
 --

ABLE

Purpose : To show whether a person is a man aged 60 or over but under 65 who does not need to be available for or able to work, under 60 and long-term sick/disabled, or under 60 and available for or able to work.

Created : 22 January 1993

Database Table : ADULT

Minimum Value : 1

Maximum Value : 3

Units : Integer

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : PSM

Amendments : VC - 5 March 1993 - Change to categories in code 2 as were too restrictive.
 : VC - 22 April 1993 - To expand definition making clear which questions have been used.
 : VC - To add the category indicating a man between 60 and 65 as do not have to be available for work.
 : VC - 9 February 1994 - No version 30 update needed
 : VE - 21 May 1996 - Initial Version 32 update needed - INJPD replaced by INJLONG
 : VE - 14 May 1996 - Initial Version 33 update - coding for long-term sick or disabled in NOLOOK and NOWANT moved from 5 to 6
 : EP - 10 August 1998 - No initial version 34 update needed
 : JC - 17 September 1999 - Security completed, no other changes for V35

Issue date : 6 June 1996

1 Definition

This variable is coded as

- 1 Indicating a man aged 60 or over but under 65 who does not need to be available for or able to work.
- 2 Indicating an adult under the age of 60 and long-term sick or disabled.
- 3 Indicating an adult under the age of 60 and available for or able to work (includes those already working).
- 1 Not applicable to this case - adults over National Insurance Retirement Pension age.
- 2 Unable to derive due to missing values.

 --

ABLE is derived from several variables in the ADULT table which indicate whether or not a person is a man aged between 60 and 65 or if he/she is long-term sick or disabled or any others under pension age who might be available for or able to work.

The following people are to be classed as long-term sick and disabled :

Those who are under Income Support pension age and not currently working ie where WORKING = 2 (no paid work within last 7 days), JOBAWAY = 2 (where the person does not have a job which they were away from) and the reason they were not looking for or did not want to work was because they are long term sick or disabled (NOLOOK/NOWANT = 6). Any other person whose illness or disability has lasted for more than 6 months, ie where INJLONG= 2.

People who are classed as being able to work include all others under Income Support pension age who have not previously been coded.

NB - The FES definition only included those known to be sick or unoccupied. The questions INJLONG and NOLOOK/NOWANT are only asked of people under NI retirement age ie 60 for women and 65 for men.

2 FRS Specification

For each adult

Code Condition

- | | |
|----|---|
| 1 | From table ADULT

If SEX = 1 and AGE >= 60 or < 65 |
| 2 | From ADULT table
If AGE < 60 and INJLONG= 2 or
If AGE < 60 WORKING = 2, JOBAWAY = 2, LIKEWK = 1 and NOLOOK = 6 or
If AGE < 60 WORKING = 2, JOBAWAY = 2, LIKEWK = 2 and NOWANT =6 |
| 3 | From table ADULT

If AGE < 60 and not coded above |
| -1 | Not applicable to this case - people over NI pension age |
| -2 | Unable to derive in this case |

3 Results

Tabulation required to show number of adults falling into each category.

--

4 Test Cases

To be added at a later date.

 --

**CAREAB, CAREAH, CARERE, CAREFR, CARECL, CAREOT, CARECB, CARECH
 HOURAB, HOURAH, HOURRE, HOURFR, HOURCL, HOUROT, HOURCB, HOURCH**

Purpose : To provide summary variables for adult and child carers
 Created : 29 April 1996
 Database Table : ADULT, CHILD
 Minimum Value :
 Maximum Value :
 Units : integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : VE - 21 May 1996 - No initial Version 32 update needed
 : VE - 14 May 1997 - To update for Version 33 by amending NEEDPER codes
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SG - 24 February 1999 – Update for V34 questionnaire changes
 : SB – 25 October 1999 – Security completed, SAS code spilt into adult &
 child
 Issue date : 25 October 1999

1 Definition

The aim of these derived variables is to provide summary information on adult and child carers within the household. For most users' needs, information is only required on *who* is being cared for and *how much time* is spent caring. Any more detailed analysis of carers and those being cared for should be done using the SAS version of the care table.

As from V34, detailed information is also collected for child carers and so these have now been incorporated into this DV.

The variables recording *who* is cared for are coded as:

For each adult or child

CAREAB total number of adults looked after in the same benefit unit (maximum value of 1 for adult carers, because can only be the adult's partner, if there is one)
CAREAH total number of adults looked after in the same household but different benefit unit
CARERE relatives outside the household looked after (maximum value of 1, since questionnaire records "relative" as a single response)
CAREFR friends and neighbours outside the household looked after (maximum value of 1, since questionnaire records "friend/neighbour" as a single response)

 --

CARECL	client of voluntary organisation outside the household looked after (maximum value of 1, since questionnaire records "client of a voluntary organisation" as a single response)
CAREOT	others outside the household looked after (maximum value of 1, since questionnaire records "other non household" as a single response)
CARECB	total number of children looked after in the same benefit unit
CARECH	total number of children looked after in the same household but different benefit unit

0 For all variables - not applicable to this case - adult or child does not look after anybody in same benefit unit/household/outside household etc.

-2 For all variables - unable to derive due to missing values.

It follows that, to calculate the total number of individuals within the household cared for by an individual, add together CAREAB, CAREAH, CARECB and CARECH. To assess whether someone carers for others outside the household, look at CARERE, CAREFR, CARECL and CAREOT.

The variables recording *how much* caring is done are coded as:

For each adult or child

HOURAB	total number of hours spent caring for adults in the same benefit unit
HOURAH	total number of hours spent caring for adults in the same household but different benefit unit
HOURRE	total number of hours spent caring for relatives outside the household
HOURFR	total number of hours spent caring for friends and neighbours outside the household
HOURCL	total number of hours spent caring for clients of voluntary organisation outside the household
HOUROT	total number of hours spent caring for others outside the household
HOURCB	total number of hours spent caring for children in the same benefit unit
HOURCH	total number of hours spent caring for children in the same household but different benefit unit

0 **For all variables** - not applicable to this case - adult or child does not look after anybody in same benefit unit/household/outside household etc.

 --

-2 **For all variables** - unable to derive due to missing values.

It follows that to calculate the total number of hours spent caring for someone within the household add together HOURAB, HOURAH, HOURCB and HOURCH. To calculate the total number of hours spent caring, add together all variables.

2 **FRS Specification**

Set all variables equal to zero.

From CARE record, for each person needing care (NEEDPER), process WHOLOO(x) for all individuals n the household looking after that person (x=1-14 household members).

If WHOLOO(x) is an adult or child in the same BU as the person needing care then

$$\begin{aligned} \text{CAREAB}(x) &= \text{CAREAB}(x) + 1 \\ \text{HOURAB}(x) &= \text{HOURAB}(x) + \text{HOUR}(x) \end{aligned}$$

If WHOLOO(x) is an adult or child in the same HH but different BU to the person needing care then

$$\begin{aligned} \text{CAREAH}(x) &= \text{CAREAH}(x) + 1 \\ \text{HOURAH}(x) &= \text{HOURAH}(x) + \text{HOUR}(x) \end{aligned}$$

If WHOLOO(x) is an adult or child looking after a relative outside the household (NEEDPER=15,16,17,18) then

$$\begin{aligned} \text{CAREER}(x) &= \text{CAREER}(x) + 1 \\ \text{HOURRE}(x) &= \text{HOURRE}(x) + \text{HOUR}(x) \end{aligned}$$

If WHOLOO(x) is an adult or child looking after a friend/neighbour outside the household (NEEDPER=19) then

$$\begin{aligned} \text{CAREFR}(x) &= \text{CAREFR} + 1 \\ \text{HOURFR}(x) &= \text{HOURFR}(x) + \text{HOUR}(x) \end{aligned}$$

If WHOLOO(x) is an adult or child looking after a client of a voluntary organisation (NEEDPER=20) then

$$\begin{aligned} \text{CARECL}(x) &= \text{CARECL}(x) + 1 \\ \text{HOURCL}(x) &= \text{HOURCL}(x) + \text{HOUR}(x) \end{aligned}$$

If WHOLOO(x) is an adult or child looking after others outside the household (NEEDPER=21) then

$$\begin{aligned} \text{CAREOT}(x) &= \text{CAREOT}(x) + 1 \\ \text{HOUROT}(x) &= \text{HOUROT}(x) + \text{HOUR}(x) \end{aligned}$$

If WHOLOO(x) is an adult or child looking after a child in the same BU then

$$\begin{aligned} \text{CARECB}(x) &= \text{CARECB}(x) + 1 \\ \text{HOURCB}(x) &= \text{HOURCB}(x) + \text{HOUR}(x) \end{aligned}$$

--

If WHOLOO(x) is an adult or child looking after a child in the same HH but different BU then
CARECH(x)=CARECH(x)+1
HOURCH(x)=HOURCH(x)+HOUR(x)

-2 If any variables are missing

 --

CURACT*, POACCT*, TESSCT*, OTBSCT*, GILTCT*, UNTRCT*, STSHCT*, NSBOCT*, SAYECT*, PRBOCT*, PEPST*

*** = I (individual); B (Benefit Unit); H (Household); C (Child)**

Purpose : To show the different types of account held by individuals to allow checking of publication tables on assets

Created :

Database Table : ADULT, BENUNIT, HOUSEHOL, CHILD

Minimum Value : 0

Maximum Value :

Units : Integer

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : FRS (publication)

Amendments : VE - 24 May 1996 - Initial update for V32 - ACCOUNT=19 no longer used
 : VE - 19 August 1996 - To update for V32
 : VE - 25 February 1997 - To include new DVs for PEPs and to add ACCOUNT=19 back in
 : SG - 7 October 1997 - No initial changes for V33
 : SG - 30 October 1997 - Yes there are changes for V33 - Account numbers
 : SG - 17 February 1998 - Correct possible child accounts
 : EP - 4 December 1998 - V34 update - Account number changes
 : EP - 21 December 1998 - Add in Child accounts where appropriate
 : JC - 21 September 1999 - Security completed, SAS code split up into sernum, benunit and person levels

Issue date : 17 April, 2003

1 Definition

These variables are coded as

CURACTI etc

- 1 Not applicable to this case (no cases should exist - either individuals, benefit units or households have or don't have accounts).
- 2 Unable to derive due to missing values (not applicable to this specification: variables are a simple count of records).

Following changes to ACCOUNT mapping for V33, OTBNCTI and OTACCTI are no longer derivable, along with their benefit unit and household level equivalents.

 --

2 FRS Specification

Code Condition

Process ACCOUNTS table via PERSON (Adults and children)

Set

CURACTI, POACCTI, TESSCTI, OTBSCTI, GILTSCTI, UNTRCTI, STSHCTI, NSBOCTI, SAYECTI, PRBOCTI, PEPSTI, **POACCTC, OTBSCTC, GILTSCTC, UNTRCTC, STSHCTC, NSBOCTC, PRBOCTC**, CURACTB, POACICTB, TESSCTB, OTBSCTB, GILTSCTB, UNTRCTB, STSHCTB, NSBOCTB, SAYECTB, PRBOCTB, PEPSTB, CURACTH, POACICTH, TESSCTH, OTBSCTH, GILTSCTH, UNTRCTH, STSHCTH, NSBOCTH, SAYECTH, PRBOCTH, PEPSTH to zero

CURACTI	If ACCOUNT=1 then CURACTI=1
POACCTI	If ACCOUNT=2 then POACCTI=POACTTI+1 If ACCOUNT=3 then POACCTI=POACCTI+1 If ACCOUNT=20 then POACCTI=POACCTI+1 If ACCOUNT=21 then POACCTI=POACCTI+1
TESSCTI	If ACCOUNT=4 then TESSCTI=1
OTBSCTI	If ACCOUNT=5 then OTBSCTI=OTBSCTI+1 If ACCOUNT=6 then OTBSCTI=OTBSCTI+1 If ACCOUNT=22 then OTBSCTI=OTBSCTI+1 If ACCOUNT=23 then OTBSCTI=OTBSCTI+1
GILTSCTI	If ACCOUNT= 8 6 then GILTSCTI=GILTSCTI+1 If ACCOUNT=24 then GILTSCTI=GILTSCTI+1
UNTRCTI	If ACCOUNT= 9 7 then UNTRCTI=UNTRCTI+1 If ACCOUNT=25 then UNTRCTI=UNTRCTI+1
STSHCTI	If ACCOUNT= 10 8 then STSHCTI=STSHCTI+1 If ACCOUNT=26 then STSHCTI=STSHCTI+1
NSBOCTI	If ACCOUNT=10 then NSBOCTI=NSBOCTI+1 If ACCOUNT=11 then NSBOCTI=NSBOCTI+1 If ACCOUNT=12 then NSBOCTI=NSBOCTI+1 If ACCOUNT=13 then NSBOCTI=NSBOCTI+1 If ACCOUNT=16 then NSBOCTI=NSBOCTI+1 If ACCOUNT=17 then NSBOCTI=NSBOCTI+1 If ACCOUNT=27 then NSBOCTI=NSBOCTI+1 If ACCOUNT=28 then NSBOCTI=NSBOCTI+1 If ACCOUNT=29 then NSBOCTI=NSBOCTI+1 If ACCOUNT=30 then NSBOCTI=NSBOCTI+1

 --

~~If ACCOUNT=32 then NSBOCTI=NSBOCTI+1~~
~~If ACCOUNT=33 then NSBOCTI=NSBOCTI+1~~
 If ACCOUNT=18 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=19 then NSBOCTI=NSBOCTI+1
~~If ACCOUNT=34 then NSBOCTI=NSBOCTI+1~~
If ACCOUNT=20 then NSBOCTI=NSBOCTI+1

SAYECTI If ACCOUNT=~~20~~ 14 then SAYECTI=SAYECTI+1

PRBOCTI If ACCOUNT=~~24~~ 15 then PRBOCTI=PRBOCTI+1
~~If ACCOUNT=31 then PRBOCTI=PRBOCTI+1~~

PEPSCTI If ACCOUNT=~~44~~ 9 then PEPSCIT=PEPSCTI+1

Repeat for any possible CHILD accounts

For each benefit unit

CURACTB equals all occurrences of CURACTI
 POACCTB equals all occurrences of POACCTI + **POACCTC**
 TESSCTB equals all occurrences of TESSCTI
 OTBSCTB equals all occurrences of OTBSCTI + **OTBSCTC**
 GILTSCTB equals all occurrences of GILTSCTI + **GILTSCTC**
 UNTRCTB equals all occurrences of UNTRCTI + **UNTRCTC**
 STSHCTB equals all occurrences of STSHCTI + **STSHCTC**
 NSBOCTB equals all occurrences of NSBOCTI + **NSBOCTC**
 SAYECTB equals all occurrences of SAYECTI
 PRBOCTB equals all occurrences of PRBOCTI + **PRBOCTC**
 PEPSCITB equals all occurrences of PEPSCIT

For each household

CURACTH equals all occurrences of CURACTB
 POACCTH equals all occurrences of POACCTB
 TESSCTH equals all occurrences of TESSCTB
 OTBSCTH equals all occurrences of OTBSCTB
 GILTSCTH equals all occurrences of GILTSCTB
 UNTRCTH equals all occurrences of UNTRCTB
 STSHCTH equals all occurrences of STSCCTB
 NSBOCTH equals all occurrences of NSBOCTB
 SAYECTH equals all occurrences of SAYECTB
 PRBOCTH equals all occurrences of PRBOCTB
 PEPSCITH equals all occurrences of PEPSCITB

ADULTB

Purpose : To show the number of adults within the Benefit Unit
Created : EP - 9 September 1998
Database Table : BENUNIT
Minimum Value : 0
Maximum Value :
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (general)
Amendments : SB - 17 April 2003 – Security completed
 - Split SAS up into benunit and household
 levels

Issued : 17 April 2003

1 Definition

This variable counts the number of adults within the Benefit Unit.

2 FRS Specification

ADULTB

Process each BENUNIT record in the household in turn, incrementing ADULTB as shown.

Code Condition

For each BENUNIT record:

1 For the first ADULT in the Benefit Unit

Then:

+1 For each further ADULT in the Benefit Unit

ADULTH

Purpose : To show the number of adults within the household
Created : EP – 18 May 1999
Database Table : HOUSEHOL
Minimum Value : 1
Maximum Value :
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (general)
Amendments : SB - 2 September 1999 – Security completed
 - Spilt SAS up into benunit and household
 levels

Issued : 2 September 1999

1 Definition

This variable counts the number of adults within the household.

2 FRS Specification

ADULTH

Process each household record in turn, incrementing ADULTH as shown.

Code Condition

For each HOUSEHOLD:

1 For the first ADULT in the household

Then:

+1 For each further ADULT in the household

 --

BOARDER

Purpose : To indicate the total weekly amount of rent paid by a boarder Benefit Unit.
 Created : VC - 12 March 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : VC 24 March 1993 - by Benefit Unit.
 : VC - 9 February 1994 updated for version 30
 : VC - 25 February 1994 To exclude any amounts using period codes 12 or: 13
 : JS - 20 February 1996 to allow skipped values of CVPD (where CVPAY has
 : been imputed)
 : VE - 23 May 1996 - No initial V32 update needed
 : VE - 14 May 1997 - To amend period codes for V33
 : SG - 7 October 1997 - correct derivation of unable to derive
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 7 September 1999 - Security completed, no other changes for V35
 : SB - 25 November 1999 - Add in housing benefit contributions
 Issue date : 7 October 1997

NB - This is a new variable produced by FRS and does not replace the FES in any way.

1 Definition

This variable is coded as

BOARDER The total weekly amount paid by a benefit unit classed as a boarder to the householder
 for a **room and food**.

-1 Not applicable to this case.

-2 Unable to derive due to missing values.

BOARDER is derived by benefit unit from the variable CONVBL (asks whether person is a boarder, lodger or neither of these). Where CONVBL = 1 which indicates that an adult in the benefit unit is a boarder the amount paid for board and lodge is to be found in CVPAY. **If the adult gets any housing benefit allowances/rates (hbothbu=1) then add these in.**

If CVPD (what period does this cover) = 90 or 95 or 97 (one off/lump sum or any other) they become unable to derive as these amounts cannot be converted to a weekly amount.

 --

Where CVPAY has been imputed, CVPD will be left as "skipped". The program needs to be amended to allow for these cases to be included

2 FRS Specification

For each BENUNIT record and for each adult in the benefit unit

<u>Code</u>	<u>Condition</u>
BOARDER	From ADULT table

If CONVBL=1 get all variables.

If CONVBL = 1 (answered boarder), and CVPD equals -1 or 1-52 the amount of BOARDER is the amount in CVPAY (amount of rent paid). **If HBOTHBU = 1 and HBOTHPD equals -1 or 1-52 add the amount of HBOTHAMT to BOARDER as well.**

If there is more than one adult in the benefit unit, the amount of BOARDER is the total amount paid from both adults.

If any person in the benefit unit is not a BOARDER (CONVBL = 2 or 3) then set BOARDER to skipped.

-1 Not applicable to this case - where CONVBL = 2 or 3 (lodgers or others) or is skipped as the question has not been asked.

-2 Unable to derive due to value of convbl or cvpay missing or where CVPD = 90, 95 or 97

3 Results

Tabulation is required to show the number of boarders paying an amount for board and lodge in weekly bands of

- Under £25
- £25 - £50
- £50 - £75
- £75 - £100
- £100 - £125
- £125 - £150
- £150 or over

4 Test Cases

None as yet

BUAGEGRP, BUETHGRP, BUAGEGR2

Purpose : To show the Age/Ethnicity of the Head of the BU
Created : 4 September 1998
Database Table : BENUNIT
Minimum Value : 1
Maximum Value : 5 (BUETHGRP), 15 (BUAGEGRP), **9 (BUAGEGR2)**
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (hotdecking)
Amendments : JC -6 September 1999 -Security completed, no other changes for V35
Issue date : 4 September 1998

1 Definition

BUAGEGRP and BUETHGRP are hotdeck Dvs used by hotdeck views to define categories for the view. They are coded as :

BUAGEGRP

- 1 Age 16-19
- 2 Age 20-24
- 3 Age 25-29
- 4 Age 30-34
- 5 Age 35-39
- 6 Age 40-44
- 7 Age 45-49
- 8 Age 50-54
- 9 Age 55-59
- 10 Age 60-64
- 11 Age 65-69
- 12 Age 70-74
- 13 Age 75-79
- 14 Age 80-84
- 15 Age 85 and above

BUETHGRP

- 1 White

- 2 Black - Caribbean, African, or neither Caribbean or African
- 3 Indian
- 4 Pakistani or Bangladeshi
- 5 Chinese or Other

BUAGEGR2 is a publication DV and is coded as follows:

- 1 **Age 16 to 24**
- 2 **Age 25 to 34**
- 3 **Age 35 to 44**
- 4 **Age 45 to 54**
- 5 **Age 55 to 59**
- 6 **Age 60 to 64**
- 7 **Age 65 to 74**
- 8 **Age 75 to 84**
- 9 **Age 85 or over**

2 FRS Specification

BUAGEGRP

From ADULT table if UPERSON=1 then

Code Condition

- 1 From BENUNIT table
AGE>=16 and AGE<=19
- 2 From BENUNIT table
AGE>=20 and AGE<=24
- 3 From BENUNIT table
AGE>=25 and AGE<=29
- 4 From BENUNIT table
AGE>=30 and AGE<=34
- 5 From BENUNIT table
AGE>=35 and AGE<=39

- 6 From BENUNIT table
AGE>=40 and AGE<=44
- 7 From BENUNIT table
AGE>=45 and AGE<=49
- 8 From BENUNIT table
AGE>=50 and AGE<=54
- 9 From BENUNIT table
AGE>=55 and AGE<=59
- 10 From BENUNIT table
AGE>=60 and AGE<=64
- 11 From BENUNIT table
AGE>=65 and AGE<=69
- 12 From BENUNIT table
AGE>=70 and AGE<=74
- 13 From BENUNIT table
AGE>=75 and AGE<=79
- 14 From BENUNIT table
AGE>=80 and AGE<=84
- 15 From BENUNIT table
AGE>=85

BUETHGRP

From ADULT table if UPERSON=1 then

- 1 From BENUNIT table
ETHGRP=1
- 2 From BENUNIT table
ETHGRP=2,3 or 4
- 3 From BENUNIT table
ETHGRP=5
- 4 From BENUNIT table

ETHGRP=6 or 7

5 From BENUNIT table

ETHGRP=8 or 9

BUAGEGR2

From BENUNIT table get BUAGEGRP

1 If BUAGEGRP in (1,2)

2 If BUAGEGRP in (3,4)

3 If BUAGEGRP in (5,6)

4 If BUAGEGRP in (7,8)

5 If BUAGEGRP in (9)

6 If BUAGEGRP in (10)

7 If BUAGEGRP in (11,12)

8 If BUAGEGRP in (13,14)

9 If BUAGEGRP in (15)

 --

BUHDBEN

Purpose :To create a variable for use in hot-decking which shows whether any income related benefits are received by a benunit
 Created :19 June 1996
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 2
 Units : Integers
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : SG - 27th June - No updates required for V33
 : EP - 7 September 1998 - Addition of BUHDBEN and HHHDBEN
 : SB - 1 September 1999 – Spilt SAS code up into adult, benunit and household levels
 Issue date : 1 September 1999

1 Definition

This variable is coded as

- 1 Income related benefits received by benunit
- 2 No income related benefits received by benunit
- 2 Unable to derive due to missing values

BUHDBEN is a derived variable in the BENUNIT table. This shows the total amount of income received each week by a benunit from income related benefits. If INIRBEN is greater than zero, a benunit is classified as receiving income related benefits. If INIRBEN is zero, an benunit is classified as not receiving income related benefits.

2 FRS Specification

For each adult

Code Condition

- 1 From ADULT table
 If INIRBEN \geq 0
- 2 From ADULT table

--

If INIRBEN=0

-2 Unable to derive due to missing values - There should be no missing values for BUHDBEN

 --

BUINC, BUEARNS, BSEINC, BUINV, BURPINC, BPENINC, BUDISBEN, BUOTHBEN, BURINC

Purpose : To show the total amount of income received by each benefit unit for use in the FRS publication.
 Created : 2 February 1996
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : INDINC, CHINCDV
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : VE - 23 May 1996 - No initial amendments needed for V32 update
 : SG - 12 November 1997 - No initial amendments needed for V33 update
 : SG - 5 June 1998 - New self-employment DV SEINCAM2
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 24 September 1999 – Security completed, no other changes for V35
 Issue date : 24 September 1999

1 Definition

This variable is coded as

BUINC The total amount of income received each week by all members of the benefit unit.
 0 No income is received by the benefit unit
 -2 Unable to derive due to missing values

This specification, also sets up the component DVs used in the publication. These are:

BUEARNS earned income
 BSEINC self-employment income
 BUINV investment income
 BURPINC retirement pension plus any income support
 BPENINC other pension income
 BUDISBEN disability benefits
 BUOTHBEN other benefits
 BURINC remaining income

2 FRS Specification

For each benefit unit

 --

<u>Code</u>	<u>Condition</u>
BUINC	<p>From ADULT table get INEARNs, SEINCAM2, ININV INRPINC INPENINC, INOTHBEN, INRINC and INDINC for all adults in the benefit unit</p> <p>From CHILD table get CHEARNs, CHINV, CHRINC and CHINDV for all children in the benefit unit</p> <p>BUINC equals total of all occurrences of INDINC and CHINDV</p>
-2	<p>where any of the components are missing</p>

Benefit unit level components are calculated as follows:

BUEARNs equals total of all occurrences of INEARNs and CHEARNs

BSEINC equals total of all occurrences of SEINCAM2

BUINV equals total of all occurrences of ININV and CHINV

BURPINC equals total of all occurrences of INRPINC

BPENINC equals total of all occurrences of INPENINC

BUDISBEN equals total of all occurrences of INDISBEN

BUOTHBEN equals total of all occurrences of INOTHBEN

BURINC equals total of all occurrences of INRINC and CHRINC

these variables should also be set to -2 if any components are missing

BUKIDS

Purpose : Count of number of children within a benefit unit for one parent and two parent families (for publication use)
 Created : 23 February 1999
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 8
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : SB - 9 November 1999 – Security completed, no other changes for V35
 Issue date : 23 February 1999

1 Definition

BUKIDS is an alternative breakdown of children within a benefit unit, splitting by the number of parents. It is for FRS publication use only and is coded as follows:

- 1 Two parent family, one child
- 2 Two parent family, two children
- 3 Two parent family, three children
- 4 Two parent family, four or more children
- 5 One parent family, one child
- 6 One parent family, two children
- 7 One parent family, three children
- 8 One parent family, four or more children

2 FRS Specification

COUNTCH is the number of children within a benefit unit.

Count the number of adults within a benefit unit.

Code Condition

If count of adults = 2 (i.e. two parent family), then

- 1 If COUNTCH = 1
- 2 If COUNTCH = 2
- 3 If COUNTCH = 3
- 4 If COUNTCH >= 4

If count of adults = 1 (i.e. one parent family)

- 5 If COUNTCH = 1

- 6 If COUNTCH = 2
- 7 If COUNTCH = 3
- 8 If COUNTCH = 4
- 1 Not applicable to this case (i.e. no children within the benefit unit)
- 2 Unable to derive BUKIDS

 --

BURENT/TUBURENT

Purpose : To show the rent eligible for housing benefit paid by a benefit unit for accommodation. This is after taking off certain service charges but before the deduction of Housing Benefit. Note: unlike HHRENT, this variable includes rent paid by BOARDERS/LODGERS.

The variable TUBURENT is exactly the same as BURENT but excludes any adjustment for rent holidays – uses TUHHRENT. For ASD3A use only.

Created : 26 January 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : HHRENT
 Children :
 Parents :
 Core variable/user : TAKE-UP
 Issue date : 17 April 2003
 Amendments : VC - 26 April 1993. To divide grsrent by benefit unit not by : household.
 : VC - 18 May 1993. To change name from GRSRENT to BURENT
 : VC - 9 February 1994 Version 30 amendment
 : VC - 25 February 1994 To exclude any records where period code is : 12 or
 13
 : BS - 8 August 1995. Spec amended to include changes to the V31
 questionnaire.
 : JS - 20 February 1996 - to make amendments to allow skipped values where
 amounts have been imputed
 : VE - 24 April 1996 - to broaden spec to include cases where HHSTAT=2 and
 no HB is received
 : VE - 5 June 1996 - No initial amendments needed for V32 update
 : SG - 28 July 1997 - correction to multi benefit unit household calculations
 : SG - 3 December 1997 - V33 updates
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 25 November 1999 - Security completed, changes to BOARDER and
 LODGER variables and removal of HB in these cases.

This variable has been produced for the HBM and for TAKE-UP as they need to know the eligible rent of a benefit unit rather than by household level.

1 Definition

This variable is coded as

 --

BURENT This is the rent eligible for housing benefit paid by a benefit unit for accommodation. This is after taking off certain service charges but before the deduction of Housing Benefit.

-1 Not applicable to this case - owners, rent-free and non-householders BU (but boarder/lodgers should be included) and squatters.

-2 Unable to derive because of missing values.

This is the total amount of eligible rent paid by a benefit unit and is derived from three other derived variables -HHRENT, BOARDER and LODGER.

The BOARDER and LODGER variables provide the amount of (gross) rent a particular boarder/lodger benefit unit pays and should be attached to the relevant benefit unit. (See comments below on food/fuel included)

All tenure types are included when looking at variables to ensure cases where the head of household is an owner occupier but has boarders/lodgers are accounted for.

HHRENT (see separate spec) derives the total amount of eligible rent paid by a household. BURENT produces this on a benefit unit basis . Therefore, if HHSTAT = 1 (indicating a conventional household of single person, couple with other family members and/or boarders/lodgers) the total amount of eligible rent produced from HHRENT should be attached to the record of benefit unit no 1, as this is the main benefit unit responsible for paying the household rent. Boarder or lodger rents are separately identified through the BOARDER and LODGER derived variables. These should be attached to the relevant BU which will not be the 1st BU. We ought to deduct amount included for food/fuel, but as set amounts for these facilities vary so much need questions in future to identify if rent includes these and if so, what value these represent.

HHSTAT=2 cases:

Where HHSTAT = 2 (indicating that the household is shared on an equal basis and the head of household is unclear or arbitrary) the household rent needs further work to produce a proportional amount of rent for each benefit unit within the household. To do this proportions are derived and applied to HHRENT using RENT, SRENTAMT, HBENAMT (the latter for 1st BU where rent is declared after HB deducted).

NB There are a very small number of cases in 94/5 where SRENTAMT does not exist pre imputation and there will be NONE post imputation so the above will always be possible.

2 FRS Specification

For each household

Code Condition

 --

BURENT From HOUSEHOL table, if TENURE = 1 2 or 3 or 4 (renting) or 5 get HHRENT and HHSTAT variables

From BENUNIT table get BOARDER and LODGER variables

If HHSTAT = 1 or -1, then

if BENUNIT=1, and not TENURE=1 or 2 BURENT = HHRENT
 if BENUNIT>1 and either BOARDER or LODGER exists BURENT for that benefit unit is the amount in BOARDER or LODGER.

If HHSTAT=2

if RENT (of 1ST BU) declared after HB deducted (HBENCHK=2) and RENTPD, HBENPD and SRENTPD<> 90, 95 AND 97 then
 $TOTRENT = (RENT + HBENAMT \text{ (for 1st BU)}) + (SRENTAMT + HBOTHAMT \text{ (for all remaining BUs)})$

(for 1st BU) set $BURENT = (RENT + HBENAMT) / TOTRENT * HHRENT$
 (for other BUs) set $BURENT = (SRENTAMT + HBOTHAMT) / TOTRENT * HHRENT$

but if RENT (of 1ST BU) declared before HB deducted (HBENCHK=1) or no HB received by 1st BU (HBENAMT<=0) then
 $TOTRENT = RENT + SRENTAMT + HBOTHAMT \text{ (for all remaining BUs)}$

(for 1st BU) set $BURENT = RENT / TOTRENT * HHRENT$
 (for other BUs) set $BURENT = (SRENTAMT + HBOTHAMT) / TOTRENT * HHRENT$

Above specification assumes that all 90, 95 AND 97 period codes have been converted, additional checks need to be put in for period codes for RENT, HBENAMT and SRENTAMT

In shared households, information on service charges etc are only collected for the first benefit unit and not for second/third benefit units. Therefore, the sum of SRENTAMT and RENT/HBENAMT may not equal total HHRENT. To maintain consistency with HHRENT it is necessary to share out total rent pro-rata.

-1 skipped (household contains no individuals paying rent)

-2 unable to derive due to missing values or period codes 90, 95 AND 97

3 Results

--

Tabulation is required to show the total amount of eligible rent a household has to pay sorted into bands of the following weekly amounts.

- Under £25
- £25 - £50
- £50 - £75
- £75 - £100
- £100 - £125
- £125 - £150
- £150 - £175
- £175 - £200
- £200 and over

 --

CHINCDV, CHEARNS, CHINV, CHRINC

Purpose : To show the total amount of income received by a child for use in the FRS publication (based on INCOMECH).
 Created : 2 February 1996
 Database Table : CHILD
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : VE - 5 June 1996 - No initial amendments needed for V32 update
 : SG - 31 December 1997 - V33 updates, period codes, account numbers
 : EP - 4 December 1998 - V34 update - change account numbers
 : SB - 17 April 2003 - Security completed, no other changes for V35
 Issue date : 17 April 2003

1 Definition

This variable is coded as

CHINCDV The total amount of income received each week by a child. This variable includes earnings from any spare time jobs, income from trust funds, from savings accounts and other assets.

0 Child has no sources of income

-2 Unable to derive due to missing values

The variable CHINCDV is derived from the variables CHAMTERN, ACCOUNT, ACCINT (which produce the amount of income from a child's investment accounts), CHTST and CHAMTTST (which produce the amount of income from trust funds). If a child possesses any or all of these incomes the variable is then the total amount received each week.

ACCINT and CHAMTTST are database variables created to hold the amounts collected in block chinc

The DV is also used to create components of gross income which are accumulated with adult versions to obtain BU and HH level variables. These are:

CHEARNS children's earnings (identical to CHAMTERN but set to zero if not applicable)

 --

CHINV child's investment earnings
 CHRINC remaining children's income

The variable has been extended to include any income from grants or scholarships. Any income from free school meals, prescriptions etc are counted in adult variables.

2 FRS Specification

For each child

<u>Code</u>	<u>Condition</u>
CHINCDV	From CHILD table set CHEARNS to zero If CHEARNS1 = 1 exists and CHPDERN equal to -1 or 1-10, 13,26 or 52 then CHEARNS=CHAMTERN From ACCOUNTS table set CHINV to zero If ACCOUNT = 2 get ACCINT and add to CHINV If ACCOUNT = 3 get ACCINT and add to CHINV If ACCOUNT = 5 get ACCINT and add to CHINV If ACCOUNT = 6 get ACCINT and add to CHINV If ACCOUNT = 7 get ACCINT and add to CHINV If ACCOUNT = 8 get ACCINT and add to CHINV From CHILD table set CHRINC to zero If CHEARNS2=1 and CHPDTST equal to -1 or 1-10, 13,26 or 52 (child has an income from a trust fund) get CHAMTTST and add to CHRINC If TOTGNTCH exists then add TOTGNTCH/52 to CHRINC CHINCDV=CHEARNS+CHINV+CHRINC
-2	Unable to derive if any components are missing

EMP, EMPHOH, SICK, SICKHOH, PENAGE, PENHOH

Purpose : To show household composition in publication
Created : 7 September 1998
Database Table : HOUSEHOL
Minimum Value : 0
Maximum Value : 1
Units :
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : ASD3E
Amendments : JC -6 September 1999 -Security completed, no other changes for V35
 : SB - 5 November 1999 – removal of JCREG variable
Issue date : 5 November 1999

1 Definition

All the following DVs have a value of 1 if the definition is true of household, else a value of 0

EMP - This designates that there are one or more unemployed adults under state pension age in the household including the head of household

EMPHOH - This designates that there are one or more unemployed adults under state pension age in the household not including the head of household

PENAGE - This designates that there are one or more adults over state pension age in the household including the head of household

PENHOH - This designates that there are one or more adults over state pension age in the household not including the head of household

SICK - This designates that there are one or more sick/disabled adults under state pension age in the household including the head of household

SICKHOH - This designates that there are one or more sick/disabled adults under state pension age in the household including the head of household

The removal of JCREG variable from the questionnaire doesn't effect the accuracy of SICK and SICKHOH. The other variables catch all the cases when JCREG=1.

2 FRS Specification

From HOUSEHOL table

EMP, EMPHOH

```
IF head of household THEN
    IF under state pension age and unemployed (EMPSTATI=5)
        THEN EMP=1
ELSE IF under state pension age and unemployed and NOT head of household
    THEN set EMPHOH flag
```

PENAGE, PENHOH

```
IF head of household THEN
```

```
IF over state pension age
    THEN PENAGE=1
ELSE IF over state pension age and NOT head of household
    THEN set PENHOH flag
```

SICK, SICKHOH

```
IF head of household THEN
    IF under state pension age and sick ((health=1 and hprob=1) or jreg=1 or lareg=1 or
        rstrct IN (1,2))
        THEN SICK=1
ELSE IF under state pension age and sick and NOT head of household
    THEN set SICKHOH flag
```


CWATAMTD

Purpose: To show total amount of council water charge paid by Scottish households
 Database Table : HOUSEHOL
 Lead User : FRS (general)
 Related variables:
 Children: HHRENT
 Parents: STDREGN
 Amendments : VE - 10 June 1996 - Amended for V32. COMMINC is no longer asked and WHYNOCT, a new question has been incorporated
 : JS - 13 November 1996, to delete cwatpd (figures only collected from annual statement and converted to weekly amounts as part of the conversion process)
 : VE - 20 November 1996 - to simplify specification
 : VE - 4 February 1996 - To bring into line with HBAI by using a lower Council Tax Band if a lower valuation band has been used
 : SG - 24 June 1997 - No changes needed for V33
 : SG - 19 November 1997 - Allow for discounts
 : SG - 8 January 1998 - document situation for when council tax band not sep valued
 : SG - 23 January 1998 - if council tax band not sep valued then set to skipped
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 7 October 1999 - Security completed, no other changes to V35
 Issue date: : 17 April 2003

1 Definition

CWATAMTD The total weekly amount of council water charge paid by Scottish Households
 -1 Not applicable to this case
 -2 Unable to derive due to missing values

This variable is set up using CWATAMT and a look up table (data supplied from the Scottish Office).

If the household receives a discount then we apply that. The discount is applied however the CWATAMT amount has been derived (ie if from respondent via bill or by lookup in this DV).

If the household has CTBAND=9 (not separately valued) then set CWATAMTD to skipped (-1).

2 FRS Specification

For each Scottish household (STDREGN=11) from HOUSEHOL table

<u>Code</u>	<u>Condition</u>
CWATAMTD	From HOUSEHOL table
	Set DISCOUNT

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

If CTDISC=1 then
 If CT25D50D=1 then DISCOUNT=25%
 Else DISCOUNT=50%
Else DISCOUNT=0%

If CWATAMT exists then CWATAMTD equals CWATAMT

Set CTBANDA

If CTBAND=9 then CWATAMTD=-1

ELSE

 If CTBAND>0 and CTBAND<9 then CTBANDA=CTBAND
 If (CTLVBAND=1 and CTLVCHK=2 and CTBAND>1) then
 CTBANDA=CTBAND-1

Else apply look up table to calculate CWATAMTD from LAC and CTBAND A.

Apply the calculated DISCOUNT to all CWATAMTD

- 1 Not applicable (see above)
- 2 Unable to derive due to missing values.

 --

DEDUCTS

Purpose : To show the total amount of deductions from pay all employment, other than
 : Income Tax, National Insurance contributions and superannuation/pension
 : deductions.
 Created : 20 January 1993
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Period : Weekly amount
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user :
 Issue date : 17 April 2003
 Amendments : VC - 11 February 1993 change to multi responses
 : VC - 8 March 1993 changed to show for main employment only
 : VC - 23 April 1993 to expand definition to show meaning of
 questions/database variables.
 : VC - 22 July 1993 Coding highlighted a problem with the charity section as
 the variable AMTTAXF is not asked after CHARITY = 1 but after CHRTAXF = 1
 indicating that the deduction has been made under the tax free scheme.
 : VC 11 February 1994 Changes to reflect version 30 changes
 : VC - 25 February 1994 To exclude any deductions which would be converted
 to a weekly amount using the pay period
 : JS - 20 February 1996 to allow for skipped values where variables have been
 imputed
 : VE - 24 May 1996 - Initial amendments for V32 - New category of
 OTHDED (OTHDED=6 - repayment of loan from employer)
 : VE - 6 November 1996 - Charity part of spec made clearer
 : VE - 27 November 1996 - To amend spec so that if OTHDED =1 but
 DEDUC=-1 (due to editing/imputation), DEDUCTS is not set to -2
 : VE - 14 May 1997 - Initial amendments for V33 - new category of OTHDED
 (OTHDED7 - private medical insurance)
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 7 October 1999 - Security completed, Deletion of CHRTAXF, AMTTAXF,
 CHROTH and AMTOTH variables and replaced by CHRAMT

NB - This does not include any deductions for superannuation, pension schemes or additional voluntary contributions as these are dealt with in a separate variable called SUPERAN.

1 Definition

This variable is coded as

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

DEDUCTS The total amount of any deductions from pay from all employment excluding Income Tax, National Insurance Contributions and deductions for superannuation/pension schemes or AVCs.

-1 Not applicable to this case

-2 Unable to derive variable

DEDUCTS is derived from the variables CHRAMT, OTHDED3 - OTHDED8, DEDUC3 -DEDUC7 and DEDOTH. CHRAMT deals with deductions paid to charities and OTHDED3 - OTHDED8 indicate whether payments are made to Trade Unions, Friendly Societies Sports and Social clubs, repayment of loans from employers, private medical insurance or any others not previously mentioned. DEDUC3 - DEDUC7 and DEDOTH store the amounts paid.

OTHDED3 - OTHDED8 are database variables produced from OTHDED, which when = 1 indicate whether that person has a particular deduction. DEDUC3 - DEDUC7 and DEDOTH hold the amount for each deduction respectively and are also database variables produced from DEDUC and DEDOTH.

NB - For information OTHDED1 = 1 would represent pension/superannuation deductions and OTHDED2 = 1 represents AVCs.

~~The original version of the spec used the question/variable CHARITY. However, a problem emerged during coding which showed that the amount held in AMTTAXF was dependent on another variable CHRTAXF which asks additionally if the deduction is through the tax free scheme and the amount collected refers to those deductions only. If analysts need information about all charitable deductions the questionnaire will have to be changed.~~

If PAYPD = 90, 95 or 97 the record must be rejected as unable to derive. This variable is used by the database conversion programme to convert the amount of deduction into a weekly rate. 90, 95 or 97 refer to less than one week/one-off/lump-sum deductions or any other period. If the variables cannot be converted the record must be rejected.

~~Where variables PAYAMT, CHRTAXF and OTHDED1-8 have been imputed, questions which follow will have been skipped and as a result DEDUCTS will be set to "unable to derive". To overcome this problem, the coding has been altered to:~~

- ~~i allow skipped PAYPD~~
- ~~ii impute AMTTAXF to overall mean CHRMEAN in constants table
(h:\asd4a\frs\dvars\docs\spreads.xls)~~
- ~~iii assume where PAYAMT has been imputed (and CHRTAXF and OTHDED1-8 skipped) no deductions are made~~

With the deletion of the base variables AMTTAXF, AMTOTH, CHROTH and CHRTAXF the code now uses CHRAMT hence solving the problem of imputed values of CHRTAXF.

--

 --

2 FRS Specification

For each ADULT

<u>Code</u>	<u>Condition</u>
DEDUCTS	<p>From JOB table, for all jobs a person has</p> <p>If PAYPD equals -1 or 1 to 52 do the following -</p> <p>If CHARITY = 1 (deductions for charities) and CHRTAXF = 1 (deductions made under tax free scheme) and AMTAXF exists then DEDUCTS = AMTAXF (amount deducted)</p> <p>Else if CHARITY = 1 (deductions for charities) and AMTAXF = -1 then DEDUCTS = DEDUCTS + CHRMEAN (from constants table)</p> <p>If CHROTH = 1 (other deductions for charities) then DEDUCTS = DEDUCTS + AMTOTH</p> <p>If CHARITY =1 (deductions for charities) and CHRAMT >= 0 then DEDUCTS = DEDUCTS + CHRAMT (amount deducted for charities)</p> <p>If OTHDED3 = 1 and DEDUC3>=0, DEDUCTS=DEDUCTS+DEDUC3 (amount of union fees) else if OTHDED3=-1 DEDUCTS=DEDUCTS</p> <p>If OTHDED4 = 1 and DEDUC4>=0, DEDUCTS=DEDUCTS+DEDUC4 (amount for friendly societies) else if OTHDED4=-1 DEDUCTS=DEDUCTS</p> <p>If OTHDED5 = 1 and DEDUC5>=0, DEDUCTS=DEDUCTS+DEDUC5 (amount for sports clubs or specialised pastimes) else if OTHDED5=-1 DEDUCTS=DEDUCTS</p> <p>If OTHDED6 = 1 and DEDUC6>=0, DEDUCTS=DEDUCTS+DEDUC6 (amount for repayment of a loan from employer) else if OTHDED6=-1 DEDUCTS=DEDUCTS</p> <p>If OTHDED7 = 1 and DEDUC7>=0, DEDUCTS=DEDUCTS+DEDUC7 (amount for private medical insurance) else if OTHDED7=-1 DEDUCTS=DEDUCTS</p> <p>If OTHDED8 = 1 and DEDOTH>=0, DEDUCTS=DEDUCTS+DEDOTH (amount of any other deductions not included above) else if OTHDED8 = -1 DEDUCTS=DEDUCTS</p>
-1	Not applicable to this case.
-2	Unable to derive as any of the above variables are missing or PAYPD = 90, 95 or 97 (less than one week/one-off/lump-sum or other period)

3 Results

Tabulation is required to show the number of adults with deductions from pay by the weekly amount of deduction sorted into bands of, for example

--

Under £5.00
£5.00 - £10.00
£10.00 - £15.00
£15.00 - £20.00
£20.00 - £25.00
Over £25.00

DEPCHLDB

Purpose : To show the number of children within the Benefit Unit
Created : EP - 9 September 1998
Database Table : BENUNIT
Minimum Value : 0
Maximum Value :
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (general)
Amendments : SB - 30 November 1999 - Security completed, no other changes for V35
Issued : 17 April 2003

1 Definition

This variable counts the number of dependent children within the Benefit Unit

2 FRS Specification

DEPCHLDB

Process each BENUNIT record in the household in turn, incrementing DEPCHLDB as shown.

