem
  AND: In loop FOR ii := 1 TO 14
  AND: In loop FOR jj := 1 TO 14
  RESERVECHECK

```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
AND: In loop FOR jj := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
AND: In loop FOR jj := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```

CHECK IF: SMeal IN FreeItem
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)

```

MedPay

^I QWelfare^I

^N

In the past 4 weeks, has anyone in this household paid for something on prescription, including anyone who has a pre-payment certificate?^N

^I^IC Only record prescriptions that were paid for in the last 4 weeks or were covered by a pre-payment certificate. Include pre-payment certificates (PPC) where prescription charges are paid 3 months or 12 months in advance.

(1) Yes

(2) No

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: (DMAge[ii] >= 16) AND (PenFlag[ii] <> Yes)

```

```

HHPer[ii] := DMName[[ii]

```

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: NOT ((DMAge[ii] >= 16) AND (PenFlag[ii] <> Yes))

```

```

HHPer[ii] := ''

```

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes

```

MedWho

^I QWelfare^I

^N

Who paid for something on prescription?^N

^I^IC Include those who pay for pre-payment certificates (PPC) where prescription charges are paid 3 months or 12 months in advance.

NB: Code ALL that apply from the list below.

SET [14] OF

- (1) ^HHPer[1]
 - (2) ^HHPer[2]
 - (3) ^HHPer[3]
 - (4) ^HHPer[4]
 - (5) ^HHPer[5]
 - (6) ^HHPer[6]
 - (7) ^HHPer[7]
 - (8) ^HHPer[8]
 - (9) ^HHPer[9]
 - (10) ^HHPer[10]
 - (11) ^HHPer[11]
 - (12) ^HHPer[12]
 - (13) ^HHPer[13]
 - (14) ^HHPer[14]
-

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
(HHPer[ii] <> '') AND INVOLVING(MedWho)

```

^I

^ii is an invalid person at MedWho. Please unselect this person no..

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```
CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK
```

RESERVECHECK

FRS1004B.QWelfare.MedQ[]

```

RECORD IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho

```

BenUnit

^I QWelfare^I
^I(^DMName[ii])^I

BU number of recipient.

0..7

```

RECORD IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho

```

Person

^I QWelfare^I
^I(^DMName[ii])^I

Person identifier.

0..14

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho

```

Person := PHHPers

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho

```

LName := DMName[Person]

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho

```

MedPrPy

^I QWelfare^I
^I(^DMName[ii])^I

^N

Can I just check, did ^LName have a pre-payment prescription certificate (PPC) covering the last 4 weeks?

- (1) Yes
- (2) No

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
AND: MedPrPy = Yes

```

MedPrPd

```

^I QWelfare^I
^I(^DMName[ii])^I

```

^N

What period did the pre-payment prescription certificate cover?^N

^I^IC If more than one 3 month pre-payment certificate mentioned code as 1 '3 month'.

- (1) 3 months
- (2) 12 months

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho

```

MedNum

```

^I QWelfare^I
^I(^DMName[ii])^I

```

^N

How many items on prescription did ^LName pay for (including any pre-payment certificate) in the last 4 weeks?^N

^I^IC Please record each item (e.g. medicine etc) on prescription not number of prescriptions. There could be more than one item per prescription.

0..97

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
AND: MedNum = RESPONSE
MedNum <= 10

```

^I

Are you sure? This seems a high number of items. Please check. If correct suppress check and explain circumstances in a note.

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
AND: MedNum = RESPONSE
(MedNum <> 0) AND INVOLVING (MedWho [, MedPay, MedNum)

```

^I

Please check. Respondent is recorded as paying for no prescription items in the last 4 weeks. Please amend the answer to MedWho 'Who pays for prescriptions' or MedPay 'Whether anyone in the household paid for a prescription in the last 4 weeks' as appropriate.

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedNum = 1

```

Were_items := 'Was this item'

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: NOT (MedNum = 1)

```

Were_items := 'Were any of these items'

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho

```

MedRep

```

^I QWelfare^I
^I(^DMName[ii])^I

^N
^Were_items on a repeat prescription?

(1)   Yes
(2)   No

```

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedNum > 1

```

MedRpNm

```

^I QWelfare^I
^I(^DMName[ii])^I

^N
How many of these items were on a repeat prescription?

0..97

```

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedNum > 1
  AND: MedRpNm > 10
  ERROR

```

```

^I
Are you sure? This seems a high number of items. Please check. If correct suppress check and explain
circumstances in a note.

```

```

CHECK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedNum > 1
  AND: (MedNum = RESPONSE) AND (MedRpNm = RESPONSE)
  (MedRpNm <= MedNum) AND INVOLVING(MedRpNm)

```

^I

Are you sure? The number of items on repeat prescription exceeds the number of items the respondent has in total on prescription? Please amend the answer.

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: NOT (MedNum > 1)

```

MedRpNm := MedNum

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm
  AND: MedRpNm = 1

```

this := 'this'

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm
  AND: NOT (MedRpNm = 1)
  AND: (mm <> 11) AND (SUBSTRING (STR (mm), LEN (STR (mm)), 1) = 1)

```

this := ('the ' + STR(mm) + 'st')

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm
  AND: NOT (MedRpNm = 1)
  AND: (mm <> 12) AND (SUBSTRING (STR (mm), LEN (STR (mm)), 1) = 2)

```

this := ('the ' + STR(mm) + 'nd')

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm
  AND: NOT (MedRpNm = 1)
  AND: (mm <> 13) AND (SUBSTRING (STR (mm), LEN (STR (mm)), 1) = 3)

```

```

this := ('the ' + STR(mm) + 'rd')

```

```

COMPUTE IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm
  AND: NOT (MedRpNm = 1)
  AND: NOT ((mm <> 13) AND (SUBSTRING (STR (mm), LEN (STR (mm)), 1) = 3))

```

```

this := ('the ' + STR(mm) + 'th')

```

```

ASK IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm

```

Med12M

^I QWelfare^I
^I(^DMName[ii])^I

^N

How many times have you paid for ^this item on repeat prescription in the last 12 months?^N

^I^IC Establish the duration of the repeated prescription item (e.g. prescription renewed and paid for every month) to help identify number of times the item has been paid for in the last 12 months.

0..97

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
  AND: MedPay = Yes
  AND: In loop FOR ii := 1 TO 14
  AND: ii IN MedWho
  AND: MedRep = Yes
  AND: MedRpNm > 0
  AND: In loop FOR mm := 1 TO MedRpNm
  AND: Med12M[mm] > 12
  ERROR

```

^I

Are you sure? This seems a high number of times for this item to be repeated in the last 12 months. Please check. If correct suppress check and explain circumstances in a note.

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK

```

RESERVECHECK

```
WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK
```

RESERVECHECK

```
WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK
```

RESERVECHECK

```
WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
AND: ii IN MedWho
RESERVECHECK
```

RESERVECHECK

FRS1004B.QWelfare (continued)

Questions about free meals etc

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN IF: ((WorkingAge > 0) AND (Wales <> Yes)) AND (NatCen <> NI)
AND: MedPay = Yes
AND: In loop FOR ii := 1 TO 14
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN ALWAYS:
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN ALWAYS:
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN ALWAYS:
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN ALWAYS:
RESERVECHECK

```

```

RESERVECHECK

```

```

WARN ALWAYS:
RESERVECHECK

```

```

RESERVECHECK

```

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

WARN ALWAYS:

RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.HStQ[Loop1].HSVPer = RESPONSE

```

```

Loop5 := QWelfare.HStQ[Loop1].HSVPer

```

```

CHECK IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.HStQ[Loop1].HSVPer = RESPONSE
    ((Loop5 > 0) AND (Loop5 <= HHSIZE)) AND ((IN(DMAge[Loop5], [16..50])) AND
    (PRC[Loop5].Sex = Female)) OR (IN(DMAge[Loop5], [0..15]))

```

^I

This code is not valid for this question.

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.HStQ[Loop1].HSVPer = RESPONSE

```

```

QWelfare.HStQ[Loop1].BenUnit := DMBU[Loop5]

```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.SMkQ[Loop1].SMkPer = RESPONSE

```

```

Loop5 := QWelfare.SMkQ[Loop1].SMkPer

```

```

CHECK IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.SMkQ[Loop1].SMkPer = RESPONSE
    ((Loop5 > 0) AND (Loop5 <= HHSIZE)) AND (IN(PRC[Loop5].TypeEd, [???]))

```

^I

This code is not valid for this question.

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.SMkQ[Loop1].SMkPer = RESPONSE

```

```

QWelfare.SMkQ[Loop1].BenUnit := DMBU[Loop5]

```

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.SMlQ[Loop1].MLPer = RESPONSE

```

```

Loop5 := QWelfare.SMlQ[Loop1].MLPer

```

```

CHECK IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.SMlQ[Loop1].MLPer = RESPONSE
    (((Loop5 > 0) AND (Loop5 <= HHSIZE)) AND (IN(DMAge[Loop5], [2..18]))) AND
    (IN(PRC[Loop5].TypeEd, [???]))

```

^I

This code is not valid for this question.

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 5
    AND: QWelfare.SMlQ[Loop1].MLPer = RESPONSE

```

```

QWelfare.SMlQ[Loop1].BenUnit := DMBU[Loop5]

```

```

WARN IF: In loop FOR Loop1 := 1 TO 5
  AND: PRec[QWelfare.SMIQ[Loop1].MLPer].TypeEd IN [Nursery, Primry, MidPri
    .. NonAdvFE]
  (IN(QWelfare.SMIQ[Loop1].SMIt, [0..5])) AND
  INVOLVING(QWelfare.SMIQ[Loop1].SMIt)

```

^I

That's ^QWelfare.SMIQ[Loop1].SMIt meals - for this type of school the weekly maximum would normally be 5 (ie. one meal per day) - please check.