Code Condition

For each BENUNIT record:

0 If no dependent children in the Benefit Unit

1 For the first CHILD in the Benefit Unit

Then:

+1 For each further CHILD in the Benefit Unit

DEPCHLDH

Purpose : To show the number of children within the household
Created : EP – 18 May 1999
Database Table : HOUSEHOL
Minimum Value : 0
Maximum Value :
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (general)
Amendments : SB - 30 November 1999 - Security completed, no other changes for V35
Issued : 1 September 1999

1 Definition

This variable counts the number of dependent children within the Household.

2 FRS Specification

DEPCHLDH

Process each household record in turn, incrementing DEPCHLDH as shown.

Code Condition

For each HOUSEHOLD:

- 0 If no dependent children in the household
- 1 For the first CHILD in the household

Then:

- +1 For each further CHILD in the household

 --

DEPDEDS

Purpose : To indicate the class of non-dependency applicable to each benefit unit.
 Created : 14 January 1993
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 9
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents : TOTHOOURS
 Core variable/user : PSM
 Amendments : VC - 28 January 1993,
 : VC - 3 February 1993 (NB2), VC - 9 February 1993 : change to multi :
 : response.
 : VC - 8 March 1993, To clarify which tables certain variables come from.
 : VC - 16 March 1993, see Ms Odwell 's minute of 10 March 1993 for :
 : comments and changes to categories.
 : VC - 24 March 1993. To add new categories of boarder and lodger and to :
 : remove head of household benefit unit and whether blind/in receipt of AA/DLA
 : VC - 25 June 1993. To split category 5 into 18 - 24 receiving IS and 25 and :
 : over and receiving IS.
 : VC - 11 February 1994 Amended to reflect changes to version 30
 : JS - 20 February 1996: amended to reflect changes to benefits for V31
 : VE - 4 June 1996 - Initial V32 changes - to amend TRAIN category for YTS
 : SG - 24 June 1997 - Initial V33 changes - to amend R01 category for non
 : family member
 : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 30 September 1999 - Security completed, no other changes for V35
 Issue date : 17 April 2003

NB Boarders/lodgers are always non-family members.

1 Definition

This variable is coded as follows, if anybody in the benefit unit meets one of the criteria. If more than one of the criteria are met, the higher code takes precedence (ie 1 is highest). Boarders or lodgers are to be coded first as they are separate conditions and only any remaining non-dependents in the household should be coded as 3 - 9.

- 1 Boarder
- 2 Lodger
- 3 Aged 18 or over and working more than 16 hours a week
- 4 Aged 18 or over and on YTS
- 5 Aged 18 to 24 and in receipt of Income Support
- 6 Aged 25 and over and in receipt of Income Support

 --

- 7 Students
- 8 Any others aged over 18
- 9 Aged 16 - 17
- 1 Not applicable to this case
- 2 Unable to derive

This variable indicates to which class of non-dependency a benefit unit belongs. DEPDEDS will be derived from several variables from the BENUNIT, ADULT and BENEFITS tables.

2 FRS Specification

For each BENEFIT UNIT from BENUNIT record where BENUNIT > 1 (indicating that the BU is not the head of household BU) and for each adult in the benefit unit.

Code Condition

- 1 From ADULT table, if R01 = 18 (indicating that the relationship to person number 1 (head of household) is a non-family member) and CONVBL = 1 (asked if boarder, lodger or other - 1 indicates a boarder).
- 2 From ADULT table, if R01 = 18 and CONVBL = 2 (as above but indicating a lodger).
- 3 From ADULT table, if AGE >= 18, WORKING = 1 (currently working) and TOTHOURS >= 16 (TOTHOURS is a derived variable indicating the total number of hours worked by an adult) or
 If AGE >= 18, WORKING = 2 (not currently working), JOBAWAY = 1 (away from normal job) and TOTHOURS >= 16.
- 4 From ADULT table, if AGE >= 18 and TRAIN = 1 (TRAIN asks if adult was on any form of govt training scheme, 1 = Youth Training).
- 5 From ADULT table, if AGE >= 18 and <= 24 and from BENEFITS table, if BENEFIT = 19
- 6 From ADULT table, if AGE >= 25 and from BENEFITS table, if BENEFIT = 19
- 7 From ADULT table, if AGE >= 18, FTED = 1 (in full-time education) and TYPEED = 7 (indicates university, polytechnic or any other college of higher education).
- 8 From ADULT table, if AGE >= 18 and not previously coded.
- 9 From ADULT table, if AGE = 16 or 17 (NB - this will pick out non-dependent adults aged 16 or 17, dependants who are aged 16 or 17 will be in the child table and are dealt with separately)
- 1 Not applicable to this case

--

-2 Unable to derive due to missing values.

3 Results

Tabulation to show total number of non-dependent benefit units and numbers falling into each category

4 Test Cases

- A Household with only head of household benefit unit
- B Household with one non-dependent, registered blind
- C Household with one non-dependent, aged 16 and not on YTS
- D Household with one non-dependent, aged 16 on YTS
- E Household with one non-dependent, aged 19, full-time student
- F Household with one non-dependent, aged 19 receiving Income Support
- G Household with one non-dependent, aged 20 earning £75 per week
- H Household with one non-dependent, aged 20 earning £105 per week
- I Household with one non-dependent, aged 20 earning £150 per week
- J Household with one non-dependent, aged 30 receiving IS
- K Household with one non-dependent, aged 30 not on IS

 --

DISHBBU

Purpose : To indicate if any person in the benefit unit receives Income Support, Housing Benefit or Family Credit for use in the disability survey.
 Created : VE - 4 February 1997
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value : 7
 Units : integer
 Validations :
 Related Variables : HBINDHH, HBINDBU
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : EP - 10 August 1998 - No initial Version 34 update needed
 : SB - 8 November 1999 - Security complete, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable is coded as

DISHBBU	Indicates that someone in the benefit unit is receiving Income Support, Housing Benefit or Family Credit as follows,
0	No HB/IS/FC
1	If receives FC only (no IS or HB)
2	If receives HB only (no IS or FC)
3	If receives IS only (no HB or FC)
4	If receives both HB and FC (no IS)
5	If receives both HB and IS (no FC)
6	If receives both FC and IS (no HB)
7	If receives HB, FC and IS
-1	Not applicable to this case: these should not occur since questions are asked of all households
-2	Unable to derive because of missing values

 --

This variable is derived from questions in the HOUSEHOL and ADULT records

Receipt of Housing Benefit is identified from two separate questions depending on the type of household:

- i HBENEFIT = 1 (has received HB in connection with last rent payment) (HOUSEHOL record)
 This question is asked BENUNIT=1 and if HHSTAT=1.
- ii HBOTHBU = 1 (receives Housing Benefit) (ADULT record). This question is asked if
 BENUNIT>1 and HHSTAT=2 or CVPAY>0.

IS receipt is identified where any person in the benefit unit answers "yes" to BEN3Q2 (ADULT record - are you at present receiving IS).

FC receipt is identified where any person in the benefit unit answers "yes" to BEN3Q3 (ADULT record - are you at present receiving FC).

2 FRS Specification

For each benefit unit in the household

Code Condition

DISHBBU

- 0 If (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=2 (HB not in receipt
 for secondary benefit units)
 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
 and for each adult in benefit unit, BEN3Q3<>1 (FC not in receipt)
- 1 If (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=2 (HB not in receipt
 for secondary benefit units)
 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
 and for at least one adult in the benefit unit, BEN3Q3=1 (FC in receipt)
- 2 If (BENUNIT=1 and (HBENEFIT=1) (HB in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=1 (HB in receipt for
 secondary benefit units)
 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
 and for each adult in benefit unit, BEN3Q3<>1 (FC not in receipt)
- 3 If (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=2 (HB not in receipt
 for secondary benefit units)
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
 and for each adult in benefit unit, BEN3Q3<>1 (FC not in receipt)

 --

- 4 If (BENUNIT=1 and HBENEFIT=1) (HB in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=1 (HB in receipt for
 secondary benefit units)
 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
 and for at least one adult in benefit unit, BEN3Q3=1 (FC in receipt)

- 5 If (BENUNIT=1 and HBENEFIT=1) (HB in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=1 (HB in receipt for
 secondary benefit units)
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
 and for each adult in benefit unit, BEN3Q3<>1 (FC not in receipt)

- 6 If (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=2 (HB in receipt for
 secondary benefit units)
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
 and for at least one adult in benefit unit, BEN3Q3=1 (FC in receipt)

- 7 If (BENUNIT=1 and HBENEFIT=1) (HB in receipt for first BU)
 or if BENUNIT>1, and (HHSTAT=2 or CVPAY>0) and HBOTHBU=1 (HB in receipt for
 secondary benefit units)
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
 and for at least one adult in benefit unit, BEN3Q3=1 (FC in receipt)

- 1 Not applicable to this case

- 2 Unable to derive as any of the above are missing.

 --

DISINDHB

Purpose : To indicate whether one or both adults in a benefit unit are blind or disabled.
 Created : 13 January 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value : 6
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : PSM
 Amendments : VC - 28 January 1993, VC - 9 February 1993 change to multi response.
 : VC - Up to date benefit code received change to AA and DLA codes.
 : VC - 29 March 1993, simplified FRS specification inserted to make coding :
 easier.
 : VC - 23 April 1993 To expand definition to show meaning of :
 questions/database variables.
 : VC - 25 June 1993 To change spec to include all benefit units and :
 increase number of categories to show 1 blind, 2 blind, 1 disabled, 2 disabled :
 or neither.
 : VC - 11 February 1994 Amended to reflect version 30 changes
 : JS - 20 February 1996 amended to reflect v31 changes
 : VE - 13 May 1996 Amended to clarify the situation when an individual
 is both blind and disabled
 : VE - 4 June 1996 - Amended to reflect initial V32 changes
 : SG - 24 June 1997 - No changes required for V33
 : EP - 10 August 1998 - No initial Version 34 update needed
 : JC - 17 September 1999 - Security completed, no other changes for V35
 Issue date : 17 April 2003

1 Definition

This variable is coded as

- 1 1 person in benefit unit blind.
- 2 2 people in benefit unit blind.
- 3 1 person in benefit unit disabled.
- 4 2 people in benefit unit disabled.
- 5 1 blind person and 1 disabled person in benefit unit.
- 6 No person in benefit unit blind or disabled.

 --

-2 Unable to derive variable due to missing values

The variables used to produce DISIND are to be found in the ADULT table and are produced for all benefit units.

The first category is fulfilled if only one person in a benefit unit is registered blind (SPCREG1 = 1) and any other member of the same benefit unit is neither blind nor disabled. SPCREG1 is a database variable which is created from the question SPCREG and indicates that the person is registered blind. However, if there are two members of the benefit unit who are blind category 2 is appropriate, in this case SPCREG1 = 1 applies to both adults.

Categories 3 and 4 are used in a similar way if there are one or two members of the benefit unit classed as disabled. This classification is fulfilled if a person is receiving the care component of Disability Living Allowance BEN2Q1=1 receiving Attendance Allowance BEN2Q3=1 or where Attendance Allowance has been awarded AA to start at a later date B2QFUT3=1.

The fifth category is used where there are two members of the benefit unit and one is blind and the other is disabled.

The sixth category is where no adult in that benefit unit fulfils any of the above categories.

An adult who appears to be classed as both blind and disabled is classified as disabled. This gives rise to the following coding system:

- A Neither blind nor disabled**
- B Disabled**
- C Blind**
- D Both blind and disabled**

Person 1	A	B	C	D
Person 2				
A	6	3	1	3
B	3	4	5	4
C	1	5	2	5
D	3	4	5	4

2 FRS Specification

For the each BENUNIT record in each household

 --

From ADULT table for each adult in the benefit unit

Count total number of adults in benefit unit where SPCREG1 = 1 = blind (temporary variable used for DISIND only).

Count total number of adults in benefit unit where BEN2Q1=1 or BEN2Q3=1 or B2QFUT3 = 1 = dis (temporary variable)

(preset temporary variables to 0)

<u>Code</u>	<u>Condition</u>
1	If blind = 1 and dis = 0
2	If blind = 2 and dis = 0
3	If blind = 0 and dis = 1
4	If blind = 0 and dis = 2
5	If blind = 1 and dis = 1
6	Any other benefit unit not previously coded (where blind = 0 and dis = 0)
-2	Unable to derive because any of the above variables have missing values.

3 Results

Tabulation needed to show number of benefit units falling into each category.

4 Test Cases

To be added at a later date.

ECOTYPBU, EMPSTATC

Purpose : To derive an HBAI type economic status indicator for each benefit unit.
 Created :
 Database Table : BENUNIT, ADULT
 Minimum Value : 1
 Maximum Value : 8
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : Take-up, **HBAI** (**bold** denotes lead user)
 Amendments : VE - 21 May 1996 - Amended for initial Version 32 changes and to change full-time hours cut-off to greater than or equal to 30 (not greater than 29)
 : VE - 20 June 1996 - To change lead user to ASD6A
 : SG - 24 June 1997 - V33 updates, Work Trial counted as training.
 : SG - 5 January 1998 - look for work changes
 : SG - 23 January 1998 - full time hours cut-off 31 hrs and training schemes count as work
 : SG - 28 April 1998 - Govt training scheme is FT work equivalent
 : EP - 11 August 1998 - V34 updates, name changes for Hours Worked variables
 : SB - 18 October 1999 - Security completed, SAS code spilt up
 DVs still derived in the same way and no changes to code routing

 Issued : 18 October 1999

1 Definition

This variable is coded by first creating an economic status variable for each adult and then from this deriving a variable for each benefit unit. As far as possible the variable has been coded to be consistent with the FES economic status variable detailed in FES Appendix 46.

The individual economic status variable EMPSTATC is coded as follows (EMPSTATC should also be added to the data base and flat file):

1 = FT SE
 2 = FT Emp
 3 = PT Emp or PT SE
 4 = Unemployed
 5 = Not working for any other reason.

EMPSTATC is derived from several variables from the adult and job records. Part time work is defined as employment or self employment for less than 31 hours.

An individual is self-employed if they record they are self-employed EMPSTAT=2 and work 31 hours or more in their main job (~~QHRSELF~~ **Hours Worked including overtime** >= 31)

An individual is employed if they record that they are an employee (EMPSTAT=1), and are either not away from work or away from work by on full pay (this is set negatively by ABSPAY<>2,3 - ie not away from work on reduced pay) and they work 31 or more hours a week in their main job ($QHRS+EMPOVT$ **Hours worked including overtime** ≥ 31), or if they are on a Govt Training Scheme.

Part Time work, either employed or self employed is set as for the above two conditions but with the hours condition set at less than 31 hours.

EMPSTATC is set as unemployed by the following conditions:

where an individual is in receipt of JSA (BEN3Q1=1);
 or where an individual is looking for work or training and wants work and can start work in next two weeks
 where an individual is waiting to start a job, either WAIT=1 or JOBAWAY=3

Anyone not coded as the above is coded as not working for any other reason.

ECOTYPBU is coded as follows:

- 1 = Self Employed
- 2 = Single or Couple all in full-time work
- 3 = Couple, one in ft work, one in pt work
- 4 = Couple, one in ft work, one not working
- 5 = One or more in pt work
- 6 = Head or Spouse aged 60+
- 7 = Head or Spouse Unemployed
- 8 = Other

The coding for this is quite straightforward once EMPSTATC is coded.

2 FRS Specification

EMPSTATC

For each adult get all values from adult table:

Tests are applied sequentially with the coding determined by the last test to be satisfied:

Code

If EVEROT = No (No overtime worked)
 then XHRS = TOTUS1
 else XHRS = DVTOTHRU

- 5 All cases first coded to not working for any other reason and then overwritten when one of the conditions apply

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

- 4 (from benefit record BEN3Q1=1) or
(LOOK=1 and LOOKWK=1 and START=1) or
WAIT=1 or
JOBAWAY=3)
- 3 ((EMPSTAT=1 or EMPSTAT=2) and
(ABSPAY<>2,3 and, from first job record, ~~QHRSELT XHRS~~ **QHRSELT XHRS < 31**) or (from first job record
~~QHRSELT XHRS < 31~~)
- 2 (TRAIN = 1,2,3,4,5,6,7) or (EMPSTAT=1 and ABSPAY<>2,3 and, from first job record,
~~QHRSELT XHRS >= 31~~)
- 1 (EMPSTAT =2 and, from first job record, ~~QHRSELT XHRS >= 31~~)

ECOTYPBU

For each Benefit Unit:

Tests are applied sequentially with the coding determined by the last test to be satisfied:

Code

- 8 All cases first coded to other and then overwritten when one of the conditions apply
- 7 EMPSTATC=4 for either the first or the second adult (if there is one)
- 6 AGE>59 for either the first or second adult (if one exists).
- 5 EMPSTATC=3 for either first or second adult (if one exists).
- 4 EMPSTATC=2 for first adult and EMPSTATC=4 or 5 for second adult, or vice versa.
- 3 EMPSTATC=2 for first adult and EMPSTATC=3 for second adult, or vice versa.
- 2 EMPSTATC=2 for first adult and EMPSTATC=2 for second adult (if one exists)
- 1 EMPSTATC=1 for either first or second adult (if one exists).

 --

ECSTATBU

Purpose : To derive an HBAI consistent variable for economic status for use in the publication.
 Created : 8 February 1996
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 9
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : VE - 21 May 1996 - Amended for initial Version 32 changes and to change full-time hours cut-off to greater than or equal to 30 (not greater than 29).
 : SG - 25 June 1997 - Amended for V33 - just text change relating to training schemes
 : SCG - 5 January 1998 - Looking for work changes
 : SCG - 23 January 1998 - Full time is over 31 hours, training is counted as work. Don't repeat coding for EMPSTATC.
 : EP - 11 August 1998 - V34 updates, name changes for Hours Worked variables
 : SB - 21 October 1999 - Security completed, no other changes for V35
 Issued : 21 October 1999

1 Definition

ECSTATBU is an extended version (ie including disability category) of the current HBAI economic status breakdown. The additional category for sick and disabled is calculated on the basis of answers to questions HEALTH, HPROB, RSTRCT, LAREG and JCREG.

The variable uses the existing coding for ECOTYPBU, but includes an initial test for disability which is then overwritten with the existing codes based on EMPSTATC tests.

ECOTYPBU is coded by first creating an economic status variable for each adult and then from this deriving a variable for each benefit unit. As far as possible the variable has been coded to be consistent with the FES economic status variable detailed in FES Appendix 46.

The individual economic status variable EMPSTATC is coded **as shown in the DV specification for ECOTYPBU**. Note that EMPSTATC should also be added to the flat file:

ECSTATBU is coded as:

 --

- 1 Self employed (benefit units where at least one adult usually works self-employed for 31 or more hours a week)
- 2 Single or couple, all in full time work *as an employee* (31 or more hours a week)
- 3 Couple, one in full time work, one in part-time work (*including part-time self employed*)
- 4 Couple, one in full time work (*as an employee*), one not working
- 5 Single or couple, one or more in part-time work (*self-employed or employee*)
- 6 Single or couple, head or spouse aged 60 or over
- 7 Single or couple, head or spouse unemployed
- 8 Single or couple, head or spouse sick or disabled (under pension age)
- 9 Others

Note: The pensioner category **is not** consistent with the HBAI family status variable or HHCOMP, using a 60+ cut off for pensioners. Moreover, working pensioners will be classified as self employed/working full or part time **before** they are classified as pensioners. The disability category may also be different from HHCOMP (although the selection criteria are the same) because of the hierarchical classification.

The full-time/part-time split for ECSTATBU is also different from the EMPSTATB classification which uses the 16 hour rule. Instead it uses the definitions as EMPSTATC, part time is defined as employment or self employment for less than 31hrs. If EVEROT = No, then XHRS = TOTUS1 else XHRS = DVTOTHRU. This is different from EMPSTATB (although in practice, figures will be very similar) which uses the TOTHOOURS DV to distinguish full/part time work. TOTHOOURS looks at hours worked from **all** jobs, ie including any hours worked as employees at second and third jobs with self-employed.

2 FRS Specification

EMPSTATC is coded in the same way as ECOTYPBU

ECSTATBU then has an additional check for disability which is included as the first test.

For each Benefit Unit:

Tests are applied sequentially with the coding determined by the last test to be satisfied:

Code

- 9 All cases first coded to other and then overwritten when one of the conditions apply

--

- 8 If (((SEX=1 and AGE<65) or (SEX=2 and AGE<60)) and ((HEALTH=1 and HPROB=1) or (RSTRCT=1 or RSTRCT=2) or (LAREG=1) or (JCREG=1)) for either the first adult or the second adult (if there is one) in the benefit unit
- 7 EMPSTATC=4 for either the first or the second adult (if there is one)
- 6 AGE>59 for either the first or second adult (if one exists).
- 5 EMPSTATC=3 for either first or second adult (if one exists).
- 4 EMPSTATC=2 for first adult and EMPSTATC=4 or 5 for second adult, or vice versa.
- 3 EMPSTATC=2 for first adult and EMPSTATC=3 for second adult, or vice versa.
- 2 EMPSTATC=2 for first adult and EMPSTATC=2 for second adult (if one exists)
- 1 EMPSTATC=1 for either first or second adult (if one exists).

 --

EMPSTATB

Purpose : To indicate the employment status of each adult.
 Created : 8 January 1993
 Database Table : ADULT
 Minimum Value : 1
 Maximum Value : 14
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Issue date : 21 October 1999
 Amendments : VC - 13 January 1993
 : VC - 20 April 1993 - includes additional categories which were incorrectly :
 : thrown out as not derivable.
 : VC - 23 April 1993 To expand definition to show meaning of :
 : questions/database variables.
 : VC - 12 May 1993 To add a category to include students and people taking :
 : industrial action.
 : VC 14 June 1993 To expand categories as more adults unclassified than :
 : necessary.
 : JS 19 January 1996 to do a belt and braces job on students
 : JS 8 March 1996 - to amend codes for reasons for absence from work
 : VE - 24 May 1996 - No initial updates needed for V32
 : VE - 25 February 1997 - To update for V33 - LOOK4 and LKYT4 removed
 : VE - 6 March 1997 - To amend for V33 - categories of ABSWHY, NOLOOK
 : and NOWANT changed and an extra value added for unpaid family workers
 : SG - 20 January 1998 - Work Trial (TRAIN=3) should be in Training category
 : SG - 11 February 1998 - Cover a couple of unusual cases
 : EP - 11 August 1998 - No initial V34 update needed
 : SB - 21 October 1999 - Security completed, added in TRAIN=7
 : AW - 24 February 2000 - change to injlong - new category =3 for V35

1 Definition

This variable is coded as

- 1 Self-employed
- 2 Full-time employee at work
- 3 Part-time employee at work
- 4 Full-time employee temporarily not working (sick for 28 weeks or less)
- 5 Part-time employee temporarily not working (sick for 28 weeks or less)
- 6 Industrial action
- 7 Unemployed
- 8 Work-related government training programme
- 9 Retired - unoccupied minimum NI age

 --

- 10 Unoccupied - under minimum NI age
- 11 Sick - temporarily sick for 28 weeks or less
- 12 Sick - long-term sick/disabled for more than 28 weeks
- 13 Students and adults in non-advanced full-time education
- 14 Unpaid family workers
- 1 Not applicable to this case
- 2 Unable to derive

The employment status variable is derived in the main from the ADULT table using a variety of variables to indicate the current employee status of all adults. Part-time and full-time work have been separated using the Income Support definition of full-time remunerative work - 16 hours a week or more.

NB - People who are at home looking after children are included in the unoccupied category (10).

Note that if individual has skipped WORKING question then we classify as unoccupied now - however from V34 should not happen as imputation checks should find such individuals.

2 FRS Specification

For each adult

Code Condition

- 1 From ADULT table, if WORKING = 1 and EMPSTAT = 2 or
 If WORKING = 2, JOBAWAY = 1 and EMPSTAT = 2.
 (Includes those not working within the last 7 days but do have a job to return to.)
- 2 From ADULT table, if EMPSTAT = 1 and TOTHOURS is greater than or equal to 16 and
 WORKING = 1 or JOBAWAY = 1 and
 TDAYWRK = 1 or {TDAYWRK = 2 or 3 and ABSWK = 2 or ABSWHY = 1 or 3}
 (Includes those not working within last 7 days but are working today.)
- 3 From ADULT table, if EMPSTAT = 1 and TOTHOURS is less than 16 and
 WORKING = 1 or JOBAWAY = 1 and
 TDAYWRK = 1 or {TDAYWRK = 2 or 3 and ABSWK = 2 or ABSWHY = 1 or 3}
 (Includes those not working within last 7 days but are working today.)
- 4 From ADULT table, if EMPSTAT = 1 and TOTHOURS is greater than or equal to 16 and
 WORKING = 1 or JOBAWAY = 1 and
 INJLONG = 1 or
 TDAYWRK = 2 or 3 and {ABSWK = 2 or ABSWHY = 2, 5, 6, 7, 8 or 9}
- 5 From ADULT table, if EMPSTAT = 1 and TOTHOURS is less than 16 and
 WORKING = 1 or JOBAWAY = 1 and
 INJLONG = 1 or
 TDAYWRK = 2 or 3 and {ABSWK = 2 or ABSWHY = 2, 5, 6, 7, 8 or 9}
- 6 From ADULT table, if EMPSTAT = 1 and

 --

- WORKING = 1 or JOBAWAY = 1 and
 TDAYWRK = 2 or 3 and ABSWHY = 4.
- 7 Code 7 for the following - for all adults who are below pension age
 From ADULT table, where AGE < 65 and SEX = 1 and where AGE < 60 and SEX = 2
 If WORKING = 2 and JOBAWAY = 3 and LOOKWK = 1, 2 or 3
 If WORKING = 2, JOBAWAY = 2 and LOOK = 1
 If WORKING = 2, JOBAWAY = 2 and WAIT = 1
 If WORKING=2, JOBAWAY=2, LIKEWK=2, NOWANT=1
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 1 and NOLOOK = 1
- 8 From ADULT table, if TRAIN = 1,2, 3,4, 5, 6 or 7
- 9 Code 9 for the following
 From ADULT table, where AGE >= 65 and SEX = 1 and where AGE >= 60 and SEX = 2
- 10 Code 10 for the following - for all adults who are below pension age
 From ADULT table, as above
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 1 and NOLOOK = 3 or 4 or 7 or 8 or 9
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 2 and NOWANT = 3 or 4 or 7 or 8 or 9
 If WORKING = 2, JOBAWAY = 2 and NOLK2 = 1. (Includes those not working as looking after children)
 If WORKING = -1
- 11 Code 11 for the following - for all adults below pension age
 From ADULT table,
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 1, NOLOOK = 5 or
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 2, NOWANT = 5 or
 If INJLONG = 1.
- 12 As above from ADULT table,
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 1, NOLOOK = 6 or
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 2, NOWANT = 6 or
 If INJLONG = 2 or 3
- 13 From ADULT table, if FTED = 1 and TYPEED = 4,5 or 7 or
 If TEA = 96 and TYPEED = 4 or 5 or 7
 If WORKING = 2 and JOBAWAY = 2 and NOWANT = 2 or NOLOOK = 2
- 14 From ADULT table, if UNPAID1=1 or UNPAID2=1
- 1 Not applicable to this case

Key to above coding

Working Any paid work in last 7 days? 1 = yes, 2 = no.

 --

Jobaway	If not doing paid work, do you have a job that away from last 7 days. 1 = yes, 2 = no, 3 = waiting take up new job.
Empstat	1 = employed 2 = self-employed (including Enterprise Allowance)
Tdaywrk	Are you going to work today. 1 = yes, 2 = no although normal working day, 3 = no not normal working day.
Tothours	Number of hours worked by an employee - a derived variable in itself.
Abswk	Have you been away from work for more than 3 working days. 1 = yes, 2 = no.
Abswhy	Reason for absence. 1 = pattern of shifts, 2 = illness/accident, 3 = holiday, 4 = strike, 5 = Laid off, 6 = maternity leave, 7 = compassionate leave, 8=other.
Injlong	How long been unable to work due to illness/injury. 1 = 6 months or less, 2 = more than 6 months.
Wait	Were you waiting to take up a job already obtained. 1 = yes, 2 = no
Likewk	Would you like to have a regular paid job. 1 = yes, 2 = no.
Nolook	Why not look. 1 = waiting results job application, 2 = student, 3 = looking after family/home, 4 = temporarily sick, 5 = long-term sick, 6 = believes no jobs available, 7 = retired, 8 = any other reason.
Nowant	Why not want. 1 = waiting results of job application, 2 = student, 3 = looking after family or home, 4 = temporarily sick/injured, 5 = long-term sick or disabled, 6 = doesn't need employment, 7 = retired from paid work, 8 = any other reason.
fted	Still in full-time education (only asked of 16 to 18 year olds). 1 = yes, 2 = no.
TEA	At what age did person leave full-time education (only asked if aged over 18). If answer is 96 the person is still receiving full-time education.
Typeed	What type of school or college does person attend. 3 = state run special school, 4 = secondary school, 5 = non-advanced further education/6th form/tertiary/further education college. (1 and 2 are nursery and junior schools)

3 Results

Tabulation is required to show the number of adults falling into each category.

EMPSTATI

Purpose : To indicate a person's employment status using the ILO definition.
 Created : VC - 17 May 1993
 Database Table : ADULT
 Minimum Value : 1
 Maximum Value : 6
 Units : Integer
 Validations :
 Related Variables : EMPSTATB (Indicates employment status by ADULT)
 Children :
 Parents :
 Core variable/user : FRS (general)
 Issue Date : 17 April 2003
 Amendments : VC 9 June 1993 To put people who are on holiday from their normal place :
 of work into category 1
 : VE - 24 May 1996 - Initial updates for V32 - amendments of
 TRAIN for new category definitions
 : SG - 25 June 1997 - Various updates for V33 - absence from work
 reasons, work trial scheme, look for work question
 : SG - 18 March 1998 - Update to full ILO definition
 : EP - 17 December 1998 - Correct values of TRAIN variable
 : SB/CWJ - 22 October 1999 - Change of definitions to more closely reflect ILO
 economic status variable, these are back dated to V34
 : SB - 5 November 1999 - Security completed, no other changes for V35

1 Definition

This variable is coded as

- 1 Full-time employee**
- 2 Part-time employee**
- 3 Full-time self-employed**
- 4 Part-time self-employed**
- 5 ILO unemployed**
- 6 Retired**
- 7 Student**
- 8 Looking after family/home**
- 9 Permanently sick/disabled**
- 10 Temporarily sick/disabled**
- 11 Other inactive**

Using broad ILO definitions, categories 1-4 are in employment, category 5 is ILO unemployed and categories 6-11 are inactive.

- 1 Not applicable to this case
- 2 Unable to derive due to missing values

EMPSTAT1 is derived from several variables in the ADULT table of the FRS database as well as the FTPT variable in the JOB table.

An employee is derived from the variables WORKING = 1 (has paid work in last 7 days) or where WORKING = 2 (no work in last 7 days) but JOBAWAY = 1 (has a job to return to). Once it has been established that the person is working, EMPSTAT = 1 then indicates that the person is an employee.

Self-employed people are derived where WORKING = 1 or WORKING = 2 and JOBAWAY = 1 (as above) and EMPSTAT = 2 (self-employed) or where they have done an odd job and not previously classified as employed.

Full or part-time work is on the basis of self assessment for the main job (FTPT). Those in unpaid work for a business they own or for a family business are classified as self-employed or employed respectively.

Those on Government Training schemes are classified as full time employees. The variable TRAIN indicates whether or not a person is on a work-related government training scheme and codes 1 to 7 indicate a government scheme (1 - Youth Training, 2 - Training for Work, 3 - Work Trial, 4 - Project work, 5 - Career Development Loans/Youth Credits, 6 - New Deal for 18-24 year olds, 7 - Other training scheme).

The ILO definition of unemployment is available for and actively seeking work. If the individual is in receipt of UB then counted as unemployed. Or where WORKING = 2 (not working) and JOBAWAY = 2 (no job to return to) we would check to see if the person is available for and looking for work. LOOK = 1 indicates that the person is looking for work or a government scheme in the last 4 weeks, WAIT = 1 indicates that he/she is waiting to start a new job, LIKEWK = 1 and NOLOOK = 1 indicates that the person is not looking for employment as he/she is waiting to start a new job. Start=1 denotes available to start in next 2 weeks.

Further classifications are based on stated reasons for not looking for work.

Individuals over state retirement age and are not currently looking for work or who state they consider themselves retired are classified as retired. Students are classified as such unless they have worked in the reference week.

The sick and disabled are those who have stated they do not want work or are not looking for work because of sickness/disability.

The final category of "other inactive" acts as a catchall.

2 FRS Specification

For each adult get all variables from ADULT table and also from main job in JOB table

Code Condition

1 If (WORKING = 1 or JOBAWAY=1) and EMPSTAT = 1 and FTPT=1 or
 If TRAIN = 1, 2, 3, 4, 5, 6 or 7 or
 If WORKING = 2 and JOBAWAY = 1 and EMPSTAT = 1 and FTPT = 1 or
 If WORKING = 2 and JOBAWAY = 2 and UNPAID2 = 1

2 If (WORKING = 1 or JOBAWAY=1) and EMPSTAT = 1 and FTPT = 2 or

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

- If JOBAWAY = 2 and EMPSTAT = 1 and FTPT = 2
- 3 If WORKING = 1 and EMPSTAT = 2 and FTPT = 1 or
If JOBAWAY = 1 and EMPSTAT = 2 and FTPT = 1 or
If JOBAWAY = 2 and UNPAID1 = 1
- 4 If WORKING = 1 and EMPSTAT = 2 and FTPT=2 or
If WORKING = 2 and JOBAWAY = 1 and EMPSTAT = 2 and FTPT=2 or
If WORKING = 2 and ODDJOB = 1
- 5 If WORKING = 2 and JOBAWAY = 3 or
If WORKING = 2 and JOBAWAY = 2 and LOOK = 1 and START = 1 or
If WORKING = 2 and JOBAWAY = 2 and WAIT = 1 or
If SEX = 1 and AGE > 64 and WORKING = 2 and JOBAWAY = 2 and LOOK = 1 or
If SEX = 2 and AGE > 59 and WORKING = 2 and JOBAWAY = 2 and LOOK = 1
- 6 If AGE >= 65 and SEX = 1 and WORKING = 2 and JOBAWAY = 2 and LOOK = 2 or
If AGE >= 60 and SEX = 2 and WORKING = 2 and JOBAWAY = 2 and LOOK = 2 or
If WORKING = 2 and JOBAWAY=2 and LIKEWK=2 and NOWANT=8
- 7 If JOBAWAY=2 and LIKEWK=1 and NOLOOK=2 or
If JOBAWAY=2 and LIKEWK=2 and NOWANT=2
- 8 If JOBAWAY=2 and LIKEWK=1 and (NOLOOK = 3 or NOWANT = 3)
- 9 If WORKING = 2 and JOBAWAY = 2 and LIKEWK = 1 and NOLOOK = 6 or
If WORKING = 2 and JOBAWAY = 2 and LIKEWK = 2 and NOWANT = 6
- 10 If WORKING = 2 and JOBAWAY = 2 and LIKEWK = 1 and NOLOOK = 5 or
If WORKING = 2 and JOBAWAY = 2 and LIKEWK = 2 and NOWANT = 5
- 11 Anyone else not previously coded.
- 1 Not applicable (should not be relevant to EMPSTATI)
- 2 Unable to derive as any of the above variables are missing

 --

EQAHCBU

Purpose : Equivalence scale after housing costs for benefit unit which will be used to calculate the equivalised income after housing costs for a benefit unit.
 Created : VC - 24 September 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents : SPCALC
 Core variable/user : HBAI
 Amendments : VC - 11 October 1993 To identify equivalence scales used.
 : VC - 11 February 1994 Amendments to reflect version 30 changes.
 : VE - 4 June 1996 - No initial amendments needed for V32 update
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : SG - 25 June 1997 - Amendments for V33
 : SG - 4 February 1998 - MS changes
 : EP - 11 August 1998 - No initial V34 update needed
 : JC - 17 September 1999 - Security completed, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable is coded as

EQAHCBU This is a scale calculated for the benefit unit depending on the number of adults and children using the McClement's scale in the same way as FES. **The constants are held in p:\frs\shared\frs34\metadata\Dvmeta34.xls**. The relevant values from the table are accumulated to provide the benefit unit scale.

1st adult (head of household)		HDBU1A
spouse of head		SPBU1A
heads of subsequent units		HDBUXA
spouse of any subsequent units		SPBUXA
each dependent aged	0 - 1	D01A
	2 - 4	D24A
	5 - 7	D57A
	8 - 10	D810A
	11 - 12	D1112A

 --

13 - 15 D1315A
 16 + D16+A

-2 Unable to derive due to missing values

The resultant scale will be used to calculate a further income variable for the benefit unit. The individual groups for the above list are identified using the benefit unit number (BENUNIT) and the variables HEAD and SPOUSE which are calculated using the SPCALC sub-programme plus the age of every child (AGE from CHILD record).

The head of the benefit unit is defined as the first adult in that benefit unit and could, therefore, be either male or female depending on the order the interviewer used during the interview.

NB - There shouldn't be any not applicable cases as all households have at least one adult.

2 FRS Specification

For each BENUNIT record, set EQAHCBU to zero

<u>Code</u>	<u>Condition</u>	
EQAHCBU	Process each ADULT record and accumulate the following for each adult in the benefit unit.	
	If BENUNIT = 1 and HEAD = 1 (head of household) add	HDBU1A
	If BENUNIT = 1 and SPOUSE = 1 (spouse/partner) add	SPBU1A
	If BENUNIT >= 2 and HEAD = 1 add	HDBUXA
	If BENUNIT >= 2 and SPOUSE = 1 add	SPBUXA
	Process each CHILD record and accumulate the following for each child in the benefit unit.	
	If AGE <= 1 add	D01A
	If AGE >= 2 and <= 4 add	D24A
	If AGE >= 5 and <= 7 add	D57A
	If AGE >= 8 and <= 10 add	D810A
If AGE >= 11 and <= 12 add	D1112A	
If AGE >= 13 and <= 15 add	D1315A	
If AGE >= 16 add	D16+A	

-2 If any of above values are missing

3 Results

None required as will be used as a component of other variables.

 --

EQBHCBU

Purpose : Equivalence scale before housing costs for benefit unit which will be used to calculate the equivalised income before housing costs for a benefit unit.
 Created : VC - 6 October 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents : SPCALC
 Core variable/user : HBAI
 Amendments : VC - 11 October 1993 To identify the equivalence scale used.
 : VC - 11 February 1994 Amendments to reflect verion 30 changes
 : VE - 4 June 1996 - No initial amendments needed for V32
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : SG - 25 June 1997 - Amended for V33 updates - new constants file
 : SG - 4 February 1998 - MS changes
 : EP - 11 August 1998 - No initial V34 update needed
 : JC - 17 September 1999 - Security completed, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable is coded as

EQBHCBU This is a scale calculated for the benefit unit depending on the number of adults and children using the McClements table of scales in the same way as FES. **The constants are held in p:/frs/shared/frs34/metadata/Dvmeta34.xls.** The relevant values from the table are accumulated to provide the benefit unit scale.

1st adult (head of household)		HDBU1B
spouse of head		SPBU1B
heads of subsequent units		HDBUXB
spouse of any subsequent units		SPBUXB
each dependent aged	0 - 1	D01B
	2 - 4	D24B
	5 - 7	D57B
	8 - 10	D810B
	11 - 12	D1112B

 --

13 - 15	D1315B
16 +	D16+B

-2 Unable to derive due to missing values

The resultant scale will be used to calculate a further income variable for the benefit unit. The individual groups for the above list are identified using the benefit unit number (BENUNIT) and the variables HEAD and SPOUSE which are calculated using the SPCALC sub-programme plus the age of every child (AGE from CHILD record).

The head of the benefit unit is defined as the first adult in that benefit unit and could, therefore, be either male or female depending on the order the interviewer used during the interview.

NB - There shouldn't be any not applicable cases as all households have at least one adult.

2 FRS Specification

For each BENUNIT record, set EQBHCBU to zero

<u>Code</u>	<u>Condition</u>
EQBHCBU	Process each ADULT record and accumulate the following for each adult in the benefit unit.
	If BENUNIT = 1 and HEAD = 1 (head of household) add HDBU1B
	If BENUNIT = 1 and SPOUSE = 1 (spouse/partner) add SPBU1B
	If BENUNIT >= 2 and HEAD = 1 add HDBUXB
	If BENUNIT >= 2 and SPOUSE = 1 add SPBUXB
	Process each CHILD record and accumulate the following for each child in the benefit unit.
	If AGE <= 1 add D01B
	If AGE >= 2 and <= 4 add D24B
	If AGE >= 5 and <= 7 add D57B
	If AGE >= 8 and <= 10 add D810B
	If AGE >= 11 and <= 12 add D1112B
	If AGE >= 13 and <= 15 add D1315B
	If AGE >= 16 add D16+B

-2 If any of above values are missing

3 Results

None required as will be used as a component of other variables.

 --

EQUIVAHC

Purpose : Household after housing costs equivalence scale which will be used to calculate the household equivalised income after housing costs. :

Created : VC - 23 September 1993

Database Table : HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables :

 Children : EQVAHCHH

 Parents :

Core variable/user : HBAI

Amendments : VC - 11 October 1993 To identify the equivalence scale used.
 : VE - 22 May 1996 - No initial amendments needed for V32
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 23 September 1996 - To clarify spouse of head as being in benunit 1
 : SG - 8 January 1998 - V33 updates - constants location
 : SG - 4 February 1998 - MS changes
 : EP - 11 August 1998 - No initial V34 update needed
 : JC - 9 September 1999 - Security completed, no other changes for V35

Issued : 17 April 2003

1 Definition

This variable is coded as

EQUIVAHC This is a scale calculated for the household depending on the number of adults and children using the McClements table of scales in the same way as FES. **The constants are held in p:\frs\shared\frs34\metadata\dvmeta34.xls.** The relevant values from the table are accumulated to provide the household scale.

1st adult (head of household)		AD1A
spouse of head		AD2A
or other 2nd adult		AD2A
3rd adult		AD3A
or any subsequent adult		SUBSADA
each dependent aged	0 - 1	DEP01A
	2 - 4	DEP24A
	5 - 7	DEP57A
	8 - 10	DEP810A

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

11 - 12	DEP1112A
13 - 15	DEP1315A
16 +	DEP16+A

-2 Unable to derive due to missing values

The resultant scale will be used to calculate a further income variable for the household. The individual groups for the above list are identified using the person number (PERSON) and the marital status of each adult (MS) and the age of every child (AGE from CHILD record).

NB - There shouldn't be any not applicable cases as all households have at least one adult.

2 FRS Specification

For each HOUSEHOLD, set EQUIVAHC to zero

<u>Code</u>	<u>Condition</u>	
EQUIVAHC	Process each ADULT record and accumulate the following for each adult in the household	
	If PERSON = 1 (head of household) add	AD1A
	If PERSON = 2 and BENUNIT =1 and ((MS = 2 and SPOUT^=1) OR COHAB=1) (spouse/partner) add	AD2A
	Else if PERSON = 2 add	AD2A
	If PERSON = 3 add	AD3A
	If PERSON > 3 add for each subsequent adult	SUBSADA
	Process each CHILD record and accumulate the following for each child in the household	
	If AGE <= 1 add	DEP01A
	If AGE >= 2 and <= 4 add	DEP24A
	If AGE >= 5 and <= 7 add	DEP57A
	If AGE >= 8 and <= 10 add	DEP810A
	If AGE >= 11 and <= 12 add	DEP1112A
	If AGE >= 13 and <= 15 add	DEP1315A
	If AGE >= 16 add	DEP16+A

-2 If any of above values are missing

3 Results

None required as will be used as a component of other variables.

 --

EQUIVBHC

Purpose : Household before housing costs equivalence scale which will be used to calculate the household equivalised income before housing costs. :

Created : VC - 23 September 1993

Database Table : HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables :

 Children : EQVBHCHH

 Parents :

Core variable/user : HBAI

Amendments : VC - 11 October 1993 To identify the equivalence scale used.
 : VE - 22 May 1996 - No initial amendments needed for V32
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 23 September 1996 - To clarify spouse of head as being PERSON=2 and BENUNIT=1
 : SG - 8 January 1998 - V33 updates - constants location
 : SG - 4 February 1998 - MS changes
 : EP - 11 August 1998 - No initial V34 update needed
 : JC - 9 September 1999 - Security completed, no other changes for V35

Issued : 17 April 2003

1 Definition

This variable is coded as

EQUIVBHC This is a scale calculated for the household depending on the number of adults and children using the McClements table of scales in the same way as FES. **The constants are held in p:\frs\shared\frs34\metadata\dvmeta34.xls.** The relevant values from the table are accumulated to provide the household scale.

1st adult (head of household)		AD1B
spouse of head		SPHDB
or other 2nd adult		AD2B
3rd adult		AD3B
or any subsequent adult		SUBSADB
each dependent aged	0 - 1	DEP01B
	2 - 4	DEP24B
	5 - 7	DEP57B

 --

FAMTHBAI

Purpose : This is the family type used for HBAI purposes for each benefit unit.
 Created : VC - 9 September 1993
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 6
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : to fit in with HBAI definitions: FAMTHBAI =1 when head of benefit unit over pensionable age; state pension definition
 : JS - 13 March 1996 - to tighten up definitions and amend spec in line with ASD6A changes
 : VE - 4 June 1996 - No initial amendments needed for V32
 : SCG - 7 January 1998 - V33 updates (MS changes)
 : EP - 11 August 1998 - V34 update, DEPCHLDB now a DV
 : JC - 9 September 1999 - Security completed, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable is coded as

FAMTHBAI	This is the definition of family type used by HBAI.
1	Pensioner couple
2	Pensioner single
3	Couple with children
4	Couple without children
5	Lone parent
6	Single without children.

FAMTHBAI is now derived independently from FAMTYPE, but needs to have checks on whether a second adult is present for couples to ensure correct definitions are set up (see FAMTYPE for further information).

According to the HBAI publication, pensioner/non-pensioner singles/couples are where they are headed by someone over/under state pension age. However, for the FES, the head of the benefit unit is always the man whereas for the FRS female heads are possible.

This causes problems for cases where MS=2 and SPOUT=1 (married, spouse not in household) and the head (partner who is in the household) is female as cases would be set to unable to derive. The female is still taken as the head in these cases. NOTE: HBAI delete these cases from their file.

 --

2 FRS Specification

Codes are hierarchical, ie if have children but head is over pension age, BUs fall in to code 1 and not 3, or code 2 and not 5 (similarly not codes 4 and 6 if no children).

ie the variable is coded sequentially from top down (1 to 6) and is set according to the first condition fulfilled

<u>Code</u>	<u>Condition</u>
1	If ((MS=1,3,4,5 or (MS=2 and SPOUT [^] =1)) and (second adult in the benefit unit exists) and man in the benefit unit is over 64 [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=2 and SEX=1 and AGE>=65)] or MS=2 and SPOUT=1 and (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60) man or woman over pensionable age where MS = married, spouse not in household
2	If ((MS=1,3,4,5 or (MS=2 and SPOUT [^] =1)) and (only one adult in the benefit unit) and person is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60)]
3	If ((MS=1,3,4,5 or (MS=2 and SPOUT [^] =1)) and (second adult in the benefit unit exists) and depchildb DEPCHLDB > 0 or MS=2 and SPOUT=1 and depchildb DEPCHLDB >0
4	If ((MS=1,3,4,5 or (MS=2 and SPOUT [^] =1)) and (second adult in the benefit unit exists) and depchildb DEPCHLDB =0 or MS=2 and SPOUT=1 and depchildb DEPCHLDB =0
5	If MS=1,3,4,5 and only one adult in the benefit unit and depchildb DEPCHLDB >0
6	If MS=1,3,4,5 and only one adult in the benefit unit and depchildb DEPCHLDB =0

3 Results

Tabulation will be required to show the number and percentage of benefit units falling into each category.

 --

FAMTYPBS

Purpose : This is the new (V33) family type DV used for publication purposes for each benefit unit. It is based on FAMTYPBU with the addition that single pensioners and single without children are split by sex.

Created : SG - 27 April 1998

Database Table : BENUNIT

Minimum Value : 1

Maximum Value : 6

Units : Integer

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : FRS (publication)

Amendments : EP - 11 August 1998 - V34 update, DEPCHLDB now a DV
 : JC - 20 September 1999 - Security completed, no other changes for V35

Issued : 17 April 2003

1 Definition

This variable is coded as

FAMTYPBS This is the definition of family type used for the publication.

- 1 Pensioner couple
- 2 Male pensioner single
- 3 Female pensioner single
- 4 Couple with children
- 5 Couple without children
- 6 Lone parent
- 7 Male single without children
- 8 Female single without children.

FAMTYPBS is now derived independently from FAMTYPE, but needs to have checks on whether a second adult is present for couples to ensure correct definitions are set up (see FAMTYPE for further information).

According to the HBAI publication, pensioner/non-pensioner singles/couples are where they are headed by a someone over/under state pension age. However, for the FES, the head of the benefit unit is always the man whereas for the FRS female heads are possible. Since for the publication, tables have been produced which show the age of the head of benefit unit, for consistency, pensioner FAMTYPBU cases are based on the age of the head, regardless of the sex.

This causes problems for cases where SPOUT=1 and MS=2 (married, spouse not in household) and the head (partner who is in the household) is female as cases would be set to unable to derive. The female is still taken as the head in these cases. NOTE: HBAI delete these cases from their file.

 --

2 FRS Specification

Codes are hierarchical, ie if have children but head is over pension age, BUs fall in to code 1 and not 3, or code 2 and not 5 (similarly not codes 4 and 6 if no children).

ie the variable is coded sequentially from top down (1 to 6) and is set according to the first condition fulfilled

<u>Code</u>	<u>Condition</u>
1	If MS=1,2,3,4,5 and SPOUT^=1 and (second adult in the benefit unit exists) and the head of the benefit unit is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60)] or SPOUT=1 and MS=2 and (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60) man or woman over pensionable age where SPOUT =married, spouse not in household
2	If MS=1,2,3,4,5 and SPOUT^=1 and (only one adult in the benefit unit) and person is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65)]
3	If MS=1,2,3,4,5 and SPOUT^=1 and (only one adult in the benefit unit) and person is over pension age [ie (UPERSON=1 and SEX=2 and AGE>=60)]
4	If MS= 1,2,3,4,5 and SPOUT^=1 and (second adult in the benefit unit exists) and depchildb DEPCHLDB > 0 or SPOUT=1 and MS=2 and depchildb DEPCHLDB >0
5	If MS=1,2,3,4,5 and SPOUT^=1 and (second adult in the benefit unit exists) and depchildb DEPCHLDB =0 or SPOUT=1 and MS=2 and depchildb DEPCHLDB =0
6	If MS=1,2,3,4,5 and SPOUT^=1 and only one adult in the benefit unit and depchildb DEPCHLDB >0
7	If MS=1,2,3,4,5 and SPOUT^=1 and only one adult in the benefit unit and sex=1 and depchildb DEPCHLDB =0
8	If MS=1,2,3,4,5 and SPOUT^=1 and only one adult in the benefit unit and sex=2 and depchildb DEPCHLDB =0

3 Results

Tabulation will be required to show the number and percentage of benefit units falling into each category.

 --

FAMTYPBU

Purpose : This is the family type used for publication purposes for each benefit unit. It is consistent with the HBAI variable FAMTHBAI except that pensioner benefit units are defined on the basis of the head of the benefit unit, be it male or female

Created : VC - 9 September 1993

Database Table : BENUNIT

Minimum Value : 1

Maximum Value : 6

Units : Integer

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : FRS (publication)

Amendments : to fit in with HBAI definitions: FAMTYPBU =1 when head of benefit unit over pensionable age; state pension definition
 : JS - 13 March 1996 - to tighten up definitions and amend spec in line with ASD6A changes
 : VE - 4 June 1996 - No initial amendments needed for V32
 : VE - 20 June 1996 - To change lead user to ASD4A in response to request from HBAI
 : SG - 3 December 1997 - V33 updates, married living outside HH now shown by SPOUT
 : EP - 11 August 1998 - V34 update, DEPCHLDB now a DV
 : JC - 20 September 1999 - Security completed, no other changes for V35

Issued : 17 April 2003

1 Definition

This variable is coded as

FAMTYPBU This is the definition of family type used for the publication.

- 1 Pensioner couple
- 2 Pensioner single
- 3 Couple with children
- 4 Couple without children
- 5 Lone parent
- 6 Single without children.

FAMTYPBU is now derived independently from FAMTYPE, but needs to have checks on whether a second adult is present for couples to ensure correct definitions are set up (see FAMTYPE for further information).

 --

According to the HBAI publication, pensioner/non-pensioner singles/couples are where they are headed by someone over/under state pension age. However, for the FES, the head of the benefit unit is always the man whereas for the FRS female heads are possible. Since for the publication, tables have been produced which show the age of the head of benefit unit, for consistency, pensioner FAMTYPBU cases are based on the age of the head, regardless of the sex.

This causes problems for cases where **SPOUT=1** and MS=2 (married, spouse not in household) and the head (partner who is in the household) is female as cases would be set to unable to derive. The female is still taken as the head in these cases. NOTE: HBAI delete these cases from their file.

2 FRS Specification

Codes are hierarchical, ie if have children but head is over pension age, BUs fall in to code 1 and not 3, or code 2 and not 5 (similarly not codes 4 and 6 if no children).

ie the variable is coded sequentially from top down (1 to 6) and is set according to the first condition fulfilled

<u>Code</u>	<u>Condition</u>
1	If MS=1,2,3,4,5,6 and SPOUT^=1 and (second adult in the benefit unit exists) and the head of the benefit unit is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60)] or SPOUT=1 and MS=2 and (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60) man or woman over pensionable age where SPOUT =married, spouse not in household
2	If MS=1,2,3,4,5,6 and SPOUT^=1 and (only one adult in the benefit unit) and person is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60)]
3	If MS= 1,2,3,4,5,6 and SPOUT^=1 and (second adult in the benefit unit exists) and depchildb DEPCHLDB > 0 or SPOUT=1 and MS=2 and depchildb DEPCHLDB>0
4	If MS=1,2,3,4,5,6 and SPOUT^=1 and (second adult in the benefit unit exists) and depchildb DEPCHLDB =0 or SPOUT=1 and MS=2 and depchildb DEPCHLDB=0
5	If MS=1,2,3,4,5,6 and SPOUT^=1 and only one adult in the benefit unit and depchildb DEPCHLDB >0
6	If MS=1,2,3,4,5,6 and SPOUT^=1 and only one adult in the benefit unit and depchildb DEPCHLDB =0

3 Results

--

Tabulation will be required to show the number and percentage of benefit units falling into each category.

 --

FAMTYPE

Purpose : Family Type Indicator for each Benefit Unit
 Created : AJG 1 December 1992
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 8
 Units : Integer
 Validations :
 Related D Variables :
 Children :
 Parents :
 Core variable/user : PSM
 Issue date : 17 April 2003
 Amendments : VC - 22 January 1993
 : VC - 23 March 1993 To change the variables used in the coding to make the
 : VC - 7 May 1993 To use income-related benefits definition of pension age ie
 : VC - 11 February 1994 Amendments to reflect version 30 changes.
 : JS - 13 March 1996, to tighten up definition
 : VE - 4 June 1996 - No initial amendments needed for V32
 : SG - 7 January 1998 - V33 updates, MS and SPOUT changes
 : SG - 11 February 1998 - Fix MS checks
 : EP - 11 August 1998 - V34 update, change COHAB to COHABIT, and
 : EP - 23 February 1999 - V34 - change condition so that single people who
 : JC - 9 September 1999 - Security completed, no other changes for V35

1 Definition

This variable is coded as

1	Under pension age	with dependants	couple
2			single
3	Under pension age	without dependants	couple
4			single
5	Over pension age	with dependants	couple
6			single
7	Over pension age	without dependants	couple
8			single
-2	Unable to derive		

For the purpose of this variable:

Pension age is taken to be 60 and over (ie the income-related benefits pension age).

 --

Couples and single people are identified by the variable marital status in the ADULT table. Couples are to include MS = 2 ~~and SPOUT=1~~ (married) or MS=1 or 2 or 3 or 4 or 5 and **COHAB COHABIT=1 (cohabiting)** and, MS = 2 and SPOUT=1 (married spouse not in household). Single people will be the rest - MS = 1 (single, never married), MS = 5 (widowed), MS = 3 (separated) and MS = 4 (divorced) It is intended that in future, difficulties arising from not knowing the status of the absentee partner involved in MS = 2, will be resolved and these cases will cover only those who should be treated as a couple for benefit purposes.

The **derived** variable DEPCHLDB, ~~which is produced by the questionnaire programme itself and is not related to any particular question,~~ indicates the total number of dependent children in a benefit unit. Dependants include those aged 15 and under, and those aged 16,17 and 18 in full time non-advanced education.

Following investigation of V30, it appears that some benefit units, marital status does not match up with benefit unit details (eg, people coded as married but where a spouse is not present). Instead of identifying singles/couples on the basis of their marital status, a check is added to see if a second adult (spouse) does/does not exist. However, for MS=2 **and SPOUT=1** (married spouse not in household), the absence of a partner is legitimate (typical example - husband working abroad on 9 month contract thereby failing the 6 month residency rule for coding members of the household) and these cases have to be allowed for under the relevant couples sections.

2 FRS Specification

For each BENUNIT record

Adult under or over pension age is determined as

If AGE >= 60 = over pension age
 If AGE < 60 = under pension age

Benefit Unit under or over is determined as any adult in the Benefit Unit over pension age = Benefit Unit over pension age.

Therefore FAMTYPE is determined as

CODE CONDITION

- 1 If under pension age and **DEPCHILDB > 0** and ((**MS=2 and second adult record in BU exists**) or (**COHAB COHABIT =1 and second adult record in BU exists**) or (MS=2 **and SPOUT=1**) where single adult record)
- 2 If under pension age and **DEPCHILDB > 0** and only one adult record in BU and MS<>2
- 3 If under pension age and **DEPCHILDB = 0** and ((**MS=2 and second adult record in BU exists**) or (**COHAB COHABIT =1 and second adult record in BU exists**) or (MS=2 **and SPOUT=1**) where single adult record)
- 4 If under pension age and **DEPCHILDB = 0** and only one adult record in BU and MS<>2

 --

- 5 If over pension age and **DEPCHILDB** > 0 ((**MS=2** and second adult record in BU exists) or (**COHAB COHABIT =1** and second adult record in BU exists) or (MS=2 and **SPOUT=1**) where single adult record)
- 6 If over pension age and **DEPCHILDB** > 0 and only one adult record in BU and MS<>2
- 7 If over pension age and **DEPCHILDB** = 0 and ((**MS=2** and second adult record in BU exists) or (**COHAB COHABIT =1** and second adult record in BU exists) or (MS=2 and **SPOUT=1**) where single adult record)
- 8 If over pension age and **DEPCHILDB** = 0 and only one adult record in BU and MS<>2
- 2 If variable cannot be derived ie if any of **DEPCHILDB** AGE, SEX or MS are missing

3 Results

Tabulate numbers in each category.

FSMBU

Purpose : This specification produces a variable that calculates the value of free school meals for each benefit unit.

Created : VC - 2 September 1993

Database Table : BENUNIT

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : FSMVAL FSMHH FSMLKVAL, FSMLKBU, FSMLKHH, FWMLKVAL, FWMLKBU & FWMLKHH

Children :

Parents :

Core variable/user : HBAI

Amendments : VC - 12 October 1993 - To set an amount for the cost of free school meals as in FES appendix 66

: VC - 2 November 1993 - To take out reference to -1 (not applicable as should be referred to as 0 in HBAI.

: VC - 11 February 1994 Amendments to reflect version 30 changes

: JS - 1 April 1996 to reflect version 31 changes

: VE - 22 May 1996 - Amendments to reflect initial V32 changes - SCHMEAL replaced by FREEITEM

: VE - 27 June 1996 - To set amount for the cost of free school meals to April 1995 level

: VE - 1 July 1996 - Amended for constants being held in a separate table

: VE - 29 October 1996 - FREEITEM in Blaise code renamed SCHMEAL etc.

: SG - 25 June 1997 - Updates for V33 - constants location

: EP - 12 August 1998 - No initial V34 update needed

:SB - 2 September 1999 – Security complete

- Spilt code up into child, benunit and household level

Issued : 2 September 1999

1 Definition

This variable is coded as

- FSMBU This is the total value of any free school meals received by each benefit unit.
- 1 Not applicable to this case - applies to all of above variables.
- 2 Unable to derive due to missing values - applies to all of above variables.

The value of FSMVAL with in each BU is summed to give total BU cost. See FSMVAL for detail on how value calculated for each individual child.

--

2 FRS Specification

Code

Condition

FSMBU

For each BENEFIT UNIT

sum each occurrence of FSMVAL for each child in benefit unit

NB - Child is FRS version of child ie 15 and under or aged 16 to 19 and in full time education.

0

Not applicable - where case has no children (NUMCHIL/DEPCHILD = 0) or no free school meals.

-2

Unable to derive due to any of above values being missing.

 --

FSMHH

Purpose : This specification produces a variable which calculates the value of free
 : school meals for each dependent in the household
 Created : VC - 2 September 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : FSMVAL FSMBU FSMLKVAL, FSMLKBU, FSMLKHH, FWMLKVAL,
 FWMLKBU & FWMLKHH
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 12 October 1993 - To set an amount for the cost of free school meals as
 : in FES appendix 66
 : VC - 2 November 1993 - To take out reference to -1 (not applicable as should
 : be referred to as 0 in HBAI.
 : VC - 11 February 1994 Amendments to reflect version 30 changes
 : JS - 1 April 1996 to reflect version 31 changes
 : VE - 22 May 1996 - Amendments to reflect initial V32 changes - SCHMEAL
 : replaced by FREEITEM
 : VE - 27 June 1996 - To set amount for the cost of free school meals to April
 : 1995 level
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 29 October 1996 - FREEITEM in Blaise code renamed SCHMEAL etc.
 : SG - 25 June 1997 - Updates for V33 - constants location
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 2 September 1999 - Security completed
 - Spilt code up into child, benunit & household levels
 Issued : 2 September 1999

1 Definition

This variable is coded as

- FSMHH This is the total value of any free school meals received by the household.
- 1 Not applicable to this case - applies to all of above variables.
- 2 Unable to derive due to missing values - applies to all of above variables.

The value of free school meals received by each dependent is derived from the SCHMEAL and SMLIT variable from the CHILD record. Where SCHMEAL = 1 (has some free school meals) the number of free meals is obtained from SMLIT. This amount is then multiplied by the cost of a school meal, which is to be found in the Tax Benefit Model to produce the total amount spent each week.

 --

The daily cost of a free school meal is given by the value COSTM.

Once the cost of free school meals has been produced for each dependent it must be accumulated for the benefit unit and then the household levels.

2 FRS Specification

Code

Condition

FSMHH

For each HOUSEHOLD

sum each occurrence of FSMBU for each benefit unit in the household

NB - Child is FRS version of child ie 15 and under or aged 16 to 19 and in full time education.

0 Not applicable - where case has no children (NUMCHIL/DEPCHILD = 0) or no free school meals.

-2 Unable to derive due to any of above values being missing.

3 Results

Tabulation will be required to show the number of children, benefit units and households by the value of their free school meals sorted into the following bands

- Under £2.50
- £2.50 - £5.00
- £5.00 - £7.50
- £7.50 - £10.00
- £10.00 - £12.50
- £12.50 - £15.00
- £15.00 - £17.50
- £17.50 and over

 --

FSMLKBU

Purpose : This specification produces a variable that calculates the value of free school milk for each benefit unit
 Created : VC - 3 September 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : FSMLKVAL FSMLKHH FSMVAL, FSMBU, FSMHH, FWMLKVAL, FWMLKBU & FWMLKHH
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 2 November 1993 - To take out reference to -1 not applicable and replace with 0
 : JS - 1 April 1996 to update for V31
 : VE - 22 May 1996 - Initial amendments for V32 - SCHMILK replaced by FREEITEM
 : VE - 14 June 1996 - To update the cost of a pint of milk to 1995-96 rates
 : VE - 30 October 1996 - FREEITEM replaced by SCHMILK
 : SG - 31 December 1997 - No initial V33 updates
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 17 April 2003 - Security completed
 - Spilt code up into child, benunit & household levels
 Issued : 17 April 2003

1 Definition

This variable is coded as

- FSMLKBU This is the total value of any free school milk received by any child in the benefit unit.
- 0 Not applicable to this case - applies to all of above variables no children or no free milk
- 2 Unable to derive due to missing values - applies to all of above variables

Once the cost of free school milk has been produced for each child it must be accumulated for each child in the benefit unit

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FSMLKBU	For each BENEFIT UNIT (for BENUNIT record)

--

sum each occurrence of FSMLKVAL for each child in benefit unit.

0 Not applicable - no school milk/no dependants

-2 Unable to derive as any of above variables are missing.

--

FSMLKHH

Purpose : This specification produces the variable that calculates the value of free school milk for each household
 Created : VC - 3 September 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : FSMVAL, FSMBU, FSMHH, FSMLKVAL, FSMLKBU, FWMLKVAL, FWMLKBU & FWMLKHH
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 2 November 1993 - To take out reference to -1 not applicable and replace with 0
 : JS - 1 April 1996 to update for V31
 : VE - 22 May 1996 - Initial amendments for V32 - SCHMILK replaced by FREEITEM
 : VE - 14 June 1996 - To update the cost of a pint of milk to 1995-96 rates
 : VE - 30 October 1996 - FREEITEM replaced by SCHMILK
 : SG - 31 December 1997 - No initial V33 updates
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 17 April 2003 – Security completed
 - Spilt code up into, child benunit & household levels
 Issued : 17 April 2003

1 Definition

This variable is coded as

FSMLKHH This is the total value of any free school milk received by any child in the household.

0 Not applicable to this case - applies to all of above variables no children or no free milk

-2 Unable to derive due to missing values - applies to all of above variables

The value of free school milk received by each person is derived from the SCHMILK and SMKIT variables from the CHILD record on the database. Where SCHMILK = 1 (has some free school milk) the number of pints of milk is obtained from SMKIT. This amount is then multiplied by the cost of each bottle of free school milk to produce the total amount spent each for that child.

The cost of a bottle of free school milk is calculated using the same method that CSO uses for FES. It may need updating each year so check with FES first before running. The cost of a third of a pint of milk is given by the value COSTMLK.

 --

Once the cost of free school milk has been produced for each child it must be accumulated for each child in the benefit unit and then the household.

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FSMLKHH	For each HOUSEHOLD (for HOUSEHOL record) sum each occurrence of FSMLKVAL for each child in household.
0	Not applicable - no school milk/no dependants
-2	Unable to derive as any of above variables are missing.

3 Results

Tabulation will be required to show the number of children, benefit units and households by the value of free school milk received sorted into the following bands

- Under £1.00
- £1.00 - £2.00
- £2.00 - £3.00
- £3.00 - £4.00
- £4.00 - £5.00
- £5.00 - £6.00
- £6.00 or over

4 Test Cases

None produced as yet.

FSMLKVAL

Purpose : This specification produces a variable that calculates the value of free school milk for each child
 Created : VC - 3 September 1993
 Database Table : CHILD
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : FSMVAL, FSMBU, FSMHH, FSMLKBU, FSMLKHH, FWMLKVAL, FWMLKBU & FWMLKHH
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 2 November 1993 - To take out reference to -1 not applicable and replace with 0
 : JS - 1 April 1996 to update for V31
 : VE - 22 May 1996 - Initial amendments for V32 - SCHMILK replaced by FREEITEM
 : VE - 14 June 1996 - To update the cost of a pint of milk to 1995-96 rates
 : VE - 30 October 1996 - FREEITEM replaced by SCHMILK
 : SG - 31 December 1997 - No initial V33 updates
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 17 April 2003 - Security completed
 - Spilt code up into child, benunit & household levels
 Issued : 17 April 2003

1 Definition

This variable is coded as

- FSMLKVAL This is the total value of any free school milk received by a child.
 0 Not applicable to this case - applies to all of above variables no children or no free milk
 -2 Unable to derive due to missing values - applies to all of above variables

The value of free school milk received by each person is derived from the SCHMILK and SMKIT variables from the CHILD record on the database. Where SCHMILK = 1 (has some free school milk) the number of pints of milk is obtained from SMKIT. This amount is then multiplied by the cost of each bottle of free school milk to produce the total amount spent each for that child.

The cost of a bottle of free school milk is calculated using the same method that CSO uses for FES. It may need updating each year so check with FES first before running. The cost of a third of a pint of milk is given by the value COSTMLK.

--

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FSMLKVAL	For each CHILD from CHILD record. If SCHMILK = 1, calculate the value of free school milk calculate FSMLKVAL = SMKIT x cost of free school milk (COSTMLK) If SCHMILK = 2, calculate FSMLKVAL = 0.
0	Not applicable - no school milk/no dependants
-2	Unable to derive as any of above variables are missing.

--

FSMVAL

Purpose : This specification produces a variable that calculates the value of free school meals for each dependent
 Created : VC - 2 September 1993
 Database Table : CHILD
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : FSMBU, FSMHH, FSMLKVAL, FSMLKBU, FSMLKHH, FWMLKVAL, FWMLKBU & FWMLKHH
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 12 October 1993 - To set an amount for the cost of free school meals as in FES appendix 66
 : VC - 2 November 1993 - To take out reference to -1 (not applicable as should be referred to as 0 in HBAI.
 : VC - 11 February 1994 Amendments to reflect version 30 changes
 : JS - 1 April 1996 to reflect version 31 changes
 : VE - 22 May 1996 - Amendments to reflect initial V32 changes - SCHMEAL replaced by FREEITEM
 : VE - 27 June 1996 - To set amount for the cost of free school meals to April 1995 level
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 29 October 1996 - FREEITEM in Blaise code renamed SCHMEAL etc.
 : SG - 25 June 1997 - Updates for V33 - constants location
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 17 April, 20039 - Security completed
 - Spilt code up into child, benunit & household level
 Issued : 17 April, 20039

1 Definition

This is the value of any free school meals received by a dependent.

- 1 Not applicable to this case
- 2 Unable to derive due to missing values

The value of free school meals received by each dependent is derived from the SCHMEAL and SMLIT variable from the CHILD record. Where SCHMEAL = 1 (has some free school meals) the number of free meals is obtained from SMLIT. This amount is then multiplied by the cost of a school meal, which is to be found in the Tax Benefit Model to produce the total amount spent each week.

 --

The daily cost of a free school meal is given by the value COSTM.

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FSMVAL	<p>For each CHILD from CHILD table,</p> <p>Set COST (of free school meals) to COSTM (Supplied to FES from Tax Benefit Model for 1994)</p> <p>If SCHMEAL = 1, calculate the value of free school meals</p> <p>calculate FSMVAL = SMLIT * COST</p>

NB - Child is FRS version of child ie 15 and under or aged 16 to 19 and in full time education.

0	Not applicable - where case has no children (NUMCHIL/DEPCHILD = 0) or no free school meals.
-2	Unable to derive due to any of above values being missing.

3 Results

Tabulation will be required to show the number of children, benefit units and households by the value of their free school meals sorted into the following bands

- Under £2.50
- £2.50 - £5.00
- £5.00 - £7.50
- £7.50 - £10.00
- £10.00 - £12.50
- £12.50 - £15.00
- £15.00 - £17.50
- £17.50 and over

FWMLKBU

Purpose : This specification produces a variable that calculates the value of free welfare milk for each benefit unit.
 Created : VC - 3 September 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : FSMVAL, FSMBU, FSMHH, FWMLKVAL, FWMLKHH, FSMLKVAL, FSMLKBU & FSMLKHH
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 2 November 1993 To remove references to -1 not applicable and replaced by 0
 : JS - 1 April 1996 - to update for V31.
 : VE - 22 May 1996 - Amendments to reflect initial V32 changes - WELFMILK replaced by FREEITEM
 : VE - 14 June 1996 - To update cost of a pint of milk to 1995-96 rates
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : SG - 25 June 1997 - Updated for V33, constants table change
 : EP - 12 August 1998 - No initial V34 update needed
 : SB – 3 September 1999 – Security completed
 - Spilt up into person, benunit & household levels
 Issued : 17 April 2003

1 Definition

This variable is coded as

FWMLKBU This is the total value of any free welfare milk received by any person in the benefit unit.
 0 Not applicable to this case - applies to all of above variables
 -2 Unable to derive due to missing values - applies to all of above variables

Once the cost of free welfare milk has been produced for each person it must be accumulated for each person in the benefit unit and then the household.

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FWMLKBU	For each BENEFIT UNIT (for BENUNIT record)

--

sum each occurrence of FWMLKVAL for each adult and child in benefit unit.

0 Not applicable - no welfare milk

-2 Unable to derive as any of above variables are missing.

 --

FWMLKHH

Purpose : This specification produces a variable, which calculates the value of free welfare milk for each household

Created : VC - 3 September 1993

Database Table : HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : FSMVAL, FSMBU, FSMHH, FWMLKVAL, FWMLKBU, FSMLKVAL, FSMLKBU & FSMLKHH

Children :

Parents :

Core variable/user : HBAI

Amendments : VC - 2 November 1993 To remove references to -1 not applicable and replaced by 0

: JS - 1 April 1996 - to update for V31.

: VE - 22 May 1996 - Amendments to reflect initial V32 changes - WELFMILK replaced by FREEITEM

: VE - 14 June 1996 - To update cost of a pint of milk to 1995-96 rates

: VE - 1 July 1996 - Amended for constants being held in a separate table

: SG - 25 June 1997 - Updated for V33, constants table change

: EP - 12 August 1998 - No initial V34 update needed

: SB - 3 September 1999 – Security completed

- Spilt up into person, benunit & household levels

Issued : 3 September 1999

1 Definition

This variable is coded as

- FWMLKHH This is the total value of any free welfare milk received by any person in the household.
- 0 Not applicable to this case - applies to all of above variables
- 2 Unable to derive due to missing values - applies to all of above variables

The value of free welfare milk received by each person is derived from the WELFMILK and WMKIT variables from the ADULT and CHILD records on the database. Where WELFMILK = 1 (has some free welfare milk) the number of pints of milk is obtained from WMKIT. This amount is then multiplied by the cost of each pint of free welfare milk (to be supplied by the Tax Benefit Model) to produce the total amount spent each for that person.

From 1994/95, it has been agreed that costs should be based on CSO figures for a pint of milk (e consistent with free school milk calculation) rather than the TBM. The cost of a pint of milk is given by the value COSTWMK.

--

Once the cost of free welfare milk has been produced for each person it must be accumulated for each person in the benefit unit and then the household.

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FWMLKHH	For each HOUSEHOLD (for HOUSEHOL record) sum each occurrence of FWMLKVAL for each adult and child in household.
0	Not applicable - no welfare milk
-2	Unable to derive as any of above variables are missing.

3 Results

Tabulation will be required to show the number of adults/children, benefit units and households by the value of free welfare milk received sorted into the following bands

Under £1.00
£1.00 - £2.00
£2.00 - £3.00
£3.00 - £4.00
£4.00 - £5.00
£5.00 - £6.00
£6.00 or over

 --

FWMLKVAL (Adult)

Purpose	: This specification produces a variable that calculates the value of free welfare milk for each adult
Created	: VC - 3 September 1993
Database Table	: ADULT
Minimum Value	: 0
Maximum Value	:
Units	: Real
Validations	:
Related Variables	: FSMVAL, FSMBU, FSMHH, FSMLKVAL, FWMLKVAL (child), FWMLKBU, FWMLKHH, FSMLKBU & FSMLKHH
Children	:
Parents :	:
Core variable/user	: HBAI
Amendments	: VC - 2 November 1993 To remove references to -1 not applicable and replaced by 0
	: JS - 1 April 1996 - to update for V31.
	: VE - 22 May 1996 - Amendments to reflect initial V32 changes - WELFMILK replaced by FREEITEM
	: VE - 14 June 1996 - To update cost of a pint of milk to 1995-96 rates
	: VE - 1 July 1996 - Amended for constants being held in a separate table
	: SG - 25 June 1997 - Updated for V33, constants table change
	: EP - 12 August 1998 - No initial V34 update needed
	: SB - 2 September 1999 – Security completed
	- Split up into Adult, child, household & benunit levels
Issued	: 2 September 1999

1 Definition

This variable is coded as

- FWMLKVAL This is the total value of any free welfare milk received by a person (adult).
- 0 Not applicable to this case - applies to all of above variables
- 2 Unable to derive due to missing values - applies to all of above variables

The value of free welfare milk received by each person is derived from the WELFMILK and WMKIT variables from the ADULT and CHILD records on the database. Where WELFMILK = 1 (has some free welfare milk) the number of pints of milk is obtained from WMKIT. This amount is then multiplied by the cost of each pint of free welfare milk (to be supplied by the Tax Benefit Model) to produce the total amount spent each for that person.

From 1994/95, it has been agreed that costs should be based on CSO figures for a pint of milk (e consistent with free school milk calculation) rather than the TBM. The cost of a pint of milk is given by the value COSTWMK.

--

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FWMLKVAL	For each ADULT from ADULT record. If WELFMILK = 1, calculate the value of free welfare milk calculate FWMLKVAL = WMKIT x COSTWMK
0	Not applicable - no welfare milk
-2	Unable to derive as any of above variables are missing.

3 Results

Tabulation will be required to show the number of adults/children, benefit units and households by the value of free welfare milk received sorted into the following bands

Under £1.00
£1.00 - £2.00
£2.00 - £3.00
£3.00 - £4.00
£4.00 - £5.00
£5.00 - £6.00
£6.00 or over

--

FWMLKVAL (Child)

Purpose : This specification produces a variable that calculates the value of free welfare milk for each child

Created : VC - 3 September 1993

Database Table : CHILD

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : FSMVAL, FSMBU, FSMHH, FSMLKVAL, FWMLKVAL (adult), FWMLKBU, FWMLKHH, FSMLKBU & FSMLKHH

Children :

Parents :

Core variable/user : HBAI

Amendments : VC - 2 November 1993 To remove references to -1 not applicable and replaced by 0
 : JS - 1 April 1996 - to update for V31.
 : VE - 22 May 1996 - Amendments to reflect initial V32 changes - WELFMILK replaced by FREEITEM
 : VE - 14 June 1996 - To update cost of a pint of milk to 1995-96 rates
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : SG - 25 June 1997 - Updated for V33, constants table change
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 2 September 1999 – Security completed
 - Split up into Adult, child, household & benunit levels

Issued : 2 September 1999

1 Definition

This variable is coded as

FWMLKVAL This is the total value of any free welfare milk received by a person (child).

0 Not applicable to this case - applies to all of above variables

-2 Unable to derive due to missing values - applies to all of above variables

The value of free welfare milk received by each person is derived from the WELFMILK and WMKIT variables from the ADULT and CHILD records on the database. Where WELFMILK = 1 (has some free welfare milk) the number of pints of milk is obtained from WMKIT. This amount is then multiplied by the cost of each pint of free welfare milk (to be supplied by the Tax Benefit Model) to produce the total amount spent each for that person.

From 1994/95, it has been agreed that costs should be based on CSO figures for a pint of milk (e consistent with free school milk calculation) rather than the TBM. The cost of a pint of milk is given by the value COSTWMK.

--

Once the cost of free welfare milk has been produced for each person it must be accumulated for each person in the benefit unit and then the household.

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FWMLKVAL	For each CHILD from CHILD record. If WELFMILK = 1, calculate the value of free welfare milk calculate FWMLKVAL = WMKIT x COSTWMK
0	Not applicable - no welfare milk
-2	Unable to derive as any of above variables are missing.

3 Results

Tabulation will be required to show the number of adults/children, benefit units and households by the value of free welfare milk received sorted into the following bands

Under £1.00
£1.00 - £2.00
£2.00 - £3.00
£3.00 - £4.00
£4.00 - £5.00
£5.00 - £6.00
£6.00 or over

GROSSCT

Purpose	: Shows gross Council Tax (Band D) for each Local Authority
Created	: 7 September 1998
Database Table	: HOUSEHOL
Minimum Value	:
Maximum Value	:
Units	: Integer
Validations	:
Related Variables	:
Children	:
Parents	:
Core variable/user	: FRS (hot-decking)
Amendments	:SB 1 September 1999 – Changed where information read from - Security completed
Issue date	: 1 September 1999

1 Definition

This is the Band 'D' Gross Council Tax for the household based on its Local Authority Code.

2 FRS Specification

The amount is read from Dvmeta35.xls.

 --

GROSSPAY

Purpose : To show the total amount of earnings received by an adult from main and subsidiary jobs as an employee, excluding any income from odd jobs. This is the gross amount before any deductions for Income Tax, National Insurance, Trade Union dues etc.

Created : VC - 10 May 1993

Database Table : ADULT

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : INEARNNS – see note in main text

Children :

Parents :

Core variable/user : PSM

Issue date : 17 April 2003

Amendments : VC - 24 June 1993 To remove references to odd jobs, baby sitting and mail order agents which are to be included in a net income variable.
 : VC - 13 September 1993 Amended using guidelines provided by Ms Honey.
 : VC - 15 February 1994 Amended to reflect version 30 changes
 : VC - 28 February 1994 To exclude any records with a period code of 12 or 13
 : JS - 20 February 1996 to allow skipped values where variables have been imputed
 : JS - 21 March 1996 to correctly consider JOBTYP=1
 : VE - 24 May 1996 - Initial updates for V32 - to amend for new category definitions for DEDUC
 : VE - 25 February 1997 - To amend for V33 - new category of OTHDED and DEDUC for private medical insurance. Also vague periods now 90, 95 and 97 rather than 12 and 13.
 : VE - 3 March 1997 - Amendment for V33 - To replace TAXINC with INCLPAY3
 : SG - 6 January 1998 - Update with V32 change to make skipped if all jobs are self employed
 : SG - 11 February 1998 - PAYS LIP changed for V33
 : EP - 12 August 1998 - No initial V34 update needed
 : EP - 11 May 1999 – Add in explanation of differences between GROSSPAY and INEARNNS
 : SB - 1 November 1999 – Security completed, replaced old charity tax variables with CHRAMT

1 Definition

This variable is coded as

GROSSPAY The total gross earnings before deductions for Income Tax, NI etc from all jobs an adult may have as an employed earner, excluding any income from odd jobs.

-1 Not applicable where an adult does not have any jobs or all are self employed.

--

-2 Unable to derive where any variables missing.

The variable GROSSPAY is derived from a variety of variables held in the ADULT and JOB tables and for each job held by that adult.

Although similar to the derived variable INEARNNS, GROSSPAY sometimes differs in value from it. This is because GROSSPAY looks at the *last pay* that the respondent received. In cases where this was not the usual pay, GROSSPAY does not give the best representation of an individual's earnings – this can be obtained from INEARNNS, as this variable includes a check on whether or not the last pay was the usual pay (using the variable UGROSS). GROSSPAY also does not include any bonuses received, which are included in INEARNNS (the variables BONAMT1-6, which are weeklyised when included), and also does not deduct other allowance/refund variables (e.g. HHA1-3, MILEAMT, MOTAMT). If any user is unclear which of these two variables to use, then refer to ASD3E.

GROSSPAY is derived from the variable GRWAGE which holds the person's gross earnings before tax, NI etc but only where the payslip has been consulted (where PAYSLIP = 1 or 2).

If the payslip has not been consulted (where PAYSLIP = 3) the amount of net pay is obtained from PAYAMT. This variable holds the total amount of net pay after all deductions have been taken off and these deductions must be added back to PAYAMT to find GROSSPAY.

The amount of income tax deducted is found in PAYE and National Insurance in NATINS these must be found in all cases. Other deductions for example trade union fees, payments to charities etc are also to be added back to PAYAMT but may not be relevant to every case. Therefore, if CHRTAXF = 1 (indicating that the person has a deduction for charities) (see deducts for reason using this not charity variable) the amount held in AMTTAXF must be added to PAYAMT and if CHROTH = 1 (indicates that there is another deduction for a charity) the amount held in AMTOTH must be added to PAYAMT.

If any of OTHDED1 to OTHDED8 = 1 there will be a deduction for pension/superannuation, union fees, friendly societies, sports social clubs, repayment of a loan from employer, private medical insurance and any other deductions with the amount held in the relevant DEDUC variable. DEDUC1 holds the amount for pension/superannuation, DEDUC2 holds the amount additional voluntary contributions, DEDUC3 holds the amount of union fees, DEDUC4 holds amount for friendly societies, DEDUC5 holds amount for sports clubs, DEDUC6 holds the amount for the repayment of a loan from employer, DEDUC7 holds the amount for private medical insurance and DEDOTH holds the amount for any other deduction not already mentioned and any occurrence of these must be added to PAYAMT.

The variables OTHDED1 to OTHDED8, DEDUC1 to DEDUC7 and DEDOTH are database variables collected from the questions OTHDED in the e-main block which asks were there any other deductions from your wage or salary and DEDUC which holds the amount of deduction in each case.

To get a person's gross earnings a check must be made to see if an income tax refund was included in PAYAMT. Therefore, if INCLPAY3 = 1 the amount held in TAXAMT has to be deducted from PAYAMT to get a true amount of gross earnings.

However, if the period code for the benefit is 90 or 95 or 97 (lump-sum/one-off or other period/less than one week) the record must be set to unable to derive as it has not been possible to convert the amount

 --

of benefit into a weekly amount during the database conversion process. Therefore, if PAYPD = 90 or 95 or 97 GROSSPAY is set to -2.

Where PAYSLIP has been imputed to yes, GRWAGE will have been skipped. Similarly where PAYAMT is imputed, PAYPD and other variables may also be skipped. This has to be catered for in the DV specification

GROSSPAY is then the total amount of earnings from all of these sources for all jobs.

2 FRS Specification

GROSSPAY If WORKING=1 or JOBAWAY=1

From JOB table for every job a person has If EMPEE=1

If first job or GROSSPAY set to skipped as first job was self employed then set GROSSPAY=0

If PAYPD equals -1 or 1-10 13, 26 or 52 and

If PAYSLIP = 1 or 2, and GRWAGE exists calculate GROSSPAY = amount in GRWAGE.

Or if PAYSLIP = 3 or (PAYSLIP=1 or 2 and GRWAGE=-1), calculate GROSSPAY as follows

GROSSPAY = sum of GROSSPAY, PAYAMT,

If PAYE exists add PAYE to GROSSPAY

If NATINS exists add NATINS to GROSSPAY

If CHRAMT exists add CHRAMT to GROSSPAY

If DEDUC1 exists add DEDUC1 to GROSSPAY

If DEDUC2 exists add DEDUC2 to GROSSPAY

If DEDUC3 exists add DEDUC3 to GROSSPAY

If DEDUC4 exists add DEDUC4 to GROSSPAY

If DEDUC5 exists add DEDUC5 to GROSSPAY

If DEDUC6 exists add DEDUC6 to GROSSPAY

If DEDUC7 exists add DEDUC7 to GROSSPAY

If DEDOTH exists add DEDOTH to GROSSPAY

--

Then if JOBTYP=1 and INCLPAY3=1 and TAXAMT exists subtract TAXAMT from total GROSSPAY

GROSSPAY is then the total of all of the above for every job a person has.

(coding should now be the same as UGRSPAY, except does not include usual earnings)

If first job is self employed then set GROSSPAY to skipped (-1)

-2 If any of the above variables are missing or PAYPD = 90 or 95 or 97

3 Results

Tabulation is required to show the number of adults falling into the following bands of weekly earnings:

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 - £350
£350 - £400
£400 - £450
£450 - £500
£500 and over

--

GVTREGN

Purpose : To indicate in which Government Office Region the interviewee lives
 Created : 27 August 1996
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 12
 Units : Integer (excludes 3)
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general use)
 Amendments : VE - 23 September 1996 - To include the change of name of the Government Office for Yorkshire and Humberside from 1 October 1995 to the Government Office for Yorkshire and Humber
 : VE - 3 March 1997 - To include separate categories for the North West and for Merseyside in line with Regional Trends publications and to correct the order of GORs to make consistent with Regional Trends
 : SG - 23 December 1997 - Changes to LAC code numbers for V33
 : SG - 23 January 1998 - Include Merseyside with Northwest following concerns over Merseyside sample stratification
 : EP - 12 August 1998 - Local Authority name changes for V34
 : EP - 5 May 1999 - general tidy up of DV for resurrection of its use in V34 (DV not on first public release, instead it was a base variable which not group Merseyside with the North West)
 : JC - 21 September 1999 - Security completed, LA name changes for V35
 : AW - 20 February 2000 - don't call on constants table
 Issue date : 17 April, 2003

1 Definition

This variable is coded as

1	North East
2	North West and Merseyside
4	Yorkshire and the Humber
5	East Midlands
6	West Midlands
7	Eastern
8	London
9	South East
10	South West
11	Wales
12	Scotland

-1 Not applicable to this case (should be no cases)

--

The government office region is derived from the Local Authority Code in the variable LAC. It should be noted that these regions are not exactly the same as the standard regions (STDREGN).

Gvtreg is provided as a base variable but the DV groups Merseyside and the Northwest. This been done following concerns that stratification of sample using standard regions leads to problems with analysis by Government Regions.

3 FRS Specification

-1 Not applicable to this case (shouldn't be any)

From the household table

If Gvtreg = 3 then Gvtreg = 2

 --

HBINDBU

Purpose : To indicate if any person in the benefit unit receives Income Support, Housing Benefit or Council Tax Benefit.
 Created : JS: 19 January 1996
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value : 7
 Units : integer
 Validations :
 Related Variables : HBINDHH
 Children :
 Parents :
 Core variable/user : Take-up
 Amendments : BS - 3 August 1995. Amended to take into account changes to V31 of the Questionnaire.
 : JS - 19 January 1996: to include additional codes for IS receipt
 : JS - 25 January: to make clear that codes for IS refer to benefit units where any adult is in receipt of IS
 : VE - 4 June 1996 - No initial amendments needed for V32 update
 : SG - 3 December 1997 - V33 update - HB receipt checked differently for
 BU>1 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 1 September 1999 - Security completed, no changes required for V35
 Issue date : 1 September 1999

Assumptions

Council tax information is collected only once, any rebate is therefore assumed to be for the householder benefit unit

1 Definition

This variable is coded as

HBINDBU	Indicates that someone in the benefit unit is receiving Income Support, Housing Benefit or Council Tax Benefit as follows,
0	No HB/CTB/IS
1	If receives CTB only (no IS or HB)
2	If receives HB only (no IS or CTB)
3	If receives IS only (no HB or CTB)

 --

- 4 If receives both HB and CTB (no IS)
- 5 If receives both HB and IS (no CTB)
- 6 If receives both CTB and IS (no HB)
- 7 If receives HB, CTB and IS
- 1 Not applicable to this case: these should not occur since questions are asked of all households
- 2 Unable to derive because of missing values

This variable is derived from questions in the HOUSEHOL and ADULT records

If the householder benefit unit receives Council Tax Benefit (CTB), this is shown by the CTREB = 1 (was any CTB allowed in connection with your last CT payment - 1 = yes, 2 = no).