```

WARN ALWAYS:
  RESERVECHECK

  RESERVECHECK

```

```

WARN ALWAYS:
  RESERVECHECK

  RESERVECHECK

```

```

WARN ALWAYS:
  RESERVECHECK

  RESERVECHECK

```

```

WARN ALWAYS:
  RESERVECHECK

  RESERVECHECK

```

```

COMPUTE IF: HStrt IN QWelfare.FreeItem

```

```

QAccomdat.HStart := Yes

```

```

COMPUTE IF: NOT (HStrt IN QWelfare.FreeItem)

```

```

QAccomdat.HStart := No

```

```

COMPUTE IF: SMeal IN QWelfare.FreeItem

```

```

QAccomdat.SchMeal := Yes

```

```

COMPUTE IF: NOT (SMeal IN QWelfare.FreeItem)

```

```

QAccomdat.SchMeal := No

```

```

COMPUTE IF: SMilk IN QWelfare.FreeItem

```

```

QAccomdat.SchMilk := Yes

```

```

COMPUTE IF: NOT (SMilk IN QWelfare.FreeItem)

```

```

QAccomdat.SchMilk := No

```

```

COMPUTE IF: AllCh > 0

```

```

childtxt := 'child'

```

```
COMPUTE IF: AllCh > 0  
      AND: AllCh > 1  
  
childtxt := 'children'
```

FRS1004B.QChCare

Questions about child care

ASK IF: AllCh > 0

Disp

^I QChCare^I

^N

The next questions are about childcare for your ^childtxt. This includes all types of childcare such as playschool or nursery school or a childminder^B as well as^B relatives or friends who look after your ^childtxt.

(1) Press <Enter> to continue.

FRS1004B.QChCare.QCh1Care

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]
    AND: DMNumParn[Index2] = 1

```

```

PNames := DMName[ [DMParent1[[Index2]] ]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]
    AND: DMNumParn[Index2] > 1

```

```

PNames := (DMName[ [DMParent1[[Index2]] ] + ' and ' +
DMName[ [DMParent2[[Index2]] ] )

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]

```

```

ChTypeEd := DMTypeEd[[Index2]]

```

FRS1004B.QChCare.QCh1Care.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

Person

^I QChCare^I

Person identifier.

0..14

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

AttTxt[[1]] := ''

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

AttTxt[[2]] := ''

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

AttTxt[[3]] := ''

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

AttTxt[[4]] := ''

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

AttTxt[[5]] := ''

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

```

AttTxt[[6] := ''

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

```

AttTxt[[9] := ''

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

```

AttTxt[[7] := 'Holiday scheme / club'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

```

AttTxt[[8] := 'Children's centres / integrated centres'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]

```

```

AttTxt[[10] := 'Other provider'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: ChAge < 6

```

```

AttTxt[[1] := 'Playgroup or pre school'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: ChAge < 6

```

```

AttTxt[[2] := 'Day nursery or workplace creche'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: ChAge < 6

```

```

AttTxt[[3] := 'Nursery school'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: ChAge IN [3 .. 5]

```

```

AttTxt[[4] := 'Infant's school'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: ChAge IN [2 .. 5]

```

```

AttTxt[[5] := 'Primary school'

```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]
    AND: ChAge >= 2
```

```
AttTxt[[6] := 'Breakfast / After school club'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: DMAge[Index2] IN [0 .. 15]
    AND: ChAge >= 2
```

```
AttTxt[[9] := 'SPONTANEOUS ONLY - Boarding school'
```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]

```

ChAtt

^I QChCare^I

^I

Count all after school activities as 'After school clubs'. Count anything organised by, or accessed via, the school. This could be on/off the school's site and could include activities at the weekend.

This includes all study support or recreational activities before or after school hours. Private tuition lessons that have not been arranged via the school should not be included.

^B

Pre-school care^B

^B

Pre-school and play groups^b

Attended by children between two and five years old.

They offer sessions from two-and-a-half hours to four hours, during term time, every day or for several days a week.

^B

Day nurseries^B

Attended by children between the ages of six weeks and five years.

Run by the council, the community, based in the workplace or privately run.

Opening times are from around 7am to 7pm, 50 weeks of the year. Children can attend full or part time.

^B

Nursery schools and classes^B

Attended by children between three and five years old.

Run as part of the state education system or by private or voluntary sector organisations.

Open during school hours, usually 9am to 3:30pm in term time for full or half-day sessions.

^B

Children's Centres^B

Children's centres may also be known as Sure Start Children's centres.

- Support children under 5 years old and their families providing easy access to a range of integrated services.

- May offer anything from short weekly sessions up to full-time day care.

- All centres in most disadvantaged areas provide full day care.

- Local Education Authorities have overall responsibility for the centres but some centres may be run by Private, Voluntary and Independent sector organisations

^B

Integrated Centres^B

Aim to meet the demands of parents in work or training and to promote training and community development.

Services offered can include:

- Full day care for 0-3 year old

- Care for 3-4 year olds before or after an early years education place such as a nursery or playgroup.

- Before or after school or holiday care for primary school children

- Before or after school or holiday activities for those aged 11-14

- Staffed play provision up to the age of 16 (for those with special needs) where children can come and go as they please.

^B

Combined Centres^B

If any respondent reports that their child attends a Combined Centre record this with Children's centres / integrated centres.

- SET [10] OF
- (1) ^AttTxt[1]
 - (2) ^AttTxt[2]
 - (3) ^AttTxt[3]
 - (4) ^AttTxt[4]
 - (5) ^AttTxt[5]
 - (6) ^AttTxt[6]
 - (7) ^AttTxt[7]
 - (8) ^AttTxt[8]
 - (9) ^AttTxt[9]
 - (10) SPONTANEOUS ONLY - Other
 - (11) None of the above

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: None IN ChAtt
ChAtt.CARDINAL = 1

^I
 'None of the above' is an exclusive code.

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: In loop FOR ii := 1 TO 9
AND: ii IN ChAtt
AttTxt[ii] <> ''

^I
 Code ^ii is not valid for this child.

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: In loop FOR ii := 1 TO 9
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: In loop FOR ii := 1 TO 9
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: ChAtt = RESPONSE
AND: (ChAge = 4) AND (ChTypeEd = Nursery)
(((IN(PlayGp, ChAtt)) OR (IN(DayNurse, ChAtt))) OR (IN(Nursery, ChAtt))) AND INVOLVING(HHG.P [].TypeEd[Person], ChAtt)

Earlier the respondent said that this child attended Nursery school/Nursery Class/ Playgroup/Pre-school. Please check whether the child attended play group, pre-school, day nursery, workplace crèche or nursery school in the seven days ending Sunday. Please check and ensure the education and childcare questions are consistent. Make a note if appropriate.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: ChAtt = RESPONSE
AND: (ChAge = 4) AND (ChTypeEd = Primry)
      ((IN(InfantS,ChAtt)) OR (IN(PrimaryS,ChAtt))) AND
      INVOLVING(HHG.P[] .TypeEd[Person],ChAtt)

```

Earlier the respondent said that this child attended a State run primary school. Please check whether the child attended either infant's school or primary school in the seven days ending Sunday. Please check and ensure the education and childcare questions are consistent. Make a note if appropriate.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: ChAtt = RESPONSE
AND: (ChAge = 4) AND (ChTypeEd = MidPri)
      (IN(PrimaryS,ChAtt)) AND INVOLVING(HHG.P[] .TypeEd[Person],ChAtt)

```

Earlier the respondent said that this child attended a Middle-deemed Primary school. Please check whether the child attended primary school in the seven days ending Sunday. Please check and ensure the education and childcare questions are consistent. Make a note if appropriate.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: ChAtt = RESPONSE
AND: (ChAge = 4) AND ((PlayGp IN ChAtt) OR (DayNurse IN ChAtt)) OR (Nursery
      IN ChAtt))
      (ChTypeEd = Nursery) AND
      INVOLVING(HHG.P[] .FTEd[Person],HHG.P[] .TypeEd[Person],ChAtt)

```

The respondent did not report that their child attended a Nursery school/Nursery Class/ Playgroup/Pre-school, earlier at TypeEd. Please check the answer to FTED and ensure the education and childcare questions are consistent. Make a note if appropriate.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: ChAtt = RESPONSE
AND: (ChAge = 4) AND ((InfantS IN ChAtt) OR (PrimaryS IN ChAtt))
      (ChTypeEd = Primry) AND
      INVOLVING(HHG.P[] .FTEd[Person],HHG.P[] .TypeEd[Person],ChAtt)

```

The respondent did not report that their child attended a State run primary school or 'middle-deemed primary school', earlier at TypeEd. Please check the answer to FTED and ensure the education and childcare questions are consistent. Make a note if appropriate.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK

```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

FRS1004B.QChCare.QCh1Care (continued)

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      RESERVECHECK

      RESERVECHECK
```

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh2Care

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
ChAge := DMAge[[Index2]]
```

FRS1004B.QChCare.QCh2Care.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]

Person

^I QChCare^I

Person identifier.

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: InfantS IN QCh1Care.Child[Person].ChAtt[]

ChInf

^I QChCare^I

^I

A Nursery class is attended by 3 to 5 year olds before attendance at a reception class.

A reception class takes children who are either almost 5 years or are 5 years old. Reception class is attended before year 1 of primary school.