Receipt of Housing Benefit is identified from two separate questions depending on the type of household:

- i HBENEFIT = 1 (has received HB in connection with last rent payment) (household record) There is no need to check whether rebate=1 because this question is only asked if HBENEFIT is set to 1
- ii If HHSTAT = 2 (shared household) or CVPAY>0 (boarders/lodgers paying rent) then from ADULT record if H BOTHBU=1 (individual receives HB)

IS receipt is identified where any person in the benefit unit answers "yes" to BEN3Q2 (ADULT record - are you at present receiving IS)

2 FRS Specification

For each benefit unit in the household

Code Condition

HBINDBU

- 0 If BENUNIT=1 (and CTREB=2) or BENUNIT>1 (CTB not in receipt) and (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU) or if HHSTAT=2 or CVPAY>0 and for each adult H BOTHBU=2 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
- 1 If BENUNIT=1 (and CTREB=1) (CTB in receipt: only possible for householder benefit unit) and (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU) or if HHSTAT=2 or CVPAY>0 and for each adult H BOTHBU=2 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
- 2 If BENUNIT=1 (and CTREB=2) or BENUNIT>1 (CTB not in receipt)

 --

- and (BENUNIT=1 and (HBENEFIT=1) (HB in receipt for first BU)
 or if HHSTAT=2 or CVPAY>0 and for at least one adult H BOTHBU=1
 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
- 3 If BENUNIT=1 (and CTREB=2) or BENUNIT>1 (CTB not in receipt)
 and (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU)
 or if HHSTAT=2 or CVPAY>0 and for each adult H BOTHBU=2
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
- 4 If BENUNIT=1 (and CTREB=1) (CTB in receipt: only possible for householder benefit unit)
 and (BENUNIT=1 and HBENEFIT=1) (HB in receipt for first BU)
 or if HHSTAT=2 or CVPAY>0 and for at least one adult H BOTHBU=1
 and for each adult in benefit unit, BEN3Q2<>1 (IS not in receipt)
- 5 If BENUNIT=1 (and CTREB=2) or BENUNIT>1 (CTB not in receipt)
 and (BENUNIT=1 and HBENEFIT=1) (HB in receipt for first BU)
 or if HHSTAT=2 or CVPAY>0 and for at least one adult H BOTHBU=1
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
- 6 If BENUNIT=1 (and CTREB=1) (CTB in receipt: only possible for householder benefit unit)
 and (BENUNIT=1 and HBENEFIT=2) (HB not in receipt for first BU)
 or if HHSTAT=2 or CVPAY>0 and for each adult H BOTHBU=2
 and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
- 7 If BENUNIT=1 (and CTREB=1) (CTB in receipt: only possible for householder benefit unit)
 and ((BENUNIT=1 and HBENEFIT=1) (HB in receipt for first BU)
 or if HHSTAT=2 or CVPAY>0 and for at least one adult H BOTHBU=1)
- and for at least one adult in benefit unit, BEN3Q2=1 (IS in receipt)
- 1 Not applicable to this case
- 2 Unable to derive as any of the above are missing.

 --

HBINDHH

Purpose : To indicate if any person in the household receives Income Support, Housing Benefit or Council Tax Benefit.
 Created : VC - 1 September 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value : 7
 Units : integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : Take-up
 Amendments : BS - 3 August 1995. Amended to take into account changes to V31 of the Questionnaire.
 : JS - 19 January 1996: to include additional codes for IS receipt
 : JS - 25 January 1996: to make completely clear that codes for IS refer to households where any adult is in receipt of IS
 : VE - 4 June 1996 - No initial amendments needed for V32 update
 : SG - 7 October 1997 - No initial amendments needed for V33 update
 : EP - 12 August 1998 - No initial V34 update needed
 : JC - 9 September 1999 - Security completed, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable is coded as

HBINDHH	Indicates that someone in the household is receiving Income Support, Housing Benefit or Council Tax Benefit as follows,
0	No HB/CTB/IS
1	If receives CTB only (no IS or HB)
2	If receives HB only (no IS or CTB)
3	If receives IS only (no HB or CTB)
4	If receives both HB and CTB (no IS)
5	If receives both HB and IS (no CTB)
6	If receives both CTB and IS (no HB)
7	If receives HB, CTB and IS

 --

-1 Not applicable to this case - not expected because questions are asked of all households

-2 Unable to derive because of missing values

This variable is derived from questions in the HOUSEHOL and ADULT records

If the household receives Council Tax Benefit (CTB), this is shown by the CTREB = 1 (was any CTB allowed in connection with your last CT payment - 1 = yes, 2 = no).

Receipt of Housing Benefit is identified from two separate questions depending on the type of household:

- i HBENEFIT = 1 (has received HB in connection with last rent payment) (household record).
- ii If HHSTAT = 2 (shared household) or CVPAY>0 (boarders/lodgers paying rent) then from ADULT record if HBOTHBU=1 (individual receives HB)

IS receipt is identified where any person in the household answers "yes" to BEN3Q2 (ADULT record - are you at present receiving IS)

2 FRS Specification

Code Condition

HBINDHH

- 0 If CTREB=2
and HBENEFIT=2
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
and for each adult in household, BEN3Q2<>1
- 1 If CTREB=1
and HBENEFIT=2
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
and for each adult in household, BEN3Q2<>1
- 2 If CTREB=2
and HBENEFIT=1
or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
and for each adult in household, BEN3Q2<>1
- 3 If CTREB=2
and HBENEFIT=2
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
and for at least one adult in household, BEN3Q2=1
- 4 If CTREB=1

 --

- and HBENEFIT=1
 - or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 - and for each adult in household, BEN3Q2<>1

- 5 If CTREB=2
 - and HBENEFIT=1
 - or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 - and for at least one adult in household, BEN3Q2=1

- 6 If CTREB=1
 - and HBENEFIT=2
 - or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
 - and for at least one adult in household, BEN3Q2=1

- 7 If CTREB=1
 - and HBENEFIT=1
 - or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 - and for at least one adult in household, BEN3Q2=1

- 1 Not applicable to this case

- 2 Unable to derive as any of the above are missing.

 --

HBSUPRAN

Purpose : To indicate the total amount of superannuation or pension contributions :
 deducted from a person's earnings from all jobs, excluding any additional
 voluntary contributions (AVCs).
 Created : January 1996
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : DEDUCTS, SUPERAN
 Children :
 Parents :
 Core variable/user :
 Issue date : 17 April 2003
 Amendments : JS - 21 February 1996 to allow for skipped values when variables have been
 imputed
 : VE - 23 May 1996 - No initial amendments needed for V32 update
 : SG - 25 June 1997 - V33 updates - change in period codes
 : EP - 12 August 1998 - No initial V34 update needed
 : JC - 9 September 1999 - Security completed, no other changes for V35

NB - will not include self-employed jobs separate base variables provided for these jobs

1 Definition

This variable is coded as

HBSUPRAN The total amount of superannuation or pension contributions deducted from a person's
 earnings from all jobs.

-1 Not applicable in this case - people who do not have superannuation deductions and
 those not working or self-employed

-2 Unable to derive variable.

The amount of superannuation or pension contributions is derived from all jobs and where OTHDED1 are
 coded 1 to show that an amount for a pension or superannuation is deducted. The variable DEDUC1 will
 then hold the amount of superannuation/pension contribution. The variables OTHDED1 and DEDUC1 are
 created in the database to hold the answers to the multi response questions OTHDED (were there any
 other deductions from your wage/salary such as 1 = pension or superannuation, 3 = Union fees etc) and
 DEDUC which holds the amounts.

However, if the period code for the deduction is 90 or 95 or 97 (less than a week, lump-sum/one-off or
 other period), from the PAYPD variable, the record must be set to unable to derive as it has not been

 --

possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if PAYPD = 90 or 95 or 97 HBSUPRAN is set to -2.

2 FRS Specification

For each ADULT with record Job for all jobs

<u>Code</u>	<u>Condition</u>
HBSUPRAN	If PAYPD equals -1 or 1-10 or 13 or 26 or 52 and - If OTHDED1 = 1 and DEDUC1 exists, get the amount of the superannuation or pension contribution deducted from variable DEDUC1 else leave as skipped.
-1	Not applicable in this case - where OTHDED1 = 2 or 3 or OTHDED1 does not exist or there are no job records
-2	If variable cannot be defined because of missing data - where there is a job record but no values or if PAYPD = 90 or 95 or 97.

 --

HDAGE

Purpose :To create a variable for use in hot-decking which shows the age range in which a respondent falls
 Created :19 June 1996
 Database Table : ADULT
 Minimum Value : 1
 Maximum Value : 6
 Units : Integers
 Validations :
 Related Variables :
 Children : HDAGECH
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : SG - 27 June 1997 - no updates required for V33
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 27 August 1999 - Security completed, no other changes required for V35
 Issue date :17 April 2003

1 Definition

This variable is coded as

- 1 Age 16 to 24
- 2 Age 25 to 34
- 3 Age 35 to 44
- 4 Age 45 to 54
- 5 Age 55 to 64
- 6 Age 65 and over
- 2 Unable to derive due to missing values

HDAGE is derived from the AGE variable in the ADULT table. It shows the age range in which a respondent falls, and is used to specify classes for hotdecking.

2 FRS Specification

For each adult

Code Condition

--

- 1 From ADULT table
 If AGE>=16 and AGE<=24
- 2 From ADULT table
 If AGE>=25 and AGE<=34
- 3 From ADULT table
 If AGE>=35 and AGE<=44
- 4 From ADULT table
 If AGE>=45 and AGE<=54
- 5 From ADULT table
 If AGE>=55 and AGE<=64
- 6 From ADULT table
 If AGE>=65
- 2 Unable to derive due to missing values - There should be no missing values for AGE

HDAGECH

Purpose : To create a variable for use in hot-decking which shows the age range in which a respondent falls

Created : 19 June 1996

Database Table : CHILD

Minimum Value : 1

Maximum Value : 4

Units : Integers

Validations :

Related Variables :

 Children : HDAGE

 Parents :

Core variable/user : FRS (hot-decking)

Amendments : SG - 27 June 1997 - no updates required for V33

 : EP - 12 August 1998 - No initial V34 update needed

 : SB – 31 August 1999 – Security completed, no other changes required for V35

Issue date : 31 August 1999

1 Definition

This variable is coded as

- 1 Age 0 to 4
- 2 Age 5 to 9
- 3 Age 10 to 14
- 4 Age 15 and over
- 2 Unable to derive due to missing values

HDAGE is derived from the AGE variable in the CHILD table. It shows the age range, in which a respondent falls, and is used to specify classes for hot-decking.

2 FRS Specification

For each child

Code Condition

- 1 From CHILD table
- If AGE<=4
- 2 From CHILD table

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

- If AGE \geq 5 and AGE \leq 9
- 3 From CHILD table
- If AGE \geq 10 and AGE \leq 14
- 4 From CHILD table
- If AGE \geq 15
- 2 Unable to derive due to missing values - There should be no missing values for AGE

 --

HDBEN

Purpose :To create a variable for use in hot-decking which shows whether any income related benefits are received by an individual
 Created :19 June 1996
 Database Table : ADULT
 Minimum Value : 1
 Maximum Value : 2
 Units : Integers
 Validations :
 Related Variables : INIRBEN
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : SG - 27th June - No updates required for V33
 : EP - 7 September 1998 - Addition of BUHDBEN and HHHDBEN
 : SB - 1 September 1999 – Spilt SAS code up into adult, benunit and household levels
 Issue date : 1 September 1999

1 Definition

This variable is coded as

- 1 Income related benefits received by individual
- 2 No income related benefits received by individual
- 2 Unable to derive due to missing values

HDBEN is derived from the INIRBEN derived variable in the ADULT table. This shows the total amount of income received each week by individuals from income related benefits. If INIRBEN is greater than zero, an individual is classified as receiving income related benefits. If INIRBEN is zero, an individual is classified as not receiving income related benefits.

2 FRS Specification

For each adult

Code Condition

- 1 From ADULT table
 If INIRBEN \geq 0
- 2 From ADULT table

--

If INIRBEN=0

-2 Unable to derive due to missing values - There should be no missing values for INIRBEN

 --

HDBORR

Purpose :To create a variable for use in hotdecking which shows the range in which BORRAMT falls
 Created :19 June 1996
 Database Table : MORTGAGE
 Minimum Value : 0
 Maximum Value : 14
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : EP - 12 August 1998 - No initial V34 update needed
 : SB - 1 September 1999 - Security completed, no other changes for V35
 Issue date : 1 September 1999

1 Definition

This variable is coded as

- 1 BORRAMT 0 to 9999.99
- 2 BORRAMT 10000 to 19999.99
- 3 BORRAMT 20000 to 29999.99
- 4 BORRAMT 30000 to 39999.99
- 5 BORRAMT 40000 to 49999.99
- 6 BORRAMT 50000 to 59999.99
- 7 BORRAMT 60000 to 69999.99
- 8 BORRAMT 70000 to 79999.99
- 9 BORRAMT 80000 to 89999.99
- 10 BORRAMT 90000 to 99999.99
- 11 BORRAMT 100000 to 124999.99
- 12 BORRAMT 125000 to 149999.99
- 13 BORRAMT 150000 to 199999.99

 --

- 14 BORRAMT 200000 and over
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

HDBORR is derived from the BORRAMT variable in the MORTGAGE table. It shows the range in which the amount of the mortgage falls, and is used to specify classes for hotdecking.

2 FRS Specification

For each mortgage

<u>Code</u>	<u>Condition</u>
1	From MORTGAGE table If BORRAMT <=9999.99
2	From MORTGAGE table If BORRAMT >=10000 and BORRAMT <=19999.99
3	From MORTGAGE table If BORRAMT >=20000 and BORRAMT <=29999.99
4	From MORTGAGE table If BORRAMT >=30000 and BORRAMT <=39999.99
5	From MORTGAGE table If BORRAMT >=40000 and BORRAMT <=49999.99
6	From MORTGAGE table If BORRAMT >=50000 and BORRAMT <=59999.99
7	From MORTGAGE table If BORRAMT >=60000 and BORRAMT <=69999.99
8	From MORTGAGE table If BORRAMT >=70000 and BORRAMT <=79999.99

--

- 9 From MORTGAGE table
If BORRAMT \geq 80000 and BORRAMT \leq 89999.99
- 10 From MORTGAGE table
If BORRAMT \geq 90000 and BORRAMT \leq 99999.99
- 11 From MORTGAGE table
If BORRAMT \geq 100000 and BORRAMT \leq 124999.99
- 12 From MORTGAGE table
If BORRAMT \geq 125000 and BORRAMT \leq 149999.99
- 13 From MORTGAGE table
If BORRAMT \geq 150000 and BORRAMT \leq 199999.99
- 14 From MORTGAGE table
If BORRAMT \geq 200000
- 1 Not applicable to this case - where there is no mortgage
- 2 Unable to derive due to missing values

--

HDINDINC, HDBUINC, HDHHINC, INC

Purpose : To create a variable for use in hot-decking which puts the total amount of income a person, benefit unit or household receives into bands.
Created : 3 February 2000
Database Table : ADULT, BENUNIT, HOUSEHOLD
Minimum Value : 1
Maximum Value : 13 (INC is an amount variable no upper or lower limit)
Units : Integers (except INC which is an amount variable)
Validations :
Related Variables : INC
 Children :
 Parents :
Core variable/user : FRS (hot-decking)
Amendments :
Issue date : 10 February 2000

1 Definition

These variables band the INC variable on a household, benefit unit and person level as follows

- 1 Less than £0
- 2 £0 to £50
- 3 £50 to £100
- 4 £100 to £150
- 5 £150 to £200
- 6 £200 to £250
- 7 £250 to £350
- 8 £350 to £450
- 9 £450 to £600
- 10 £600 to £800
- 11 £800 to £1000
- 12 £1000 to £2000
- 13 £2000 and over

 --

2 FRS Specification

For each adult

Code Condition

INC From the ADULT table it takes the code of the following DVs INEARNs, SEINCAM2, ININV, INRPINC, INPENINC, INDISBEN, INOTHBEN and INRINC. These are added together in the same way as INDINC except for the following changes:

Merge all individual datasets

Replace any INEARNs = .D with INEARNs = 0

If SE1, SE2 or INTDATE are missing then set to 0

If any other missing values then set SEINCAM2 to 0

If WEEKHOL or HBENAMT missing then set XINOTHBEN to 0

If CTREBAMT (council tax rebate) is missing or CTREBPD is missing or one off then don't change XINOTHBEN

If BENAMT missing then add 0 to temporary variable (XINDISBE, XINRPINC, XINOTHB2)

If TOTGRANT = .D then set INRINC to 0

If any missing values for account interest then set ININV to 0

HDINDINC For the person level the INC variable on the ADULT table is banded into the above bands

HDBUINC Get all occurrences of INC within each Benefit unit.
 Sum up to get totals for each Benefit unit
 Band totals into same bands as above

HDHHINC Get all occurrences of INC within each Household.
 Sum up to get totals for each Household
 Band totals into same bands as above

There should be no missing values

 --

HDPAY, HDGRWAG, HDUNETT, HDUGROSS, HDQHRS

Purpose :To create a variable for use in hotdecking which shows the range in which PAYAMT falls
 Created :19 June 1996
 Database Table : JOB
 Minimum Value : 0
 Maximum Value : 15
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : EP - 12 August 1998 - Addition of HDGRWAG, HDQHRS, HDQHRSSE, HDUGROSS, HDUNETT
 SG - 4 January, 1999 - Changes in hours worked variables
 EP - 9 April 1999 - Remove HDQHRSSE as no longer separate questions for hours worked as self-employed
 : SB - 31 August 1999 - Security completed, no other changes required for V35
 Issue date :31 August 1999

1 Definition

HDPAY is coded as

- 1 PAYAMT 0 to 49.99
- 2 PAYAMT 50 to 99.99
- 3 PAYAMT 100 to 149.99
- 4 PAYAMT 150 to 199.99
- 5 PAYAMT 200 to 249.99
- 6 PAYAMT 250 to 299.99
- 7 PAYAMT 300 to 349.99
- 8 PAYAMT 350 to 399.99
- 9 PAYAMT 400 to 499.99
- 10 PAYAMT 500 to 599.99

 --

- 11 PAYAMT 600 to 699.99
- 12 PAYAMT 700 to 799.99
- 13 PAYAMT 800 to 899.99
- 14 PAYAMT 900 to 999.99
- 15 PAYAMT 1000 and over
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

HDPAY is derived from the PAYAMT variable in the JOB table. It shows the range in which take-home pay falls, and is used to specify classes for hotdecking.

The following DVs also use the above coding:

HDGRWAG is derived from the GRWAGE variable in the JOB table. It shows the gross wage band.

HDUNETT is derived from the UNETT variable in the JOB table. It shows the nett pay band.

HDUGROSS is derived from the UGROSS variable in the JOB table. It shows the gross pay band.

HDQHRS is coded as:

- 1 hours worked ~~QHRS~~ 0 to < 16
- 2 hours worked ~~QHRS~~ 16 to < 30
- 3 hours worked ~~QHRS~~ 30 to < 40
- 4 hours worked ~~QHRS~~ 40 to < 50
- 5 hours worked ~~QHRS~~ 50 to < 60
- 6 hours worked ~~QHRS~~ >= 60

HDQHRS is derived from the hours worked ~~QHRS~~ variables in the JOB table. It shows the weekly hours worked band.

2 FRS Specification

For each job

 --

<u>Code</u>	<u>Condition</u>
1	From JOB table If PAYAMT <=49.99
2	From JOB table If PAYAMT >=50 and PAYAMT <=99.99
3	From JOB table If PAYAMT >=100 and PAYAMT <=149.99
4	From JOB table If PAYAMT >=150 and PAYAMT <=199.99
5	From JOB table If PAYAMT >=200 and PAYAMT <=249.99
6	From JOB table If PAYAMT >=250 and PAYAMT <=299.99
7	From JOB table If PAYAMT >=300 and PAYAMT <=349.99
8	From JOB table If PAYAMT >=350 and PAYAMT <=399.99
9	From JOB table If PAYAMT >=400 and PAYAMT <=499.99
10	From JOB table If PAYAMT >=500 and PAYAMT <=599.99
11	From JOB table If PAYAMT >=600 and PAYAMT <=699.99
12	From JOB table

 --

- If PAYAMT >= 700 and PAYAMT <= 799.99

13 From JOB table

If PAYAMT >= 800 and PAYAMT <= 899.99

14 From JOB table

If PAYAMT >= 900 and PAYAMT <= 999.99

15 From JOB table

If PAYAMT >= 1000

-1 Not applicable to this case

-2 Unable to derive due to missing values

HDGRWAG, HDUNETT AND HDUGROSS are derived as above.

For HDQHRS:

**If EVEROT=1 then QHRS=TOTUS1
 Else if EVEROT=2 then QHRS=USUHR**

- 1 From JOB table

If QHRS < 16
- 2 From JOB table

If QHRS >= 16 and QHRS < 30
- 3 From JOB table

If QHRS >= 30 and QHRS < 40
- 4 From JOB table

If QHRS >= 40 and QHRS < 50
- 5 From JOB table

If QHRS >= 50 and QHRS < 60
- 6 From JOB table

If QHRS >= 60

 --

HDPEN

Purpose : To create a variable for use in hot-decking which shows the range in which
 PENPAY falls
 Created : 19 June 1996
 Database Table : PENSION
 Minimum Value : 0
 Maximum Value : 11
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : SG - 17 February 1998 - no updates required for V33
 : EP - 12 August 1998 - no initial V34 update needed
 : JC - 1 September 1999 - Security completed, no other changes for V35
 Issue : 1 September 1999

1 Definition

This variable is coded as

- 1 PENPAY 0 to 14.99
- 2 PENPAY 15 to 29.99
- 3 PENPAY 30 to 49.99
- 4 PENPAY 50 to 69.99
- 5 PENPAY 70 to 99.99
- 6 PENPAY 100 to 149.99
- 7 PENPAY 150 to 199.99
- 8 PENPAY 200 to 299.99
- 9 PENPAY 300 to 399.99
- 10 PENPAY 400 to 499.99
- 11 PENPAY 500 and over
- 1 Not applicable to this case

 --

-2 Unable to derive due to missing values

HDPEN is derived from the PENPAY variable in the PENSION table. It shows the range in which the last payment of a respondent's pension falls, and is used to specify classes for hotdecking.

2 FRS Specification

For each pension

<u>Code</u>	<u>Condition</u>
1	From PENSION table If PENPAY<=14.99
2	From PENSION table If PENPAY>=15 and PENPAY<=29.99
3	From PENSION table If PENPAY>=30 and PENPAY<=49.99
4	From PENSION table If PENPAY>=50 and PENPAY<=69.99
5	From PENSION table If PENPAY>=70 and PENPAY<=99.99
6	From PENSION table If PENPAY>=100 and PENPAY<=149.99
7	From PENSION table If PENPAY>=150 and PENPAY<=199.99
8	From PENSION table If PENPAY>=200 and PENPAY<=299.99
9	From PENSION table If PENPAY>=300 and PENPAY<=399.99

--

- 10 From PENSION table
 If $PENPAY \geq 400$ and $PENPAY \leq 499.99$
- 11 From PENSION table
 If $PENPAY \geq 500$
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

HDPROFIT

Purpose : Creates banded profit variable used in hotdecking OWNAMT
Created : 7 September 1998
Database Table : JOB
Minimum Value : 1
Maximum Value : 15
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (hot-decking)
Amendments :JC -1 September 1999 - Security completed, no other changes for V35
Issue date : 1September 1999

1 Definition

HDPROF creates profit bands which are coded as follows

- 1 PROFIT £0 to < £50
- 2 PROFIT £50 to < £100
- 3 PROFIT £100 to < £150
- 4 PROFIT £150 to < £200
- 5 PROFIT £200 to < £250
- 6 PROFIT £250 to < £300
- 7 PROFIT £300 to < £350
- 8 PROFIT £350 to < £400
- 9 PROFIT £400 to < £500
- 10 PROFIT £500 to < £600
- 11 PROFIT £600 to < £700
- 12 PROFIT £700 to < £800
- 13 PROFIT £800 to < £900
- 14 PROFIT £900 to < £1000
- 15 PROFIT £1000 and above
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

2 FRS specification

<u>Code</u>	<u>Condition</u>
1	From JOB table PROFIT=0 and PROFIT<50
2	From JOB table PROFIT=50 and PROFIT<100
3	From JOB table PROFIT=100 and PROFIT<150
4	From JOB table PROFIT=150 and PROFIT<200
5	From JOB table PROFIT=200 and PROFIT<250
6	From JOB table PROFIT=250 and PROFIT<300
7	From JOB table PROFIT=300 and PROFIT<350
8	From JOB table PROFIT=350 and PROFIT<400
9	From JOB table PROFIT=400 and PROFIT<500
10	From JOB table PROFIT=500 and PROFIT<600
11	From JOB table PROFIT=600 and PROFIT<700
12	From JOB table PROFIT=700 and PROFIT<800
13	From JOB table PROFIT=800 and PROFIT<900
14	From JOB table PROFIT=900 and PROFIT<1000

- 15 From JOB table
PROFIT \geq 1000
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

 --

HDPURC

Purpose : To create a variable for use in hotdecking which shows the range in which PURCAMT falls
 Created : 19 June 1996
 Database Table : ~~MORTGAGE~~ **OWNER**
 Minimum Value : 0
 Maximum Value : 14
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : EP - 12 August 1998 - No initial V34 update needed
 : EP - 16 November 1998 - Change from MORTGAGE table to OWNER table
 : JC - 1 September 1999 - Security completed no other changes for V35
 Issue date : 1 September 99

1 Definition

This variable is coded as

- 1 PURCAMT 0 to 9999.99
- 2 PURCAMT 10000 to 19999.99
- 3 PURCAMT 20000 to 29999.99
- 4 PURCAMT 30000 to 39999.99
- 5 PURCAMT 40000 to 49999.99
- 6 PURCAMT 50000 to 59999.99
- 7 PURCAMT 60000 to 69999.99
- 8 PURCAMT 70000 to 79999.99
- 9 PURCAMT 80000 to 89999.99
- 10 PURCAMT 90000 to 99999.99
- 11 PURCAMT 100000 to 124999.99
- 12 PURCAMT 125000 to 149999.99

 --

- 13 PURCAMT 150000 to 199999.99
- 14 PURCAMT 200000 and over
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

HDBORR is derived from the PURCAMT variable in the ~~MORTGAGE~~ **OWNER** table. It shows the range in which the amount of the purchase price of the house/flat falls, and is used to specify classes for hotdecking.

2 FRS Specification

For each mortgage

Code Condition

- 1 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT <=9999.99
- 2 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT >=10000 and PURCAMT <=19999.99
- 3 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT >=20000 and PURCAMT <=29999.99
- 4 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT >=30000 and PURCAMT <=39999.99
- 5 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT >=40000 and PURCAMT <=49999.99
- 6 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT >=50000 and PURCAMT <=59999.99
- 7 From ~~MORTGAGE~~ **OWNER** table
 If PURCAMT >=60000 and PURCAMT <=69999.99
- 8 From ~~MORTGAGE~~ **OWNER** table

 --

- If PURCAMT >= 70000 and PURCAMT <= 79999.99

9 From ~~MORTGAGE~~ **MORTGAGE OWNER** table

If PURCAMT >= 80000 and PURCAMT <= 89999.99

10 From ~~MORTGAGE~~ **MORTGAGE OWNER** table

If PURCAMT >= 90000 and PURCAMT <= 99999.99

11 From ~~MORTGAGE~~ **MORTGAGE OWNER** table

If PURCAMT >= 100000 and PURCAMT <= 124999.99

12 From ~~MORTGAGE~~ **MORTGAGE OWNER** table

If PURCAMT >= 125000 and PURCAMT <= 149999.99

13 From ~~MORTGAGE~~ **MORTGAGE OWNER** table

If PURCAMT >= 150000 and PURCAMT <= 199999.99

14 From ~~MORTGAGE~~ **MORTGAGE OWNER** table

If PURCAMT >= 200000

-1 Not applicable to this case - where there is no mortgage

-2 Unable to derive due to missing values

 --

HDRENT

Purpose : To create a variable for use in hot-decking which shows the rent range in which a household falls
 Created : 22 September 1997
 Database Table : RENTER
 Minimum Value : 1
 Maximum Value : 5
 Units : Integers
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (hot-decking)
 Amendments : EP - 12 August 1998 - No initial V34 update needed
 : JC - 1 September 1999 - Security completed, no other changes for V35
 Issue date :17 April 2003

1 Definition

This variable is coded as

- 1 0
- 2 >0 and <40
- 3 >=40 and <80
- 4 >=80 and <120
- 5 >=120
- 2 Unable to derive due to missing values

HDRENT is derived from the RENT variable in the RENTER table. It shows the rent range in which a household falls, and is used to specify classes for hotdecking.

2 FRS Specification

For each adult

Code Condition

- 1 From RENTER table
- If RENT=0

--

- 2 From RENTER table
 If RENT>0 and RENT<40
- 3 From RENTER table
 If RENT>=40 and RENT<80
- 4 From RENTER table
 If RENT>=80 and RENT<120
- 5 From RENTER table
 If RENT>=120

- 2 Unable to derive due to missing values

HDSEINC

Purpose : Creates banded profit variable used in hot-decking NIDAMT and TAXDAMT
Created : 7 September 1998
Database Table : JOB
Minimum Value : 1
Maximum Value : 15
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (hot-decking)
Amendments : JC -1 September 1999- Security completed no other changes for V35
Issue date : 1 September 1999

1 Definition

HDSEINC creates profit bands which are coded as follows

- 1 PROFIT £0 to < £50
- 2 PROFIT £50 to < £100
- 3 PROFIT £100 to < £150
- 4 PROFIT £150 to < £200
- 5 PROFIT £200 to < £250
- 6 PROFIT £250 to < £300
- 7 PROFIT £300 to < £350
- 8 PROFIT £350 to < £400
- 9 PROFIT £400 to < £500
- 10 PROFIT £500 to < £600
- 11 PROFIT £600 to < £700
- 12 PROFIT £700 to < £800
- 13 PROFIT £800 to < £900
- 14 PROFIT £900 to < £1000
- 15 PROFIT £1000 and above
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

2 FRS Specification

<u>Code</u>	<u>Condition</u>
1	From JOB table PROFIT=0 and PROFIT<50
2	From JOB table PROFIT=50 and PROFIT<100
3	From JOB table PROFIT=100 and PROFIT<150
4	From JOB table PROFIT=150 and PROFIT<200
5	From JOB table PROFIT=200 and PROFIT<250
6	From JOB table PROFIT=250 and PROFIT<300
7	From JOB table PROFIT=300 and PROFIT<350
8	From JOB table PROFIT=350 and PROFIT<400
9	From JOB table PROFIT=400 and PROFIT<500
10	From JOB table PROFIT=500 and PROFIT<600
11	From JOB table PROFIT=600 and PROFIT<700
12	From JOB table PROFIT=700 and PROFIT<800
13	From JOB table PROFIT=800 and PROFIT<900
14	From JOB table

PROFIT=900 and PROFIT<1000

15 From JOB table

PROFIT>=1000

-1 Not applicable to this case

-2 Unable to derive due to missing values

 --

HHCOMP

Purpose : To indicate household composition for use in the FRS publication.
 Created : 29 January 1996
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 21
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Issue date : 17 April 2003
 Amendments : JS - 14 March to collapse some of the households with children categories
 : 4 June 1996 - No initial amendments needed for V32 update
 : 27 June 1997 - No initial amendments needed for V33 update
 : EP - 12 August 1998 - V34 update
 : EP - 11 January 1999 - set up count variables to avoid use of ADULTH and
 DEPCHLDH
 : JC - 20 September 1999 - Security completed, no other changes for V35

HHCOMP categorises households on the basis of whether *dependent* children are present, number of adults and whether adults are over or under state retirement age. Some categories may yield very few cases. These will be combined in the final publication.

Using state retirement age is consistent with FAMTYPBU categories (although these only relate to whether the head is under/over pension age) but not ECSTATBU, which considers whether one or more adult is aged over 60.

As well as the DV, HHCOMP tables include a number of subtotals relating to pensioners, the sick and disabled and the unemployed (including work-related government training schemes). Definitions of cases to be included in these categories are also given below. These categories may be overlapping.

1 Definition

HHCOMP is derived by looking at the **temporary count** variables ~~DEPCHLDH~~ **CADULT** (number of dependent children in HH) and ~~ADULTH~~ **CCHILD** (number of adults in HH), and variables PERSON (person number), AGE and SEX.

ADULTH and DEPCHLDH not used in V34 due to incorrect values of these variables

Additional subtotals used in tables are identified by age and EMPSTATB, and responses to the questions HEALTH, HPROB, RSTRCT, LAREG and JCREG.

The definition of sick and disabled is consistent with the proposed follow up study of the disabled, **except** that it excludes receipt of disability benefits. This will help ensure that figures are comparable over time (rules for benefits may change).

--

The coding for HHCOMP is:

- 1 One adult, no children over pension age
- 2 One adult, no children, under pension age
- 3 Two adults, no children, both over pension age
- 4 Two adults, no children, one over pension age
- 5 Two adults, no children, both under pension age
- 6 Three or more adults, no children
- 7 One adult, one child
- 8 One adult, two children
- 9 One adult, three or more children
- 10 Two adults, one child
- 11 Two adults, two children
- 12 Two adults, three or more children
- 13 Three or more adults, one child
- 14 Three or more adults, two children
- 15 Three or more adults, three or more children

The sub totals required are:

- A Total households without children
- B Total households with children
- C Total households with one or more adults over pension age including the head
- D Total households with one or more adults over pension age excluding the head
- E Households with one or more sick/disabled adults under pension age including the head
- F Households with one or more sick/disabled adults under pension age excluding the head
- G Households with one or more unemployed adults under pension age including the head
- H Households with one or more unemployed adults under pension age excluding the head

 --

2 FRS Specification

Households without children, codes 1-6 all where ~~DEPCHLDH~~ CCHILD =0

- 1 ~~ADULTH~~ CADULT =1 and (SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)
- 2 ~~ADULTH~~ CADULT =1 and (SEX=1 and AGE<65) or (SEX=2 and AGE<60)
- 3 ~~ADULTH~~ CADULT =2 and (PERSON=1 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
 and (PERSON=2 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
- 4 ~~ADULTH~~ CADULT =2 and (PERSON=1 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
 and (PERSON=2 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
 or (PERSON=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
 and (PERSON=2 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
- 5 ~~ADULTH~~ CADULT =2 and (PERSON=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
 and (PERSON=2 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
- 6 ~~ADULTH~~ CADULT >=3
- 7 ~~ADULTH~~ CADULT =1 and ~~DEPCHLDH~~ CCHILD =1
- 8 ~~ADULTH~~ CADULT =1 and ~~DEPCHLDH~~ CCHILD =2
- 9 ~~ADULTH~~ CADULT =1 and ~~DEPCHLDH~~ CCHILD >=3
- 10 ~~ADULTH~~ CADULT =2 and ~~DEPCHLDH~~ CCHILD =1
- 11 ~~ADULTH~~ CADULT =2 and ~~DEPCHLDH~~ CCHILD =2
- 12 ~~ADULTH~~ CADULT =2 and ~~DEPCHLDH~~ CCHILD >=3
- 13 ~~ADULTH~~ CADULT >=3 and ~~DEPCHLDH~~ CCHILD =1
- 14 ~~ADULTH~~ CADULT >=3 and ~~DEPCHLDH~~ CCHILD =2
- 15 ~~ADULTH~~ CADULT >=3 and ~~DEPCHLDH~~ CCHILD >=3

 --

The subtotals are calculated as:

- A Cases where HHCOMP=1-6
- B Cases where HHCOMP=7-21
- C Cases where PERSON 1 is over pension age:
 (SEX=1 and AGE>=65) or (SEX=2 and AGE>=60))
- D Cases where PERSON 1 is under pension age and at least one other adult is over pension age
- E Cases where PERSON 1 is under pension age and has a long standing illness or restricted in what they do ie:
 PERSON=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60))
 and ((HEALTH=1 and HPROB=1) or (RSTRCT=1 or RSTRCT=2) or (LAREG=1) or
 (JCREG=1))
- F Where PERSON 1 is not restricted in what they do, but that one or more other adults in the household meet the criteria:
 (PERSON=1 and (HEALTH=2 and RSTRCT=3))
 and for all other adults in HH at least one is:
 ((SEX=1 and AGE<65) or (SEX=2 and AGE<60))
 and ((HEALTH=1 and HPROB=1) or (RSTRCT=1 or RSTRCT=2) or (LAREG=1)
 or (JCREG=1))
- Note: HPROB, LAREG and JCREG are only asked if HEALTH=1, so (HEALTH=2 and RSTRCT=3) should be sufficient to identify adults who are **not** sick/disabled*
- G Where for PERSON 1, EMPSTATB=7 (unemployed) or EMPSTATB=8 (govt training scheme)
- H Where for PERSON 1, EMPSTATB<>7 and EMPSTATB<>8 and at least one other adult in the household has (EMPSTATB=7 or EMPSTAT=8)

 --

HHCOMPS

Purpose : To indicate household composition for use in the FRS publication, same as HHCAMP except additional split for sex.
 Created : 6 May 1998
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 21
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Issue date : 17 April 2003
 Amendments : EP - 12 August 1998 - V34 update
 : EP – 11 January 1999 - set up count variables to avoid use of ADULTH and DEPCHLDH
 : JC – 20 September 1999 – Security completed, no other changes for V35

HHCAMP categorises households on the basis of whether *dependent* children are present, number of adults and whether adults are over or under state retirement age and sex for single adults without children. Some categories may yield very few cases. These will be combined in the final publication.

Using state retirement age is consistent with FAMTYPBU categories (although these only relate to whether the head is under/over pension age) but not ECSTATBU, which considers whether one or more adult is aged over 60.

As well as the DV, HHCAMP tables include a number of subtotals relating to pensioners, the sick and disabled and the unemployed (including work-related government training schemes). Definitions of cases to be included in these categories are also given below. These categories may be overlapping.

1 Definition

HHCAMP is derived by looking at **temporary count** variables ~~DEPCHLDH~~ **CADULT** (number of dependent children in HH), ~~ADULTH~~ **CCHILD** (number of adults in HH), and variables PERSON (person number), AGE and SEX.

Additional subtotals used in tables are identified by age, EMPSTATB and responses to questions HEALTH, HPROB, RSTRCT, LAREG and JCREG.

The definition of sick and disabled is consistent with the proposed follow up study of the disabled, except that it excludes receipt of disability benefits. This will help ensure that figures are comparable over time (rules for benefits may change).

The coding for HHCAMP is:

--

- 1 One adult male, no children over pension age
- 2 One adult female, no children over pension age
- 3 One adult male, no children, under pension age
- 4 One adult female, no children, under pension age
- 5 Two adults, no children, both over pension age
- 6 Two adults, no children, one over pension age
- 7 Two adults, no children, both under pension age
- 8 Three or more adults, no children
- 9 One adult, one child
- 10 One adult, two children
- 11 One adult, three or more children
- 12 Two adults, one child
- 13 Two adults, two children
- 14 Two adults, three or more children
- 15 Three or more adults, one child
- 16 Three or more adults, two children
- 17 Three or more adults, three or more children

The sub totals required are:

- A Total households without children
- B Total households with children
- C Total households with one or more adults over pension age including the head
- D Total households with one or more adults over pension age excluding the head
- E Households with one or more sick/disabled adults under pension age including the head
- F Households with one or more sick/disabled adults under pension age excluding the head
- G Households with one or more unemployed adults under pension age including the head
- H Households with one or more unemployed adults under pension age excluding the head

--

2 FRS Specification

Households without children, codes 1-8 all where ~~DEPCHLDH~~ CCHILD=0

- 1 ~~ADULTH~~ CADULT =1 and (SEX=1 and AGE>=65)
- 2 ~~ADULTH~~ CADULT =1 and (SEX=2 and AGE>=60)
- 3 ~~ADULTH~~ CADULT =1 and (SEX=1 and AGE<65)
- 4 ~~ADULTH~~ CADULT =1 and (SEX=2 and AGE<60)
- 5 ~~ADULTH~~ CADULT =2 and (PERSON=1 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
and (PERSON=2 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
- 6 ~~ADULTH~~ CADULT =2 and (PERSON=1 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
and (PERSON=2 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
or (PERSON=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
and (PERSON=2 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)))
- 7 ~~ADULTH~~ CADULT =2 and (PERSON=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
and (PERSON=2 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)))
- 8 ~~ADULTH~~ CADULT >=3
- 9 ~~ADULTH~~ CADULT =1 and ~~DEPCHLDH~~ CCHILD =1
- 10 ~~ADULTH~~ CADULT =1 and ~~DEPCHLDH~~ CCHILD =2
- 11 ~~ADULTH~~ CADULT =1 and ~~DEPCHLDH~~ CCHILD >=3
- 12 ~~ADULTH~~ CADULT =2 and ~~DEPCHLDH~~ CCHILD =1
- 13 ~~ADULTH~~ CADULT =2 and ~~DEPCHLDH~~ CCHILD =2
- 14 ~~ADULTH~~ CADULT =2 and ~~DEPCHLDH~~ CCHILD >=3
- 15 ~~ADULTH~~ CADULT >=3 and ~~DEPCHLDH~~ CCHILD =1
- 16 ~~ADULTH~~ CADULT >=3 and ~~DEPCHLDH~~ CCHILD =2
- 17 ~~ADULTH~~ CADULT >=3 and ~~DEPCHLDH~~ CCHILD >=3

--

The subtotals are calculated as:

- A Cases where HHCOMP=1-8
- B Cases where HHCOMP=9-17
- C Cases where PERSON 1 is over pension age:
(SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)
- D Cases where PERSON 1 is under pension age and at least one other adult is over pension age
- E Cases where PERSON 1 is under pension age and has a long standing illness or restricted in what they do ie:

PERSON=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60))
and ((HEALTH=1 and HPROB=1) or (RSTRCT=1 or RSTRCT=2) or (LAREG=1) or
(JCREG=1))
- F Where PERSON 1 is not restricted in what they do, but that one or more other adults in the household meet the criteria:

(PERSON=1 and (HEALTH=2 and RSTRCT=3))
and for all other adults in HH at least one is:
((SEX=1 and AGE<65) or (SEX=2 and AGE<60))
and ((HEALTH=1 and HPROB=1) or (RSTRCT=1 or RSTRCT=2) or (LAREG=1)
or (JCREG=1))
- Note: HPROB, LAREG and JCREG are only asked if HEALTH=1, so (HEALTH=2 and RSTRCT=3) should be sufficient to identify adults who are **not** sick/disabled*
- G Where for PERSON 1, EMPSTATB=7 (unemployed) or EMPSTATB=8 (govt training scheme)
- H Where for PERSON 1, EMPSTATB<>7 and EMPSTATB<>8 and at least one other adult in the household has (EMPSTATB=7 or EMPSTAT=8)

HHAGEGRP, HHETHGRP, HHAGEGR2

Purpose : To show the Age/Ethnicity of the Head of the BU
Created : 7 September 1998
Database Table : HOUSEHOL
Minimum Value : 1
Maximum Value : 5 (HHETHGRP), 15 (HHAGEGRP), **9 (HHAGEGR2)**
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (hot-decking)
Amendments : EP - 22 February 1999 – Addition of new publication DV –
 HHAGEGR2
 : EP – 10 March 1999 – Harmonisation of age bands
 : JC -6 September 1999 -Security completed, no other changes for V35
Issue date : 7 September 1998

1 Definition

HHAGEGRP and HHETHGRP are hot-deck DVs used by hot-deck views to define categories for the view. They are coded as:

HHAGEGRP

- 1 Age 16-19
- 2 Age 20-24
- 3 Age 25-29
- 4 Age 30-34
- 5 Age 35-39
- 6 Age 40-44
- 7 Age 45-49
- 8 Age 50-54
- 9 Age 55-59
- 10 Age 60-64
- 11 Age 65-69
- 12 Age 70-74
- 13 Age 75-79
- 14 Age 80-84
- 15 Age 85 and above

HHETHGRP

-
- 1 White
 - 2 Black - Caribbean, African, or neither Caribbean or African
 - 3 Indian
 - 4 Pakistani or Bangladeshi
 - 5 Chinese or Other

HHAGEGR2 is a publication DV and is coded as follows:

- 1 **Age 16 to 24**
- 2 **Age 25 to 34**
- 3 **Age 35 to 44**
- 4 **Age 45 to 54**
- 5 **Age 55 to 59**
- 6 **Age 60 to 64**
- 7 **Age 65 to 74**
- 8 **Age 75 to 84**
- 9 **Age 85 or over**

2 **FRS Specification**

HHAGEGRP

Code Condition

- 1 From HOUSEHOL table
AGE>=16 and AGE<=19
- 2 From HOUSEHOL table
AGE>=20 and AGE<=24
- 3 From HOUSEHOL table
AGE>=25 and AGE<=29
- 4 From HOUSEHOL table
AGE>=30 and AGE<=34
- 5 From HOUSEHOL table
AGE>=35 and AGE<=39

-
- 6 From HOUSEHOL table
AGE \geq 40 and AGE \leq 44
- 7 From HOUSEHOL table
AGE \geq 45 and AGE \leq 49
- 8 From HOUSEHOL table
AGE \geq 50 and AGE \leq 54
- 9 From HOUSEHOL table
AGE \geq 55 and AGE \leq 59
- 10 From HOUSEHOL table
AGE \geq 60 and AGE \leq 64
- 11 From HOUSEHOL table
AGE \geq 65 and AGE \leq 69
- 12 From HOUSEHOL table
AGE \geq 70 and AGE \leq 74
- 13 From HOUSEHOL table
AGE \geq 75 and AGE \leq 79
- 14 From HOUSEHOL table
AGE \geq 80 and AGE \leq 84
- 15 From HOUSEHOL table
AGE \geq 85

HHETHGRP

- 1 From HOUSEHOL table
ETHGRP=1
- 2 From HOUSEHOL table
ETHGRP=2,3 or 4
- 3 From HOUSEHOL table
ETHGRP=5
- 4 From HOUSEHOL table
ETHGRP=6 or 7

5 From HOUSEHOL table

ETHGRP=8 or 9

HHAGEGR2

From HOUSEHOL table get HHAGEGRP

- 1 If HHAGEGRP in (1,2)
- 2 If HHAGEGRP in (3,4)
- 3 If HHAGEGRP in (5,6)
- 4 If HHAGEGRP in (7,8)
- 5 If HHAGEGRP in (9)
- 6 If HHAGEGRP in (10)
- 7 If HHAGEGRP in (11,12)
- 8 If HHAGEGRP in (13,14)
- 9 If HHAGEGRP in (15)

 --

HHHDBEN

Purpose :To create a variable for use in hot-decking which shows whether any income related benefits are received by an individual

Created :19 June 1996

Database Table : HOUSEHOLD

Minimum Value : 1

Maximum Value : 2

Units : Integers

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : FRS (hot-decking)

Amendments : SG - 27th June - No updates required for V33
 : EP - 7 September 1998 - Addition of BUHDBEN and HHHDBEN
 : SB - 1 September 1999 – Spilt SAS code up into adult, benunit and household levels
 - Security completed

Issue date : 1 September 1999

1 Definition

This variable is coded as

- 1 Income related benefits received by individual
- 2 No income related benefits received by individual
- 2 Unable to derive due to missing values

HHHDBEN is a derived variable in the HOUSEHOL table. This shows the total amount of income received each week by a household from income related benefits. If INIRBEN is greater than zero, a household is classified as receiving income related benefits. If INIRBEN is zero, a household is classified as not receiving income related benefits.