- (1) ^N Reception class?
- (2) ^N Nursery class?
- (3) ^N None of the above

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: PrimaryS IN QCh1Care.Child[Person].ChAtt[]

```

ChPri

^I QChCare^I

^I

A Nursery class is attended by 3 to 5 year olds before attendance at a reception class.

A reception class takes children who are either almost 5 years or are 5 years old. Reception class is attended before year 1 of primary school.

- (1) ^N Reception class?
- (2) ^N Nursery class?
- (3) ^N None of the above

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: (((((((PlayGp IN QCh1Care.Child[Person].ChAtt[]) OR (DayNurse IN
QCh1Care.Child[Person].ChAtt[])) OR (Nursery IN
QCh1Care.Child[Person].ChAtt[])) OR (Breakfst IN
QCh1Care.Child[Person].ChAtt[])) OR (Holiday IN
QCh1Care.Child[Person].ChAtt[])) OR (ChCentre IN
QCh1Care.Child[Person].ChAtt[])) OR (Boarding IN
QCh1Care.Child[Person].ChAtt[])) OR (Other IN
QCh1Care.Child[Person].ChAtt[])) OR (ChInf IN [Recept, Nursery])) OR
(ChPri IN [Recept, Nursery])
AND: ChAge >= 3

```

CTrm

^I QChCare^I

^I(Can I just check),^I^N for ^ChName was that week in term time or was it a school holiday^N^I...

- (1) ^N Term time
- (2) ^N Half term
- (3) ^N Or other school holiday?
- (4) ^N Not applicable

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK

```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh2Care (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh3Care

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
```

```
ChAge := DMAge[[Index2]]
```

FRS1004B.QChCare.QCh3Care.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]

Person

^I QChCare^I

Person identifier.

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]

ChPeo

^I QChCare^I

^I^IS C2^I

^N

And during those seven days (ending Sunday the ^DatLSun) were there any other people who looked after ^ChName?^N

^I^IC Other than resident parent/guardian, and staff contact whilst at places previously mentioned. Code babysitters as 8 'other non-relative.' Ensure that respondent has included all types of paid and unpaid childcare, including childcare given by non-resident parents.

SET [9] OF

- (1) Child's grandparents
- (2) Child's non-resident parent/an ex-spouse/an ex-partner
- (3) Child's brother or sister
- (4) Other relatives
- (5) Childminder
- (6) Nanny/Au pair (includes both live-in and day nannies)
- (7) Friends or neighbours
- (8) Other non-relatives (includes babysitters)
- (9) SPONTANEOUS ONLY - Does not require minding
- (10) None of the above

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: None IN ChPeo
ChPeo.CARDINAL = 1
```

^I
'None' exclusive for this question.

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

FRS1004B.QChCare.QCh3Care (continued)

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare (continued)

Questions about child care

COMPUTE IF: AllCh > 0

AskNoChCr := No

COMPUTE IF: AllCh > 0

AND: In loop FOR Idx1 := 1 TO HHSize

AND: (None IN QCh1Care.Child[Idx1].ChAtt) AND ((NotMind IN QCh3Care.Child[Idx1].ChPeo) OR (None IN QCh3Care.Child[Idx1].ChPeo))

AskNoChCr := Yes

ASK IF: AllCh > 0

AND: AskNoChCr = Yes

NoChCr

^I QChCare^I

^N

Can I just check, what arrangements do you make to look after your ^childtxt when you are working?^N

^I^IC Code all that apply.

SET [5] OF

- (1) Looked after by husband/wife/partner
 - (2) Old enough to look after themselves
 - (3) I only work during school hours
 - (4) I work at home
 - (5) Other type of arrangement
 - (6) Does not apply: Parent(s) are not working
-

WARN IF: AllCh > 0

AND: AskNoChCr = Yes

NOT (IN (Other, NoChCr)) AND INVOLVING (Disp, NoChCr)

^I^IC Please check that the arrangement mentioned is not an informal or formal childcare arrangement that should be recorded in the childcare section e.g child looked after by grandparents, ex-partner, childminder, friend etc.

If an informal or formal arrangement is mentioned go back to the childcare section and ask the childcare questions.

WARN IF: AllCh > 0

AND: AskNoChCr = Yes

NOT (IN (DNA, NoChCr)) AND INVOLVING (NoChCr)

^I^IC Only use this code if BOTH parents in household are not working.

CHECK IF: AllCh > 0

AND: AskNoChCr = Yes

AND: DNA IN NoChCr

NoChCr.CARDINAL = 1

^I

Does not apply is an exclusive code at this question.

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh4Care

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: DMAge[Index2] IN [0 .. 15]
```

```
ChAge := DMAge[[Index2]]
```

FRS1004B.QChCare.QCh4Care.Child[]

```
RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
```

BenUnit

^I QChCare^I

BU number of person

0..7

```
RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
```

Person

^I QChCare^I

Person identifier.

0..14

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: ChMind IN QCh3Care.Child[Person].ChPeo[]
```

provider := 'childminder'

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: ChMind IN QCh3Care.Child[Person].ChPeo[]

```

Registrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers, for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: ChMind IN QCh3Care.Child[Person].ChPeo[]

```

with1 := 'with'

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: ChMind IN QCh3Care.Child[Person].ChPeo[]

```

EmplProv

^I QChCare^I

^I^ChName^I

^N

Is the childcare ^with1 the ^provider provided by your employer?

- (1) Yes
- (2) No
- (3) Does not apply - child's parents/guardians have no employer

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: PlayGp IN QCh1Care.Child[Person].ChAtt[]
```

provider := 'playgroup or pre-school'

```
ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: PlayGp IN QCh1Care.Child[Person].ChAtt[]
```

Registrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers, for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: DayNurse IN QCh1Care.Child[Person].ChAtt[]
```

provider := 'day nursery or creche'

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: DayNurse IN QCh1Care.Child[Person].ChAtt[]

```

Registrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers, for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: DayNurse IN QCh1Care.Child[Person].ChAtt[]

```

with1 := 'in'

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
AND: DayNurse IN QCh1Care.Child[Person].ChAtt[]

```

EmplProv

^I QChCare^I

^I^ChName^I

^N

Is the childcare ^with1 the ^provider provided by your employer?

- (1) Yes
- (2) No
- (3) Does not apply - child's parents/guardians have no employer

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: Nursery IN QCh1Care.Child[Person].ChAtt[]

```

```

provider := 'nursery school'

```

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: Nursery IN QCh1Care.Child[Person].ChAtt[]

```

Regstrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers, for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DMAge[Index2] IN [0 .. 15]
  AND: QCh2Care.Child[Person].ChInf IN [Recept, Nursery]

```

```

provider := 'infant's school'

```

```
ASK IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      AND: QCh2Care.Child[Person].ChInf IN [Recept, Nursery]
```

Registrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers, for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```
COMPUTE IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      AND: QCh2Care.Child[Person].ChPri IN [Recept, Nursery]
```

provider := 'primary school'

```
ASK IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      AND: QCh2Care.Child[Person].ChPri IN [Recept, Nursery]
```

Registrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers, for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```
COMPUTE IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: DMAge[Index2] IN [0 .. 15]
      AND: Nanny IN QCh3Care.Child[Person].ChPeo[]
```

provider := 'nanny/au-pair'

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: Nanny IN QCh3Care.Child[Person].ChPeo[]

```

Registrd

^I QChCare^I

^I

Registered or approved childcare can include:

Registered childminders, nurseries and play schemes.

Out of hours clubs on school premises run by a school of local authority.

Childcare schemes run by approved providers , for example, an out of school hours scheme or a provider approved under a Ministry of Defence accreditation scheme.

A registered childminder, nursery or childcare scheme is one that is registered by:

OFSTED or the Social Care Inspection Commission in England

The National Assembly for Wales (through the Care Standards Inspectorate for Wales) in Wales

The Scottish Commission for the Regulation for Care in Scotland

A Health and Social Services Trust in Northern Ireland.

Approved childcare in England is childcare approved by a body acting under the authority of the Secretary of State.

Childcare providers who are eligible to apply for approval in England include:

Childminders who are not required to register.

Nannies or au pairs

- (1) Registered OR Approved
- (2) Not registered OR Not approved

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: Nanny IN QCh3Care.Child[Person].ChPeo[]

```

with1 := 'with'

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: DMAge[Index2] IN [0 .. 15]
AND: Nanny IN QCh3Care.Child[Person].ChPeo[]

```

EmplProv

^I QChCare^I

^I^ChName^I

^N

Is the childcare ^with1 the ^provider provided by your employer?

- (1) Yes
- (2) No
- (3) Does not apply - child's parents/guardians have no employer

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

```
CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: DMAge[Index2] IN [0 .. 15]
RESERVECHECK
```

RESERVECHECK

FRS1004B.QChCare.QCh4Care (continued)

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

CHECK IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care1

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care1.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt []

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt []

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt []

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her playgroup or pre-school?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PlayGp IN QCh1Care.Child[Index2].ChAtt []
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: PlayGp IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care1 (continued)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care2

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care2.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her Day nursery or workplace creche?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: DayNurse IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care2 (continued)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care3

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care3.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her nursery school?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Nursery IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care3 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care4

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care4.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatL.Sun did ^ChName spend in ^his_her infant's school?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: InfantS IN QCh1Care.Child[Index2].ChAtt[]
AND: QCh2Care.Child[Index2].ChInf IN [Recept, Nursery]
RESERVECHECK
```

RESERVECHECK

FRS1004B.QChCare.QCh5Care4 (continued)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care5

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care5.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatL.Sun did ^ChName spend in ^his_her primary school?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
  AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
 AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
 AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
 AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: PrimaryS IN QCh1Care.Child[Index2].ChAtt[]
 AND: QCh2Care.Child[Index2].ChPri IN [Recept, Nursery]
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care5 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care6

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care6.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her Breakfast / After school club?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Breakfst IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care6 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care7

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care7.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her Holiday scheme / club?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Holiday IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care7 (continued)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care8

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
    AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
    AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
    AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care8.Child[]

```
RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
```

BenUnit

^I QChCare^I

BU number of person

0..7

```
RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
```

Person

^I QChCare^I

Person identifier

0..14

```
ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her Children"s centres / integrated centres?

0..60

```
WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55
```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: ChCentre IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care8 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care9

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care9.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend in ^his_her boarding school?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Boarding IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh5Care9 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care10

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Other IN QCh1Care.Child[Index2].ChAtt[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care10.Child[]

RECORD IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: Other IN QCh1Care.Child[Index2].ChAtt[]

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: Other IN QCh1Care.Child[Index2].ChAtt[]

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: Other IN QCh1Care.Child[Index2].ChAtt[]

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend with
^his_her other childcare provider?

0..60

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: Other IN QCh1Care.Child[Index2].ChAtt[]
 AND: Edit = No
 AND: ChHr1 = RESPONSE
 ChHr1 < 55

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Other IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Other IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Other IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: Other IN QCh1Care.Child[Index2].ChAtt[]
      RESERVECHECK
```

RESERVECHECK

FRS1004B.QChCare.QCh5Care10 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care11

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care11.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatL.Sun did ^ChName spend with
^his_her grandparents?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

```

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's grandparents during the daytime; or at night; or both in the day^B
and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Granps IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

FRS1004B.QChCare.QCh5Care11 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care12

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care12.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend with
^his_her non-resident parent/ex-spouse/ex-partner?

0..60

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
AND: Edit = No
AND: ChHr1 = RESPONSE
ChHr1 < 55

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

```

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's non-resident parent/ex-spouse/ex-partner during the daytime; or at night; or both in the day^B and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: NonRes IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

FRS1004B.QChCare.QCh5Care12 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care13

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care13.Child[]

```
RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
```

BenUnit

^I QChCare^I

BU number of person

0..7

```
RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
```

Person

^I QChCare^I

Person identifier

0..14

```
ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend with
^his_her brother or sister?

0..60

```
WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55
```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
 AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's brother or sister during the daytime; or at night; or both in the day^B
and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: BroSis IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care13 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care14

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care14.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend with
^his_her other relatives?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

```

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's other relatives during the daytime; or at night; or both in the day^B
and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Rels IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

FRS1004B.QChCare.QCh5Care14 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care15

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care15.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatL.Sun did ^ChName spend with
^his_her childminder?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
 AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's childminder during the daytime; or at night; or both in the day^B
and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: ChMind IN QCh3Care.Child[Index2].ChPeo[]
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care15 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care16

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care16.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatL.Sun did ^ChName spend with
^his_her nanny/au pair?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

```

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's nanny/au pair during the daytime; or at night; or both in the day^B
and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Nanny IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

FRS1004B.QChCare.QCh5Care16 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care17

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh5Care17.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend with
^his_her friends or neighbours?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

```

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's friends or neighbours during the daytime; or at night; or both in the day^B and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: Friends IN QCh3Care.Child[Index2].ChPeo[]
RESERVECHECK

```

RESERVECHECK

FRS1004B.QChCare.QCh5Care17 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh5Care18

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

```

ChAge := DMAge[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh5Care18.Child[]

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

BenUnit

^I QChCare^I

BU number of person

0..7

```

RECORD IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

Person

^I QChCare^I

Person identifier

0..14

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]

```

ChHr1

^I QChCare^I

@>^I^Help_F9^I@<

^N

About how many hours in the seven days ending Sunday the ^DatLSun did ^ChName spend with
^his_her other non-relatives?

0..60

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  AND: Edit = No
  AND: ChHr1 = RESPONSE
  ChHr1 < 55

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that
your figure is correct. If so, suppress warning and continue.

```

ASK IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  AND: (ChHr1 = RESPONSE) AND (ChHr1 > 0)

```

PMChk

^I QChCare^I

^I^IC Ask or record:

^ChName^I

^N

Was the time spent with ^ChName's other non-relatives during the daytime; or at night; or both in the day^B and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night (include overnight only stays)

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: NonRels IN QCh3Care.Child[Index2].ChPeo[]
  RESERVECHECK

```

RESERVECHECK

FRS1004B.QChCare.QCh5Care18 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care1

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
        (QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
        (QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
        (QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
        (QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
        (QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
        (QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care1.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's playgroup or pre-school, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care1.Child[Index2].ChHr1 > 0) OR
(QCh5Care1.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care1 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care2

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
        (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
        (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
        (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
        (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
        (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
        (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care2.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
(QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
(QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
(QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's day nursery or workplace creche, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
 (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
 (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
 (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care2.Child[Index2].ChHr1 > 0) OR
 (QCh5Care2.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care2 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care3

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
        (QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
        (QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
        (QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
        (QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
        (QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
        (QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care3.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's nursery school, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care3.Child[Index2].ChHr1 > 0) OR
(QCh5Care3.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care3 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care4

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
        (QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
        (QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
        (QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
        (QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
        (QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
        (QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care4.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's infant school, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care4.Child[Index2].ChHr1 > 0) OR
(QCh5Care4.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care4 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care5

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
        (QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
        (QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
        (QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
        (QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
        (QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
        (QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care5.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's primary school, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care5.Child[Index2].ChHr1 > 0) OR
(QCh5Care5.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care5 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care6

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
        (QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
        (QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
        (QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
        (QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
        (QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
        (QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care6.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's Breakfast / After school club, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care6.Child[Index2].ChHr1 > 0) OR
(QCh5Care6.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care6 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care7

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
        (QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
        (QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
        (QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
        (QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
        (QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
        (QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care7.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's Holiday scheme / club, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care7.Child[Index2].ChHr1 > 0) OR
(QCh5Care7.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care7 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care8

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
        (QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
        (QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
        (QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
        (QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
        (QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
        (QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care8.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's Children's centres / integrated centres, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care8.Child[Index2].ChHr1 > 0) OR
(QCh5Care8.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care8 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care9

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
        (QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
        (QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
        (QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
        (QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
        (QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
        (QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care9.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's boarding school, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

- (1) Yes
- (2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care9.Child[Index2].ChHr1 > 0) OR
(QCh5Care9.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care9 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care10

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
    (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
    (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
    (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
    (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
    (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
    (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care10.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
(QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
(QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
(QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare in ^ChName's (other) childcare provider, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
 (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
 (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
 (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care10.Child[Index2].ChHr1 > 0) OR
 (QCh5Care10.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care10 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care11

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
        (QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
        (QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
        (QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
        (QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
        (QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
        (QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care11.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's grandparents, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care11.Child[Index2].ChHr1 > 0) OR
(QCh5Care11.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care11 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care12

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
    (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
    (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
    (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
    (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
    (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
    (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care12.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
(QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
(QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
(QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's non-resident parent/ex-spouse/ex-partner, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
 (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
 (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
 (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care12.Child[Index2].ChHr1 > 0) OR
 (QCh5Care12.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care12 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care13

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
        (QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
        (QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
        (QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
        (QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
        (QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
        (QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care13.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's brother or sister, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care13.Child[Index2].ChHr1 > 0) OR
(QCh5Care13.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK
```

RESERVECHECK

FRS1004B.QChCare.QCh6Care13 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care14

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
        (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
        (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
        (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
        (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
        (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
        (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care14.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
(QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
(QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
(QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's other relatives, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
 (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
 (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
 (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care14.Child[Index2].ChHr1 > 0) OR
 (QCh5Care14.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care14 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care15

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
        (QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
        (QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
        (QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
        (QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
        (QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
        (QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care15.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's childminder, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care15.Child[Index2].ChHr1 > 0) OR
(QCh5Care15.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care15 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care16

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
        (QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
        (QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
        (QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
        (QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
        (QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
        (QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care16.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's nanny/au-pair, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: (QCh5Care16.Child[Index2].ChHr1 > 0) OR
(QCh5Care16.Child[Index2].ChHr1 = DONTKNOW)
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care16 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care17

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
        (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
        (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
        (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
        (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
        (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
        (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh6Care17.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
(QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
(QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
(QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's friends or neighbours, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
 (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
 (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
 (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care17.Child[Index2].ChHr1 > 0) OR
 (QCh5Care17.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care17 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care18

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
        (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].BenUnit := DMBU[[Index2]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
        (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
        (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)

```

```

ChName := DMName[[Index2]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
        (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
        (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
        (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh6Care18.Child[]

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
(QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
(QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0

AND: In loop FOR Index2 := 1 TO HHSIZE

AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
(QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)

Cost

^I QChCare^I

^I^ChName^I

^N

Thinking of the childcare with ^ChName's other non-relatives, does this cost you anything?

^I^IC Record the total cost to the parents/guardian.