2 FRS Specification

For each adult

Code Condition

- 1 From ADULT table
- If INIRBEN \geq 0

--

2 From ADULT table

If INIRBEN=0

-2 Unable to derive due to missing values - There should be no missing values for HHHDBEN

 --

HHINC, HEARNS, HSEINC, HHINV, HHRPINC, HPENINC, HHDISBEN, HBENINC, HHOTHBEN, HHRINC, HOTHINC

Purpose : To show the total amount of income received by each household for use in the FRS publication.
 Created : 2 February 1996
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents : INDINC, BUINC (see specs for these variables for what goes into HHINC)
 Core variable/user : FRS (publication)
 Amendments : VE - 4 June 1996 - No initial amendments needed for V32 update
 : SG - 27 June 1997 - No initial amendments needed for V33 update
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 24 September 1999 - Security Completed, no other changes for V35
 Issued : 24 September 1999

1 Definition

This variable is coded as

HHINC The total amount of income received each week by all members of the household.
 0 No income is received by the household
 -2 Unable to derive due to missing values

This specification, also sets up the component DVs used in the publication. These are:

HEARNS earned income
 HSEINC self-employment income
 HHINV investment income
 HHRPINC retirement pension plus any income support
 HPENINC other pension income
 HHDISBEN disability benefits
 HHOTHBEN other benefits
 HHRINC remaining income

In addition, there are two other OPCS variables included in this specification

HBENINC Household benefit income
 HOTHINC Other household income

 --

2 FRS Specification

For each household

<u>Code</u>	<u>Condition</u>
HHINC	From BENUNIT table get BUEARNS, BSEINC, BUINV, BURPINC, BPENINC, BUDISBEN, BUOTHBEN, BURINC and BUINC for all benefit units in the household HHINC equals total occurrences of BUINC
-2	If any components are missing

Component variables are calculated as follows:

- HEARNNS equals total occurrences of BUEARNS
- HSEINC equals total occurrences of BSEINC
- HHINV equals total occurrences of BUINV
- HHRPINC equals total occurrences of BUPRINC
- HPENINC equals total occurrences of BPENINC
- HHDISBEN equals total occurrences of BUDISBEN
- HHOTHBEN equals total occurrences of BUOTHBEN
- HHRINC equals total occurrences of BURINC

- HBENINC = HHRPINC+HHDISBEN+HHOTHBEN
- HOTHINC=HHINV+HHRINC

components are set to -2 if any variables are missing

HHINCBND

Purpose :
 Created : 22 February 1999
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 8
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents : HHINC
 Core variable/user : FRS (publication)
 Amendments : SB - 9 November 1999 - Security completed, no other changes for V35
 Issue date : 22 February 1999

1 Definition

HHINCBND is derived for the FRS publication and it is a categorical breakdown of household income. It is derived from HHINC.

HHINCBND is coded as follows:

- 1 Under £100 a week
- 2 £100 and less than £200
- 3 £200 and less than £300
- 4 £300 and less than £400
- 5 £400 and less than £500
- 6 £500 and less than £600
- 7 £600 and less than £700
- 8 £700 and above

2 FRS Specification

For each household

<u>Code</u>	<u>Condition</u>
1	If HHINC < 100 and HHINC not in (.D)
2	If HHINC >= 100 and HHINC < 200
3	If HHINC >= 200 and HHINC < 300
4	If HHINC >= 300 and HHINC < 400
5	If HHINC >= 400 and HHINC < 500
6	If HHINC >= 500 and HHINC < 600
7	If HHINC >= 600 and HHINC < 700

- 8 If HHINC >= 700
- 1 Not applicable to this case – should not happen to this variable
- 2 Unable to derive HHINCBND

HHKIDS

Purpose : Further breakdown of household composition (for publication use)
 Created : 18 February 1999
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 7
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents : HHCOMPS
 Core variable/user : FRS (publication)
 Amendments : SB - 8 November 1999 - Security completed, no other changes for V35

 Issue date : 18 February 1999

1 Definition

HHKIDS is a further breakdown of household composition for publication use. It is derived from the derived variable HHCOMPS.

HHKIDS is coded as follows:

- 1 Household with children, one adult
- 2 Household with children, two adults
- 3 Household with children, three or more adults
- 4 Household without children, one male adult
- 5 Household without children, one female adult
- 6 Household without children, two adults
- 7 Household without children, three or more adults

2 FRS Specification

For each household

<u>Code</u>	<u>Condition</u>
1	If HHCOMPS in (9,10,11) then HHKIDS=1
2	If HHCOMPS in (12,13,14) then HHKIDS=2
3	If HHCOMPS in (15,16,17) then HHKIDS=3
4	If HHCOMPS in (1,3) then HHKIDS=4
5	If HHCOMPS in (2,4) then HHKIDS=5
6	If HHCOMPS in (5,6,7) then HHKIDS=6
7	If HHCOMPS in (8) then HHKIDS=7
-1	Not applicable to this case – should not happen to this variable

-2

Unable to derive HHKIDS

 --

HHRENT/TUHHRENT

Purpose : To show the rent eligible for Housing Benefit paid by a household for accommodation before the deduction of Housing Benefit but after taking off extras such as service charges.

The variable TUHHRENT is exactly the same as HHRENT but excludes any adjustment for rent holidays. This variable should be used by ASD3A only.

Created : 26 January 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children : BURENT, WATSEWRT
 Parents :
 Core variable/user : PSM
 Issue date : 17 April, 2003
 Amendments : VC - 26 April 1993. To divide GRSRENT by benefit unit not by household.
 : VC - 18 May 1993. To produce separate household and benefit unit gross :
 rent variables
 : VC - 9 June 1993. To include rent free weeks.
 : AJG - 21 June 1993. If 100% rebate received, then HHRENT = housing :
 benefit amount.
 : VC - 15 February 1994. Amendments due to version 30 update.
 : VC - 1 March 1994 To exclude period codes 12 and 13.
 : BS - 2 August 1995 To include changes to questionnaire for V31.
 : JS - 21 February 1996 to allow skipped values where variables have been
 imputed and to make CWATAMTD change explicit
 : JS - 27 March - to amend values included in rent for rent holiday cases
 : VE - 10 June 1996 - Amendments for V32 - COMMINC no longer asked,
 WORSINC replaces WATERINC and SEWERINC, section relating to
 CWATAMTD removed
 : VE - 27 August 1996 - To add SCHBAMT for shared household cases as a
 result of discussions with Martin Uglow and Graham Farrant (SCPR). It is
 assumed from this date that SRENTAMT is the figure for rent after HB has
 been deducted (ie a net amount).
 : VE - 28 August 1996 - To deduct service charges for cases in receipt of HB
 where a statement is consulted.
 : VE - 29 November 1996 - To amend so that contributions to rent are only
 added back to current rent when the person paying is not the DSS (all DSS
 payments relate to rent arrears). Also to amend so that the amount can be
 added when RENT=0, ie ACCCHK=-1.
 : VE - 12 December 1996 - to change HBSERAMT to HBSERA01, HBSERA02
 etc.

 --

: VE - 3 February 1997 - to amend so that rent free cases are included to pick up rent paid by outsiders
 : SG - 18 July 1997 - correct check for deducting WATSEWRT from HHRENT - previously missing certain cases
 : SG - 29 December 1997 - V33 updates
 : EP - 12 August 1998 - No initial V34 update needed
 : SB - 11 November 1999 – Security completed, data inputted from constants table after base variables deleted from questionnaire

1 Definition

This variable is coded as

HHRENT This is the total amount of rent eligible for HB paid by a household, before the deduction of any Housing Benefit but after taking off certain expenses such as service charges, council tax etc which are included in the rent.

-1 Not applicable to this case.

-2 Unable to derive because of missing values.

HHRENT is derived from the variables RENT (rent actually paid for household), HBENEFIT (housing benefit receipt y/n), HBENCHK (HB included before/after original amount in rent), HBENAMT (amount of HB), ACCNONHH (anyone outside HH paying rent for you y/n), ACCCHK (before or after original amount), ACCAMT (amount paid by other person), SERINC (rent include any services) and SERAMT (amount of services, see constant's table).

First omit owner-occupiers, boarders and lodgers. Note rent-free cases have been checked to make sure not including 100% HB cases.

In shared households (HHSTAT=2) we need to aggregate individual payments (RENT, SRENTAMT) before checking for rent holidays etc.

Next the weekly amount of rent has to be checked as the household may have some rent free weeks (RENTHOL = 1) and the total rent has to be apportioned over this period. The rent should be multiplied by the number of weeks over which it is actually paid (52 weeks minus the number of rent free weeks (WEEKHOL) and then divided by 52 for the weekly amount.

If someone pays part of the rent on behalf of the tenant and this has not been included in the original amount for rent declared, this must be added to the declared rent. (Not including HB at this point). This is only for non-DSS contributors to the rent.

(Reordered)

Other variables ask whether certain service charges are included in the rent amount stated which should be deducted in full from the gross rent amount. (These amounts have been converted to a weekly amount using period codes)

For HB cases (non 100%), if Housing Benefit is received and the original amount of rent is said to be after Housing Benefit has been deducted, ie where HBENCHK = 2 (after), the Housing Benefit has to be

 --

added back on to the rent. HHRENT is then the result of the total amount of rent paid less any ineligible services (council tax, community water rates or any other service charges), plus any HB. In case (confused?) responses have led to HHRENT being negative a check is applied and HHRENT set to zero before HB is added.

A further check makes sure that HHRENT is not less than HB.

If the household pays no rent because of 100% HB ie where REBATE = 1 (receive 100% HB) the amount of HB should be substituted for the rent from HBENAMT.

If the period code for any of the above variables is 90, 95 or 97 (lump-sum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if RENTPD, COMPD, HBENPD, ACCPD = 90, 95 or 97 HHRENT is set to -2. Since 94/5 all code 12 are edited so this problem should be reduced.

The value of services included in rent is now inputted from the constant's spreadsheet in p:\frs\shared\frs35\metadata\Dvmeta35.xls. The information is obtained from the Survey of English housing (ONS).

2 FRS Specification

For each household

<u>Code</u>	<u>Condition</u>
HHRENT	<p>From HOUSEHOL table, if TENURE = 4 or 5 (renting or rent free) get HHRENT and HHSTAT variables. Else if TENURE = 1 or 2 set HHRENT to .A</p> <p>From RENTER table, if RENTPD equals -1 or 1-10, 13, 26 or 52 get RENT variable and calculate HHRENT = RENT</p> <p>From HOUSEHOL record if HHSTAT = 2 then from ADULT record for all BUs > 1 if SRENTPD equals -1 or 1-10, 13, 26 or 52 and SRENTAMT exists add SRENTAMT to HHRENT.</p> <p>If RENTHOL = 1 and WEEKHOL exists (has some rent free weeks) multiply HHRENT by the number of weeks actually paid (52 - WEEKHOL) and divide the result by 52 to get the overall weekly rent else do not change HHRENT.</p> <p>If RENTHOL =1 and WEEKHOL exists and HBENEFIT=1(HB in receipt), adjust HBENAMT as above, ie multiply HBENAMT by (52-WEEKHOL) and divide by 52. (Note: Because HB is only paid for the weeks rent is paid)</p> <p>From RENTCONT record if ACCNONHH = 1 and ACCPAY ≠ 1 (2, 3, 4 or 5) and ACCCHK ≠ 1 (2 or -1) and ACCPD equals -1 or 1-10, 13, 26 or 52 add ACCAMT to rent to get HHRENT (Someone outside of HH pays part of rent and original rent given after</p>

 --

this deducted. ACCCHK=-1 refers to those cases where RENT=0, in which case ACCCHK is not asked). If ACCNONHH = 1 and ACCCHK = 1 or ACCNONHH = 2, do not change amount in HHRENT.

If SERINC(1 to 5) = 1 and RENTHOL = 1 and WEEKHOL exists, then adjust SERAMT (inputted value of SERINC(1 to 5) from constants table) for rent holidays as above and subtract from HHRENT. Else if SERINC(1 to 5)=1 and no rent holiday deduct SERAMT from HHRENT.

If WSINC = 1 or ~~HBWSAMT > 0 (statement exists and is amount to deduct)~~ or WSINCAMT > 0 (no statement and is amount to deduct), and RENTHOL=1 and WEEKHOL exists adjust WATSEWRT as above and deduct WATSEWRT from HHRENT; else if no rent holiday deduct unadjusted WATSEWRT from HHRENT. If WORSINC=2, do not change HHRENT.

(At this stage possible that HHRENT is negative if confusion over inclusive charges. This is most likely for HB cases where rent has been declared after HB deducted). So check

If HHRENT < 0 then HHRENT = 0

If HBENEFIT = 1 and HBENCHK = 2 or HBENCHK = .A and HBENPD equals .A or 1-11, add HBENAMT to HHRENT (HB rec'd and adjusted for rent holidays where applicable & original rent given after HB and ineligible services deducted).

If HBENEFIT = 1 and HBENCHK = 1
 If HBENAMT > HHRENT then HHRENT = HBENAMT

Otherwise if HBENEFIT = 2, do not change amount in HHRENT.

If REBATE = 1 and HBENPD equals .A or 1-10, HHRENT = HBENAMT. In this circumstance the declared rent will have been zero.

From HOUSEHOL record if HHSTAT = 2 then from ADULT record for all BUs > 1 if HBOTHPD equals -1 or 1-10 or 13, 26 or 52 and HBOTHAMT exists add HBOTHAMT to HHRENT.

Set HBSERAMT = Sum of services included in rent

If HBSERAMT > 0 then

If HBENEFIT = 1 and RENTDOC = 1 and RENTHOL = 1 and WEEKHOL not equal to (.A, .B, .C, .D) then subtract (adjust for rent holiday) HBSERAMT from HHRENT

Else if HBENEFIT = 1 and RENTDOC = 1 then subtract (not adjusted as no rent free holiday) HBSERAMT from HHRENT

(At this stage possible that HHRENT is negative if confusion over inclusive charges. This is most likely for HB cases where rent has been declared after HB deducted). So check

--

If HHRENT<0 then HHRENT=0

HHSIZE

Purpose : To show the number of people within the household
Created : 7 September 1998
Database Table : HOUSEHOL
Minimum Value : 0
Maximum Value : 7
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (hot-decking)
Amendments : EP – 11 January 1999 – set up count variables to avoid use of
ADULTH and DEPCHLDH for v34
: JC -6 September 1999- Security completed, no other changes for V35
Issue date : 7 September 1998

1 Definition

HHSIZE shows the number of people within the household and is coded as follows:

- | | |
|---|------------------|
| 1 | 1 person |
| 2 | 2 people |
| 3 | 3 people |
| 4 | 4 people |
| 5 | 5 people |
| 6 | 6 people |
| 7 | 7 or more people |

2 FRS Specification

Code Condition

From HOUSEHOL table

HHSIZE = ~~ADULTH + DEPCHLDH~~ CADULT + CCHILD

- | | |
|---|--------------------------------|
| 1 | HHSIZE = 1 |
| 2 | HHSIZE = 2 |
| 3 | HHSIZE = 3 |
| 4 | HHSIZE = 4 |
| 5 | HHSIZE = 5 |
| 6 | HHSIZE = 6 |
| 7 | IF HHSIZE >= 7 then HHSIZE = 7 |

 --

HHSTATUS

Purpose : Householder Status of Benefit Unit
 Created : AJG 1 December 1992
 Database Table : BENUNIT
 Minimum Value : 1
 Maximum Value : 3
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : VC - 5 February 1993
 : VC - 15 February 1994 Amended to reflect version 30 changes
 : VE - 4 June 1996 - No initial amendments needed for V32 update
 : SG - 11th July 1997 - No initial amendments needed for V33 update
 : EP - 12 August 1998 - No initial V34 update needed
 : JC - 9 September 1999 - Security completed, no other changes for V35
 Issue date : 11th July 1997

1 Definition

Identifies whether the Benefit Unit is the head of household Benefit Unit.

This variable is coded as

- 1 Single Benefit Unit household.
- 2 Multi Benefit Unit household, head of household Benefit Unit
- 3 Multi Benefit Unit household, not head of household Benefit Unit
- 2 Unable to derive.

The assumption is made that the first Benefit Unit is the head of household Benefit Unit.

2 FRS Specification

For Benefit Unit in each household

CODE CONDITION

Get variables BENUNIT and BENUNITS from BENUNIT and HOUSEHOL tables respectively.

- 1 If BENUNIT = 1 and BENUNITS = 1
- 2 If BENUNIT= 1 and BENUNITS > 1
- 3 If BENUNIT > 1
- 2 If unable to derive variable.

--

NB - BENUNIT indicates which benefit unit within the household and BENUNITS shows the total number of benefit units within that household.

3 Results

Tabulate numbers falling into each category.

4 Test Cases

Households:

- A 1 Benefit Unit
- B 2 Benefit Units
- C 3 Benefit Units

 --

HOURCARE

Purpose : To show the number of hours of care an adult receives from all helpers
 Created : 6 February 1997
 Database Table : ADULT
 Minimum Value :
 Maximum Value :
 Units : integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user :
 Amendments : EP -12 August 1998 - No initial V34 update needed
 SG – 12 March 1999 – Changes in data recorded
 SB – 8 November 1999 – Security completed, no other changes for V35
 Issue date : 17 April, 2003

1 Definition

This derived variables shows the number of hours of care an adult receives from all helpers, and is for use in the disability trailer. HOUR01 17 gives the number of hours a week each helper provides, so these variables should be totalled for each person cared for to give HOURCARE

HOURCARE The number of hours of care an adult receives from all helpers

- 0 For all variables - not applicable to this case - adult is not cared for by anyone
- 2 For all variables - unable to derive due to missing values.

2 FRS Specification

Set HOURCARE to zero.

From CARE record, for each adult in the household needing care (NEEDPER1-8), process HOUR(xx) for all people looking after that person (xx=00- 17).

$$\text{HOURCARE}=\text{HOUR01}+\text{HOUR01}+\dots+\text{HOUR16}+\text{HOUR17}$$

- 2 If any variables are missing

 --

HPERSON (adult) HPERSON (child)

Purpose : To show the person number within the Household
 Created : AJG 10 September 1993
 Database Table : ADULT and CHILD
 Minimum Value : 1
 Maximum Value : 20
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VE - 23 May 1996 - No amendments needed for initial V32 update
 : SG - 12 November 1997 - No amendments needed for initial V33 update
 : EP - 12 August 1998 - No initial V34 update needed
 : EP - 18 May 1999 - Correct for V34 to ensure that head of household always
 has the value HPERSON=1
 : JC - 8 September 1999 - Security completed, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable assigns the value 1 to the head of household and then increments by one for each adult and each child, by Benefit Unit, i.e. incrementing by one for all individuals (adults then children) in the head of household Benefit Unit, then for all individuals in other Benefit Units in the household.

2 FRS Specification

Process each Benefit Unit in the household in turn (starting with the Benefit Unit in which the head of household lies), incrementing HPERSON as shown.

The HPERSON variable exists on both the ADULT and the CHILD datasets. If dealing with the whole household then these need to be merged together.

Code Condition

(This part rewritten for FRS 1997-98 (V34) – see 1996-97 (V33) spec for previous coding)

- 1 If BENUNIT contains the Head of Household (HOH BENUNIT) and HOH = 1
 Then in the following priority:
 +1 If BENUNIT = HOH BENUNIT, and other ADULT in the BENUNIT (if they exist),
 +1 If BENUNIT = HOH BENUNIT, for each dependent in descending order of age

--

For each subsequent Benefit Unit:

- +1 For each adult in PERSON number order,
- +1 For each dependent in descending order of age.

3 Results

What tabulation should be produced to check the results?

--

HSCOSTHH

Purpose : Housing costs paid by a household.
 Created : VC - 24 September 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children : INCAHCHH INCAHCBU HHCOSTBU
 Parents : HHRENT WATSEWRT MORTINT
 Core variable/user : FRS (Publication)
 Amendments : VC - 12 October 1993 To emphasise that this is by household
 : VC - 16 February 1994 Amended to reflect changes to version 30
 : VC - 1 March 1994 To exclude period codes 12 and 13
 : JS - 6 February 1996 to amend calculation of structural insurance and change
 names of period codes
 : JS - to allow for skipped values where variables have been imputed and to
 make CWATAMTD change explicit
 : VE - 5 June 1996 - No initial amendments needed for V32 update
 : VE - 2 September 1996 - To adjust CWATAMTD and WSEWRT for rent free
 holidays
 : VE - 20 November 1996 - To remove the rent holiday adjustment to
 CWATAMTD (because it is not included in rent, and doesn't feed into HHRENT)
 : VE - 20 November 1996 - To amend STRAMT to STRAMT1 and STRAMT2
 where appropriate
 : VE - 3 February 1997 - To bring in line with HBAI by assuming that 2/3 of
 combined contents and structural insurance relates to structure
 : VE - 6 February 1997 - To reference household charge variables from the
 HOUSEHOL table rather than the OWNER table
 : VE - 10 March 1997 - To amend the structural insurance part of the spec
 : SG - 7 October 1997 - Set unable to derive if missing values
 : SG - 30 October 1997 - V33 updates, period codes
 : EP - 12 August 1998 - V34 update, amend CHAMT1-8 and CHPD1-8 to
 CHRGAMT1-8 and CHRGPD1-8 respectively
 : SB - 17 April 2003 - Security completed, no other changes for V35

 Issued : 17 April 2003

1 Definition

This variable is coded as

HSCOSTHH This is the total amount spent on housing costs by each household regardless of whether they are in rented or owned accommodation.

 --

0 Not applicable as have no housing costs

-2 Unable to derive due to missing values.

The variable is produced from the total amount of rent paid after any charges for TVs etc, the amount of any mortgage interest paid, the amount of water and sewerage paid, any insurance on the structure of the accommodation, any community water charge per household and any other regular housing payments (eg ground rent, feu duty etc).

The gross rent is calculated by the derived variable HHRENT which produces the household's gross rent after taking off any expenses such as community charge or service charges for lifts, TV etc which might have been included in the rent.

Mortgage interest is found in the derived variable MORTINT which calculates the total amount of mortgage interest paid for all mortgages that have been taken out to buy the property.

The amount of water and sewerage rates which have to be paid by the household are also found in another derived variable - WATSEWRT. This collects both water and sewerage rates for the whole household. WATSEWRT should be adjusted for rent-free holidays before adding into HSCOSTHH.

Insurance on the structure of the property, on the other hand, must be derived within this variable. The variables to do this are to be found in the OWNER record. The amount of insurance paid on the structure is held in a variable called STRINTH regardless of how the interviewee pays the insurance premium. Therefore, if STRINTH exists it should be added to HSCOSTHH. STRINTH needs to be amended to include structural insurance premiums paid separately from mortgage payments. For combined contents and structural insurance, it is assumed that 2/3 of the insurance relates to structure.

The amount of community water charge should also be added into HSCOSTHH and is found in the DV CWATAMTD.

Finally any other housing costs have to be added into HSCOSTHH. These are collected from the question CHARGE which asks if the household pays ground rent, feu duty, chief rent, service charge, any compulsory maintenance charge, site rent (for caravans) or any other not mentioned. These are in turn put into the database under the following variables - CHARGE1, CHARGE2, CHARGE3, CHARGE4, CHARGE5, CHARGE6 and CHARGE7 and **CHARGE8** respectively. If the answer to any of the above is yes, the amount of each charge is held in the database variables ~~CHAMT1~~ **CHRGAMT1**, ~~CHAMT2~~ **CHRGAMT2**, ~~CHAMT3~~ **CHRGAMT3**, ~~CHAMT4~~ **CHRGAMT4**, ~~CHAMT5~~ **CHRGAMT5**, ~~CHAMT6~~ **CHRGAMT6** and ~~CHAMT7~~ **CHRGAMT7** and ~~CHAMT8~~ **CHRGAMT8**.

However, if the period code for charges is 90,95 or 97 (lump-sum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if ~~CHARGE1~~ **CHRGPD1**= 90,95 or 97 HSCOSTHH is set to -2. However, a value of -1 (skipped) is acceptable, since this implies that the amount has been imputed. As the other parts of HSCOSTHH are derived using previously derived variables the period codes need not be looked at for these variables as they are already excluded.

2 FRS Specification

 --

If any of components missing (refused or don't know) then set to unable to derive, also if period codes set to 90,95 or 97.

For each HOUSEHOLD, set HSCOSTHH to zero.

<u>Code</u>	<u>Condition</u>
HSCOSTHH	<p>From HOUSEHOL and RENTER records,</p> <p>If HHRENT exists <u>and</u> does not equal -1 or -2 (not applicable or unable to derive) add the amount in HHRENT into HSCOSTHH.</p> <p>If RENTHOL=1 and WEEKHOL exists and WATSEWRT exists <u>and</u> does not equal -1 or -2 (as above) multiply WATSEWRT by (52-WEEKHOL) and divide by 52 to get the overall weekly rent, else do not change WATSEWRT. Add the amount in WATSEWRT into HSCOSTHH.</p> <p>If MORTINT exists <u>and</u> does not equal -1 or -2 (as above) add the amount in MORTINT into HSCOSTHH.</p> <p>From HOUSEHOL record get STRCOV, STRAMT1, STRAMT2, STRPD1, STRPD2, STROTHS, COVOTHS, TENURE</p> <p style="padding-left: 40px;">set STRINTH equal to zero</p> <p style="padding-left: 40px;">If STRCOV=1 and STRAMT1 exists (structural insurance) and STRPD1=-1 or 1-10,13,26,52 STRINTH=STRAMT1</p> <p style="padding-left: 40px;">If STRCOV=3 and STRAMT1 exists (structural insurance combined with furniture and contents as part of mortgage payment) and STRPD1=-1 or 1-10,13,26,52 STRINTH=STRAMT1*0.6666 (to bring in line with HBAI)</p> <p style="padding-left: 40px;">If STROTHS=1 and COVOTHS=2 and STRAMT2 exists (insurance premium paid on structure of accommodation separately from any mortgage payments and STRPD2=-1 or 1-10,13,26,52 STRINTH=STRAMT2*0.6666</p> <p style="padding-left: 40px;">If STROTHS=1 and COVOTHS=1 and STRAMT2 exists (insurance premium paid on structure of accommodation separately from any mortgage payments) and STRPD2=-1 or 1-10,13,26,52 STRINTH=STRAMT2</p> <p style="padding-left: 40px;">Add STRINTH to HSCOSTHH</p> <p style="padding-left: 40px;">From owner HOUSEHOL record get CHARGE1-78, CHRGAMT1-78, CHARGE1-78 CHRGPD1-78</p> <p style="padding-left: 40px;">If CHARGE1=1 and CHRGPD1=-1, or 1-10,13,26,52 add CHRGAMT1 CHRGAMT1 to HSCOSTHH</p>

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

If CHARGE2=1 and ~~CHPD~~ **CHRGPD2=-1**, or 1-10,13,26,52 add ~~CHAMT~~
CHRGAMT2 to HSCOSTHH

If CHARGE3=1 and ~~CHPD~~ **CHRGPD3=-1**, or 1-10,13,26,52 add ~~CHAMT~~
CHRGAMT3 to HSCOSTHH

If CHARGE4=1 and ~~CHPD~~ **CHRGPD4=-1**, or 1-10,13,26,52 add ~~CHAMT~~
CHRGAMT4 to HSCOSTHH

If CHARGE5=1 and ~~CHPD~~ **CHRGPD5=-1**, or 1-10,13,26,52 add ~~CHAMT~~
CHRGAMT5 to HSCOSTHH

If CHARGE6=1 and ~~CHPD~~ **CHRGPD6=-1**, or 1-10,13,26,52 add ~~CHAMT~~
CHRGAMT6 to HSCOSTHH

If CHARGE7=1 and ~~CHPD~~ **CHRGPD7=-1**, or 1-10,13,26,52 add ~~CHAMT~~
CHRGAMT7 to HSCOSTHH

If **CHARGE8=1** and **CHRGPD8=-1**, or **1-10,13,26,52** add ~~CHAMT~~
CHRGAMT8 to HSCOSTHH

From ~~RENTER~~ **HOUSEHOL** record,

If CWATAMTD exists add the amount in CWATAMTD into HSCOSTHH.

- 0 Not applicable as household has no housing costs.
- 2 Unable to derive - if any of above variables are missing or where a derived variable has already been set to -2.

3 Results

Tabulation is required to show the number of households by the total weekly amount of housing costs they pay divided into the following bands

£50 or under
£50.01 - £100
£100.01 - £150
£150.01 - £200
£200.01 - £250
£250.01 - £300
£300.01 - £350
£350.01 or over

IAGEGRP, IAGEGR2

Purpose : Age groups of individuals for the publication
 Created : 22 February 1999
 Database Table : ADULT, CHILD
 Minimum Value : 1
 Maximum Value : 18 (IAGEGRP), 12 (IAGEGR2)
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : EP – 10 March 1999 - Harmonise age bands and also create IAGEGR2
 : SB - 9 November 1999 – Security completed, no other changes for V35
 Issue date : 22 February 1999

1 Definition

IAGEGRP creates places an individual, adult or child, into 5-year age bands for FRS publication purposes. IAGEGR2 mainly uses 10-year age bands. It is coded as follows:

<u>IAGEGRP</u>		<u>IAGEGR2</u>		
1	4 and under	1	4 and under	} CHILD table
2	5 to 10	2	5 to 10	
3	11 to 15	3	11 to 15	
4	16 to 19	4	16 to 24	} ADULT table
5	20 to 24	5	25 to 34	
6	25 to 29	6	35 to 44	
7	30 to 34	7	45 to 54	
8	35 to 39	8	55 to 59	
9	40 to 44	9	60 to 64	
10	45 to 49	10	65 to 74	
11	50 to 54	11	75 to 84	
12	55 to 59	12	85 or over	
13	60 to 64			
14	65 to 69			
15	70 to 74			
16	75 to 79			
17	80 to 84			
18	85 or over			

There are two separate pieces of code for each DV. They both have the same name but one is on the CHILD table and the other is on the ADULT table.

2 FRS Specification

IAGEGRP

<u>Code</u>	<u>Condition</u>
1	From CHILD table If (Age <= 4)
2	From CHILD table If (Age >= 5 and Age <=10)
3	From CHILD table If (Age >= 11 and Age <=15)
4	From CHILD or ADULT table If (Age >= 16 and Age <=19)
5	From ADULT table If (Age >= 20 and Age <=24)
6	From ADULT table If (Age >= 25 and Age <=29)
7	From ADULT table If (Age >= 30 and Age <=34)
8	From ADULT table If (Age >= 35 and Age <=39)
9	From ADULT table If (Age >= 40 and Age <=44)
10	From ADULT table If (Age >= 45 and Age <=49)
11	From ADULT table If (Age >= 50 and Age <=54)
12	From ADULT table If (Age >= 55 and Age <=59)
13	From ADULT table If (Age >= 60 and Age <=64)
14	From ADULT table If (Age >= 65 and Age <=69)
15	From ADULT table If (Age >= 70 and Age <=74)
16	From ADULT table If (Age >= 75 and Age <=79)
17	From ADULT table If (Age >= 80 and Age <=84)

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

- 18 From ADULT table
If (Age >= 85)
- 1 Not applicable in this case – should not happen to this variable
- 2 Unable to derive IAGEGRP

IAGEGR2

<u>Code</u>	<u>Condition</u>
1	From CHILD table If (Age <= 4)
2	From CHILD table If (Age >= 5 and Age <=10)
3	From CHILD table If (Age >= 11 and Age <=15)
4	From CHILD or ADULT table If (Age >= 16 and Age <=24)
5	From ADULT table If (Age >= 25 and Age <=34)
6	From ADULT table If (Age >= 35 and Age <=44)
7	From ADULT table If (Age >= 45 and Age <=54)
8	From ADULT table If (Age >= 55 and Age <=59)
9	From ADULT table If (Age >= 60 and Age <=64)
10	From ADULT table If (Age >= 65 and Age <=74)
11	From ADULT table If (Age >= 75 and Age <=84)
12	From ADULT table If (Age >= 85)
-1	Not applicable in this case – should not happen to this variable
-2	Unable to derive IAGEGRP

 --

INCSEO2, SEINCAM2, NINCSEO2, NINSEIN2

Purpose : To calculate the total income/earnings from self-employment - Option 2 based on profit or income/drawings, GROSS and NET versions
 Created : SG - 30 March 1998
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : EP - 12 August 1998 - No initial V34 update needed
 : SB - 17 April 2003 – Security completed, no other changes for V35
 Issued : 17 April 2003

1 Definition

This variable is coded as

INCSEO2 This is the total amount of income received from self-employment GROSS of tax and national insurance payments, based on profits where individual considers themselves as running a business, on estimated earnings/drawings otherwise.
 -1 Not applicable to this case
 -2 Unable to derive due to missing values

SEINCAM2 This is same as INCSEO2 except set to zero for people with no self employment income, rather than -1 (for use in INDINC)
 -2 Unable to derive due to missing values

NINCSEO2 This is the total amount of income received from self-employment NET of tax and national insurance payments, based on profits where individual considers themselves as running a business, on estimated earnings/drawings otherwise.
 -1 Not applicable to this case
 -2 Unable to derive due to missing values

NINSEIN2 This is same as NINCSEO2 except set to zero for non self employed, rather than -1 (for use in NINDINC)

 --

-2 Unable to derive due to missing values

The questions on self-employment income were thoroughly revised from April 1996. Information on profit/loss is now only collected where a respondent keeps business accounts which are prepared for Inland Revenue for tax purposes. Instead, questions are asked about money drawn for non-work purposes (for those with separate business accounts) or income from job/business after paying for any materials, equipment etc used in work. This definition of self employed income uses the information on profits where a figure is available and where the individual considers themselves to be running a business. Where the individual considers themselves as having a job, or where profit accounts are not available, self employment income is based on estimated income coming in to the household, or in the absence of that drawings from business accounts.

Initially a check is made as to whether a particular job counts as self employed. This is derived from several variables in the ADULT and JOB tables. Firstly, the person must be working or temporarily absent from his/her self-employment, so where WORKING = 1 (did paid work in last 7 days) or where JOBAWAY = 1 (not worked in last 7 days but has job to return to) and EMPEE = 2 (indicating that the job is self-employed).

Self employed respondents are asked as to whether they consider themselves as having a job (JOBBUS=1) or business (JOBBUS=2) or neither (JOBBUS=3). Those who answer "neither" are treated as having a business.

Most of the specification relates to both GROSS and NET versions. Certain sections are labelled { } where appropriate to refer to adding back or removing tax and/or NI to produce the net and gross versions.

Respondents with a business and profit data

If a business then the profit data is used where it has been completed. The variables used to indicate whether a profit or loss has been made and the amount of profit or loss are - PROFIT1 which holds the total amount of profit or loss made by the business and PROFIT2 which indicates whether it is a profit (PROFIT2 = 1) or a loss (PROFIT2 = 2).

If the respondent has given a figure net of tax and/or Class IV NI then the amount given in PRBEFORE will be used to provide gross profit.

For calculating net profit tax and national insurance are deducted as appropriate.

Once the total amount of profit has been established it has to be increased by an uprating factor as it is likely to be from a particular accounting period which may not be consistent with the interview date. The period that the earnings have been taken from are defined by the variables SE1 and SE2 which give the start and the end of the accounting period. These are used to the nearest full month (not day) and the total number of months that this covers is worked out from the start and end date of SE1 and SE2.

The end of the self-employment period is first constrained to end no more than six years before the FRS year. An uprating factor is established from the tables by adding together each index for the months

 --

covered by SE1 and SE2 and then dividing by the period. A second average uprating factor is also calculated from the total indices for the equivalent period before the interview date (INTDATE) divided by PERIOD. The amount of earnings in PROFIT1 is then multiplied by the result of dividing these two indices to provide the uprated profit. NB A loss or drawings from the business will not be affected by uprating factors.

Respondents with a "job"

Where respondents consider themselves as having a job INCSEO2 is initially set to SEINCAMT (income from job/business after paying for any materials, equipment or goods used for work). This is then checked to see whether income tax or regular NI have been deducted at source (CHECKTAX) and whether the average income quoted was before or after tax/NI was deducted (CHKINCOM). Where appropriate, amounts in TAXDAMT and NIDAMT are added back in to INCSEO2. *(Note it is not possible to have both tax and NI deducted at source. Whereas some self-employed have tax deducted, eg construction workers on the SC60 scheme; and others might have NI deducted, eg actors on contract; anyone with both tax and NI deducted is almost certainly an employee. By preventing both being coded in the questionnaire, it is hoped that employees who think they are self-employed, eg company directors, will be stopped from going any further down the wrong route.)*

For own account cases, if a regular NI contribution (SENIREG) is made, the amount at SENIRAMT is added back into INCSEO3. Similarly, any tax and NI lump sum payments are also added back.

Remaining cases

For those who do not answer the profit or income questions information on drawings from business accounts for non-business purposes will be used. INCSEO3 is first set to OWNAMT (amount taken out for non-business purposes which is net of any tax or NI). Then, if the respondent receives any other income from the job/business for personal use (OWNOTHER=1) then the amount in OWNOTAMT is also added to INCSEO3. NI and tax are then added back.

2 FRS Specification

For each ADULT, get ADULT and JOB records.

Code Condition

INCSEO2

From ADULT table, If WORKING = 1 or JOBAWAY = 1

If first.jobtype

 If EMPEE = 2 (self-employed)

To avoid adding back tax and NI payments twice into income cases, set three temporary variables to zero

 --

xincse2p=0 {fpr profit cases}
 xincse2b=0 (for cases with separate business accounts)
 xincse2i=0 (for other cases)
 xincse2=0

If (PROFIT1 exists and JOBBUS=2 or 3) or (SEINCAMT < 0 and JOBBUS = 1 and PROFIT1 exists) then

if PROFIT2=1 (a profit) then XWRKPROF = PROFIT1
 else XWRKPROF = -PROFIT1

{{ FOR GROSS VERSION }

if PRBEFORE>0 then UPROFIT=PRBEFORE

}

Calculate the uprating factor and accumulate profits as follows -

Self-employment period is constrained to end no more than six years before the FRS year:

if SE2<April 1992 then SE2=March 1992, SE1=April 1991

if SE2 after INTDATE, change SE2 to INTDATE and adjust SE1

Calculate PERIOD referred to by SE1 and SE2 = (SE2 - SE1)/30.416666 and then round to the nearest month. This will produce the number of months between SE1 and SE2 as accurately as possible.

Calculate SEINDEX1 as sum of each index over period from SE1 to SE2 divided by PERIOD (as above) using look up table of uprating values

Calculate SEINDEX2 as sum of each index over (PERIOD-1) months prior to date of interview (INTDATE) divided by PERIOD.

Calculate uprated profit UPROFIT = UPROFIT * (SEINDEX2/SEINDEX1)

{{ FOR NET VERSION }

if PROFTAX=1 and SETAX=1 then remove SETAXAMT from UPROFIT
 if PROFNI=1 and SENIREG=1 then remove SENIRAMT from UPROFIT
 if PROFNI=1 and SENILUMP=1 then remove SENILUMP from UPROFIT

if PROFTAX=2 and PROFNI<>2 and SENIREG=1 then remove SENIRAMT from UPROFIT

if PROFTAX=2 and PROFNI<>2 and SENILUMP=1 then remove SENILAMT from UPROFIT

 --

if PROFTAX=2 and PROFNI=2 and SENIREG=1 then remove SENIRAMT
 from UPROFIT

}

Calculate xincse2p = UPROFIT + xincse2p

Else If SEINCAMT exists then

xincse2i=xincse2i+SEINCAMT

{{ FOR GROSS VERSION }

If (CHECKTAX=1 and CHKINCOM=2 and TAXDAMT exists and TAXDPD<>90,95,97)
 then xincse2i=xincse2i+TAXDAMT

If (CHECKTAX=2 and CHKINCOM=2 and NIDAMT exists and NIDPD<>90,95,97)
 then xincse2i=xincse2i+NIDAMT

}

{{ FOR NET VERSION }

If (CHECKTAX=1 and CHKINCOM=1 and TAXDAMT exists and TAXDPD<>90,95,97)
 then xincse2i=xincse2i-TAXDAMT

If (CHECKTAX=2 and CHKINCOM=1 and NIDAMT exists and NIDPD<>90,95,97)
 then xincse2i=xincse2i-NIDAMT

If (SENIREG=1 and SENIRAMT exists and SENIRPD<>90,95,97) then
 xincse2i=xincse2i-SENIRAMT

If (SENILUMP=1 and SENILAMT exists) then xincse2i=xincse2i-SENILAMT

If (SETAX=1 and SETAXAMT exists) then xincse2i=xincse2i-SETAXAMT

}

Else {seincamt not collected}

If OWNSUM=1 and OWNAMT exists then xincse2b=xincse2b+OWNAMT

If OWNOTHER=1 and OWNOTAMT exists then xincse2b=xincse2b+OWNOTAMT

{{ FOR GROSS VERSION }

If (SENIREG=1 and SENIRAMT exists and SENIRPD<>90,95,97) then
 xincse2b=xincse2b+SENIRAMT

If SETAX=1 and SETAXAMT exists then xincse2b=xincse2b+SETAXAMT

If SENILUMP=1 and SENILAMT exists then xincse2b=xincse2b+SENILAMT

}

If SENILUMP=1 and SENILAMT exists then xincse2b=xincse2b+SENILAMT
 for those with separate business accounts who draw money from them

--

If last.jobtype

output INCSEO2=sum(xincse2p,xincse2b,xincse2i) (ADULT record)

-1 Not applicable - people who are not working or are employees (WORKING <>1 or
 JOBAWAY<>1 OR EMPEE<>2)

-2 Underivable - where any of above values are missing

SEINCAM2

Initialise SEINCAM2 =0.

If working and INCSEO2 ^= -1 THEN SEINCAM2=INCSEO2

NINSEIN2

Initialise NINSEIN2 =0.

If working and NINCSEO2 ^= -1 THEN NINSEIN2=NINCSEO2

 --

INDINC, INEARN, ININV, INRPINC, INPENINC, INDISBEN, INOTHBEN, INRINC

Purpose : To indicate the amount of gross income received by an adult for use in the FRS publication (based on GROSSINC).

Created : 31 January 1996

Database Table : ADULT

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : **GROSSPAY (related to INEARN) – refer to note in main text**

Children :

Parents :

Core variable/user : FRS (general)

Amendments : JS - 21 March to correct INRINC for luncheon vouchers (only asked about first job); INPENINC to allow for skipped tax (where penpay has been imputed or TU pension) and INEARN for questions which are only asked of the first job

:JS - 12 April 1996 - to add in income for contributions to rents/mortgages from those outside the household

VE - 17 April 1996 - to correct ININV to add in tax for those accounts where interest is received after tax

: VE - 23 May 1996 - Initial amendments for V32 - Still need to look at accounts and benefits to check definitions are the same as for V31

: VE - 16 July 1996 - To amend typo in adjustments for SSP and SMP and to include SSP and SMP in benefits block for cases where the benefit is recorded in the benefits table but not in the job section

: VE - 19 August 1996 - To update ACCOUNTS table definitions in ININV for V32

: VE - 20 August 1996 - to amend benefit definitions for V32

: JS - 12 November 1996 - to amend SSP/SMP adjustment

: VE - 20 November 1996 - to amend INRINC

: VE - 20 November 1996 - to replace IND TAX with ACCTAX

: VE - 28 November 1996 - to amend so that UBONAMT is only subtracted from INEARN when UBONINC=1

: VE - 11 December 1996 - to amend the OJAMT step to take into account the fact that for V32 OJPD=12 or 13 have been weeklyised

: VE - 23 January 1997 - to adjust HBENAMT and SCHBAMT for any rent free holidays as in HHRENT

: VE - 31 January 1997 - to make adjustments to bring into line with HBAI

: SG - 12 November 1997 - V33 changes, also implications of NINDINC (net income version)

: SG - 3 December 1997 - include maintenance even if paid via DSS/CSA

: SG - 30 December 1997 - no period codes for odd jobs

: SG - 11 February 1998 - change in PAYSLIP for V33

: SG - 17 March 1998 - use 20% tax rate

: SG - 23 April 1998 - new self employment DV

: SG - 1 June 1998 - add extended HB/CTB where appropriate

: SG - 8 June 1998 - Social Fund crisis loan should not be included

 --

: EP – 30 November 1998 – V34 – change head of household from person =1 to HOH = 1
 : EP – 8 December 1998 – remove references to UB, and replace references to benefit 20 with references to benefits 65 and 66
 : EP – 9 December, 1998 – add in Benefit 61 for V34
 : EP – 15 December 1998 – include income for students from their parents in INRINC
 : EP – 11 May 1999 – Add in explanation of difference between INEARNs and GROSSPAY
 : SB - 30 November 1999 – Security completed, change to charity tax relief.
 : SB - 5 January 2000 - moved baby-sitting and mail order from ADULT table to ODDJOB table
 : SB - 5 January 2000 - moved ROYAL1-3 down one and added in a new ROYAL1 category
 : SB - 5 January 2000 - remove child income from rent
 : SB - 5 January 2000 - remove deleted benefits
 : SB - 5 January 2000 – SSP/SMP no longer asked on benefits table table
 : AW – 15 February 2000 – checking of dv – changes made to add inearn code bottom of innirben code so that SMP/SSP adjustment is correctly calculated
 :
 Issued : 17 April, 2003

1 Definition

This variable is coded as

INDINC The total amount of gross income received by an adult from all sources.