Include in the parent/guardian's total childcare cost any money received via Tax Credits to pay for the childcare (i.e. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

(1) Yes

(2) No

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
 (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
 (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
 (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: (QCh5Care18.Child[Index2].ChHr1 > 0) OR
 (QCh5Care18.Child[Index2].ChHr1 = DONTKNOW)
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh6Care18 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care1

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care1.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care1.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care1.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care1.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care1.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care1.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh7Care1.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's playgroup or pre-school?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: Edit = Yes
ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care1.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care1.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care1 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care2

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care2.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care2.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care2.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care2.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care2.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care2.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care2.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's day nursery or workplace creche?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care2.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care2.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care2.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care2.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care2.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care2.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care2 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care3

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care3.Child[Index2].Cost = Yes
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care3.Child[Index2].Cost = Yes
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care3.Child[Index2].Cost = Yes
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care3.Child[Index2].Cost = Yes
    AND: PRec[Index2].Sex = Male
```

```
his_her := 'his'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care3.Child[Index2].Cost = Yes
    AND: PRec[Index2].Sex = Female
```

```
his_her := 'her'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care3.Child[Index2].Cost = Yes
    AND: NOT (PRec[Index2].Sex = Female)
```

```
his_her := 'his/her'
```

FRS1004B.QChCare.QCh7Care3.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's nursery school?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

HMissVar := (HMissVar + 1)

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```
ASK IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care3.Child[Index2].Cost = Yes
      AND: ChAmt1 > 0
      AND: ChPd = LessWeek
```

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

```
ASK IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care3.Child[Index2].Cost = Yes
      AND: ChAmt1 > 0
      AND: ChPd = Note
```

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care3.Child[Index2].Cost = Yes
      AND: ChAmt1 > 0
      AND: Edit = Yes
      ChPd <> Note
```

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care3.Child[Index2].Cost = Yes
      RESERVECHECK
```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care3.Child[Index2].Cost = Yes
      RESERVECHECK
```

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care3.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care3 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care4

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care4.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care4.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care4.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care4.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care4.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care4.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care4.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care4.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care4.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care4.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's infant school?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care4.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care4.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care4.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care4.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care4.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care4.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care4.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care4.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care4.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care4 (continued)

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care5

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care5.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care5.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care5.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care5.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care5.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care5.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh7Care5.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's primary school?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care5.Child[Index2].Cost = Yes
      AND: ChAmt1 > 0
      AND: ChPd = LessWeek

```

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

```

ASK IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care5.Child[Index2].Cost = Yes
      AND: ChAmt1 > 0
      AND: ChPd = Note

```

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care5.Child[Index2].Cost = Yes
      AND: ChAmt1 > 0
      AND: Edit = Yes
      ChPd <> Note

```

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

```

WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care5.Child[Index2].Cost = Yes
      RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSIZE
      AND: QCh6Care5.Child[Index2].Cost = Yes
      RESERVECHECK

```

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care5.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care5 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care6

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: QCh6Care6.Child[Index2].Cost = Yes
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: QCh6Care6.Child[Index2].Cost = Yes
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: QCh6Care6.Child[Index2].Cost = Yes
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: QCh6Care6.Child[Index2].Cost = Yes
    AND: PRec[Index2].Sex = Male
```

```
his_her := 'his'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: QCh6Care6.Child[Index2].Cost = Yes
    AND: PRec[Index2].Sex = Female
```

```
his_her := 'her'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSIZE
    AND: QCh6Care6.Child[Index2].Cost = Yes
    AND: NOT (PRec[Index2].Sex = Female)
```

```
his_her := 'his/her'
```

FRS1004B.QChCare.QCh7Care6.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's Breakfast / After school club?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care6.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care6.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care6.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care6.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care6.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care6.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care6 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care7

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care7.Child[Index2].Cost = Yes
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care7.Child[Index2].Cost = Yes
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care7.Child[Index2].Cost = Yes
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care7.Child[Index2].Cost = Yes
    AND: PRec[Index2].Sex = Male
```

```
his_her := 'his'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care7.Child[Index2].Cost = Yes
    AND: PRec[Index2].Sex = Female
```

```
his_her := 'her'
```

```
COMPUTE IF: AllCh > 0
    AND: In loop FOR Index2 := 1 TO HHSize
    AND: QCh6Care7.Child[Index2].Cost = Yes
    AND: NOT (PRec[Index2].Sex = Female)
```

```
his_her := 'his/her'
```

FRS1004B.QChCare.QCh7Care7.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's Holiday scheme / club?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care7.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care7.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care7.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care7.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care7.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care7.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care7 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care8

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care8.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care8.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care8.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care8.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care8.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care8.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care8.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's Children's centres / integrated centres?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care8.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = LessWeek

```

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care8.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = Note

```

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care8.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: Edit = Yes
ChPd <> Note

```

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care8.Child[Index2].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care8.Child[Index2].Cost = Yes
RESERVECHECK

```

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care8.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care8 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care9

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care9.Child[Index2].Cost = Yes
```

```
Child[Index2].BenUnit := DMBU[[Index2]]
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care9.Child[Index2].Cost = Yes
```

```
Child[Index2].Person := Index2
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care9.Child[Index2].Cost = Yes
```

```
ChName := DMName[[Index2]]
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care9.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male
```

```
his_her := 'his'
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care9.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female
```

```
his_her := 'her'
```

```
COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care9.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)
```

```
his_her := 'his/her'
```


FRS1004B.QChCare.QCh7Care9.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care9.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care9.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care9.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare in ^ChName's boarding school?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care9.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care9.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care9.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care9.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care9.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care9.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care9.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care9.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care9.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care9 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care10

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care10.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care10.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care10.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care10.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care10.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care10.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care10.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care10.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care10.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care10.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's other provider?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care10.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care10.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care10.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care10.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care10.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care10.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care10.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care10.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: QCh6Care10.Child[Index2].Cost = Yes
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh7Care10 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care11

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care11.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care11.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care11.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care11.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care11.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care11.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care11.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care11.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care11.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care11.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's grandparents?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care11.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care11.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care11.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care11.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care11.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care11.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care11.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care11.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: QCh6Care11.Child[Index2].Cost = Yes
      RESERVECHECK

      RESERVECHECK
```


FRS1004B.QChCare.QCh7Care11 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care12

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care12.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care12.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care12.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care12.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care12.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care12.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care12.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's non-resident parent/ex-spouse/ex-partner?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care12.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = LessWeek

```

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care12.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = Note

```

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care12.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: Edit = Yes
ChPd <> Note

```

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care12.Child[Index2].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care12.Child[Index2].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```
WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care12.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK
```

FRS1004B.QChCare.QCh7Care12 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care13

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care13.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care13.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care13.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care13.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care13.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care13.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh7Care13.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care13.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care13.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care13.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's brother or sister?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care13.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care13.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care13.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care13.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care13.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care13.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care13.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care13.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: QCh6Care13.Child[Index2].Cost = Yes
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh7Care13 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care14

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE

```

```

PerNo := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care14.Child[PerNo].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care14.Child[PerNo].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care14.Child[PerNo].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care14.Child[PerNo].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care14.Child[PerNo].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care14.Child[PerNo].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care14.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care14.Child[PerNo].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care14.Child[PerNo].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care14.Child[PerNo].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's other relatives?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care14.Child[PerNo].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care14.Child[PerNo].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care14.Child[PerNo].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care14.Child[PerNo].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = LessWeek

```

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care14.Child[PerNo].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = Note

```

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care14.Child[PerNo].Cost = Yes
AND: ChAmt1 > 0
AND: Edit = Yes
ChPd <> Note

```

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care14.Child[PerNo].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care14.Child[PerNo].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: QCh6Care14.Child[PerNo].Cost = Yes
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh7Care14 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care15

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care15.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care15.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care15.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care15.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care15.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care15.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care15.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's childminder?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: Edit = Yes
ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care15.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: QCh6Care15.Child[Index2].Cost = Yes
      RESERVECHECK

      RESERVECHECK
```


FRS1004B.QChCare.QCh7Care15 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care16

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care16.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care16.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care16.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care16.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care16.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care16.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care16.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's nanny/au pair?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care16.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care16.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care16.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care16.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care16.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care16.Child[Index2].Cost = Yes
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care16 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care17

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care17.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care17.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care17.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care17.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care17.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSize
  AND: QCh6Care17.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```


FRS1004B.QChCare.QCh7Care17.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care17.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care17.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care17.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's friends or neighbours?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice.

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care17.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care17.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care17.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care17.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = LessWeek

```

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care17.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: ChPd = Note

```

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care17.Child[Index2].Cost = Yes
AND: ChAmt1 > 0
AND: Edit = Yes
ChPd <> Note

```

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care17.Child[Index2].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
AND: QCh6Care17.Child[Index2].Cost = Yes
RESERVECHECK

```

RESERVECHECK

```
WARN IF: AllCh > 0
      AND: In loop FOR Index2 := 1 TO HHSize
      AND: QCh6Care17.Child[Index2].Cost = Yes
      RESERVECHECK

      RESERVECHECK
```

FRS1004B.QChCare.QCh7Care17 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care18

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care18.Child[Index2].Cost = Yes

```

```

Child[Index2].BenUnit := DMBU[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care18.Child[Index2].Cost = Yes

```

```

Child[Index2].Person := Index2

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care18.Child[Index2].Cost = Yes

```

```

ChName := DMName[[Index2]]

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care18.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Male

```

```

his_her := 'his'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care18.Child[Index2].Cost = Yes
  AND: PRec[Index2].Sex = Female

```

```

his_her := 'her'

```

```

COMPUTE IF: AllCh > 0
  AND: In loop FOR Index2 := 1 TO HHSIZE
  AND: QCh6Care18.Child[Index2].Cost = Yes
  AND: NOT (PRec[Index2].Sex = Female)

```

```

his_her := 'his/her'

```

FRS1004B.QChCare.QCh7Care18.Child[]

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care18.Child[Index2].Cost = Yes

BenUnit

^I QChCare^I

BU number of person

0..7

RECORD IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care18.Child[Index2].Cost = Yes

Person

^I QChCare^I

Person identifier

0..14

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care18.Child[Index2].Cost = Yes

ChAmt1

^I QChCare^I

^I^ChName^I

^N

How much was your most recent payment for the childcare with ^ChName's other non-relatives?^N

^I^IC If unable to attribute costs per child, then enter an estimate by dividing total childcare costs by number of children.

Record the total cost to the parents/guardian.

Include in the total childcare cost to the parent/guardian any money received via Tax Credits to pay for the childcare (e.g. Childcare element of Working Tax Credit).

Exclude payments made by others e.g. grandparents or where payment is made with childcare vouchers unless received as a salary sacrifice..

0.00..99997.00

```

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care18.Child[Index2].Cost = Yes
AND: Edit = No
AND: ChAmt1 = RESPONSE
ChAmt1 < 130

```

^I

Warning: The answer is much higher than the figures usually given at this question. Please check that your figure is correct. If so, suppress warning and continue.

```

COMPUTE IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care18.Child[Index2].Cost = Yes
AND: ChAmt1 = NONRESPONSE

```

```

HMissVar := (HMissVar + 1)

```

```

ASK IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSize
AND: QCh6Care18.Child[Index2].Cost = Yes
AND: ChAmt1 > 0

```

ChPd

^I QChCare^I

^I^ChName^I

^N

And what period did that payment cover?^N

^I^IC If hourly rate code as 90 'Less than one week'.

^I^IC If payment period is per school term code as 13 'Three months/13 weeks'.

- (1) One week
- (2) Two weeks
- (3) Three weeks
- (4) Four weeks
- (5) Calendar month
- (7) Two Calendar months
- (8) Eight times a year
- (9) Nine times a year
- (10) Ten times a year
- (13) Three months/13 weeks
- (26) Six months/26 weeks
- (52) One Year/12 months/52 weeks
- (90) Less than one week
- (95) One off/lump sum
- (97) None of these ^I(Explain in a note)

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care18.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = LessWeek

Hourly

^I QChCare^I

^I^IC ASK OR RECORD

^I ^N

Is the payment an hourly rate?

(1) Yes

(2) No

ASK IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care18.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: ChPd = Note

ChPx

^I QChCare^I

^I^IC ^Pd97Txt

OPEN

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care18.Child[Index2].Cost = Yes
 AND: ChAmt1 > 0
 AND: Edit = Yes
 ChPd <> Note

^IEditor: Code 97 must be re-coded into existing list.

If you temporarily suppress this check you must come back to resolve it.

If there is an interviewer note to say that the amount covers one term, code as 13 'Three months/13 weeks'.

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care18.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSIZE
 AND: QCh6Care18.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
 AND: In loop FOR Index2 := 1 TO HHSize
 AND: QCh6Care18.Child[Index2].Cost = Yes
 RESERVECHECK

RESERVECHECK

FRS1004B.QChCare.QCh7Care18 (continued)

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
AND: In loop FOR Index2 := 1 TO HHSIZE
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B.QChCare (continued)**Questions about child care**

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

WARN IF: AllCh > 0
RESERVECHECK

RESERVECHECK

COMPUTE IF: In loop FOR Loop1 := 1 TO 14
AND: (((QChCare.QCh4Care.Child[Loop1].Registrd[1] = Registered) OR
(QChCare.QCh4Care.Child[Loop1].Registrd[2] = Registered)) OR
(QChCare.QCh4Care.Child[Loop1].Registrd[3] = Registered)) OR
(QChCare.QCh4Care.Child[Loop1].Registrd[4] = Registered)) OR
(QChCare.QCh4Care.Child[Loop1].Registrd[5] = Registered)

ChRegis := (ChRegis + 1)

COMPUTE IF: In loop FOR Loop1 := 1 TO 14
AND: DMAge[Loop1] IN [0 .. 15]
AND: DMNumParn[Loop1] = 1

LoneParent := Yes

COMPUTE IF: In loop FOR Loop1 := 1 TO 14
AND: DMAge[Loop1] IN [0 .. 15]
AND: DMNumParn[Loop1] = 1

NCDVLP := Yes

COMPUTE IF: In loop FOR Loop1 := 1 TO 14
AND: DMAge[Loop1] IN [0 .. 15]
AND: DMNumParn[Loop1] > 1

LoneParent := No

FRS1004B.QCare

Questions about carers/cared for

ASK ALWAYS :

NeedHelp

^I QCare^I

^I^B

Keeping an eye out, 'being there':^B

Being available if needed

Making your whereabouts known so you can be contacted if needed

^B

Social support and assistance:^B

Sitting with

Chatting with/ listening to/ reading to

Making/receiving telephone calls to talk to them

Encouraging them to do things for themselves

^B

Accompanying on trips out to go:^B

Shopping

To hospital/ GP/ optician/ dentist/ chiroprapist

To the park/ place of worship/ restaurant

^B

Paperwork/ official/ financial:^B

Helping with paperwork

Dealing with 'officials' (including by phone)

^B

Home and garden:^B

Making meals

Going shopping for someone

Washing/ ironing/ changing sheets

Cleaning/ housework

Gardening

Odd jobs/ maintenance

^B

Medical:^B

Collecting prescriptions/ giving medication

Changing dressings

^B

Moving about the home: Giving help with^B

Getting up and down stairs

Moving from room to room

Getting in and out of bed

^B

Personal care: help with^B

Getting dressed

Feeding

Washing/ bathing/ using the toilet

(1) Yes

(2) No

ASK ALWAYS :

GiveHelp

^I QCare^I

^I^B

Keeping an eye out, 'being there':^B

Being available if needed

Making your whereabouts known so you can be contacted if needed

^B

Social support and assistance:^B

Sitting with

Chatting with/ listening to/reading to

Making/receiving telephone calls to talk to them

Encouraging them to do things for themselves

^B

Accompanying on trips out to go:^B

Shopping

To hospital/ GP/ optician/ dentist/ chiropodist

To the park/ place of worship/ restaurant

^B

Paperwork/ official/ financial:^B

Helping with paperwork

Dealing with 'officials' (including by phone)

^B

Home and garden:^B

Making meals

Going shopping for someone

Washing/ ironing/ changing sheets

Cleaning/ housework

Gardening

Odd jobs/maintenance

^B

Medical:^B

Collecting prescriptions/giving medication

Changing dressings

^B

Moving about the home: Giving help with^B

Getting up and down stairs

Moving from room to room

Getting in and out of bed

^B

Personal care: help with^B

Getting dressed

Feeding

Washing/ bathing/ using the toilet

(1) Yes

(2) No

FRS1004B.QCare.QRecHelp

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: NeedHelp = Yes
  AND: In loop FOR Idy := 1 TO HHSIZE
```

```
LName[Idy] := DMName[[Idy]
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[15] := 'Parent outside household'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[16] := 'Other Parent outside household'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[17] := 'Child outside household'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[18] := 'Spouse outside household'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[19] := 'Other relative'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[20] := 'Friend/neighbour'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[21] := 'Client of voluntary organisation'
```

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: GiveHelp = Yes
```

```
LName[22] := 'Other non-household'
```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)

QNeedPer

^I QCare^I

^I

If they provide help or give help for^B more than one^B individual in any one category of:

- 15: Parent outside household
- 16: Other parent outside household
- 18: Child outside household
- 19: Relative
- 20: Friend/neighbour
- 21: Client of voluntary organisation
- 22: Other non-household

Please make a Note here <Ctrl+M> to tell us how many^B more^B people are involved.

SET [5] OF

- (1) ^LName[1]
- (2) ^LName[2]
- (3) ^LName[3]
- (4) ^LName[4]
- (5) ^LName[5]
- (6) ^LName[6]
- (7) ^LName[7]
- (8) ^LName[8]
- (9) ^LName[9]
- (10) ^LName[10]
- (11) ^LName[11]
- (12) ^LName[12]
- (13) ^LName[13]
- (14) ^LName[14]
- (15) ^LName[15]
- (16) ^LName[16]
- (17) ^LName[17]
- (18) ^LName[18]
- (19) ^LName[19]
- (20) ^LName[20]
- (21) ^LName[21]
- (22) ^LName[22]

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR ixy := 1 TO 22
AND: ixy IN QNeedPer
 (LName[ixy] <> '') AND INVOLVING(QNeedPer)

^I

Record a valid code for person cared for.

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: Per22 IN QNeedPer

```

NeedPerO

^I QCare^I

^N

Who is the other person outside the household receiving help or being looked after?

STRING[40]

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: QNeedPer[Idx] IN [Per1 .. Per22]

```

NeedNum := ORD(QNeedPer[Idx])

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: NeedNum IN [1 .. 14]

```

NeedName := DMName[NeedNum]

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: QNeedPer[Idx] = Per15

```

NeedName := 'the PARENT'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: QNeedPer[Idx] = Per16

```

NeedName := 'the OTHER PARENT'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: QNeedPer[Idx] = Per17

```

NeedName := 'the CHILD'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: QNeedPer[Idx] = Per18

```

NeedName := 'the SPOUSE'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: QNeedPer[Idx] = Per19

```

NeedName := 'the RELATIVE'

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: QNeedPer[Idx] = Per20
```

NeedName := 'the FRIEND/NEIGHBOUR'

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: QNeedPer[Idx] = Per21
```

NeedName := 'the CLIENT of a voluntary organisation'

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: QNeedPer[Idx] = Per22
```

NeedName := UPCASE(NeedPer0)

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
```

Recip[Idx].NeedPer := NeedNum

FRS1004B.QCare.QRecHelp.Recip[]

```

RECORD IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL

```

NeedPer

^I QCare^I

Who is receiving help/being looked after.

0..22

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL

```

LNeedPer := NeedPer

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL

```

LNeedName := PNeedName

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: PRec[LNeedPer].Sex = Male

```

need_hisher := 'his'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: PRec[LNeedPer].Sex = Male

```

need_heshe := 'he'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: PRec[LNeedPer].Sex = Male

```

need_himher := 'him'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: PRec[LNeedPer].Sex = Female

```

need_hisher := 'her'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: PRec[LNeedPer].Sex = Female

```

```

need_heshe := 'she'

```

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: PRec[LNeedPer].Sex = Female

```

```

need_himher := 'her'

```

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (PRec[LNeedPer].Sex = Female)

```

```

need_hisher := 'his/her'

```

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (PRec[LNeedPer].Sex = Female)

```

```

need_heshe := 'he/she'

```

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (PRec[LNeedPer].Sex = Female)

```

```

need_himher := 'him/her'

```

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL

```

Freq

^I QCare^I

^N

How frequently does ^LNeedName receive such help?