0 Not applicable as adult does not have any gross income.

-2 Unable to derive due to missing values.

The total amount of gross income is derived from numerous variables from the ADULT, JOB, BENEFITS, ODDJOB and PENSIONS records which when added together form the person's total gross income. It includes gross normal earnings (cf with GROSSPAY which is last pay), self-employed earnings, tax paid on pensions annuities, other income in the form of benefit income, income in kind, royalties, other allowances, income from trust funds and odd jobs etc. Income Support is included gross of any direct payments or social fund repayments. Income from boarders/lodgers has been excluded to avoid double counting at a household level and to simplify definitions (this will need to be discussed in the publication).

Additional amounts for direct expenses from absent partners, regular contributions from household members, education grants and other deductions from pension income have also been included.

Private benefit schemes are included unless they are one off/lump sum payments (these are ignored).

Income from free school milk and meals and free welfare milk are also included (allocated to the head of benefit unit).

 --

Amounts of maintenance are also checked to make sure they are usual.

If the period code for the benefit is 90 or 95 or 97 (lump-sum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, for example, if PAYPD = 90 or 95 or 97 INDINC is set to -2. Additionally, coding has been changed to allow skipped values for period codes: this will occur where the (weekly) amount has been imputed. It has been decided to leave period codes as "skipped" in these cases since this may help flag imputation.

The addition of HB is adjusted for any rent-free holidays as in the derivation of HHRENT. This adjustment is done for conventional households and for shared households, but not for boarders and lodgers. The reasoning behind this is that boarders and lodgers would probably pay rent every week even if the household had some rent-free weeks.

Note: additional categories are now included for personal pensions

The DV spec is also used to set up components of gross income which are accumulated to obtain BU and HH level variables. These are:

INEARNS	earned income (see note below)
SEINCAM2	self employment income (identical to INCSEO2 but set to zero if not applicable)
ININV	investment income
INRPINC	retirement pension plus any income support
INPENINC	other pensions
INDISBEN	disability benefits
INOTHBEN	other benefits
INRINC	remaining income

For pensioners, any retirement pension is taken together with IS to avoid any issues of misreporting. Analyses of pensioner income by ASD3 also take these two together.

Disability benefits comprise war disablement benefit, DWA, SDA, AA, DLA (mob and care); IIDB and IB.

Although similar to the derived variable GROSSPAY, INEARNS sometimes differs in value from it. This is because GROSSPAY looks at the last pay that the respondent received. In cases where this was not the usual pay, GROSSPAY does not give the best representation of an individual's earnings – this can be obtained from INEARNS, as this variable includes a check on whether or not the last pay was the usual pay (using the variable UGROSS). GROSSPAY also does not include any bonuses received, which are included in INEARNS (the variables BONAMT1-6, which are weeklyised when included), and also does not deduct other allowance/refund variables (e.g. HHA1-3, MILEAMT, MOTAMT). If any user is unclear which of these two variables to use, then refer to ASD3E.

2 FRS Specification

For each ADULT

 --

Code Condition

INDINC Gross earnings: INEARNs

From ADULT record, set INEARNs to zero

If WORKING = 1 or JOBAWAY = 1 (worked in last 7 days or away from work in the last 7 days) process each JOB record for that person and

If ABSWHY ne 3 (To bring in line with HBAI)

If EMPEE = 1

(Rewording of following section)

(Coding includes a check to see if UGROSS exists, where it doesn't exist, gross pay is calculated dependent on whether a payslip has been consulted, where it does (which can only be JOBTYP=1, PAYUSL=no), UGROSS is used)

If UGROSS does not exist or is missing:

 and If PAYSLIP = 1 or 2 (consulted)

 If GRWAGE exists and PAYPD equals -1 or 1 to 10 or 13 or 26 or 52, add it into INEARNs

 If it is missing set INDINC and INEARNs to -2

 else if PAYSLIP = 3 or 4 (or PAYSLIP=1 or 2 and GRSWAGE=-1) and PAYPD equals -1 or 1 to 10 or 13 or 26 or 52 ,

 If PAYAMT exists, add it into INEARNs.

 If it is missing set INDINC and INEARNs to -2

 If PAYE exists, add it into INEARNs.

 If it is missing do not change INEARNs

 If NATINS exists, add it into INEARNs.

 If it is missing do not change INEARNs

 If CHRAMT exists, add it into INEARNs.

 If it is missing do not change INEARNs

 If OTHDED1 = 1 add DEDUC1 to INEARNs

 If OTHDED2 = 1 add DEDUC2 to INEARNs

 If OTHDED3 = 1 add DEDUC3 to INEARNs

 If OTHDED4 = 1 add DEDUC4 to INEARNs

 If OTHDED5 = 1 add DEDUC5 to INEARNs

 If OTHDED6 = 1 add DEDUC6 to INEARNs

 If OTHDED7 = 1 add DEDUC7 to INEARNs

 If OTHDED* = 1 add DEDOTH to INEARNs

Else if UGROSS exists

 --

If UGROSS exists add UGROSS to INEARNNS

Adjustments to gross earnings for HBAI consistency:

deduction of income tax refunds, mileage and motoring allowances, refunds for items of household expenditure

if INEARNNS ne-2 (other conditions relating to PAYAMT and PAYPD will have been met by this point if INEARNNS has not been set to -2)

and UGROSS does not exist (ie all jobtypes except jobtype=1 where pay not usual)

and JOBTYP=1 and TAXAMT exists INEARNNS=INEARNNS-TAXAMT
 (TAXAMT only asked for first job)

and MILEAMT exists INEARNNS=INEARNNS-MILEAMT

and MOTAMT exists INEARNNS=INEARNNS-MOTAMT

and HHA1 exists INEARNNS=INEARNNS-HHA1

and HHA2 exists INEARNNS=INEARNNS-HHA2

and HHA3 exists INEARNNS=INEARNNS-HHA3

Adjustments to gross earnings for HBAI consistency:

addition of bonuses received in last 12 months divided by 52

for up to 6 bonuses i=1-6:

If BONAMT(i) exists and BONTAX(i) (before tax)=1
 INEARNNS=INEARNNS+((BONAMT(i)/52))

If BONAMT(i) exists and BONTAX(i)=2 or -1 (after tax or skipped where BONAMT imputed)

INEARNNS=INEARNNS+((BONAMT(i)/52)/0.77)

{questions about bonuses are asked regardless of whether pay usual or not, however, if UGROSS has been taken, have to make sure that bonus is not double counted. If bonus is after tax then need to make adjustment to add tax

 --

back in because looking at gross pay, before tax*0.77=net tax therefore net tax/0.77=before tax.)

If UGROSS exists
 and UBONINC=1 and UBONAMT exists
 $INEARNS=INEARNS-(UBONAMT/0.77)$

(UBONAMT is on a net basis, assume that if included in net pay ie UBONINC = yes. then it will also have been in usual gross pay; UBONAMT is only asked if UBONINC = yes.

Adjustment for possible receipt of SSP or SMP

NOTE : The section of code which actually works out the adjustment is contained within innirben code. The inearns code just takes of the adjustment which has been calculated earlier on.

Set ADJUST (temporary variable) and INOTHBEN to zero

For JOBTYP=1 only (SSP/SMP questions only asked once)

**If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYSLIP = 1 or 2
 Calculate ADJUST = SSPAMT**

**If JOBAWAY = 1 and ABSWHY = 6 and (SSPSMP = 1 or 3) and PAYSLIP = 1 or 2
 Calculate ADJUST = ADJUST + SMPAMT**

**If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYSLIP= 3 and
 SSPRATE=1 or 2 Calculate ADJUST =52.50**

**If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSLIP= 3 and
 SMPRATE=1 or 2 Calculate ADJUST=ADJUST+52.50**

If ADJUST>INEARNS, then leave INEARNNS. This is in line with HBAI assumptions that where SSP is bigger than the pay it's supposedly in then the respondent has made a mistake and has actually quoted their pay after SSP/SMP.

**Else INEARNNS=INEARNNS-ADJUST.
 Add ADJUST to INOTHBEN**

This adjustment is to check whether any SMP/SSP is included in gross earnings and to switch it to benefit income. Previous specifications of GROSSINC reduced earnings in respect of SSP/SMP where a payslip was consulted but did not include amounts with benefits because the benefits table did not record them They therefore undercounted income by the rates of SSP/SMP in appropriate cases. Within the pay block, if a payslip is consulted (payslip=1 or 2), respondents are asked for the amount

--

included (SSPAMT/SMPAMT) otherwise, they are asked what rate was in payment (SSPRATE/SMPRATE). These are standard rates –one standard rate for SSP and two rates for SMP (but higher SMP is used as this is in line with HBAI). These standard rates are input into DVMETA.XLS which contains constants – so that they can be easily updated year on year Earned income equals amount calculated at INEARNs less the estimated SMP/SSP (the adjustment). ADJUST is added to INOTHBEN.

Other sources of income: INRINC

Set INRINC to zero

Income as occasional work, professional advice

From ODDJOB record, if ODDTYPE=3 and OJNOW=1 then add OJAMT to INRINC

Income as a baby-sitter

From ODDJOB record, if ODDTYPE=1 and OJNOW=1 then add OJAMT to INRINC
~~From ADULT record, if BABY1 = 1 and BABNOW=1 (doing work as a baby sitter currently) add BABPAY into INRINC~~

Income as a mail order agent

From ODDJOB record, if ODDTYPE=2 and OJNOW=1 then add OJAMT to INRINC
~~From ADULT record, if BABY2 = 1 and BABNOW=1 add BABPAY into INRINC~~

Allowance from absent spouse

From ADULT record, if ABSPAR = 1 and APPD equals -1 or 1 to 10 or 13 or 26 or 52, add APAMT to INRINC.

If APDIR=1 and APDPD equals -1 or 1 to 10 or 13 or 26 or 52 add APDAMT to INRINC

Allowances from spouse in forces, friends other relatives etc

From ADULT record, if ALLOW1 = 1 and ALLPD1 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY1 to INRINC.

Allowance from an organisation

From ADULT record, if ALLOW2 = 1 and ALLPD2 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY2 to INRINC.

Allowance from a Local Authority for a foster child

-7-

--

From ADULT record, if ALLOW3 = 1 and ALLPD3 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY3 to INRINC.

Allowance from a Local Authority for an adopted child.

From ADULT record, if ALLOW4 = 1 and ALLPD4 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY4 to INRINC.

Income in kind

From JOB record, if JOBTYP=1 and LVAMT exists add amount in LVAMT into INRINC (luncheon vouchers) else do not change INRINC (*lvamt only asked if lunchv=1 and lv7dy=1, and only asked of first job*)

Rent from any property before tax

From ADULT record, if ROYAL1 = 1 add ROYYR1 into INRINC

Royalties

From ADULT record, if ROYAL2 = 1 add ROYYR2 into INRINC

Income as a sleeping partner

From ADULT record, if ROYAL3 = 1 add ROYYR3 into INRINC.

Pension from an overseas Government

From ADULT record, if ROYAL4 = 1 add ROYYR4 into INRINC.

Maintenance

From ADULT record, if MNTREC = 1 and MNTPD1 equals -1 or 1 to 10 or 13 or 26 or 52, add MNTAMT1 into INRINC.

From ADULT record, if MNTREC = 1 and MNTPD2 equals -1 or 1 to 10 or 13 or 26 or 52, add MNTAMT2 into INRINC.

~~**Odd jobs**~~

~~From ADULT record if ODDJOB =1
for all occurrences of OJAMT in ODDJOB record
(note that OJAMT converted to weekly amount in conversion process)~~

~~**Income from property**~~

~~If PROPARENT exists add PROPARENT to INRINC.~~

Income from sub-tenants

-8-

 --

If SUBLET = 1, add SUBRENT into INRINC for HOH = 1 (head of household).

Income from those outside the household paying towards rents/mortgages

For rented property, all contributions are included excluding contributions from DSS, consistent with HBAI. The income is assumed to be for the head of household
 From RENTCONT record if ACCNONHH =1 and ACCPAY <> 1 and ACCAMT exists and RENTHOL=2 and ACCPD=-1 or 1-10 then $INRINC=INRINC+ACCAMT$ for HOH = 1 only (head of household). If ACCNONHH=1 and ACCPAY <> 1 and ACCAMT exists and ACCPD=-1 or 1-11 and WEEKHOL exists multiply ACCAMT by the number of weeks rent actually paid (52-WEEKHOL) and divide the result by 52. Add this to INRINC for **HOH = 1** only

For those buying their house with a mortgage, contributions from outside are included for all cases except the DSS. This is to avoid double counting of direct payments which are already included as part of benefit income. Amounts are included for all types of mortgage (having been explicitly added back for endowment mortgages where appropriate and implicit in repayment mortgages calculation)
 From MORTCONT record if OUTSMORT=1 and OUTSPAY=2,3,4 or 5 (ie not equal to 1) OUTSAMT exists and OUTSPD=-1 or 1-11 then $INRINC=INRINC+OUTSAMT$ for HOH = 1 only.

Income from education grants

If TOTGRANT exists add TOTGRANT/52 to INRINC

Income from free welfare milk, free school meals and free school milk

(these are assumed to be income of the head of benefit unit - largely for convenience, individual amounts are not held on the data base)

If from BENUNIT record FWMLKBU exists add FWMLKBU to INRINC *for head of benefit unit only*

If from BENUNIT record FSMBU exists add FSMBU to INRINC *for head of benefit unit only*

If from BENUNIT record FSMLKBU exists add FSMLKBU to INRINC *for head of benefit unit only*

Income from student loans

If from ADULT record TUBORR exists add TUBORR/52 to INRINC

Interest/income from savings accounts or investments: ININV

set ININV to zero

From ACCOUNTS record,

--

if ACCOUNT = 1 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 1 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 2 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 2 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 3 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 3 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 4, add amount in ACCINT to ININV
 if ACCOUNT = 5 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 5 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 6 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 6 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 7, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 8, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 9, add amount in ACCINT to ININV

Note: Accounts 7 (Unit Trusts) and 8 (Stocks & Shares) are assumed net of tax, so we do need to add the tax back in. There is no ACCTAX check for these accounts.

Income from parents to students

If from ADULT record PareAmt exists,
 add PareAmt / 52 to INRINC if ParePd in (95,97)
 else add PareAmt to INRINC

Personal pensions: INPENINC

set INPENINC to zero

Occupational pensions

From PENSIONS record, if PENTYPE = 1 (occupational pension) and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC). If PENOTH exists and equal to 1 and POINC exists and =2 (other deductions from pension not included in figure at PENPAY) and POAMT exists add POAMT to INPENINC otherwise, do not change INPENINC.

(this is the same approach as taken in OCCPEN to deal with skipped values)

Widow's employee pension

From PENSIONS record, if PENTYPE = 2 (widow's employee pension) and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=INPENINC+PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC). If PENOTH exists and equal to 1 and POINC exists and =2 (other deductions from pension not included in figure at PENPAY) and POAMT exists add POAMT to INPENINC otherwise, do not change INPENINC.

--

Personal pension

From PENSIONS record, if PENTYPE = 3 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=INPENINC+PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC)

Trade union Friendly society pensions

From PENSIONS record, if PENTYPE = 4 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

get amount from PENPAY and add to INPENINC If PTINC = 2 add PTAMT then add result into INPENINC

Annuity pension

From PENSIONS record, if PENTYPE = 5 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=INPENINC+PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC)

Trust/covenant

From PENSIONS record, if PENTYPE = 6 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

get amount from PENPAY and add to INPENINC. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC)

Income from benefits: INRPINC, INDISBEN, INOTHBEN (plus bits of INRINC held in BENEFITS record)

set INRPINC, INDISBEN to zero

Housing Benefit

if BENUNIT = 1 and PERSON = 1: From RENTER record, if RENTHOL = 1 and WEEKHOL exists and HBENEFIT = 1 and HBENPD equals -1 or 1 to 11 or 13 or 26 or 52, multiply HBENAMT by number of weeks rent actually paid (52 - WEEKHOL) and divide the result by 52. Add to INOTHBEN. If RENTHOL ne 1 and HBENEFIT = 1 and HBENPD equals -1 or 1 to 10 or 13 or 26 or 52, add HBENAMT to INOTHBEN.

From HOUSEHOL record, (if HHSTAT = 2 or CVPAY>0 (boarders/lodgers paying rent)) then from ADULT record if H BOTHBU=1 (individual qualifies for HB) and CVPD equals -1 or 1 to 10 or 13 or 26 or 52, then add H BOTHAMT to INOTHBEN.

 --

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to 10 or 13 or 26 or 52 add CTREBAMT into INOTHBEN only for PERSON = 1.

Income from benefits

From BENEFITS record, if BENPD equals -1 or 1 to 10 or 13 or 26 or 52, and

If BENEFIT = 1 add BENAMT to INDISBEN (DLA Care)

If BENEFIT = 2 add BENAMT to INDISBEN (DLA Mob)

If BENEFIT = 3 add BENAMT to INOTHBEN (CHB)

If BENEFIT = 5 add BENAMT to INRPINC (RP)

If BENEFIT = 6 add BENAMT to INOTHBEN (Widows Pension)

If BENEFIT = 7 add BENAMT to INOTHBEN (Widowed Mothers allowance)

If BENEFIT = 8 add BENAMT to INDISBEN (War Disablement Pension)

If BENEFIT = 9 add BENAMT to INOTHBEN (War Widows Pension)

If BENEFIT = 10 add BENAMT to INDISBEN (SDA)

If BENEFIT = 11 add BENAMT to INDISBEN (DWA)

If BENEFIT = 12 add BENAMT to INDISBEN (AA)

If BENEFIT = 13 add BENAMT to INOTHBEN (Invalid Care Allowance)

If BENEFIT = ~~13~~ 14 add BENAMT to INOTHBEN JSA

If BENEFIT = 15 add BENAMT to INDISBEN (Industrial Injuries)

~~If BENEFIT = 16 add BENAMT to INOTHBEN (SSP)~~

If BENEFIT = 17 add BENAMT to INDISBEN (Incapacity Benefit)

If BENEFIT = 18 add BENAMT to INOTHBEN (FC)

If BENEFIT = 60 add BENAMT / 52 to INOTHBEN (widow's payment)

If BENEFIT = 19 and (SEX=1 and AGE>=65) or (SEX=2 and AGE>=60) add BENAMT to INRPINC (IS)

else add BENAMT to INOTHBEN

 --

If BENEFIT = 65 and VAR2=2
 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)) add BENAMT to
 INRPINC

else if BENEFIT= 65 and VAR2=2 add BENAMT to INOTHBEN
*(adjustment for IS quoted after any direct payments: pensioners income must be
 included in INRPINC whilst others are in INOTHBEN)*

If BENEFIT = 66 and VAR2=2 add BENAMT to INOTHBEN
 If BENEFIT = 21 add BENAMT to INOTHBEN (Maternity Allowance)

~~If BENEFIT = 23 add BENAMT to INOTHBEN (SMP)~~

~~If BENEFIT = 28 and HBINDBU in (0,1,2,3,5,6) and HOH = 1 add BENAMT / 4 to
 INOTHBEN (Extended HB or extended HB and CTB)~~

~~If BENEFIT = 29 and HBINDBU in (0,2,3,5) and HOH = 1 add BENAMT / 4 to
 INOTHBEN (Extended CTB)~~

If BENEFIT = 78,80 and HBINDBU = 0→3,5,6 and UPERSON=1 add BENAMT / 4

If BENEFIT = 79 and HBINDBU = 0,2,3,5 and HOH =1 add BENAMT / 4

If BENEFIT = 30 add BENAMT to INOTHBEN (Any other DSS benefits)

If BENEFIT = 37 add BENAMT to INOTHBEN (Guardians Allowance)

If BENEFIT = 69 and var2=1 subtract BENAMT from INOTHBEN (social fund repayment
 - IS)

If BENEFIT = 70 subtract BENAMT from INOTHBEN (social fund repayment – JSA)
 (for v34 the question Sflnc, “was the amount of IS/JSA you said earlier before or after
 taking off your Social Fund Loan repayments?” was only asked if they didn’t know or
 refused to say how much they were repaying. Anyone not asked this question is
 assumed to have quoted an amount that was after any repayments have been taken off)

If BENEFIT = 22 and BENAMT/26 to INOTHBEN (Maternity Grant)

If BENEFIT = 24 and BENAMT/26 to INOTHBEN (Funeral Grant)

If BENEFIT = 25 and BENAMT/26 to INOTHBEN (Comm Care Grant)

If BENEFIT = 41 and BENAMT/26 to INOTHBEN (FC Lump Sum)

If BENEFIT = 26 and BENAMT/26 to INOTHBEN (Back to work bonus)

Note: following benefits are included in INRINC

If BENEFIT = 31 and PRES = 1 and BENPD=-1 or 1-10 or 13 or 26 or 52 add BENAMT
 to INRINC (Trade Union sick) else if PRES=1 and BENPD=90, 95 or 97 do not change
 INRINC

 --

If BENEFIT = 32 and PRES = 1 and BENPD=-1 or 1-10 or 13 or 26 or 52 add BENAMT to INRINC (Friendly sick) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 33 and PRES = 1 add BENAMT to INRINC (Private sick) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT =34 and PRES = 1 add BENAMT to INRINC (Accident) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 35 and PRES = 1 add BENAMT to INRINC (Hospital savings) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 36 add BENAMT to (Training) INRINC

If BENEFIT = **61** and PRES = 1 and BENPD=-1 or 1-11 or **13 or 26 or 52** add BENAMT to INRINC (**Unemployment/Redundancy Insurance**) else if PRES=1 and BENPD=**90, 95 or 97** do not change INRINC

INDINC will then be calculated as follows for each ADULT -

$INEARNS + SEINCAM2 + INSEINC + ININV + INRPINC + INPENINC + INDISBEN + INOTHBEN + INRINC$

-2 If any of above variables are missing or if a period code is 90 or 95 or 97 (skipped is OK) also applied to individual component variables ININV, INRPINC, INPENINC, INDISBEN, INOTHBEN and INRINC which are calculated in this spec (missing INEARN components are already documented above).

 --

INIRBEN, INNIRBEN, BUIRBEN, BUNIRBEN, HHIRBEN, HHNIRBEN**(See also HDBEN spec for INIRBEN)**

Purpose : To show the total amount of income received from income (means tested) and non-income related (non-means tested) benefits at an individual, benefit unit and household level for use in the FRS publication.

Created : 5 February 1996

Database Table : ADULT, BENUNIT, HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : FRS (publication)

Amendments : VE - 5 June 1996 - To note that on April 13 1995 Incapacity Benefit replaced Invalidity Benefit and NI Sickness Benefit. Specification not changed because schema has not yet been amended to reflect this change.

 : 20 August 1996 - updated for V32

 : VE - 12 March 1997 - To update for V33 - new HB questions for sharers/boarders, new question numbers, new vague periods

 : SCG - 3 November 1997 - fix period codes, retain HDBEN properly

 : SCG - 21 November 1997 - HB rent holidays, take out social fund

 : SCG - 31 December 1997 - correct benefits according to current code and consistency with INDISBEN, INOTHBEN

 : SCG - 11 February 1998 - PAYSLIP changed in V33

 : SCG - 13 March 1998 - Check spec against code

 : SCG - 27 March 1998 - Correct routing if RENTHOL=1 and WEEKHOL=.A

 : SCG - 1 June 1998 - Add extended HB/CTB where appropriate

 : SCG - 8 June 1998 - Social Fund crisis loan should not be included, BTW bonus is income related

 : SCG - 17 June 1998 - Rename BU and HH level flag variables

 : SCG - 13 October 1998 - JSA income related for certain cases

 : EP - 22 October 1998 - removal of DV_const call for V34

 : EP - 30 November 1998 - change head of household from person=1 to hoh=1

 : EP - 8 December 1998 - For JSA, add check to only include contributory based

 : EP - 8 December, 1998 - Change benefit 20 references to benefits 65 and 66 and subtract Social Fund Loan Repayments (IS) from INIRBEN and (JSA) from INNIRBEN

 : SB - 23 September 1999 - Security completed, no other changes for V35

Issued : 17 April, 2003

1 Definition

This variable is coded as

 --

INIRBEN The total amount of income received each week by individuals from income related benefits.

0 No income is received from income related benefits

-2 Unable to derive due to missing values

INNIRBEN The total amount of income received each week by individuals from non-income related benefits

0 No income is received from non-income related benefits

-2 Unable to derive due to missing values

Income related benefits are: IS, FC, DWA, HB, CTB **and Social Fund payments and Back to Work Bonus**

Non-income related benefits are: SSP, SMP, DLA, CHB, OPB, RP, Widows' pension, War disablement pension, SDA, AA, ICA, UB, Industrial Injuries, Incapacity Benefit, Maternity benefit, Guardians Allowance

Other DSS benefits are included under non-income related benefits

Note: It may be possible to code these variables as part of INDINC as most of the coding is the same. If not, these DVs should be run after INDINC because they use INEARNs for SSP/SMP

INIRBEN+INNIRBEN=INRPINC+INDISBEN+INOTHBEN
 INRPINC includes both mean and non-income related benefits

The variables are used in tabulations as analysis and categorical variables, eg proportion of income from income related benefits and in receipt of income related benefits. In receipt is identified as where the amount is >0.

2 FRS Specification

For each adult

Code Condition

Set INIRBEN and INNIRBEN to zero and get INEARNs

SSP or SMP

--

set ADJUST (temporary variable) to zero

If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYSLIP = 1 or 2
Calculate ADJUST = SSPAMT

If JOBAWAY = 1 and ABSWHY = 6 and (SSPSMP = 1 or 3) and PAYSLIP = 1 or 2
Calculate ADJUST = ADJUST + SMPAMT

If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYSLIP= 3 and
SSPRATE=1 Calculate ADJUST =52.50

If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYSLIP= 3 and
SSPRATE=2 Calculate ADJUST=47.80

If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSLIP= 3 and
SSPRATE=1 Calculate ADJUST=ADJUST+52.50

If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSLIP= 3 and
SSPRATE=2 Calculate ADJUST=ADJUST+48.80

If ADJUST >= to employment income calculated in INEARNs, reset ADJUST to
INEARNs

Add ADJUST to **INNIRBEN**

Housing Benefit

From RENTER record, if BENUNIT = 1 and HBENPD equals -1 or 1 to 52, and
HBENEFIT = 1 add in HBENAMT to **INNIRBEN** for PERSON = 1 and **HOH = 1**.

From HOUSEHOL record, if HHSTAT=2 or CVPAY>0 then from ADULT record if
HBOHBU=1 and HBOHHPD equals -1 or 1 to 52 then add HBOHAMT to **INNIRBEN**.

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to 52 add CTREBAMT to **INNIRBEN** for
PERSON = 1 only.

Other benefits

From BENEFITS record, if BENPD equals -1 or 1 to 52 and

If BENEFIT = 1 add BENAMT to **INNIRBEN** (DLA Care)

If BENEFIT = 2 add BENAMT to **INNIRBEN** (DLA Mob)

If BENEFIT = 3 add BENAMT to **INNIRBEN** (CHB)

 --

If BENEFIT = 4 add BENAMT to **INNIRBEN** (OPB)

If BENEFIT = 5 add BENAMT to **INNIRBEN** (RP)

If BENEFIT = 6 add BENAMT to **INNIRBEN** (Widows Pension)

If BENEFIT = 7 add BENAMT to **INNIRBEN** (Widowed Mother's Allowance)

If BENEFIT = 8 add BENAMT to **INNIRBEN** (War Disablement Pension)

If BENEFIT = 9 add BENAMT to **INNIRBEN** (War Widow's Pension)

If BENEFIT = 10 add BENAMT to **INNIRBEN** (SDA)

If BENEFIT = 11 add BENAMT to **INIRBEN** (DWA)

If BENEFIT = 12 add BENAMT to **INNIRBEN** (AA)

If BENEFIT = 13 add BENAMT to **INNIRBEN** (Invalid Care Allowance)

If BENEFIT = 14 and VAR2=1 add BENAMT to **INNIRBEN** (~~UB~~ Jobseeker's Allowance)

If BENEFIT = 15 add BENAMT to **INNIRBEN** (Industrial Injuries)

If BENEFIT = 17 add BENAMT to **INNIRBEN** (Incapacity Benefit)

If BENEFIT = 18 add BENAMT to **INIRBEN** (FC)

If BENEFIT = 19 add BENAMT to **INIRBEN** (IS)

~~If BENEFIT = 20 and VAR2=2 add BENAMT to **INIRBEN** (Direct payments not included in quoted IS)~~

If BENEFIT = 21 add BENAMT to **INNIRBEN** (Maternity Allowance)

If BENEFIT=22 add BENAMT/26 to **INIRBEN** (Soc Fund Grant for Maternity Expenses)

If BENEFIT=24 add BENAMT/26 to **INIRBEN** (Soc Fund Grant for Funeral Expenses)

If BENEFIT=25 add BENAMT/26 to **INIRBEN** (Comm Care Grant)

If BENEFIT = 26 add BENAMT/26 to ~~INNIRBEN~~ INIRBEN (Back to work bonus)

If BENEFIT = 28 and HBINDBU in (0,1,2,3,5,6) and PERSON=1 add BENAMT / 4 to **INIRBEN** (Extended HB or Extended HB and CTB)

 --

If BENEFIT = 29 and HBINDBU in (0,2,3,5) and PERSON=1 add BENAMT / 4 to INIRBEN (Extended CTB)

If BENEFIT = 30 and PRES = 1 add BENAMT to **INNIRBEN** (Any other DSS benefits)

If BENEFIT = 37 add BENAMT to INNIRBEN (Guardians Allowance)

If BENEFIT=~~38~~ **69** and VAR2=~~2~~ **1** ~~add~~ **subtract** BENAMT ~~to~~ **from** **INIRBEN** (Social Fund loan repayments IS)

If BENEFIT = 70 and VAR2=1 subtract BENAMT from INNIRBEN (social fund loan repayments JSA)

(for v34 the question Sflnc, "was the amount of IS/JSA you said earlier before or after taking off your Social Fund Loan repayments?" was only asked if they didn't know or refused to say how much they were repaying. Anyone not asked this question is assumed to have quoted an amount that was after any repayments have been taken off)

If BENEFIT= 41 add BENAMT/26 to **INIRBEN** (FC Lump Sum)

If BENEFIT = 16 add BENAMT to INNIRBEN (SSP)

If BENEFIT = 23 add BENAMT to INNIRBEN (SMP)

If BENEFIT = 65 add BENAMT to INIRBEN (DSS direct payments – IS)

If BENEFIT = 66 and income-related add BENAMT to INIRBEN (DSS direct payments – JSA)

If BENEFIT = 66 and non-income related add BENAMT to INNIRBEN (DSS direct payments – JSA)

-2 variables set to unable to derive if any components are missing or period codes equal 90 or 95 or 97

The benefit unit variables are calculated as:

For each benefit unit

from ADULT table get INIRBEN and INNIRBEN

BUIRBEN equals total occurrences of INIRBEN

BUNIRBEN equals total occurrences of INNIRBEN

-2 if any components are missing

--

The household variables are calculated as:

For each household

from BENUNIT table get BUIRBEN and BUNIRBEN

HHIRBEN equals total occurrences of BUIRBEN

HHNIRBEN equals total occurrences of BUNIRBEN

-2 if any components are missing

 --

KIDS0BU, KIDS1BU,.....,KIDS18BU, KID04, KID510, KID1115, KID1618

Purpose : Total number of dependants aged under 1, aged from 1 to 2 years,....., and to indicate the number of children in the following age bands, 0 to 4, 5 to 10,...

Created : NM - 3 November 1992

Database Table : BENUNIT

Minimum Value : 0

Maximum Value :

Units : Integer

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : FRS (publication)

Amendments : VC - 9 February 1993 updated to version 29
 : VC - 11 May 1993 amended to show coding
 : VC - 17 February 1994 Amended to reflect version 30 changes
 : VE - 22 May 1996 - No initial Version 32 update needed
 : SG - 12 November 1997 - No initial Version 33 update needed
 : EP - 13 August 1998 - No initial V34 update needed
 : JC - 17 September 1999 - Security completed, no other changes for V35

Issue date : 17 April 2003

1 Definition

This variable is coded as

KIDS0BU	Number of dependants under age 1
KIDS1BU	Number of dependants between age 1 and 2
KIDS2BU	Number of dependants between age 2 and 3
KIDS3BU	Number of dependants between age 3 and 4
KIDS4BU	Number of dependants between age 4 and 5
KIDS5BU	Number of dependants between age 5 and 6
KIDS6BU	Number of dependants between age 6 and 7
KIDS7BU	Number of dependants between age 7 and 8
KIDS8BU	Number of dependants between age 8 and 9
KIDS9BU	Number of dependants between age 9 and 10
KIDS10BU	Number of dependants between age 10 and 11
KIDS11BU	Number of dependants between age 11 and 12
KIDS12BU	Number of dependants between age 12 and 13
KIDS13BU	Number of dependants between age 13 and 14
KIDS14BU	Number of dependants between age 14 and 15
KIDS15BU	Number of dependants between age 15 and 16
KIDS16BU	Number of dependants between age 16 and 17
KIDS17BU	Number of dependants between age 17 and 18
KIDS18BU	Number of dependants between age 18 and 19
KID04	This is the number of children in each benefit unit aged 0 to 4 years inclusive
KID510	This is the number of children in each benefit unit aged 5 to 10 years inclusive

 --

KID1115 This is the number of children in each benefit unit aged 11 to 15 years inclusive
 KID1618 This is the number of dependants in each benefit unit aged 16 to 18 years inclusive

-1 Not applicable to this case.
 -2 Unable to derive due to missing values.

The above are derived from the variable age in the CHILD table

2 FRS Specification

For each BENUNIT record

Code Condition

Using variable AGE from CHILD table which holds details of all children aged 15 and under and all 16 to 18 year olds in full-time, non-advanced education.

KIDS0BU The sum of all dependants in the benefit unit where age < 1
 KIDS1BU The sum of all dependants in the benefit unit where age >= 1 and < 2
 KIDS2BU The sum of all dependants in the benefit unit where age >= 2 and < 3
 KIDS3BU The sum of all dependants in the benefit unit where age >= 3 and < 4
 KIDS4BU The sum of all dependants in the benefit unit where age >= 4 and < 5
 KIDS5BU The sum of all dependants in the benefit unit where age >= 5 and < 6
 KIDS6BU The sum of all dependants in the benefit unit where age >= 6 and < 7
 KIDS7BU The sum of all dependants in the benefit unit where age >= 7 and < 8
 KIDS8BU The sum of all dependants in the benefit unit where age >= 8 and < 9
 KIDS9BU The sum of all dependants in the benefit unit where age >= 9 and < 10
 KIDS10BU The sum of all dependants in the benefit unit where age >= 10 and < 11
 KIDS11BU The sum of all dependants in the benefit unit where age >= 11 and < 12
 KIDS12BU The sum of all dependants in the benefit unit where age >= 12 and < 13
 KIDS13BU The sum of all dependants in the benefit unit where age >= 13 and < 14
 KIDS14BU The sum of all dependants in the benefit unit where age >= 14 and < 15
 KIDS15BU The sum of all dependants in the benefit unit where age >= 15 and < 16
 KIDS16BU The sum of all dependants in the benefit unit where age >= 16 and < 17
 KIDS17BU The sum of all dependants in the benefit unit where age >= 17 and < 18
 KIDS18BU The sum of all dependants in the benefit unit where age >= 18 and < 19

KID04 For each child in benefit unit, from CHILD table
 If age >= 0 and <= 4, count total number of children where age falls in this range

KID510 For each child in benefit unit, from CHILD table.
 If age >= 5 and <= 10, count total number of children where age falls in this range

KID1115 For each child in benefit unit, from CHILD table
 If age >= 11 and <= 15, count number of children where age falls in this range

KID1618 For each child in benefit unit, from CHILD table

--

If age ≥ 16 and ≤ 18 , count total number of children where age falls in this range.

NB - The CHILD table includes all children aged 15 and under and those aged 16 to 18 inclusive who are in non-advanced full-time education.

-1 Not applicable to this case - should not occur for these variables.

-2 Unable to derive due to missing values.

3 Results

Tabulation is required to show the number of dependants in each benefit unit by age of the dependent.

4 Test Cases

None as yet

LASTWORK

Purpose : Time since the head of a benefit unit, where the head or spouse is unemployed, last worked
Created : 22 February 1999
Database Table : BENUNIT
Minimum Value : 1
Maximum Value : 8
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (publication)
Amendments : SB - 9 November 1999 – Security completed, no other changes to V35
Issue date : 22 February 1999

1 Definition

LASTWORK shows the length of time since the head of a BENUNIT last worked, if the head or spouse of a BENUNIT is unemployed. It is derived from the variables EVERWRK, LSTWRK1, LSTWRK2, EMPSTATI and UPERSON on the ADULT table, INTDATE on the HOUSEHOL table, and ECSTATBU and FAMTYPBU on the BENUNIT table. It is coded as follows:

- | | |
|---|-------------------------------|
| 1 | Head currently in work |
| 2 | Head never worked |
| 3 | Less than 6 months |
| 4 | 6 months and less than a year |
| 5 | 1 year and less than 2 years |
| 6 | 2 years and less than 5 years |
| 7 | 5 years or more |
| 8 | Missing |

2 FRS Specification

For each benefit unit

<u>Code</u>	<u>Condition</u>
-------------	------------------

If ECSTATBU=7 (Single or couple, head or spouse unemployed)	
and FAMTYPBU in (3,4,5,6)	(Couple with children, couple without children, lone parent, single without children)

then do

LAST = MDY (LSTWRK1, 1, LSTWRK2)
MONTHS = INTCK ('MONTH', LAST, INTDATE)

- | | |
|---|---|
| 1 | If EMPSTATI in (1,2,3,4) (Full-time employee, part-time employee, full-time self-employed, part-time self-employed) |
| 2 | If EVERWRK=2 |

-
- 3 If MONTHS \geq 0 and MONTHS $<$ 6
- 4 If MONTHS $>$ 5 and MONTHS $<$ 12
- 5 If MONTHS $>$ 11 and MONTHS $<$ 24
- 6 If MONTHS $>$ 23 and MONTHS $<$ 60
- 7 If MONTHS $>$ 59
- 8 If ECSTATBU=7 and FAMTYPBU in (3,4,5,6) and LASTWORK not in (1,2,3,4,5,6,7)
- 1 Not applicable in this case – (If (If ECSTATBU=7 and FAMTYPBU in (3,4,5,6)) does not hold)
- 2 Unable to derive LASTWORK

 --

LODGER

Purpose : To indicate the total weekly amount of rent paid by a lodger.
 Created : VC - 12 March 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : VC - 24 April 1993 Change from adult to BU variable
 : VC - 27 April 1993. To expand definition to include more details about the
 : questions/database variables used.
 : VC - 17 February 1994 Amended to reflect changes to version 30
 : VC - 1 March 1994 To exclude period codes 12 or 13
 : JS - 21 February 1996 to allow skipped CVPD where CVPAY has been
 : imputed
 : VE - 23 May 1996 - No initial V32 update needed
 : SG - 3 December 1997 - V33 update
 : EP - 10 September 1998 - No initial V34 update needed
 : SB - 7 September 1999 - Security completed, no other changes for V35
 : SB - 25 November 1999 - Add in housing benefit rates
 Issue date : 17 April 2003

NB - This is a new variable produced by FRS and does not replace FES in any way.

1 Definition

This variable is coded as

LODGER The total weekly amount paid by a benefit unit classed as a lodger to the householder
 for a **room but no food**.

-1 Not applicable to this case.

-2 Unable to derive due to missing values.

LODGER is derived by benefit unit from the variable CONVBL which asks whether the person is a boarder or lodger or neither of these. Where CONVBL = 2 indicating that the adult is a lodger, the amount paid for lodging is to be found in CVPAY. **If the benefit unit receives any housing benefit then this can also be added in.**

However, if the period code for the amount of rent paid by the lodger is 90, 95 or 97 (lump-sum/one-off or other period) the record must be set to unable to derive, as it has not been possible to convert the

 --

amount of benefit into a weekly amount during the database conversion process. Therefore, if CVPD = 90, 95 or 97 LODGER is set to -2.

2 FRS Specification

For each BENUNIT record and for each adult in the benefit unit

<u>Code</u>	<u>Condition</u>
LODGER	From ADULT table If CONVBL = 2 and CVPD equals -1 or 1-10,13,26 or 52, the amount of LODGER is the amount in CVPAY. If CONVBL = 2 and HBOTHBU = 1 (receives housing benefit) then add HBOTHAMT to LODGER if HBOTHPD equals -1 or 1-10,13,26 or 52. If there is more than one adult in the benefit unit, the amount of LODGER is the total amount paid by both adults. If only one adult in a two adult benefit unit is a lodger then LODGER is set to .A
-1	Not applicable to this case - where CONVBL = 1 or 3 or is missing as the question has not been asked.
-2	Unable to derive due to value of CVPAY missing or CVPD = 90, 95 or 97.

3 Results

Tabulation is required to show the number of lodgers paying rent in weekly bands of

Under £25
 £25 - £50
 £50 - £75
 £75 - £100
 £100 - £125
 £125 - £150
 £150 or over

4 Test Cases

None as yet

LONDON

Purpose : To flag those households in inner / outer London
 Created : 8 March 1999
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 3
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : Take-Up
 Amendments : JC -6 September 1999- Security completed, no other changes for V35
 Issue date : 8 March 1999

1 Definition

This derived variable flags the households that are located in inner or outer London:

- 1 Inner London
- 2 Outer London
- 3 Household not in London

2 FRS Specification

For each Household

Code Condition

- 1 INNER=0;
- If LAC in (649 City of London
 647 Camden
 650 Hackney
 356 Hammersmith
 654 Haringey
 648 Islington
 449 Kensington
 961 Lambeth
 962 Lewisham
 651 Newham
 963 Southwark
 652 Tower Hamlets
 257 Wandsworth
 450 Westminster)
- Then INNER=1;
- If INNER=1 Then LONDON=1;
- 2 OUTER=0;
- If LAC in (555 Barking/Dagenham
 352 Barnet
 958 Bexley
 353 Brent)

960 Bromley
162 Croydon
355 Ealing
653 Enfield
959 Greenwich
354 Harrow
556 Havering
448 Hillingdon
447 Hounslow
163 Kingston-U-Thames
165 Merton
655 Redbridge
164 Richmond-U-Thames
166 Sutton
656 Waltham Forest)

Then OUTER=1;

If OUTER=1 Then LONDON=2;

3 If INNER=0 and OUTER=0

Then LONDON=3;

-2 Unable to Derive (should not happen in this case)

--

MARITAL

Purpose : To show marital status for publication
Created : 1 June 1998
Database Table : ADULT
Minimum Value : 1
Maximum Value : 6
Units : Integer
Validations :
Related Variables :
Children :
Parents :
Core variable/user : FRS (publication)
Amendments : EP - 13 August 1998 - No initial V34 update, though ADULTB now a DV
JC - 8 September 1999 - Security completed, no other changes for V35
Issue date : 1 June 1998

1 Definition

This variable is coded as

- 1 Married
- 2 Cohabiting
- 3 Single
- 4 Widowed
- 5 Separated
- 6 Divorced

MARITAL is derived from two variables in the ADULT table which indicate legal marital status and also whether an individual is cohabiting. In combination these variables provide the necessary information to derive this DV which is primarily for publication purposes.

The MS variable has the following values :

- 1 Single, never married
- 2 Married, living with spouse
- 3 Married and separated
- 4 Divorced

--

5 Widowed
-2 Unable to derive

In most cases these map straight to values for the DV. Where the individual is single and never married, or divorced, separated or widowed a check is made as to whether COHAB=1, in which case they are now cohabiting. Such individuals are treated as cohabiting. Same sex couples, whilst classified as cohabiting in the FRS must be treated as single (or divorced or separated) as appropriate. This can be checked by the value of ADULTB (adults in benefit unit). If married and spouse living in household (MS=2) but no other adult in the benefit unit as actually living away (SPOUT=1) then set as married.

2 FRS Specification

For each adult

Code Condition

MARITAL= -2
if MS = (1 or 3 or 4 or 5) and (ADULTB>1 and COHAB=1) then MARITAL=2
else if MS=1 and ADULTB=1 then MARITAL=3
else if MS=3 and ADULTB=1 then MARITAL=5
else if MS=4 and ADULTB=1 then MARITAL=6
else if MS=5 and ADULTB=1 then MARITAL=4
else if MS=2 and (ADULTB>1 or SPOUT<2) then MARITAL=1

3 Results

Tabulation required to show number of adults falling into each category.

--**MORTCOST, MORTPAY, ENDOWPAY, STRUINS, SERVPAY**

Purpose : To show weekly housing expenditure for owner occupiers for use in the FRS publication.

Created : 2 February 1996

Database Table : HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : HSCOSTHH, MORTINT

Children :

Parents : MORTPAY, ENDOWPAY, STRUINS, SERVPAY (also derived here)

Core variable/user : **FRS (publication)**, Regional Trends (**Bold** indicates lead user)

Amendments : To correct identification of mortgage protection policies
: VE - 4 June 1996 - No amendments needed for SERVPAY in initial V32 update
: VE - 5 June 1996 - No amendments needed for other DVs in V32 update
: VE - 20 November 1996 - To amend STRAMT to STRAMT1 and STRAMT2 where appropriate
: VE - 17 December 1996 - to amend INCM PAMT and INCM PPD to INCM PAM1, 2 and 3 and INCM PPD1, 2 and 3
: SG - 29 October 1997 - V33 changes, tenure codes, period codes
: EP - 13 August 1998 - V34 update - change CHAMT1-7 to CHR GAMT1-7
: SB - 15 October 1999 - Security completed, no other changes for V35

Issue date : 17 April 2003

1 Definition

This variable is coded as

MORTCOST Total weekly housing (mortgage) costs of owner occupiers, including mortgage payments, endowment policies, structural insurance and service payments

0 Owner occupiers who have no housing costs

-1 Not applicable to this case (renter households)

-2 Unable to derive due to missing values

MORTCOST is derived from variables which occur in HSCOSTHH. The total is broken down into components which are to be used in a table for Regional Trends. The differences between HSCOSTHH for owner occupiers and MORTCOST are:

mortgage protection policies are included as part of mortgage payments
Endowment policies are included as a separate category

 --

Water and Sewerage charges are excluded

Mortgage payments are equal to MORTINT plus mortgage protection policies. MORTINT excludes any amounts for policies included in last payment so it can simply be added back in. Similarly, there is no problem of double counting endowment policies because these are also excluded from MORTINT.