- (1) Continuously
- (2) Several times a day
- (3) Once or twice a day
- (4) Several times a week
- (5) Once a week
- (6) Less frequently

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: Freq IN [Continuously, SevDay]

```

DayNight

^I QCare^I

^N

And does ^LNeedName receive help during the daytime; or at night; or both in the day^B and^B at night?

- (1) Daytime only
- (2) At night only
- (3) Both day and night

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: LNeedPer > 14

```

Rel := ''

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: LNeedPer > 14

```

Fri := ''

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: LNeedPer > 14

```

LAH := ''

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: LNeedPer > 14

```

Dom := ''

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: LNeedPer > 14

```

Nur := ''

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: LNeedPer > 14

```

Hel := ''

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)

```

Rel := 'Relatives'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)

```

Fri := 'Friends/Neighbours'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)
  AND: NatCen = NI

```

LAH := 'Social Services home help or home care worker'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)
  AND: NOT (NatCen = NI)

```

LAH := 'Local Authority home help or home care worker'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)

```

Dom := 'Private domestic help'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)

```

Nur := 'District nurse, health visitor or other kind of nurse'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: NOT (LNeedPer > 14)

```

Hel := 'Other outside helpers'

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE

WhoLook

^I QCare^I

^N

Who looks after, or provides help for ^LNeedName? ^Anyone_else?^N

^I^IC Code all that apply.

SET [5] OF

- (1) ^DMName[1]
- (2) ^DMName[2]
- (3) ^DMName[3]
- (4) ^DMName[4]
- (5) ^DMName[5]
- (6) ^DMName[6]
- (7) ^DMName[7]
- (8) ^DMName[8]
- (9) ^DMName[9]
- (10) ^DMName[10]
- (11) ^DMName[11]
- (12) ^DMName[12]
- (13) ^DMName[13]
- (14) ^DMName[14]
- (15) ^Rel
- (16) ^Fri
- (17) ^LAH
- (18) ^Dom
- (19) ^Nur
- (20) ^Hel

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR ixx := 1 TO 14
AND: ixx IN WhoLook
(DMName[[ixx] <> '') AND INVOLVING(WhoLook)

^I

Record valid code for carer.

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: Relative IN WhoLook
(Rel <> '') AND INVOLVING(WhoLook)

^I

Record valid code for carer.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: Friends IN WhoLook
(Fri <> '') AND INVOLVING(WhoLook)

```

^I

Record valid code for carer.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: LAHelp IN WhoLook
(LAH <> '') AND INVOLVING(WhoLook)

```

^I

Record valid code for carer.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: Domestic IN WhoLook
(Dom <> '') AND INVOLVING(WhoLook)

```

^I

Record valid code for carer.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: Nurse IN WhoLook
(Nur <> '') AND INVOLVING(WhoLook)

```

^I

Record valid code for carer.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: Helpers IN WhoLook
(Hel <> '') AND INVOLVING(WhoLook)

```

^I

Record valid code for carer.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: LNeedPer > 14
NOT((((IN(Relative,WhoLook)) OR (IN(Friends,WhoLook))) OR
(IN(LAHelp,WhoLook))) OR (IN(Domestic,WhoLook))) OR (IN(Nurse,WhoLook)))
OR (IN(Helpers,WhoLook))

```

^M^IC That code is invalid

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: LNeedPer <= 14
NOT (IN (LNeedPer, WhoLook))

```

^I^IC You've included ^LNeedName as looking after ^need_himherself. Please remove ^need_himher from the answer at WhoLook.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 14
AND: Count IN WhoLook
DMAge[[Count] >= 0

```

^I^IC Code ^Count is not valid for this question.

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 14
RESERVECHECK

```

RESERVECHECK

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 14
RESERVECHECK

```

RESERVECHECK

```

CHECK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 14
RESERVECHECK

```

RESERVECHECK

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL
AND: WhoLook[Count] IN [Per1 .. Per14]

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] IN [Per1 .. Per14]

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] IN [Per1 .. Per14]

```

helper := DMName [[ORD (WhoLook [Count])]]

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] IN [Per1 .. Per14]

```

HCount := ORD (WhoLook [Count])

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Relative

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Relative

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Relative

```

helper := 'the relative'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Relative

```

HCount := 15

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Friends

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Friends

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Friends

```

helper := 'the friend'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Friends

```

HCount := 16

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp
  AND: NatCen = NI

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp
  AND: NatCen = NI

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp
  AND: NatCen = NI

```

helper := ('the Social Services home help or ' + 'home care worker')

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp
  AND: NOT (NatCen = NI)

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp
  AND: NOT (NatCen = NI)

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp
  AND: NOT (NatCen = NI)

```

helper := 'the LA home help or home care worker'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = LAHelp

```

HCount := 17

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Domestic

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Domestic

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Domestic

```

helper := 'the private domestic help'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Domestic

```

HCount := 18

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Nurse

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Nurse

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Nurse

```

helper := 'the nurse'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Nurse

```

HCount := 19

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Helpers

```

does := 'does'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Helpers

```

has := 'has'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Helpers

```

helper := 'the outside helper'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: WhoLook[Count] = Helpers

```

HCount := 20

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: HCount <= 14
  AND: PRec[HCount].Sex = Male

```

help_hisher := 'his'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: HCount <= 14
  AND: PRec[HCount].Sex = Male

```

help_heshe := 'he'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: HCount <= 14
  AND: PRec[HCount].Sex = Female

```

help_hisher := 'her'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: HCount <= 14
  AND: PRec[HCount].Sex = Female

```

help_heshe := 'she'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: HCount <= 14
  AND: NOT (PRec[HCount].Sex = Female)

```

help_hisher := 'his/her'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: HCount <= 14
  AND: NOT (PRec[HCount].Sex = Female)

```

help_heshe := 'he/she'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: NOT (HCount <= 14)

```

help_hisher := 'his/her'

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: NOT (HCount <= 14)

```

help_heshe := 'he/she'

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL

```

Hour

^I QCare^I

^I^IS D2^I

^N

About how many hours a week, on average, does ^helper spend actually providing help for or looking after ^LNeedName?

- (1) 0-4 hours per week
- (2) 5-9 hours per week
- (3) 10-19 hours per week
- (4) 20-34 hours per week
- (5) 35-49 hours per week
- (6) 50-99 hours per week
- (7) 100 or more hours per week
- (8) Varies - under 20 hours per week
- (9) Varies - 20-34 hours per week
- (10) Varies - 35 hours a week or more

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: Edit <> Yes
  AND: HCount <= 14
  AND: DMAge[HCount] IN [1 .. 15]
  Hour[Count] <> over100

```

^I

Are you sure that the child(ren) are looking after ^LNeedName for 100 hours a week or more?
If so, suppress warning.

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL

```

HowLng

^I QCare^I

^I^IS D3^I

^N

Thinking of when ^helper first began providing help for or looking after ^LNeedname, about how long ^has ^helper been providing help for or looking after ^LneedName?

^I^IC Prompt as necessary.

Record how long care has been given regardless of the number of hours the carer gives now or has given in the past.

- (1) Less than 6 months
- (2) 6 months, less than 1 year
- (3) 1 year, less than 3 years
- (4) 3 years, less than 5 years
- (5) 5 years, less than 10 years
- (6) 10 years, less than 15 years
- (7) 15 years, less than 20 years
- (8) 20 years or more

```

RECORD IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL
AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
(Hour[Count] IN [upto49, upto99, over100, var35])

```

ChCAPx

^I QCare^I

^I^IC ^Pd97Txt

OPEN

```

RECORD IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL
AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
(Hour[Count] IN [upto49, upto99, over100, var35])

```

ChCAEx

^I QCare^I

^I
^IC ^SuppTxt

OPEN

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL
AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
(Hour[Count] IN [upto49, upto99, over100, var35])
AND: ChCA[Count] = Yes

```

ChCAAmt

^I QCare^I

^N
How much did ^help_heshe get last time?

0.01..99997.00

```

RECORD IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL
AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
(Hour[Count] IN [upto49, upto99, over100, var35])
AND: ChCA[Count] = Yes

```

ChCAPx

^I QCare^I

^I^IC ^Pd97Txt

OPEN

```
ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]
  AND: ChCAPd[Count] = Note
```

ChCAPx

^I QCare^I

^I^IC ^Pd97Txt

OPEN

FRS1004B.QCare.QRecHelp.Recip[.Weekly()

Procedure Call

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
  (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]
```

PdConW[1] := 1

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
  (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]
```

PdConW[2] := 2

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
  (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]
```

PdConW[3] := 3

```
COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
  (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]
```

PdConW[4] := 4

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[5] := 4.333

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[7] := 8.67

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[8] := 6.5

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[9] := 5.78

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[10] := 5.2

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[13] := 13

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[26] := 26

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]

```

PdConW[52] := 52

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]
  AND: (PAmount > 0) AND (PPeriod IN [OneWeek .. Year])

```

PWeekly := (PAmount / PdConW[ORD(PPeriod)])

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAAmt[Count] IN [0.01 .. 99997]
  AND: NOT ((PAmount > 0) AND (PPeriod IN [OneWeek .. Year]))

```

PWeekly := 0

FRS1004B.QCare.QRecHelp.Recip[] (continued)

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]

```

LPd := ChCAPd[Count]

```

COMPUTE IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]
  AND: LPd IN [OneWeek .. Year]

```

ChCAMax := 53.1

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
  AND: In loop FOR Idx := 1 TO 5
  AND: Idx <= QNeedPer.CARDINAL
  AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
  AND: In loop FOR Count := 1 TO 5
  AND: Count <= WhoLook.CARDINAL
  AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
    (Hour[Count] IN [upto49, upto99, over100, var35])
  AND: ChCA[Count] = Yes
  AND: ChCAmt[Count] IN [0.01 .. 99997]
  AND: LPd IN [OneWeek .. Year]
  (LWeekly < ChCAMax) AND INVOLVING(ChCAPd[Count], ChCAmt[Count])

```

(LWeekly < ChCAMax) AND INVOLVING(ChCAPd[Count], ChCAmt[Count])