Payments of structural insurance only should be included. However, where the premium includes furniture and contents in combination with the structure, the total is taken. Where payments are made separately from mortgage payments (STROTHS=1), questions are directed to those in rented accommodation as well as owner occupiers. Only cases where TENURE=1 or 2 or 3 are included as part of MORTCOST.

If the property is owned outright (TENURE=1), MORTINT will be skipped and MORTPAY should be set to zero. Similarly, there will be no endowment premium costs which will also remain as zero. However, any structural insurance or other payments should be included and MORTCOST should not be set to -2.

2 FRS Specification

For each household where TENURE=1 or 2 or 3

Code Condition

MORTCOST

Mortgage payments

From HOUSEHOL record get MORTINT
 From MORTGAGE get INCMAMP1, INCMAMP2, INCMAMP3, INCMPPD1,
 INCMPPD2, INCMPPD3

Set MORTPAY equal to MORTINT

If MORTINT <>-2 and MORTPROT=1 and INCMAMP1 or INCMAMP2 or
 INCMAMP3 exists and INCMPPD1, INCMPPD2, INCMPPD3
 equal to -1 or 1-10,13,26,52 for all mortgages
 then MORTPAY=MORTPAY+INCMAMP1+INCMAMP2+INCMAMP3

Else if TENURE=1 (owned outright) MORTPAY=0

Endowment premiums

From ENDOWMNT table get MENPOLAM, MENPOLPD

Set ENDOWPAY equal to zero

For each mortgage, if TENURE=2 or 3 and MENPOLPD equal to -1 or 1-
 10,13,26,52

 --

then ENDOWPAY = total of MENPOLAM else do not change ENDOWPAY

(if TENURE=1, ENDOWPAY remains set as zero)

Structural insurance

From HOUSEHOL record get STRCOV, STRAMT1, STRPD1, STROTHS, TENURE

set STRUINS equal to zero

If STRCOV=1 or STRCOV=3 (structural insurance only or combined with furniture and contents as part of mortgage payment) and STRPD1=-1 or 1-10,13,26,52 STRUINS=STRAMT1 else do not change STRUINS

If STROTHS=1 and TENURE=1, 2 or 3 (insurance premium paid on structure of accommodation separately from any mortgage payments, owner occupiers only), and STRPD2=-1 or 1-10,13,26,52 STRUINS=STRUINS+STRAMT2 else do not change STRUINS

Service payments

From OWNER record get CHARGE1-7, ~~CHAMT1-7~~ **CHRGAMT1-7**, CHARGE1-7

set SERVPAY=0

If CHARGE1=1 and CHARGE1=-1, or 1-10,13,26,52 add **CHAMT1** **CHRGAMT1** to SERVPAY else do not change SERVPAY

If CHARGE2=1 and CHARGE2=-1, or 1-10,13,26,52 add **CHAMT2** **CHRGAMT2** to SERVPAY else do not change SERVPAY

If CHARGE3=1 and CHARGE3=-1, or 1-10,13,26,52 add **CHAMT3** **CHRGAMT3** to SERVPAY else do not change SERVPAY

If CHARGE4=1 and CHARGE4=-1, or 1-10,13,26,52 add **CHAMT4** **CHRGAMT4** to SERVPAY else do not change SERVPAY

If CHARGE5=1 and CHARGE5=-1, or 1-10,13,26,52 add **CHAMT5** **CHRGAMT5** to SERVPAY else do not change SERVPAY

If CHARGE6=1 and CHARGE6=-1, or 1-10,13,26,52 add **CHAMT6** **CHRGAMT6** to SERVPAY else do not change SERVPAY

If CHARGE7=1 and CHARGE7=-1, or 1-10,13,26,52 add **CHAMT7** **CHRGAMT7** to SERVPAY else do not change SERVPAY

MORTCOST=MORTPAY+ENDOWPAY+STRUINS+SERVPAY

-1 not applicable to this case - TENURE ≠ 1, 2 or 3.

-2 MORTCOST=-2 if MORTPAY is equal to -2 (-2 cases for ENDOWPAY, STRUINS and SERVPAY should not exist)

MORTINT

Purpose : The amount of mortgage interest paid by each household.
 Created : 13 January 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : PSM, ASD3E
 Issued : 17 April, 20039
 Amendments : VC - 27 January 1993 Added more groups to those picked out of tenure type.
 : VC - 2 March 1993 Added sorting bands for tabulation.
 : VC - 19 October 1993 Changes to specification as repayment mortgages :
 : were calculating an average interest paid over the last 12 months and not the :
 : last payment of interest made. See Andrew Ray's specification 13 October :
 : 1993
 : VC - 6 December 1993 Emphasises use of intdate to calculate mortgage :
 : interest for repayment mortgages
 : VC - 1 March 1994 To exclude any period codes 12 or 13
 : JS - 6 February 1996 to exclude structural insurance payments included as
 : part of mortgage payments
 : JS - 12 March 1996 to update variable names and include different interest
 : rates and MIRAS adjustments
 : JS - 12 April 1996 to allow for contributions made from anyone outside the
 : household
 : VE - 5 June 1996 - Initial amendments for V32 update - adding in PEP
 : mortgages to MORTTYPE
 : VE - 14 June 1996 - To update mortgage interest rates for 1995-96
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 19 August 1996 - To amend MNTHCODE to read January - December
 : VE - 20 November 1996 - To amend STRAMT to STRAMT1
 : VE - 2 December 1996 - To take off any amounts relating to top-up loans
 : unrelated to housing costs
 : VE - 3 February 1997 - To amend the top-up section of the variable
 : VE - 28 February 1997 - To amend MIRAS adjustment to include second
 : mortgages for the purpose of house purchase
 : VE - 1 May 1997 - To amend MIRAS adjustment to 15%
 : SG - 28 October 1997 - V33 mods - tenure code changes, invalid period

 codes : SG - 12 February 1998 - adjust to cover imputation problem cases
 : EP - 13 August 1998 - V34 update, top-up sections removed, reasons for
 : second mortgage change to variable name and codes
 : SG - 17 December 1998 - V34 update again - removal of TOPUPs
 : SB - 15 October 1999 - Security completed, no other changes for V35
 : SB - 14 January 2000 - minor changes to correct spec

1 Definition

This variable is coded as

MORTINT The total amount of mortgage interest payable by a household

-1 Not applicable to this case

-2 Unable to derive because of missing values

The amount of mortgage interest is derived from several variables in the HOUSEHOL, OWNER, MORTGAGE and ENDOWMNT tables. The variable TENURE in the HOUSEHOL table indicates whether the person owns the property and whether it is owned outright or with a mortgage. The variable MORTTYPE in the MORTGAGE table then indicates whether the mortgage is an endowment mortgage (including pension mortgages) or a repayment one. Once this has been established the total amount of mortgage interest can be calculated.

Endowment Mortgages

MORINPAY holds the amount of interest paid for endowment mortgages. However, if an insurance premium or a mortgage protection policy has been included in the amount of interest recorded in MORINPAY, the amount has to be adjusted. If MENPOL = 1 and INCININT = 1 (from the ENDOWMNT table) indicating that an amount for an insurance premium is included in MORINPAY, the amount of the premium has to be deducted from the amount of mortgage interest. The amount of insurance premium is found in MENPOLAM.

Also if MORTPROT = 1 and INCMP = 1 (both from the MORTGAGE table) indicating that a mortgage protection policy is included in MORINPAY, the amount in INCMPAMT is deducted from the amount of mortgage interest.

Adjustments are also required for contributions from outside the household if mortgage interest has been quoted after these amounts.

Finally, in block g_Insur of the questionnaire, respondents are asked whether their last payment included amounts of any insurance on the structure or contents of the accommodation (STRMORT). If yes, total amount STRAMT1 should be removed. The value of any structural insurance is added back in HSCOSTHH. (STRAMT2 relates to insurance payments not paid as part of mortgage payments).

Repayment Mortgages

Repayment mortgages are slightly different and the amount of interest paid on a repayment mortgage is calculated separately using the amount of mortgage still outstanding (MORTLEFT) multiplied by the rate of interest current for the month in which the interview took place. These interest rates are taken from the Central Statistics Office's report of Financial Statistics which will be held on the FRS database as a standard table of values and updated every year. **The constants are held in p:\frs\shared\frs35\metadata\dvmeta35.xls.** The average rate is used in preference to the basic rate because it is a better indicator of the interest rate charged on all mortgages (the basic rate excludes

discounts for first time buyers and is therefore slightly higher). Note: since October 1995 the eligible mortgage interest on IS has been calculated on the basis of the standard rate which is set using the basic rather than average rate.

As this will calculate the total amount of mortgage interest any mortgage protection policies and/or insurance premiums may be ignored.

However, if the period code for any period is 90, 95 or 97 (lumpsum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if MORINPD, MENPOLPD, and INCMPPD = 90, 95 or 97 MORTINT is set to -2. Cases where period codes have been skipped (where amount has been imputed) are allowed.

MIRAS

Allowance also has to be made for MIRAS which is deducted at 15% of the interest rate on the first £30,000 of mortgages taken out for the express purpose of buying a house. All payments in MORTINT are shown net of MIRAS. This means that for ENDOWMENT mortgages an adjustment is required for any payments quoted gross of MIRAS (TAXRELF=2). For REPAYMENT mortgages, where interest is calculated, an adjustment has to be applied to all cases.

Tax relief up to £30,000 is allowed on the residence and so MIRAS on a second mortgage will only be allowed if the allowance has not been used up by the first mortgage. From V34 we keep a record as each mortgage processed of how much of MIRAS has been used.

SECOND MORTGAGES

The FRS asks for details of up to two mortgages. For those buying their house with a mortgage respondents are routed to the question on the basis of "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 on this property?". This will include mortgages which have been secured on the property but which were not for house purchase (OTHPUR=1,2,4,6 or 7). These type of loans are not eligible for MIRAS. Rules for calculation of eligible mortgage interest for Income Support were tightened up in October 1995 and also exclude these type of loans.

It has been agreed that:

- i the MIRAS adjustment will only be applied to the first mortgage and to the second mortgage where OTHPUR3 or 5 = 1
- but ii that MORTINT will continue to included interest from all loans secured on the property (this is also consistent with DoE's calculation of equity which looks at *all* loans secured on a property).

Note: those who own their property outright are also asked if they are using their property as security for a mortgage or loan of any kind but these are EXCLUDED from MORTINT.

2 FRS Specification

For each HOUSEHOLD

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

<u>Code</u>	<u>Condition</u>
MORTINT	From HOUSEHOL table if TENURE- - 2 (buy with mortgage) or 3 (part own/rent) process MORTGAGE record If no MORTGAGE record but TENURE in (2,3) then MORTINT=0

Calculate INTRATE (temporary variable) = relevant interest rate for month of interview from standard table (look up table for basic rate relevant to month of interview: used in both endowment and repayment calculations so moved to beginning of program)

If MNTHCODE=1	INTM1
If MNTHCODE=2	INTM2
If MNTHCODE=3	INTM3
If MNTHCODE=4	INTM4
If MNTHCODE=5	INTM5
If MNTHCODE=6	INTM6
If MNTHCODE=7	INTM7
If MNTHCODE=8	INTM8
If MNTHCODE=9	INTM9
If MNTHCODE=10	INTM10
If MNTHCODE=11	INTM11
If MNTHCODE=12	INTM12

From MORTGAGE table, for each mortgage get all variables

If MORTTYPE = 1 or 3 or 4 or 5 (endowment, pension mortgage, PEP or Unit Trust mortgage or other type of mortgage) and MORINPD not equal 90,95 or 97 (but may equal -1) and MORINPAY not equal -1

Calculate MORTINT = MORINPAY

If MORTINT<0 set MORTINT=-2

If MORTPROT = 1 and INCMP1 = 1 and INCMPPD1 not equal 90,95 or 97 (but may equal -1), or INCMP2 = 1 and INCMPPD2 not equal 90,95 or 97 (but may equal -1) or INCMP3 = 1 and INCMPPD3 not equal 90,95 or 97 (but may equal -1) calculate MORTINT = MORTINT - (INCMPPD1+INCMPPD2+INCMPPD3). (mortgage protection policies)

From ENDOWMNT table, for each endowment policy get all variables

If INCININT = 1 (insurance premium included in MORINPAY) and MENPOLPD not equal 90,95 or 97 (but may equal -1), MORTINT = MORTINT - MENPOLAM

From MORTCONT table

If OUTSINCL=2 (amount of contribution to interest not included in amount mentioned earlier) and OUTSPD=-1 or 1-10, 13, 26, 52 then MORTINT=MORTINT+OUTSAMT

If STRMORT=1 and STRPD1=-1, 1-10,13,26,52 and STRAMT1<MORTINT then
MORTINT=MORTINT-STRAMT1

Else if STRMORT=1 and STRPD1=-1, 1-10,13,26,52 and STRAMT1>MORTINT
then do not change MORTINT

(this should only be applied to endowment mortgages because repayment
mortgages are calculated and do not use respondents' answers. If, however,
the amount in STRAMT1 is greater than MORTINT, assume that an error has
been made and do not change MORTINT)

MIRAS adjustment is moved to the last step *after* structural mortgage
payments adjustment

MIRAS adjustment to first mortgage and second mortgage where
OTHPUR3 or 5 = 1.

If MORTSEQ=1 then ALLOW=30,000

If MORTSEQ=1 or (MORTSEQ=2 and OTHPUR3 or 5 = 1) and ALLOW>0

If TAXRELF=2 (interest quoted does not include MIRAS arrangements)
and MORTLEFT>ALLOW

then MORTINT=MORTINT-((ALLOW*INTRATE*0.15)/52)
ALLOW=0

Else if TAXRELF=2 and MORTLEFT<=ALLOW

then MORTINT=MORTINT-((MORTLEFT*INTRATE*0.15)/52)
ALLOW=ALLOW-MORTLEFT

(MORINPAY will be held as a weekly amount, but the MIRAS adjustment also
has to be converted)

If MORTTYPE = 2 (repayment mortgage) or (MORTTYPE = 1 or 3 or 4 or 5, and
MORINPAY=-1) (where MORTTYPE has been imputed)

Calculate MORTINT = (MORTLEFT * INTRATE)/52

MIRAS adjustment to first mortgage and second mortgage where
OTHPUR= 3 or 5

If MORTSEQ=1 or (MORTSEQ=2 and ~~SECMPUR=1 or 6~~ OTHPUR=3 or 5)
and ALLOW>0

If MORTLEFT >ALLOW

then MORTINT=MORTINT-((ALLOW *INTRATE*0.15)/52)
ALLOW=0

Else if MORTLEFT <=ALLOW

then MORTINT=MORTINT-((MORTLEFT *INTRATE*0.15)/52)

ALLOW=ALLOW-MORTLEFT

- 1 Not applicable to this case - property not owned with a mortgage.
- 2 Unable to derive variable because of any missing values or MORINPD, INCMPPD or MENPOLPD = 90 or 95 or 97 (but -1 OK).

NB. FRS does not collect the rate of interest charged on a mortgage the interviewee's answer is taken to be correct.

 --

NDDCTB, NDDISHC

Purpose : To calculate total amount of non-dependent deductions for each household :
 using HBAI specifications - for Council Tax Benefit, rent rebates and JSA (IB)

Created : VC - 28 February 1994

Database Table : HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables :

 Children :

 Parents : DEPDEDS, INDINC - both are derived variables

Core variable/user : Take up of income related benefits

Amendments : VC - 28 February 1994 To reflect changes to version 30
 : JS - 3 April 1996 to reflect changes for V31
 : VE - 19 April 1996 - to replace GROSSINC with INDINC
 : VE - 4 June 1996 - No initial amendments needed for V32 update
 : VE - 27 June 1996 - To update income levels
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : SG - 3 December 1997 - No initial amendments for V33 (except constants
 table updated)
 : EP - 23 February 1999 - Update constants and income levels for V34 - two
 new income bands
 : SB - 30 September 1999 - Security completed, no other changes for V35
 : Re-written to combine rent rebates and JSA (IB) into one variable and only to
 include conventional households.

Issued : 30 September 1999

1 Definition

This variable is coded as

NDDISHC This is the total amount of non-dependent deductions for each household from rent
 rebate and JSA (IB) using the HBAI specification. **The constants and income levels
 are held in the table in p:\frs\shared\frs35\metadata\Dvmeta35.xls.**

NDDCTB This is the total amount of non-dependent deductions for each household for Council tax
 benefit using the HBAI specification. **The constants and income levels are held in
 the table in p:\frs\shared\frs35\metadata\Dvmeta35.xls.**

2 FRS Specification

A Household that does not receive any non-dependant deductions is set to 0 and not .A

For each HOUSEHOL table - set NDDCTB to zero, then

--

Create a flag to show exemptions for blind people and adults in receipt of DLA or AA for first BU
If BENUNIT = 1 and BENEFIT = 1, 12 or SPCREG1 = 1 then XFLAGHH = 1

Calculate ADULT income for each BU using INDINC
Sum through each BU adding INDINC to create BUINC (not the same as BUINC DV)

Create and set floating variables to 0
XNDDISHC = 0
XNDDCTB = 0

Calculate deductions for 2nd BU and above
If BENUNIT > 1 then do

If over 18 and other (DEPDEDS = 8) then add other amount for NDD amount
If DEPDEDS = 8
XNDDISHC = 7 (constants)
XNDDCTB = 2 (constants)

If over 18 and working over 16 hrs a week (DEPDEDS = 3) then add relevant amount
If DEPDEDS = 3

Firstly add rent rebate and JSA (IB) amounts to XNDDISHC
If BUINC < 78 (INCL1IS) add 7 (C1IS) (taken from constants table)
Else If BUINC < 116 (INCL2IS) add 16 (C2IS)
Else If BUINC < 152 (INCL3IS) add 22 (C3IS)
Else If BUINC < 200 (INCL4IS) add 36 (C4IS)
Else If BUINC < 250 (INCL5IS) add 41 (C5IS)
Else If BUINC >= 250 (INCL5IS) add 45 (C6IS)

Secondly add Council Tax Benefit amounts to XNDDCTB
If BUINC < 116 (INCLEV1) add 2 (C1CTB)
Else If BUINC < 200 (INCLEV2) add 4 (C2CTB)
Else If BUINC < 250 (INCLEV3) add 5 (C3CTB)
Else If BUINC >= 250 (INCLEV1) add 6 (C4CTB)

Create another flag to show whether a person is sub renting
If BEUNIT = 1 and SUBLTAMT > 0 then SUBLTFLG = 1
Otherwise SUBLTFLG = 0

Finally create NDDISHC and NDDCTB variables on a HH level
Set NDDISHC and NDDCTB to 0
If conventional HH (HHSTAT = 1) and above flags = 0 (XFLAGHH = 0 and SUBLTFLG=0) then add above amounts of rent rebate, JSA (AB) and CTB to variables for each BU.

NDDISHC = NDDISHC + XNDDISHC
NDDCTB = NDDCTB + XNDDCTB

 --

NETOCPEN

Purpose : To show the amount of income received from all forms of occupational pensions from former employers net of tax (REVISED OCCUPPEN)
 Created : January 1995
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : **EMPOCCP, WIDOCCP, TOTOCCP**
 Children :
 Parents :
 Core variable/user : PSM
 Amendments : JS - 21 February 1996 to allow for skipped values where amounts have been imputed
 : VE - 23 May 1996 - No initial amendments needed for V32 update
 : SG - 6 January 1998 - V33 updates - period codes
 : EP - 13 August 1998 - No initial V34 update needed
 : EP - 18 December 1998 - change related variables
 : JC - 8 September 1999 - Security completed, no other changes for V35
 : SB - 2 November 1999 - Changed ROYAL3 to ROYAL4
 Issue date : 2 November 1999

1 Definition

This variable is coded as

NETOCPEN This is the total gross amount received from **all** occupational pensions paid by a person's former employer. It includes any occupational pension which is being paid by an overseas government/company paid in foreign currency (converted to sterling).

-1 Not applicable to this case - adults who do not have occupational pensions

-2 Unable to derive variable

NETOCPEN will be derived from variables PENTYPE, PENPAY, ROYAL4 AND ROYYR4. PENTYPE and PENPAY indicate that the person is in receipt of a pension from a previous employer. The variable PENTYPE is a database variable created to indicate which of the incomes listed in ANYPEN the person has and which one this particular record refers to. A person may have up to 5 occupational pensions and as a result NETOCPEN must be the total of all pensions.

To get the net amount, the amount held in PENPAY must be looked at in relation to the variables which ask if any tax has been deducted at source or whether any other deductions had been taken into account when the original amount of PENPAY had been given. These variables are PENTAX (has tax been deducted at source y/n), PTINC (was the original amount before or after this was deducted 2 =

 --

after), PTAMT (amount deducted), PENOTH (any other deductions y/n), POINC (original amount before/after deduction 2 = after) and POAMT (amount deducted). Consequently, if the original amount in PENPAY was after either of these amounts had been deducted they must be added back to get the gross amount.

ROYAL4 and ROYYR4 deal with an occupational pension paid by an overseas government or company which is paid in a foreign currency. This amount is taken to be a gross amount for OCCUPPEN but must be considered to be net of tax for NETOCPEN as the questionnaire does not collect information regarding any deductions from the amount held in ROYYR3.

ROYAL4 and ROYYR4 are also database variables which hold the information obtained from questions royal and ROYYR (NB - royal1 = royalties, royal2 = sleeping partners and royal4 = occ pen from o/s govt). However, if the period code for the pension is 12 or 13 (lump-sum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if PENPD = 12 or 13 NETOCPEN is set to -2.

2 FRS Specification

For each ADULT with a pension record

Code

Condition

NETOCPEN From PENSION table, for each pension calculate a temporary variable OCCUP

If PENTYPE = 1 (indicating an occupational pension is being received) and PENPD equal to -1 or 1-10, 13,26 or 52, get variables PENPAY, PENTAX, PTAMT, PTINC, PENOTH, POAMT and POINC.

Compute OCCUP = PENPAY.

If PENTAX = equal to 1 or has been skipped (tax deducted at source) and PTINC = 2 (original amount declared after amount of tax deducted) do not change OCCUP

If PENTAX=1 (tax deducted at source) and PTINC=1 and PTAMT exists (original amount before amount of tax deducted) subtract PTAMT from OCCUP else do not change OCCUP

If PENOTH = 1 (other deductions) and POINC = 2 (original amount declared after deduction), add POAMT to OCCUP.

From ADULT table

If ROYAL4 = 1 (pension from an overseas government) get amount from ROYYR4 and add to OCCUP.

NETOCPEN will then be the sum of all occurrences of OCCUP as each adult is able to have up to 5 occupational pensions.

--

-1 Not applicable to this case -

-2 Unable to derive because any of the above variables are missing or PENPD = 90, 95 or 97.

 --

NINDINC, NINEARNS, NININV, NINPENIN

Purpose : To indicate the amount of net income received by an adult for use in the FRS publication (based on INDINC).
 Created : 11 November 1997
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : 20 November 1997 - SCG - Correct treatment of social fund loans, update spec with regard to back to work bonus
 : 30 December 1997 - SCG - no period codes for odd jobs
 : 13 March 1998 - SCG - Don't remove assumed tax from investment income not taxed at source
 : 23 April 1998 - SCG - Change in DV for self employment income
 : EP - 13 August 1998 - No initial V34 update needed
 : JC - 8 September 1999 - Security completed, no other changes for V35
 : SB - 21 September 1999 - Adjusted gross bonus amount received
 Issued : 17 April, 2003

1 Definition

This variable is coded as

NINDINC The total amount of net income received by an adult from all sources.

0 Not applicable as adult does not have any net income.

-2 Unable to derive due to missing values.

NINDINC is very similar to INDINC (Gross adult income) except it does not include income tax and NI contributions. The specification is closely based on that for INDINC, which is in line with HBAI definitions.

Of the components of NINDINC, several are also used as components of INDINC. These are INDISBEN, INOTHBEN, INRINC, INRPINC. These are detailed in the specification for INDINC.

The other components exist both as Gross and Net versions, with the Gross versions being added into INDINC, and the Net versions into NINDINC. These are INEARNNS, INPENINC and ININV (Gross versions) and NINEARNS, NINPENIN and NININV (Net versions).

The total amount of net income is derived from numerous variables from the ADULT, JOB, BENEFITS, ODDJOB and PENSIONS records which when added together form the person's total net income. It

 --

includes net normal earnings, net self-employed earnings, net pensions and annuities, other income in the form of benefit income, income in kind, royalties, other allowances, income from trust funds and odd jobs etc.

Income Support is included gross of any direct payments or social fund repayments. Income from boarders/lodgers has been excluded to avoid double counting at a household level and to simplify definitions (this will need to be discussed in the publication).

Additional amounts for direct expenses from absent partners, regular contributions from household members, education grants and other deductions from pension income have also been included.

Private benefit schemes are included unless they are one off/lump sum payments (these are ignored).

Income from free school milk and meals and free welfare milk are also included (allocated to the head of benefit unit).

Amounts of maintenance are also checked to make sure they are usual.

However, if the period code for the benefit is 90 or 95 or 97 (lump-sum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, for example, if PAYPD = 90 or 95 or 97 NINDINC is set to -2. Additionally, coding has been changed to allow skipped values for period codes: this will occur where the (weekly) amount has been imputed. It has been decided to leave period codes as "skipped" in these cases since this may help flag imputation.

The addition of HB is adjusted for any rent-free holidays as in the derivation of HHRENT. This adjustment is done for conventional households and for shared households, but not for boarders and lodgers. The reasoning behind this is that boarders and lodgers would probably pay rent every week even if the household had some rent-free weeks.

This DV spec is also used to set up components of net income which are accumulated to obtain BU and HH level variables. These are:

NINEARNS	net earned income
NININV	net investment income
NINPENIN	net other pensions

In most cases investment income will have been taxed at source and so will be net amounts. Where this is not the case we are not in a position to make assumptions about taxpayer status and so we do not attempt to calculate and remove an amount for tax.

Other components of net income are specified elsewhere. These are :

NINSEIN2	net self-employment income (specified separately)
INDISBEN	disability benefits (specified with INDINC)
INOTHBEN	other benefits (specified with INDINC)
INRINC	remaining income (specified with INDINC)
INRPINC	retirement pension plus any income support (specified with INDINC)

 --

For pensioners, any retirement pension is taken together with IS to avoid any issues of misreporting. Analyses of pensioner income by ASD3 also take these two together.

Disability benefits comprise war disablement benefit, DWA, SDA, AA, DLA (mob and care); IIDB and ICB

**NINEARNS has been corrected and corrections have been backdated to 1997-98 release d. The problem was to do with the way gross bonus' were converted into net figures. For further information, see paper on ninearns: G:\Public\frs\docs\1997_8\Ninearns.doc
 P:\frs\shared\Frs34\derived\Ninearns.doc**

2 FRS Specification

For each ADULT

Code

Condition

NINDINC

Net earnings: NINEARNS

From ADULT record, set NINEARNS to zero

If WORKING = 1 or JOBAWAY = 1 - process each JOB record for that person and

If ABSWHY \neq 3 (To bring in line with HBAI)

If EMPEE = 1

(PAYUSL, and therefore UNETT if PAYUSL equals "no" is only asked for JOBTYP=1, therefore coding has been changed so that a check is made to see if UNETT exists. Where it doesn't, net pay is calculated dependent on whether a payslip has been consulted, where it does - which can only be JOBTYP=1 - UNETT is used)

If UNETT does not exist

PAYSLIP = 1 or 2 or 3 and PAYPD equals -1 or 1 to 11 or 13 or 26 or 52 ,

If PAYAMT exists, add it into NINEARNS.

If it is missing set NINDINC and NINEARNS to -2

If CHRAMT exists add it into NINEARNS

If it is missing do not change NINEARNS

If OTHDED1 = 1 add DEDUC1 to NINEARNS

If OTHDED2 = 1 add DEDUC2 to NINEARNS

If OTHDED3 = 1 add DEDUC3 to NINEARNS

If OTHDED4 = 1 add DEDUC4 to NINEARNS

If OTHDED5 = 1 add DEDUC5 to NINEARNS

If OTHDED6 = 1 add DEDUC6 to NINEARNS

If OTHDED7 = 1 add DEDUC7 to NINEARNS

 --

If OTHDED8 = 1 add DEDOTH to NINEARNS

Else if UNETT exists

If UNETT exists add UNETT to NNINEARNS

If it is missing do not change NINEARNS (e use PAYAMT calculation if it exists)

Adjustments to net earnings for HBAI consistency:

income tax refunds, mileage and motoring allowances, refunds for items of household expenditure

if NINEARNS<>-2 (other conditions relating to PAYAMT and PAYPD will have been met by this point if NINEARNS has not been set to -2)

and UNETT does not exist (e all jobtypes except jobtype=1 where pay not usual)

and JOBTYP=1 and TAXAMT exists $NINEARNS=NINEARNS-TAXAMT$ (TAXAMT only asked for first job)

and MILEAMT exists $NINEARNS=NINEARNS-MILEAMT$

and MOTAMT exists $NINEARNS=NINEARNS-MOTAMT$

and HHA1 exists $NINEARNS=NINEARNS-HHA1$

and HHA2 exists $NINEARNS=NINEARNS-HHA2$

and HHA3 exists $NINEARNS=NINEARNS-HHA3$

Adjustments to net earnings for HBAI consistency:

addition of bonuses received in last 12 months divided by 52

for up to 6 bonuses $i=1-6$:

If BONAMT(i) exists and BONTAX(i) (after tax)=2 or -1 (after tax or skipped where BONAMT imputed)

$NINEARNS=NINEARNS+((BONAMT(i)/52))$

If BONAMT(i) exists and BONTAX(i)=1 (before tax)

$NINEARNS=NINEARNS+((BONAMT(i)/52)*0.77)$

 --

(questions about bonuses are asked regardless of whether pay usual or not, however, if UGROSS has been taken, have to make sure that bonus is not double counted)

[If UGROSS exists
 and UBONINC=1 and UBONAMT exists
 NINEARNS=NINEARNS-(UBONAMT)]

(UBONAMT is on a net basis, assume that if included in net pay ie UBONINC = yes. then it will also have been in usual gross pay; UBONAMT is only asked if UBONINC = yes. Need the condition 'if UBONINC=2' to account for cases where UBONINC has been edited)

Self - employment income: INCSE2

Use INCSE2

Interest/income from savings accounts or investments: NININV

set NININV to zero

From ACCOUNTS record,

if ACCOUNT = 1, add amount in ACCINT to NININV
 if ACCOUNT = 2, add amount in ACCINT to NININV
 if ACCOUNT = 3, add amount in ACCINT to NININV
 if ACCOUNT = 4, add amount in ACCINT to NININV
 if ACCOUNT = 5, add amount in ACCINT to NININV
 if ACCOUNT = 6, add amount in ACCINT to NININV
 if ACCOUNT = 7, add amount in ACCINT to NININV
 if ACCOUNT = 8, add amount in ACCINT to NININV
 if ACCOUNT = 9, add amount in ACCINT to NININV

Personal pensions: NINPENIN

set NINPENIN to zero

Occupational pensions

From PENSIONS record, if PENTYPE = 1 (occupational pension) and PENPD equals - 1 or 1 to 10 or 13 or 26 or 52,

NNINPENIN=PENPAY. If PTINC exists and = 1 and PTAMT exists remove PTAMT then add result into NINPENINC (otherwise, do not change NINPENINC). If PENOTH exists and equal to 1 and POINC exists and =2 (other deductions from pension not included in figure at PENPAY) and POAMT exists add POAMT to NINPENINC otherwise, do not change NINPENINC.

 --

(this is the same approach as taken in OCCUPPEN to deal with skipped values)

Widow's employee pension

From PENSIONS record, if PENTYPE = 2 (widow's employee pension) and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

$NINPENIN = NINPENIN + PENPAY$. If PTINC exists and = 1 and PTAMT exists remove PTAMT then add result into NINPENIN (otherwise, do not change NINPENIN). If PENOTH exists and equal to 1 and POINC exists and =2 (other deductions from pension not included in figure at PENPAY) and POAMT exists add POAMT to NINPENIN otherwise, do not change NINPENIN.

Personal pension

From PENSIONS record, if PENTYPE = 3 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

$NINPENIN = NINPENIN + PENPAY$. If PTINC exists and = 1 and PTAMT exists remove PTAMT then add result into NINPENIN (otherwise, do not change NINPENIN)

Trade union Friendly society pensions

From PENSIONS record, if PENTYPE = 4 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

get amount from PENPAY and add to NINPENIN If PTINC = 1 remove PTAMT then add result into NINPENIN

Annuity pension

From PENSIONS record, if PENTYPE = 5 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

$NINPENIN = NINPENIN + PENPAY$. If PTINC exists and = 1 and PTAMT exists remove PTAMT then add result into NINPENIN (otherwise, do not change NINPENIN)

Trust/covenant

From PENSIONS record, if PENTYPE = 6 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

get amount from PENPAY and add to NINPENIN. If PTINC exists and = 1 and PTAMT exists remove PTAMT then add result into NINPENIN (otherwise, do not change NINPENIN)

NINDINC will then be calculated as follows for each ADULT -

$INCSE2 + NININV + INRPINC + NINPENIN + INDISBEN + INOTHBEN + INRINC + NINEARNS$

--

-2 If any of above variables are missing or if a period code is 90, 95 or 97 (skipped is OK) also applied to individual component variables NININV, INRPINC, NINPENIN, INDISBEN, NINEARNS, INOTHBEN and INRINC which are calculated in this spec (missing components are already documented above).

 --

EMPOCCP, WIDOCCP, TOTOCPP

Purpose : To show the amount of income received from different forms of occupational pensions from former employers
 Created : 16 December 1998
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : PSM
 Amendments : JC – 8 September 1999 – Security completed, no other changes for V35
 : SB – 2 November 1999 – Royyr3 changed to Royyr4 (category change only)
 Issue date : 17 April, 2003

1 Definition

This variable is coded as

EMPOCCP This is the gross amount received from an employee pension paid by a person's former employer. It includes any occupational pension which is being paid by an overseas government/company paid in foreign currency (converted to sterling).

WIDOCCP This is the gross amount received from a widow's employee pension paid by the former employer of the deceased spouse or relative.

TOTOCPP This is the total gross amount received from any occupational pensions paid by a person's former employer. It includes any occupational pension which is being paid by an overseas government/company paid in foreign currency (converted to sterling).

-1 Not applicable to this case - adults who do not have occupational pensions

-2 Unable to derive variable

All 3 DVs are derived in a similar way:

EMPOCCP will be derived from variables PENTYPE, PENPAY, and **ROYR4**

WIDOCCP from PENTYPE and PENPAY

And TOTOCPP from PENTYPE, PENPAY and **ROYR4**.

PENTYPE and PENPAY indicate that the person is in receipt of a pension from a previous employer.

The variable PENTYPE is a database variable created to indicate which of the incomes listed in ANYPEN the person has and which one this particular record refers to.

 --

To get the gross amount, the amount held in PENPAY must be looked at in relation to the variables which ask if any tax has been deducted at source or whether any other deductions had been taken into account when the original amount of PENPAY had been given. These variables are PENTAX (has tax been deducted at source y/n), PTINC (was the original amount before or after this was deducted 2 = after), PTAMT (amount deducted), POINC (original amount before/after deduction 2 = after) and POAMT (amount deducted). Consequently, if the original amount in PENPAY was after either of these amounts had been deducted they must be added back to get the gross amount.

ROYR4 deals with an occupational pension paid by an overseas government or company which is paid in a foreign currency. This amount has to be taken to be a gross amount as the questionnaire does not collect information regarding any deductions from the amount held in ROYR4

If the period code for the pension is 90 or 95 or 97 (less than one week / lump sum / one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if PENPD = 90 or 95 or 97, EMPOCCP, WIDOCCP or TOTOCCP is set to -2.

Where values have been imputed, answers to questions which follow will remain as skipped. To overcome this problem, the specification needs to be amended to allow:

- i skipped values of PENPD and PENTAX where PENPAY has been imputed (PENTAX assumed to be equal to 2 for these cases)
- ii missing PTAMT where PENTAX has been imputed to yes
- iii missing PTINC where PTAMT has been skipped (assumed to be after)

2 FRS Specification

For each ADULT with a pension record

<u>Code</u>	<u>Condition</u>
EMPOCCP	<p>From PENSION table, for each pension</p> <p>If PENTYPE = 1 (indicating an occupational pension is being received) and PENPD equal to -1 or (1 – 52), get variables PENPAY, PENTAX, PTAMT, PTINC, POAMT and POINC.</p> <p>Compute EMPP = PENPAY.</p> <p>If PENTAX exists and = 1 (tax deducted at source) and PTINC exists and equals to 2 (original amount declared <u>after</u> amount of tax deducted), and PTAMT exists add PTAMT to EMPP (otherwise, do not change EMPP).</p> <p>If POAMT >= 0 (other deductions) and POINC = 2 (original amount declared <u>after</u> deduction), add POAMT to EMPP.</p> <p>From ADULT table</p>

 --

If ROYYR4 \geq 0 (pension from an overseas government) get amount from ROYYR4 and add to EMPP.

EMPOCCP will then be the sum of all occurrences of EMPP as each adult is able to have up to 4 occupational pensions.

-1 Not applicable to this case

-2 Unable to derive because any of the above variables are missing or PENPD = 90 or 95 or 97.

WIDOCCP From PENSION table, for each pension

If PENTYPE = 2 (indicating a widow's employee pension is being received) and PENPD equal to -1 or (1 - 52), get variables PENPAY, PENTAX, PTAMT, PTINC, POAMT and POINC.

Compute WIDP = PENPAY.

If PENTAX exists and = 1 (tax deducted at source) and PTINC exists and equals to 2 (original amount declared after amount of tax deducted), and PTAMT exists add PTAMT to WIDP (otherwise, do not change WIDP).

If POAMT \geq 0 (other deductions) and POINC= 2 (original amount declared after deduction), add POAMT to WIDP.

From ADULT table

WIDOCCP will then be the sum of all occurrences of WIDP as each adult is able to have up to 3 widow's employee pensions.

-1 Not applicable to this case

-2 Unable to derive because any of the above variables are missing or PENPD = 90 or 95 or 97.

TOTOCCP From PENSION table, for each pension

If PENTYPE in (1,2) (indicating ANY occupational pension is being received) and PENPD equal to -1 or (1 - 52), get variables PENPAY, PENTAX, PTAMT, PTINC, POAMT and POINC.

Compute TOTP = PENPAY.

If PENTAX exists and = 1 (tax deducted at source) and PTINC exists and equals to 2 (original amount declared after amount of tax deducted), and PTAMT exists add PTAMT to TOTP (otherwise, do not change TOTP).

--

If POAMT \geq 0 (other deductions) and POINC= 2 (original amount declared after deduction), add POAMT to TOTP.

From ADULT table

If ROYYR4 \geq 0 (pension from an overseas government) get amount from ROYYR4 and add to TOTP.

TOTOCCP will then be the sum of all occurrences of TOTP.

-1 Not applicable to this case

-2 Unable to derive because any of the above variables are missing or PENPD = 90 or 95 or 97.

3 Results

Tabulation is required to show the number of people with occupational pensions by the total amount of pension received sorted into the following bands

Under £25
£25 - £50
£50 - £75
£75 - £100
£100 - £125
£125 - £150
£150 - £175
£175 - £200
£200 - £250
£250 and over

4 Test Cases

None as yet

 --

OCCUPNUM

Purpose : To show the total number of occupational pensions a person receives.
 Created : VC - 13 July 1993
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value : 6
 Units : Integer
 Validations :
 Related Variables : **EMPOCCP, WIDOCCP, TOTOCCP** - total amount of occupational pensions
 Children :
 Parents :
 Core variable/user : PSM
 Amendments : VC - 17 February 1994 Amended to reflect version 30 changes
 : VE - 23 May 1996 - No initial amendments needed for V32 update
 : SG - 6 January 1998 - No initial amendments needed for V33 update
 : EP - 13 August 1998 - No initial V34 update needed
 : EP - 18 December 1998 - Change related variables
 : SB - 2 November 1999 - Security completed, ROYAL3 change to ROYAL4
 Issue date : 6 January 1998

1 Definition

This variable is coded as

OCCUPNUM The total number of occupational pensions a person receives from both a former employer or from any pensions from overseas governments or companies.

-1 Not applicable to this case

-2 Unable to derive due to missing values.

This variable is a simple count of the number of occupational pensions a person receives and is derived from processing the PENSION record in the database and counting the number of these records where PENTYPE = 1 (PENTYPE is a database variable indicating a record holding information about occupational pensions). A person may have up to five of this type of pension record.

The variable should also include pensions paid by overseas governments or companies. Information about these pensions are to be found where ROYAL4 = 1. ROYAL4 is a database variable produced from the question ROYAL where royal1 = royalties, royal2 = sleeping partners and royal4 = occupational pensions. Only one extra pension need be counted as the questionnaire only collects information about one of these pensions.

2 FRS Specification

--

For each adult

Code

Condition

OCCUPNUM Set OCCUPNUM to zero

From PENSION table, count number of records where PENTYPE = 1

From ADULT record, if ROYAL4 = 1 add one to total number of pension records calculated above.

3 Results

Tabulation is required to show the number of people by the number of pensions they receive sorted into bands of

- No occupational pensions
- One pension
- Two pensions
- Three pensions
- Four pensions
- Five pensions
- Six or more pensions

4 Test Cases

None produced yet - to be added at a later date.

 --

PACCTYPE

Purpose : To indicate the number of households in any specific accommodation type for use in the FRS publication.
 Created : 29 January 1996 (although similar coding existed for 1993/94 publication)
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 5
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Issue date : 17 April 2003
 Amendments : VE - 5 June 1996 - No initial amendments needed for V32 update
 : SG - 6 January 1998 - V33 update
 : EP - 13 August 1998 - No initial V34 update needed
 : EP - 10 May 1999 - Include skipped values of TYPEACC into PACCTYPE=5
 : JC - 17 September 1999 - Security completed, no other changes for V35

1 Definition

PACCTYPE is a cut down version of TYPEACC in the household table. A completely new DV is probably not necessary. It is coded as:

- 1 Detached
- 2 Semi-detached
- 3 Terraced (including end of terrace)
- 4 Flat/maisonette (including part of house/converted flat etc)
- 5 Other

2 FRS Specification

- 1 TYPEACC=1
- 2 TYPEACC=2
- 3 TYPEACC=3
- 4 TYPEACC=4 or 5

--

- 5 TYPEACC in (6, 7, or .A) – (Note: TYPEACC is skipped (.A) when the household's main type of accommodation is a single room)

--

PTENTYPE

Purpose : To indicate the number of households in any specific tenure type for use in the FRS publication.
Created : 29 January 1996 (although coding existed for 1993/94 publication)
Database Table : HOUSEHOL
Minimum Value : 1
Maximum Value : 5
Units : Integer
Validations :
Related Variables : TENTYPE
 Children :
 Parents :
Core variable/user : FRS (publication)
Issue date : 17 April 2003
Amendments : JS - 21 February 1996 - to allow for missing values where variables have been imputed
 : JS - 15 March 1996 to bring in line with SEH categories (at request of ASD3F)
 : VE - 19 August 1996 - To amend so that part owner/part renter cases are now included with cases buying with a mortgage
 : VE - 10 October 1996 - To bring in line with Harmonisation of Outputs categories
 : SG - 31 October 1997 - V33 mods - changes to TENTYPE coding
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 21 October 1999 - Security completed, no other changes needed for V35

1 Definition

PTENTYPE is derived using variables TENURE (household record), FURNISH and LANDLORD (renter). Codes are effectively a re-code of the TENTYPE breakdown. (*All cases where TENTYPE=7 and LANDLORD=1, 2 or 3 will be edited to rebate cases.*) FURNISH is not asked for TENTYPE=8 so assumed to be rented furnished.

This variable is coded as

- 1 Rented from council
- 2 Rented from housing association
- 3 Rented privately unfurnished
- 4 Rented privately - furnished
- 5 Owned with mortgage (including part rent/part own)

--

6 Owned outright
-2 unable to derive due to missing values

2 FRS Specification

1 TENTYPE=1
2 TENTYPE=2
3 TENTYPE=3 or (TENTYPE=7 and FURNISH=3) or (TENTYPE=8 and FURNISH=3)
4 TENTYPE=4 or ((TENTYPE=8 or TENTYPE=7) and (FURNISH=1 or FURNISH=-2 or FURNISH=.A))
5 If TENTYPE=5
6 If TENTYPE=6

 --

RELHOH (Adult), RELHOH (Child)

Purpose : The relationship of an individual to head of household.
 Created : 18 December 1998
 Database Table : ADULT, CHILD
 Minimum Value :
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Issued : 17 April, 2003
 Amendments : JC – 6 September 1999 – Security completed, no other changes for V35

1 Definition

This variable is coded as

RELHOH The relationship of an individual to the head of household

- 1 Spouse
- 2 Cohabitee
- 3 Son/daughter (incl. adopted)
- 4 Step-son/daughter
- 5 Foster child
- 6 Son-in-law/daughter-in-law
- 7 Parent
- 8 Step-parent
- 9 Foster parent
- 10 Parent-in-law
- 11 Brother/sister (incl. adopted)
- 12 Step-brother/sister
- 13 Foster brother/sister
- 14 Brother/sister-in-law
- 15 Grand-child
- 16 Grand-parent
- 17 Other relative
- 18 Other non-relative

-1 Not applicable to this case

-2 Unable to derive because of missing values

From V34 onwards the head of household is not necessarily person 1. This DV can therefore be used where it is necessary to know the relationship and the variable R01 has previously been used.

2 FRS Specification

--

For each ADULT and CHILD

Code

Condition

From the HOUSEHOL table get HOHNUM.

From the ADULT table use HOHNUM as an index into R01 to R14 to read the relationship.