```

ASK IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
AND: Count <= WhoLook.CARDINAL
AND: ((DMAge[HCount] IN [16 .. 19]) AND (PRec[HCount].FtEd <> Yes)) AND
(Hour[Count] IN [upto49, upto99, over100, var35])
AND: ChCA[Count] = Yes
AND: ChCAAmt[Count] IN [0.01 .. 99997]
AND: LPd IN [OneWeek .. Year]
AND: (ChCARs[Count] = Suppressed) OR ChCAEx[Count] <> EMPTY

```

ChCAEx

^I QCare^I

^I
^IC ^ISuppTxt

OPEN

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
AND: (Freq IN [Continuously .. OWeek]) OR Freq = NONRESPONSE
AND: In loop FOR Count := 1 TO 5
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
RESERVECHECK

```

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
AND: Idx <= QNeedPer.CARDINAL
RESERVECHECK

RESERVECHECK

FRS1004B.QCare.QRecHelp (continued)

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Idx := 1 TO 5
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
RESERVECHECK

RESERVECHECK

FRS1004B.QCare (continued)

Questions about carers/cared for

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: QRecHelp.QNeedPer <> EMPTY AND (NeedHelp = Yes)
((( ((( ((( ((( (IN(Per1,QRecHelp.QNeedPer)) OR
(IN(Per2,QRecHelp.QNeedPer))) OR (IN(Per3,QRecHelp.QNeedPer))) OR
(IN(Per4,QRecHelp.QNeedPer))) OR (IN(Per5,QRecHelp.QNeedPer))) OR
(IN(Per6,QRecHelp.QNeedPer))) OR (IN(Per7,QRecHelp.QNeedPer))) OR
(IN(Per8,QRec

```

^I

Please include the household member who receives regular help, or change 'NeedHelp' to 'No'.

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 1 TO HHSIZE
AND: NeedHelp <> EMPTY AND (Index1 IN QRecHelp.QNeedPer)
NeedHelp = Yes

```

^I

You have coded a household member as receiving regular help, so please change 'NeedHelp' to 'Yes', or remove the household member from 'QNeedPer'.

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: QRecHelp.QNeedPer <> EMPTY AND (GiveHelp = Yes)
((( ((( ((( (IN(Per15,QRecHelp.QNeedPer)) OR (IN(Per16,QRecHelp.QNeedPer))) OR
(IN(Per17,QRecHelp.QNeedPer))) OR (IN(Per18,QRecHelp.QNeedPer))) OR
(IN(Per19,QRecHelp.QNeedPer))) OR (IN(Per20,QRecHelp.QNeedPer))) OR
(IN(Per21,QRecHelp.QNeedPer))) OR (IN(Per22,QR

```

^I

Please include the non-household member receiving help from someone in the household, or change 'GiveHelp' to 'No'.

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
AND: GiveHelp <> EMPTY AND (Index1 IN QRecHelp.QNeedPer)
GiveHelp = Yes

```

^I

You have coded a non-household member as receiving help, so please change 'GiveHelp' to 'Yes', or remove the non-household member from 'QNeedPer'.

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

```

RESERVECHECK

```

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

```

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

RESERVECHECK

WARN IF: (NeedHelp = Yes) OR (GiveHelp = Yes)
AND: In loop FOR Index1 := 15 TO 22
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

WARN ALWAYS:
RESERVECHECK

RESERVECHECK

FRS1004B (continued)

FAMILY RESOURCES SURVEY 2010-2011

```

CHECK IF: In loop FOR Loop1 := 1 TO 14
          AND: Loop1 IN QCare.QRecHelp.QNeedPer
          DMAge[Loop1] >= 0

```

^M^IC Code ^Loop1 is not valid for this question.

```

COMPUTE IF: In loop FOR Loop1 := 1 TO 14
            AND: In loop FOR Loop2 := 1 TO 5
            AND: Loop1 IN QCare.QRecHelp.Recip[Loop2].WhoLook

```

DMCarer[Loop1] := Yes

```

CHECK IF: In loop FOR Loop1 := 1 TO 14
          AND: In loop FOR Loop2 := 1 TO 5
          RESERVECHECK

```

RESERVECHECK

```

CHECK IF: In loop FOR Loop1 := 1 TO 14
          AND: In loop FOR Loop2 := 1 TO 5
          RESERVECHECK

```

RESERVECHECK

```

CHECK IF: In loop FOR Loop1 := 1 TO 14
          RESERVECHECK

```

RESERVECHECK

```

CHECK IF: In loop FOR Loop1 := 1 TO 14
          RESERVECHECK

```

RESERVECHECK

```

CHECK ALWAYS:
          RESERVECHECK

```

RESERVECHECK

```

CHECK ALWAYS:
          RESERVECHECK

```

RESERVECHECK

```

CHECK ALWAYS:
          RESERVECHECK

```

RESERVECHECK

```

CHECK ALWAYS:
          RESERVECHECK

```

RESERVECHECK

ASK ALWAYS :

EndDisp

^I^IC@|-@|End of 'Household' Schedule.
 Now administer 'Benefit Unit' Schedule(s).
 B.U. members
 ^NameInBU[1]^NameInBU[2]^NameInBU[3]^NameInBU[4]^NameInBU[5]
 ^NameInBU[6]^NameInBU[7]

Total number of Benefit Units = ^NewBU

Press <Ctrl + Enter> to select Benefit Unit or to fill in Admin details.
 Alternatively, press <1> and <Enter> to start the first Benefit Unit.

1..1

RECORD ALWAYS :

HHTime

^I
 Time taken from interview start to end of household grid.

Only visible for testing purposes, just press <Enter>.

TIME

RECORD ALWAYS :

HHMins

^I
 Total minutes in household grid.

Only visible for testing purposes, just press <Enter>.

0.00..1440.00

COMPUTE IF: HHTime = EMPTY AND EndDisp <> EMPTY

HHTime := SYSTIME

COMPUTE IF: HHTime = EMPTY AND EndDisp <> EMPTY

HHMins := ((HHTime.ABSTIME - QSignIn.IntSTime.ABSTIME) / 60000)

ASK IF: Test = Yes

HHTime

^I
 Time taken from interview start to end of household grid.

Only visible for testing purposes, just press <Enter>.

TIME

ASK IF: *Test = Yes*

HHMins

^I

Total minutes in household grid.

Only visible for testing purposes, just press <Enter>.

0.00..1440.00

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

CHECK ALWAYS:

RESERVECHECK

RESERVECHECK

COMPUTE ALWAYS:

NBusRooms := 0

COMPUTE IF: *In loop FOR Loop1 := 1 TO NewBU*

AdInBU[1] := 1

COMPUTE IF: *In loop FOR Loop1 := 1 TO NewBU*

AdInBU[2] := 1

COMPUTE IF: *In loop FOR Loop1 := 1 TO NewBU*

Child1 := 0

COMPUTE IF: *In loop FOR Loop1 := 1 TO NewBU*

Child2 := 0

COMPUTE IF: *In loop FOR Loop1 := 1 TO NewBU*

Child3 := 0

COMPUTE IF: *In loop FOR Loop1 := 1 TO NewBU*

Child4 := 0

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU

Child5 := 0

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU

Child6 := 0

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU

Child7 := 0

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU

Child8 := 0

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
 AND: In loop FOR Loop2 := 1 TO HHSIZE
 AND: Loop1 = ABen[Loop2]
 AND: PRec[Loop2].Depend = Adult
 AND: AdInBU[1] = EMPTY

AdInBU[1] := Loop2

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
 AND: In loop FOR Loop2 := 1 TO HHSIZE
 AND: Loop1 = ABen[Loop2]
 AND: PRec[Loop2].Depend = Adult
 AND: AdInBU[2] = EMPTY

AdInBU[2] := Loop2

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
 AND: In loop FOR Loop2 := 1 TO HHSIZE
 AND: Loop1 = ABen[Loop2]
 AND: PRec[Loop2].Depend IN [DepAd .. Child]
 AND: Child1 = 0

Child1 := Loop2

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
 AND: In loop FOR Loop2 := 1 TO HHSIZE
 AND: Loop1 = ABen[Loop2]
 AND: PRec[Loop2].Depend IN [DepAd .. Child]
 AND: Child2 = 0

Child2 := Loop2

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
 AND: In loop FOR Loop2 := 1 TO HHSIZE
 AND: Loop1 = ABen[Loop2]
 AND: PRec[Loop2].Depend IN [DepAd .. Child]
 AND: Child3 = 0

Child3 := Loop2

COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
 AND: In loop FOR Loop2 := 1 TO HHSIZE
 AND: Loop1 = ABen[Loop2]
 AND: PRec[Loop2].Depend IN [DepAd .. Child]
 AND: Child4 = 0

Child4 := Loop2

```
COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
  AND: In loop FOR Loop2 := 1 TO HHSIZE
  AND: Loop1 = ABen[Loop2]
  AND: PRec[Loop2].Depend IN [DepAd .. Child]
  AND: Child5 = 0
```

Child5 := Loop2

```
COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
  AND: In loop FOR Loop2 := 1 TO HHSIZE
  AND: Loop1 = ABen[Loop2]
  AND: PRec[Loop2].Depend IN [DepAd .. Child]
  AND: Child6 = 0
```

Child6 := Loop2

```
COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
  AND: In loop FOR Loop2 := 1 TO HHSIZE
  AND: Loop1 = ABen[Loop2]
  AND: PRec[Loop2].Depend IN [DepAd .. Child]
  AND: Child7 = 0
```

Child7 := Loop2

```
COMPUTE IF: In loop FOR Loop1 := 1 TO NewBU
  AND: In loop FOR Loop2 := 1 TO HHSIZE
  AND: Loop1 = ABen[Loop2]
  AND: PRec[Loop2].Depend IN [DepAd .. Child]
  AND: Child8 = 0
```

Child8 := Loop2