From the CHILD table use HOHNUM as an index into R01 to R14 to read the relationship.

 --

SUBLTAMT

Purpose : To show the amount of rent received by a benefit unit from sub-letting.
 Created : VC - 11 January 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : VC - 27 April 1993. To expand the definition to make meaning of each question/database variable clear.
 : VC - 14 September 1993 To make by benefit unit not household
 : VC - 17 February 1994 Amended to reflect version 30 changes
 : VE - 4 June 1996 - No initial amendments needed for V32
 : SG - 28 October 1997 - No initial amendments needed for V33
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 2 September 1999 - Security completed, no other changes for V35
 Issue date : 2 September 1999

NB - This variable has changed slightly from the FES variable as FES uses the amount of "profit" from sub-letting, whereas FRS uses the actual rent charged.

1 Definition

This variable is coded as

SUBLTAMT The amount of rent received by a benefit unit from sub-letting part of the property to a person who is not a member of the household or a boarder/lodger.

-1 Not applicable in this case.

-2 Unable to derive variable

The amount of rent received from sub-letting is to be found in the HOUSEHOL table in the variable SUBLET. If SUBLET = 1 (indicating that there is a sub-letting arrangement in the household) the amount of rent charged is to be found in SUBRENT.

As this variable is collected on a household basis, the amount received in SUBRENT is attributed to a benefit unit. In FES this was attached to the record of benefit unit number 1 (head of household BU) and this will be duplicated by FRS. Any other benefit unit will be designated not applicable.

2 FRS Specification

--

For each BENUNIT record, set SUBLTAMT to zero.

Code

Condition

SUBLTAMT If BENUNIT = 1, process HOUSEHOL record and
 If SUBLET = 1 get the amount of rent charged from SUBRENT.

-1 Not applicable to this case where BENUNIT > 1 or SUBLET = 2

-2 Unable to derive in this case - where any of the above variables are missing

3 Results

Tabulation is required to show the number of benefit units which have income from sub-tenants by the amount of rent received each week sorted into bands of, for example,

Under £25
£25 - £50
£50 - £75
£75 - £100
£100 - £125
£125 - £150
£150 - £175
£175 - £200
Over £200

 --

SUPERAN

Purpose : To indicate the total amount of superannuation or pension contributions
 : deducted from a person's earnings from all jobs.
 Created : VC
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : Deducts
 Children :
 Parents :
 Core variable/user : PSM
 Issue date :17 April 2003
 Amendments : VC - 9 February 1993 change to multi response.
 : VC - 11 May 1993 amended to show superannuation or pension payments
 made from all income from jobs.
 : VC - 23 August 1993 amended to emphasise that for employees only - also to
 indicate which groups are not applicable
 : VC - 17 February 1994 - Amended to reflect version 30 changes - it has an
 extra category detailing the contribution to an additional voluntary pension
 which has now been included.
 : VC - 1 March 1994 - To exclude any period codes 12 or 13
 : JS - 21 February 1996 - to allow skipped values of PAYPD where PAYAMT
 has been imputed
 : VE - 23 May 1996 - No amendments needed for initial V32 update
 : SG - 19 June 1997 - To allow for changes in PAYPD definitions
 : EP - 13 August 1998 - No initial V34 update needed
 : JC - 8 September 1999 - Security completed, no other changes for V35

NB - will not include self-employed jobs separate base variables provided for these jobs

1 Definition

This variable is coded as

- SUPERAN The total amount of superannuation or pension contributions deducted from a person's earnings from all jobs also includes contributions to additional voluntary pension.
- 1 Not applicable in this case - people who do not have superannuation deductions and those not working or self-employed
- 2 Unable to derive variable

The amount of superannuation or pension or additional voluntary pension contributions is derived from all jobs and where OTHDED1 or OTHDED2 are coded 1 to show that an amount for a pension or

 --

superannuation is deducted. The variable DEDUC1 will then hold the amount of superannuation/pension contribution and DEDUC2 holds the amount of additional voluntary contributions. The variables OTHDED1, OTHDED2, DEDUC1 and DEDUC2 are created in the database to hold the answers to the multi response questions OTHDED (were there any other deductions from your wage/salary such as 1 = pension or superannuation, 3 = Union fees etc) and DEDUC which holds the amounts.

However, if the period code for the deduction is 90 or 95 or 97 (lump-sum/one-off or other period), from the PAYPD variable, the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, if PAYPD = 90 or 95 or 97 SUPERAN is set to -2.

2 FRS Specification

For each ADULT with record Job for all jobs

<u>Code</u>	<u>Condition</u>
SUPERAN	If PAYPD equals -1 or 1-10, 13, 26 or 52 and - If OTHDED1 = 1 and DEDUC1 exists, get the amount of the superannuation or pension contribution deducted from variable DEDUC1. If OTHDED2 = 1, and DEDUC2 exists get the amount of the additional voluntary contribution deducted from variable DEDUC2.
-1	Not applicable in this case - where OTHDED1 = 2 or 3, OTHDED2 = 2 or 3 or OTHDED1/OTHDED2 not exist or there are no job records
-2	If variable cannot be defined because of missing data - where there is a job record but no values or if PAYPD = 90 or 95 or 97.

3 Results

Tabulation is required to show the weekly amount of superannuation or pension contribution deducted from earnings sorted into bands, for example,

Under £5.00
 £5.00 - £10.00
 £10.00 - £15.00
 £15.00 - £20.00
 £20.00 - £25.00
 Over £25.00

 --

TAXPAYER

Purpose : To show if an adult should be paying tax on their income
 Created : Ed Pickering – 28 May 1999
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value : 1
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : HBAI
 Issue Date : 17 April 2003
 Amendments : JC – 6 September 1999 – Security completed
 : SB – 8 November 1999 – Updated tax rates/allowances (See Dvmeta35) and
 replaced PROPARENT with ROYAL1, Input tax rates at start of code
 : SB – 6 January 2000 – Moved SMP and SSP from benefit table to job table

1 Definition

The derived variable, TAXPAYER, identifies whether or not an individual *should* be paying tax on their income. It is not possible to determine if the individual is *actually* paying the tax – but this DV should provide a reasonably accurate indication of the proportion of adults who do pay tax.

1997-99 Personal Allowances:

	1997-98 (£)	1998-98 (£)
Personal allowance – under 65	4,045	4,195
- 65 to 74	5,220	5,410
- 75 and over	5,400	5,600
Married couple’s allowance – under 65	1,830*	1,900*
- 65 to 74	3,185*	3,305*
- 75 and over	3,225*	3,345*
Income limit for age-related allowances	15,600	16,200
Additional personal allowance for single parent’s etc.	1,830*	1,900*
Widow’s bereavement allowance	1,830*	1,900*
Blind person’s allowance	1,280	1,380
* These allowances attract tax relief at only 15%		

Income tax rates on taxable income:

Rate	1997-98		1998-99	
	Band (£)	Tax (£)	Band (£)	Tax (£)

 --

Lower (20%)	0 - 4,100	820	0 – 4,300	860
Basic (**)	4,101 - 26,100	5,060	4,301 – 27,100	6,095
Higher (40%)	Over 26,100		Over 27,100	
** 23% for 1997-98				

The tax allowances and rates are held in the temp informats sheet of the table in
 p:\frs\shared\frs35\metadata\Dvmeta35.xls

2 FRS Specification

For each adult:

Calculate taxable income:

Add together (annual amounts)

- Earnings from employment
 - If self-employed, then use SEINCAM2
 - For all other EMPSTATI then use INEARN\$
- Retirement Pension
 - Use BENAMT for BENEFIT=5
- Jobseeker's Allowance (Contributions based)
 - Use BENAMT for BENEFIT=14 where JSATYPE in (1,3)
- Widow's Benefits
 - Use BENAMT for BENEFITS 6 and 7
- Occupational Pension
 - Use amount held in TOTOCCP
- Annuity (Gross payments from personal pensions and trusts)
 - PENPAY for PENTYPE in (3,4,5,6)
- Taxable interest (Total interest before tax, including rental income from other property)
 - ININV + ROYAL1
- Statutory Sick Pay
 - Use SSPAMT

--

- Statutory Maternity Pay

Use SMPAMT or SMPRATE if SMPAMT unknown

Also add

- Boarder/Lodger payments (only if they are greater than the tax free allowance - £4,250)

Use amounts held in BOARDER and LODGER

Add all these together to get TAXINC

Then subtract

- Any other deductions from pay, i.e. deductions for pension or superannuation, and AVCs.

Subtract DEDUC1-2 from taxable income

Calculate total allowances:

Add together

- Personal Allowance (depending on age)

Everyone entitled - use AGE variable

Then, if applicable, also add

- Married couple's allowance

Married people – If only one of the couple is working then assign MCA to that person. If both are working, then assign MCA to highest earner. If this makes the highest earner a non-taxpayer then assign excess MCA to lower earner. If lower earner is still a non-taxpayer then reassign MCA to higher earner.

If MARITAL=1 and INEARNNS>0 for one person only, then allocate all MCA to that person.

Else if Marital=1 and INEARNNS>0 for both people, then allocate MCA to higher earner.

Then if higher earner is a non-taxpayer, allocate excess MCA to lower earner.

Then if lower earner is still a non-taxpayer then re-allocate excess MCA to higher earner to make them a non-taxpayer also.

If married, but spouse not in household then assign full MCA to the person in the household

If MARITAL=1 and SPOUT=1 then assign full MCA

 --

Widowed in last year (also applicable for those divorced & separated, but no info on this held in FRS)

If the difference between current age (AGE) and the age when widowed (W1) is less than or equal to 1, then assign full MCA to widow.

Lone parents

FAMTYPBU=5, receive allowance of £1,830

- Cohabiting couples

Cohabiting couples with children are entitled to MCA, so treat as for married people above (but MARITAL=2)

- Blind person's allowance

If SPCREG1=1

- Youth Training Allowance

If AGE=16 or AGE=17 and TRAIN=1 then additional allowance of £29.50/£35.00 per week.

- Bereavement Allowance

If widowed in last 2 years (AGE-W1 <=1) then entitled to bereavement allowance

Is income high enough to affect age allowance?

Once income exceeds the limit set (£15,600), age allowances are reduced by 50% on excess until eliminated. Allowances can only be removed up to a maximum of the value of age additions. Personal allowances are reduced first, then MCA's. There is no reduction to the spouse's MCA. However, because this DV does not show the amount of tax being paid, this can be ignored, as these people will be flagged as taxpayers anyway due to their level of income.

Taxpayer (Y/N)?

If

TAXABLE INCOME – TOTAL ALLOWANCES > 0 then TAXPAYER=1

Else if

TAXABLE INCOME – TOTAL ALLOWANCES <= 0 then TAXPAYER=0

 --

TENTYPE

Purpose : To indicate the number of households in any specific tenure type.
 Created : 14 December 1992
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 7
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Issue date : 17 April 2003
 Amendments : VC - 28 January 1993
 : VC - 15 March 1993 Amendments to insert the part-own part-renter category
 : into the appropriate renter category, to put tenure = 2 or 3 into the owner :
 : categories and to exclude tenure = 5 as these tenants are not eligible for :
 : Housing Benefit.
 : VC - 18 May 1993. Amended to re-insert tenure = 5 as a separate :
 : category and also to add another category for co-ownership schemes : as both
 : of these groups are not eligible for HB.
 : VC - 9 June 1993. Amended to insert extra category for shared ownership.
 : BS - 3 August 1995. Amended to include changes to V31 of the
 : questionnaire. New var FURNISH asked if TENURE equals new codes 6, 7 or
 : 8.
 : JS - to clarify tentype=3 category and to allow skipped values of LANDLORD,
 : FURNISH and OWNHOW (in cases where LANDLORD or TENURE have been
 : imputed)
 : JS - to note categories are actually slightly different from the name suggested!
 : VE - 22 May 1996 - No initial Version 32 update needed
 : VE - 10 June 1996 - To bring specification in line with Harmonisation of Survey
 : Outputs: part owner/part renter cases are now included with cases buying with
 : a mortgage and lettings which go with the job of someone in the household are
 : now classified as rented privately.
 : SG - 30 October 1997 - V33 changes - values of Tenure, Landlord, Othway
 : EP - 13 August 1998 - V34 update, Tenancy agreement question (SHORT)
 : split up into SHORT1 and SHORT2
 : JC - 20 September 1999 - Security completed, no other changes for V35

1 Definition

This variable is coded as

- 1 LA / New Town / Council rented
- 2 Housing Association / Co-Op / Trust rented
- 3 Other private rented unfurnished
- 4 Other private rented furnished

 --

- 5 Owned with mortgage (**Includes part rent/part own**)
- 6 Owned outright
- 7 Rent-free
- 8 Squats

- 1 Not applicable to this case
- 2 Unable to derive variable

Tenure type is derived from the variable tenure in the HOUSEHOL record which when combined with a second variable from either the OWNER (furnish) or the RENTER (landlord) records will indicate to which category of tenure that particular household belongs.

All tenants whose accommodation goes with the job of someone in the household are allocated to rented privately, even if landlord is a local authority or housing association or Trust, or if rent free.

FURNISH is only asked where landlord=6-8, coding for TENTYPE=3 needs to reflect this

2 FRS Specification

For each household, process HOUSEHOL record

Code Condition

Where TENURE = 4, process RENTER record for details of type of landlord

- 1 If TENURE = 4 and LANDLORD = 1 (council/New Town Corporation) and ACCJOB=2
- 2 If TENURE = 4 and LANDLORD = 2 (Housing Association/Co-Op/Trust) and ACCJOB=2
- 3 If TENURE = 4 and (LANDLORD= 3,4,5,6,or 7 and FURNISH = 2) or (ACCJOB=1 and FURNISH=2) (other unfurnished)
- 4 If TENURE = 4 and ((LANDLORD = 3,4,5,6,or 7 and (FURNISH = 1 or FURNISH= -1)) or LANDLORD=-1) or (ACCJOB=1 and FURNISH=1) (other furnished)

Where TENURE = 1 or 3, process OWNER record for details of how the property is owned

- 5 If TENURE = 2 or 3 (owns/bought with mortgage or loan/part own-rent)
- 6 If TENURE = 1 (owns/outright) (*this closes down the route*)

Where TENURE = 5

- 7 If TENURE = 5 (rent-free).

Where TENURE = 6 (squatting).

--

8 If TENURE = 6 (squatting)

-1 Not applicable to this case

-2 Unable to derive - if any of the above variables are missing.

3 Results

Tabulation is required to show the numbers of households which fall into each category.

4 Test Cases

A Single person living in Local Authority housing.

B Single person living in Housing Association accommodation.

C Single person living in an unfurnished flat rented from a private landlord.

D Single person living in a furnished flat rented from a private landlord.

E Single person living in his/her own home with a mortgage outstanding on the property.

F Single person living in his/her own home with no outstanding mortgage.

G Single person living rent-free.

Repeat all of the above for couples.

 --

TOTCAPBU

Purpose : To show the total amount of capital an adult possesses
 Created : 15 January 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : TOTCAPCH - total amount of child's capital
 Children :
 Parents :
 Lead User : Take-up
 Amendments: : updated to V31: totsav now held on BU record
 : VE - 5 June 1996 - Initial amendments for V32 - Removed SAMPLQTR
 information because no longer needed
 : VE - 4 July 1996 - Amended to refer to constants table
 : VE - 3 December 1996 - Amended to clarify the situation where TOTSAVE is
 missing and to use HOWMUCHE for assettype 8 where it exists
 : SG - 16 December 1997 - V33 changes, asset numbers
 : SG - 17 February 1998 - National Savings account amounts to be based on
 NSAMT band
 : EP - 13 August 1998 - No initial V34 update needed
 : SG - 16 February 1999 - Allow for skipped HOWMUCH for current accounts
 : EP - 18 May 1999 - General V34 tidy up
 : SB - 3 November 1999 - Security completed, no other changes for V35
 Issue date: : 3 November 1999

1 Definition

This variable is coded as

TOTCAPBU The total amount of capital adults in a benefit unit possess from all sources.

-1 Not applicable to this case.

-2 Unable to derive due to missing values.

The total amount of capital a person has is calculated using two methods. Firstly, for those cases which have an assets record from several variables that have information about the number and type of different assets/accounts. The variable ASSETYPE indicates whether or not a person has a particular asset so if this variable is coded 1-10 or 13-19 there should be an additional variable which will hold the amount. If a person has more than one asset, an individual's capital will be the total of all of these variables. TOTCAPBU is derived by adding together amounts for the head and, where appropriate, the spouse.

Note: during the FRS interview, for asset types 6, 7, 8, 9, 10, 14, 17 (Gilts, stocks and shares, unit / investment trusts, SAYE, PEPs, national savings capital/deposit bonds,) respondents are asked for an

 --

estimate of their holdings (coded at HOWMUCH). During the office edit, actual values are looked up and stored in HOWMUCHE. Coding therefore is amended to look at HOWMUCHE if it exists and otherwise, HOWMUCH. (This is reflected in the flat file which holds HOWMUCHE in preference to HOWMUCH where appropriate.) However, for national savings certificates, issue value is used in place of HOWMUCH so coding is unaffected.

The asset questions in FRS are only asked of individuals within benefit units who have declared that the total value (all adults combined) have capital between £1,500 and £20,000. For benefit units that have answered that they have assets of less than £1,500 or over £20,000 no further details about assets are collected. This is also the case for benefit units who have refused to answer any more questions about their assets. For these BUs, therefore, the total amount of capital it possesses needs to be calculated in another way.

This second method uses the amount of interest received on a person's capital investments to calculate an approximate amount of capital per individual which could have generated that amount of interest. The interest is divided by an appropriate (weekly) interest rate and then multiplied by 100 to get a capital amount. This figure then needs to be multiplied by 52 to produce an annual figure. The rates used have been produced by ASD3A and follow the traditional FES approach to calculating capital. This method gives a capital amount for people who have not answered any further questions about their assets. Again, these are totalled for head and spouse to calculate TOTCAPBU. **The constants r*** are held in the spreadsheet p:\frs\shared\frs35\metadata\Dvmeta35.xls.**

However, there is a mismatch between the interest/dividend payments and the amounts collected in the assets block. Whilst amounts for National Savings Certificates, SAYE, premium bonds and National Savings Bonds, First Option Bonds and Yearly Plan and Pensioners Guaranteed Income Bonds are included, respondents are only asked whether they have these type of savings at question OTINVA. Therefore, there are no interest/dividend amounts on which to calculate holdings. From 1996-97 individuals are asked to state the banded amount of their holdings in such accounts (NSAMT) and the band mid-point is now used to increase TOTCAP. These band midpoints are read as informats from CONST33.XLS. Note that the top band is £30,000+ so a mid-point cannot be calculated. This is read in as £30,000 so note that analysis of savings should not use bands above £30,000. If the individual has TOTSAV=5 ("coy") then they do not get asked NSAMT and we use the old method of factoring up TOTCAPBU by 20%.

The coding of TOTSAV was changed at the beginning of July 1994 (SAMPLQTR=2). The four codes used in 1997/98 were split into 5 categories:

was	from	
	July 94	
1	1	less than £1,500
2	2	£1,500 to £8,000
2	3	£8,000 to £20,000
3	4	£20,000 and over
4	5	does not wish to say

The routing remains the same, but for the final quarter's data the program has to be altered to correctly identify the method to calculate TOTCAPBU

 --

2 FRS Specification

Code

Condition

TOTCAPBU

From BENUNIT record, if TOTSAV = 2 or TOTSAV=3

From ASSETS record, for each asset held for each adult

If ASSETYPE = 1, get amount of capital in current accounts from HOWMUCH (ignore if skipped).

If ASSETYPE = 2, get amount of capital in savings accounts from HOWMUCH.

If ASSETYPE = 3, get amount of capital in savings accounts from HOWMUCH.

If ASSETYPE = 4, get amount of capital in TESSAs from HOWMUCH.

If ASSETYPE = 5, get amount of capital in other accounts from HOWMUCH.

If ASSETYPE = 6, get HOWMUCHE if it exists else use HOWMUCH (gilts)

If ASSETYPE = 7, get HOWMUCHE if it exists else use HOWMUCH (trusts)

If ASSETYPE = 8, get HOWMUCHE if it exists else use HOWMUCH (stocks/shares)

If ASSETYPE = 9, get HOWMUCHE if it exists else use HOWMUCH (PEPs)

If ASSETYPE = 11 or 12, get HOWMUCHE if it exists else use the issue value of National Savings Certs from ISSVAL.

If ASSETYPE = 13, get amount of capital from HOWMUCH (Pensioners Guaranteed Bonds).

If ASSETYPE = 14, get HOWMUCHE if it exists else use HOWMUCH (SAYE)

If ASSETYPE = 15, get amount of capital in premium bonds from HOWMUCH.

If ASSETYPE = 16, get amount of capital in Nat Sav Income Bonds from HOWMUCH.

If ASSETYPE = 10 or 17, get HOWMUCHE if it exists else use HOWMUCH (Nat Sav Capital/Deposit Bonds).

If ASSETYPE = 18, get HOWMUCH (First Option Bonds).

If ASSETYPE = 19, get HOWMUCH (Yearly Plan).

TOTCAPBU is then the total of any occurrences of the above

Else if from BENUNIT record, TOTSAV=1,4 or 5 or missing

Set the interest rates for each type of account

Then calculate amounts as follows -

From ACCOUNTS record

If ACCOUNTS = 1 and TOTSAV = 1 calculate CAP = ACCINT/r01a%

Else if ACCOUNTS = 1 and TOTSAV = 4 calculate CAP = ACCINT/r01b%

Else if ACCOUNTS = 1 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r01c% (ADULT CURRENT)

If ACCOUNTS = 2 and TOTSAV = 1 calculate CAP = ACCINT/r02a%

Else if ACCOUNTS = 2 and TOTSAV = 4 calculate CAP = ACCINT/r02b%

Else if ACCOUNTS = 2 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r02c% (ADULT NSB/PO)

 --

If ACCOUNTS = 3 and TOTSAV = 1 calculate CAP = ACCINT/r03a%
 Else if ACCOUNTS = 3 and TOTSAV = 4 calculate CAP = ACCINT/r03b%
 Else if ACCOUNTS = 3 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r03c% (ADULT NSB/PO INVEST)

If ACCOUNTS = 4 and TOTSAV = 1 calculate CAP = ACCINT/r04a%
 Else if ACCOUNTS = 4 and TOTSAV = 4 calculate CAP = ACCINT/r04b%
 Else if ACCOUNTS = 4 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r04c% (TESSA)

If ACCOUNTS = 5 and TOTSAV = 1 calculate CAP = ACCINT/r05a%
 Else if ACCOUNTS = 5 and TOTSAV = 4 calculate CAP = ACCINT/r05b%
 Else if ACCOUNTS = 5 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r05c% (Other SAVINGS/INVESTMENT ACCOUNTS)

If ACCOUNTS = 6 and TOTSAV = 1 calculate CAP = ACCINT/r06a%
 Else if ACCOUNTS = 6 and TOTSAV = 4 calculate CAP = ACCINT/r06b%
 Else if ACCOUNTS = 6 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r06c% (GILTS)

If ACCOUNTS = 7 and TOTSAV = 1 calculate CAP = ACCINT/r07a%
 Else if ACCOUNTS = 7 and TOTSAV = 4 calculate CAP = ACCINT/r07b%
 Else if ACCOUNTS = 7 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r07c% (UNIT TRUST)

If ACCOUNTS = 8 and TOTSAV = 1 calculate CAP = ACCINT/r08a%
 Else if ACCOUNTS = 8 and TOTSAV = 4 calculate CAP = ACCINT/r08b%
 Else if ACCOUNTS = 8 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r08c% (STOCKS & SHARES)

If ACCOUNTS = 9 and TOTSAV = 1 calculate CAP = ACCINT/r09a%
 Else if ACCOUNTS = 9 and TOTSAV = 4 calculate CAP = ACCINT/r09b%
 Else if ACCOUNTS = 9 and TOTSAV = 5 or missing calculate CAP =
 ACCINT/r09c% (PEP)

TOTCAPBU is then the total of each occurrence of CAP

If ACCOUNT=10,11,12,13,14,15,16,17,18,19
 then Read midpoint of band that NSAMT represents (if exists) into NSVALUE
 IF NSAMT exists TOTCAPBU=TOTCAPBU+NSVALUE
 ELSE TOTCAPBU=TOTCAPBU*1.2

Then TOTCAPBU=TOTCAPBU*52

-2 Unable to derive because of missing values.

 --

TOTCAPCH

Purpose : To calculate the total amount of capital a child possesses.
 Created : 9 December 1992
 Database Table : CHILD
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : TOTCAP- total amount of adult's capital
 Children :
 Parents :
 Core variable/user : Take-up
 Amendments : VC - 22 March 1993 - To amend variable names from asset table which hold
 amount of each type of asset.
 : VC - 6 October 1993 To add further coding to for children where TOTSAVE is
 less than £1500 or greater than £20000 as these were missing from original.
 : VE - 5 June 1996 - No initial updates possible for V32
 : VE - 4 July 1996 - To amend for standard constants table
 : VE - 8 November 1996 - To remove the 'hole' between £8000 and £20000
 : VE - 3 December 1996 - To clarify the situation where TOTSAVE is missing
 and to use HOWMUCHE for ASSETTYPE 7
 : JS - 25 March 1997 to remove restrictions on TOTCAP/TOTSAVE
 : SG - 16 December 1997 - V33
 : EP - 13 August 1998 - No initial V34 update needed
 : JC - 8 September 1999 - Security completed, no other changes needed for V35
 Issue date : 17 April, 2003

1 Definition

This variable is coded as

TOTCAPCH The total amount of capital a child possesses from all sources.

-1 Not applicable to this case.

-2 Unable to derive because of missing values.

The total amount of a child's capital is derived from several variables which hold information about the number and type of different accounts/assets. The variable ASSETTYPE indicates whether or not a child has a particular asset so if this variable equals 3, 5, 7, 10, 12 or 14 or 19 there should be an additional variable which will hold the amount. If the child has more than one asset TOTCAPCH will be the total of all of the assets.

The asset questions in FRS are only asked of people and children who have declared that they have capital between £1500 and £20000. For cases where it has been said that the child has assets under £1500, over £20000 or has refused to say (TOTSAVE = 1, 4 or 5) no further details about that child's

 --

assets is collected. In these cases, therefore, the total amount of capital must be calculated in another way.

The amount of interest received on a child's capital investments is used to calculate an approximate amount of capital which could have generated that amount of interest. The interest is divided by an appropriate interest rate percentage to get a capital amount. The rates used have been produced by Mr Ray and follow the traditional FES approach to calculating capital (**held in the spreadsheet p:\frs\shared\frs35\metadata\DVmeta35.xls.**). This method also gives a capital amount for people who have refused to give any further details about their assets which also follows the FES.

2 FRS Specification

For each CHILD

Code

Condition

TOTCAPCH From CHILD table, if TOTSAVE = 2 or 3

From ASSETS record, for each asset held for each child

If ASSETYPE = 3, get amount of capital in NSB ordinary, NSB investment, Building Society, bank accounts from HOWMUCH.

If ASSETYPE = 5, get amount of capital in Gilts, Unit Trusts, Stocks and Shares) from HOWMUCHE if exists, else use HOWMUCH.

If ASSETYPE = 7, get amount from HOWMUCHE if it exists else use issue value of all Index-linked and Fixed National Savings Certificates from ISSVAL

If ASSETYPE = 10, get amount of capital in Premium Bonds from HOWMUCH.

If ASSETYPE = 12, get amount of capital in National Savings Income Bonds from HOWMUCH.

If ASSETYPE = 14, get amount of capital in National Savings Capital Bonds and Children's Bonus Bonds from HOWMUCHE if exists, else use HOWMUCH.

If ASSETYPE = 19, get amount of capital in Child Yearly Plan from HOWMUCH.

TOTCAPCH is then the sum of the above where TOTSAVE = 2 or 3.

If TOTSAVE = 1, 4 or 5 or missing

Then calculate amount as follows -

From ACCOUNTS record

If ACCOUNTS = 20 and TOTSAV = 1, calculate CAP = ACCINT/r02a%

Else if ACCOUNTS = 20 and TOTSAV = 4, calculate CAP = ACCINT/r02b%

Else if ACCOUNTS = 20 and TOTSAV = 5 or missing, calculate CAP =
 ACCINT/r02c%

If ACCOUNTS = 21 and TOTSAV = 1, calculate CAP = ACCINT/r03a%

Else if ACCOUNTS = 21 and TOTSAV = 4, calculate CAP = ACCINT/r03b%

--

Else if ACCOUNTS = 21 and TOTSAV = 5 or missing , calculate CAP =
ACCINT/r03c%
If ACCOUNTS = 22 or ACCOUNTS = 23 and TOTSAV = 1, calculate CAP =
ACCINT/r01a%
Else if ACCOUNTS = 22 or ACCOUNTS = 23 and TOTSAV = 4, calculate CAP
= ACCINT/r01b%
Else if ACCOUNTS = 22 or ACCOUNTS = 23 and TOTSAV = 5 or missing,
calculate CAP = ACCINT/r01c%
If ACCOUNTS = 24, calculate CAP = ACCINT/r11a%
If ACCOUNTS = 25, calculate CAP = ACCINT/r13a%
If ACCOUNTS = 26, calculate CAP = ACCINT/r14a%

TOTCAP is then the total of each occurrence of CAP.

- 1 Not applicable to this case.
-2 Unable to derive because of missing values.

3 Results

Tabulations to show the total number of adults by the amount of capital they possess sorted into bands,
for example

Under £500
£500 - £1000
£1000 - £2000
£2000 - £4000
£4000 - £6000
£6000 - £8000
£8000 - £10000
£10000 - £12000
£12000 - £14000
£14000 - £16000
£16000 - £18000
£18000 - £20000
£20000 and over

NB - Test cases removed for immediate future as test dataset still to be finalised.

 --

TOTGNTCH

Purpose : To show the total amount of educational maintenance grants or scholarships
 : received **directly** by a child.
 Created : 15 July 1993
 Database Table : CHILD
 Minimum Value : 0
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables : TOTGRANT - total grant received by an adult
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : AG 9 August 1993. The FRS database will now keep the grant amounts as
 annual figures.
 : JS - 7 March 1996: amended to look at payments actually received by
 students, rather than the total awarded
 : VE - 22 May 1996 - No initial amendments needed for V32
 : JS - 24 March 1997 - add in where 3 grants recorded at GRTNUM
 : SG - 23 December 1997 - No initial amendments needed for V33
 : EP - 13 August 1998 - No initial V34 update needed
 : SP - 6 September 1999 – Security completed, no other changes for V35
 Issue date : 6 September 1999

NB - This will be provided as an annual amount and the individual models will have to calculate weekly entitlement for the duration of the academic year.

1 Definition

This variable is coded as

TOTGNTCH The total amount of income received by a child from educational grants, maintenance grants or scholarships.

-1 Not applicable to this case

-2 Unable to derive

TOTGNTCH will be derived from the variables GRTNUM, GRTSCE1, GRTSCE2, GRTDIR1, GRTDIR2. GRTNUM gives the number of grants/scholarships received and the amount of the grant then depends on whether the grant/scholarship is funded by the state or from a private or overseas source. This will be an annual amount as requested by ISM.

2 FRS Specification

For each CHILD where FTED = 1 (in full time education)

 --

<u>Code</u>	<u>Condition</u>
TOTGNTCH	<p>From CHILD table,</p> <p>If GRTNUM = 1 (Number of grants = one) and GRTSCE1 = 1 (source is state), get amount from GRDIR1 (amount of grant paid directly to student). If GRTNUM = 1 (Number of grants = one) and GRTSCE1 = 2 or 3 (source is private or overseas), get amount from GRDIR1 (amount of grant paid directly to student).</p> <p>If GRTNUM = 2 or 3 (two or more grants) get the amount for first grant as above in addition to:- If GRTSCE2 = 1 (source is state) get amount from GRDIR2. If GRTSCE2 = 2 or 3 (source is private or overseas), get amount from GRDIR2.</p> <p>TOTGNTCH will then be the sum of the two grants. This is equivalent to summing all occurrences of GRDIR1 and GRDIR2 for each child in full time education</p> <p>(NB - this does not include top-up loans as they are not payable to children)</p>
-1	Not applicable to this case
-2	Unable to derive as any of the above variables are missing.

3 Results

Tabulation is required to show the number of children receiving educational grants, maintenance grants or scholarships by the amount received sorted into bands of

- Under £500
- £5100 - £1000
- £1000 - £1500
- £1500 - £2000
- £2000 - £2500
- £2500 - £3000
- £3000 - £4000
- £4000 - £5000
- £5000 and over

4 Test Cases

None as yet

 --

TOTGRANT

Purpose : To show the total amount of educational maintenance grants or scholarships
 : for higher education received **directly** by an adult.
 Created : 22 January 1993
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : VC - 8 March 1993. To specify an annual amount as requested by ISM.
 : VC - 22 March 1993. To amend the sort bands in the tabulation.
 : AG - 9 August 1993. The database will store grant amounts as annual figures
 : so no multiplication required.
 : JS - 19 January 1996: amended to include all students in further education
 : JS - 7 March 1996: amended to look at payments actually received by
 : students, rather than the total awarded (whether including or excluding fees)
 : VE - 22 May 1996 - No initial amendments needed for V32
 : VE - 4 December 1996 - To set TOTGRANT to zero initially
 : VE - 31 January 1997 - To remove the condition for TOTGRANT only to be
 : derived for students in line with HBAI
 : JS - 24 March 1997 - to add in where 3 or more grants recorded
 : SG - 14 November 1997 - No initial changes for V33
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 6 September 1999 – Security completed, no other changes for V35
 Issue date : 6 September 1999

NB - This will be provided as an annual amount and the individual models will have to calculate weekly entitlement for the duration of the academic year. Totgrant replaces FES variable IN411 as FRS cannot differentiate between a grant or a scholarship, therefore, the total amount of grants or scholarships are included.

1 Definition

This variable is coded as

TOTGRANT The total amount of income received by an adult from educational grants, maintenance grants or scholarships.

-1 Not applicable to this case

-2 Unable to derive

 --

TOTGRANT will be derived from the variables GRTNUM, GRTSCE1, GRTDIR1, GRTSCE2, and GRTDIR2. GRTNUM gives the number of grants/scholarships received and the amount of the grant then depends on whether the grant/scholarship is funded by the state or from a private or overseas source. This will be an annual amount as requested by ISM.

2 FRS Specification

For each adult

Code Condition

Set TOTGRANT to ZERO

TOTGRANT From ADULT table,

If GRTNUM = 1 (Number of grants = one) and GRTSCE1 = 1 (source is state), get amount from GRTDIR1 (amount of grant).

If GRTNUM = 1 (Number of grants = one) and GRTSCE1 = 2 or 3 (source is private or overseas) get amount from GRTDIR1 (amount of grant) .

If GRTNUM = 2 or 3 (two or more grants) get the amount for first grant as above in addition to:-

If GRTSCE2 = 1 (source is state) get amount from GRTDIR2.

If GRTSCE2 = 2 or 3 (source is private or overseas) get amount from GRTDIR2

TOTGRANT will then be the sum of the two grants.

(NB - this does not include top-up loans)

-1 Not applicable to this case - adult not in full-time education.

-2 Unable to derive as any of the above variables are missing.

3 Results

Tabulation is required to show the number of adults receiving educational grants, maintenance grants or scholarships by the amount received sorted into bands of

Under £500
 £5100 - £1000
 £1000 - £1500
 £1500 - £2000
 £2000 - £2500
 £2500 - £3000
 £3000 - £4000
 £4000 - £5000

--

£5000 and over

4 Test Cases

None as yet

 --

TOTHOOURS

Purpose : To indicate the total number of hours a person works each week (main and subsidiary)
 Created : 10 February 1993
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : VC - 16 March 1993, to change the database table.
 : VE - 22 May 1996 - No initial amendments needed for V32
 : SG - 12 November 1997 - No initial amendments needed for V33
 : EP - 9 September 1998 - Replacement of QHRS, QHRSSELF, EMPOVT variables for V34
 : SB - 3 September 1999 – Security completed, no other changes for V35
 Issue date : 3 September 1999

NB - This replaces SEFTPT as this was only produced for self-employed people - TOTHOOURS also includes employees .

1 Definition

This variable is coded as

- TOTHOOURS The total number of hours worked by an adult from all jobs whether main or subsidiary, employed or self-employed including all regular overtime.
- 1 Not applicable to this case
- 2 Unable to derive due to missing values

TOTHOOURS is derived from the variables ~~QHRS, QHRSSELF and EMPOVT~~ **EVEROT, TOTUS1 and DVTOTHRU**. These are to be obtained for each job a person has and the total number of hours are then calculated. The variables include the number of hours as an employed earner, as a self-employed person and from any regular overtime.

2 FRS Specification

For each ADULT, set TOTHOOURS to zero

<u>Code</u>	<u>Condition</u>
TOTHOOURS	For each adult access all JOB records

 --

~~If EMPEE = 1 (employed person), get QHRS (number of hours worked by an employee) and EMPOVT (number of hours of overtime worked by an employee) and add into TOTHOURS~~

~~If EMPEE = 2 (self-employed), get QHRSSELF (number of hours worked by a self-employed person) and add into TOTHOURS~~

~~If EVEROT = yes, and DVTOTHRU not in (.A.,.B.,.C) then~~

~~TOTHOURS + DVTOTHRU~~

~~Else If TOTUS1 not in (.A.,.B.,.C) then~~

~~TOTHOURS + TOTUS1~~

~~TOTHOURS will then be the sum of all occurrences of QHRS, EMPOVT and QHRSSELF.~~

3 Results

Tabulation is required to show adults by the number of hours worked each week sorted into bands of for example

- Under 16
- 16 - 24
- 24 - 30
- Over 30

4 Test Cases

None as yet

TOTSAVBU

Purpose : To create variable consistent with V30 TOTSAVBU with 4 categories
 Created :
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables : TOTCAPBU - total amount of benefit unit's capital
 Children :
 Parents : TOTSAV
 Core variable/user : FRS (publication)
 Amendments : VE - 5 June 1996 - To tidy up spec for V32
 : SG - 12 November 1997 - No initial updates for V33
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 7 October 1999 - Security completed, no other changes to V35
 Issue date : 7 October 1999

1 Definition

This variable is coded as

- 1 Less than £1,500
- 2 Over £1,500 and up to £20,000
- 3 Over £20,000
- 4 Does not wish to say
- 1 Not applicable to this case (where TOTSAV has been skipped: question only asked where respondents have already stated interest on savings)
- 2 Unable to derive due to missing values (should only occur where "don't know" has been input, "refused" cases should be included as part of code 4 - does not wish to say).

The question TOTSAV is asked of adults and gives respondents' estimate of their total savings held by partner/spouse. From July 1995, the coding was changed to include an additional category for £1,500 to £8,000 such that:

- 1 Less than £1,500
- 2 Over £1,500 and up to £8,000
- 3 Over £8,000 and up to £20,000
- 4 Over £20,000
- 5 Does not wish to say

This split coding is held on the database in TOTSAV. TOTSAVBU condenses the 5 codes.

2 FRS specification

From table BENUNIT

<u>Code</u>	<u>Condition</u>
1	TOTSAV=1
2	TOTSAV=2 or TOTSAV=3
3	TOTSAV=4
4	TOTSAV=5 or TOTSAV= -8
-1	TOTSAV= -1
-2	TOTSAV= -9

 --

TTWCOSTS

Purpose : To show weekly travel to work costs for each adult
 Created :
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : JS: for V31, to change calculation of weekly travel pass costs
 : JS 18/12/95 to take on changes made to V30 (additional methods of
 transport, coding of variable work costs cases)
 : JS - 21 February 1996 to allow for skipped values where variables have been
 imputed
 : JS - 15 March 1996, to stop weeklyising of pass within program (already
 weekly on the data base)
 : JS - 17 April 1996 to amend cases where TTWCOST or TTWPAY has been
 skipped to use calculated amount for car/bike (currently no change made)
 : VE - 5 June 1996 - to include new category of TTWCODE for V32
 : VE - 14 June 1996 - To amend for 1995-96 rates for cost per mile
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 29 October 1996 - to include TTWREC
 : VE - 9 December 1996 - To amend spec for change in data structure
 regarding TTWCODE1 and TTWCODE2
 : VE - 1 April 1997 - To tidy up spec to bring in line with SAS code for V32
 : SG - 8 January 1998 - V33 updates (CONST33.XLS location)
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 17 August 1999 - Change of source of mileage rates data. Motorbikes
 changed from 9p to 25.3p, See Dvmeta35
 : SB - 28 October 1999 - Security completed, TTWMOD1-6 replaced by
 TTWMOD
 Issue date : 28 October 1999

1 Definition

This variable is coded as

TTWCOSTS Total weekly travel to work costs paid by adult

0 Free travel

 --

- 1 Not applicable to this case - including adults with no usual place of work, or coded "other" form of transport (no information collected)
- 2 Unable to derive due to missing values

Travel to work costs are based on the number of round trips per week (TTWFRQ). Adults are only asked about their main method of transport and costs are collected on all methods of transport except walk/cycle or "other" (TTWMOD). Public transport costs are calculated using the costs of bus passes or season tickets, recorded fare, contributions to drivers of shared cars, or on costs per mile of journeys in cars or on motorcycle. DSS rates for costs per mile have been used. **The rates (CARRATE and BIKERATE) are held on the table in p:\frs\shared\frs35\metadata\Dvmeta35.xls**

Questionnaire asks about total distance travelled to usual place of work. Where respondents use both car/motorcycle and train/bus/tube, only the main method is recorded. Therefore we assume most of the journey is by one method of transport.

Where PSSAMT has been imputed, PSSDATE1/2 will have been skipped. Amount held in PSSAMT is taken as weekly. Where TTWCOST has been imputed, TTWCOST is skipped: do not change TTWCOSTS in these cases

2 FRS Specification

TTWCOSTS From adult record, add up costs for each mode of transport TTWMOD 2 - 5

TTWMOD in (4,5,6,7) (bus/train/tube/light rail) or TTWMOD=1 (works bus/company transport)

If TTWPSS=1 (Yes) and PSSAMT exists TTWCOSTS = PSSAMT
 Else if PSSDATE1/2 is skipped then cost of pass equals PSSAMT

Else if respondent does not have a season ticket, calculate cost of round trip and multiply by number of trips each week

If TTWPSS=2 (No)
 If ONEWAY=1 then TTWCOSTS=FARE*2*TTWFRQ
 else If ONEWAY=2 then TTWCOSTS=FARE*TTWFRQ

Else if travels in a car/van, first check whether car used in combination with bus/train/tube or works bus/company transport, if yes, assume average journey is 2 miles

If TTWMOD=1 and TTWPAY ≠ 3 (pays all/some of costs of taking car/van to work)
 then TTWCOSTS=TTWCOSTS+(4*TTWFRQ*CARRATE)

Else if only car/van used (or used in combination with walking/bicycle) calculate total costs per mile per week (CARCOST - based on midpoints of TTWFAR categories),

 --

If TTWMOD=1 (car or van)
 and TTWFAR=3 then CARCOST=0.5*2*TTWFRQ*CARRATE
 and TTWFAR=4 then CARCOST=2*2*TTWFRQ*CARRATE
 and TTWFAR=5 then CARCOST=4*2*TTWFRQ*CARRATE
 and TTWFAR=6 then CARCOST=7.5*2*TTWFRQ*CARRATE
 and TTWFAR=7 then CARCOST=17.5*2*TTWFRQ*CARRATE
 and TTWFAR=8 then CARCOST=32.5*2*TTWFRQ*CARRATE

Then calculate TTWCOSTS, taking into account any contributions

If TTWMOD=1
 If TTWPAY=1 (all) then TTWCOSTS=TTWCOSTS+CARCOST
 If TTWPAY=2 (some) and TTWCODE1=1 then TTWCOSTS=TTWCOSTS+TTWCOST
 If TTWPAY=2 and TTWCODE2=1
 then TTWCOSTS=TTWCOSTS+(CARCOST - TTWREC)
 (if CARCOST < TTWREC then set TTWCOSTS to zero)

else if TTWCOST and TTWREC have been skipped (where TTWCODE1/TTWCODE2 are set) or TTWPAY has been skipped or if TTWCODE1=3 and TTWCODE2=3 then
 TTWCOSTS=TTWCOSTS+CARCOST

Else if respondent drives a motorcycle, use similar approach.

If TTWMOD3=1 and (TTWMOD4=1 or TTWMOD5=1) and TTWPAY ≠ 3 (pays all/some of costs of taking motorcycle to work)
 then TTWCOSTS=TTWCOSTS+(4*TTWFRQ*BIKERATE)

Else if only motorbike used (or in combination with walking/bicycle) calculating variable BIKECOST

If TTWMOD=2 (motorcycle)
 and TTWFAR=3 then BIKECOST=0.5*2*TTWFRQ*BIKERATE
 and TTWFAR=4 then BIKECOST=2*2*TTWFRQ*BIKERATE
 and TTWFAR=5 then BIKECOST=4*2*TTWFRQ*BIKERATE
 and TTWFAR=6 then BIKECOST=7.5*2*TTWFRQ*BIKERATE
 and TTWFAR=7 then BIKECOST=17.5*2*TTWFRQ*BIKERATE
 and TTWFAR=8 then BIKECOST=32.5*2*TTWFRQ*BIKERATE

Then calculate TTWCOSTS, taking into account any contributions

If TTWMOD=2
 If TTWPAY=1 (all) then TTWCOSTS=TTWCOSTS+BIKECOST
 If TTWPAY=2 (some) and TTWCODE1=1, then TTWCOSTS=TTWCOSTS+TTWCOST
 If TTWPAY=2 and TTWCODE2=1
 then TTWCOSTS=TTWCOSTS+(BIKECOST - TTWREC)
 (if BIKECOST < TTWREC then set TTWCOSTS to zero)

 --

else if TTWCOST or TTWREC have been skipped (where TTWCODE1/TTWCODE2 are set) or TTWPAY has been skipped or if TTWCODE1=3 or TTWCODE2=3 then TTWCOSTS=TTWCOSTS+BIKECOST

- 0 TTWFAR=1 (work at home, live at work, no work journey)
 If TTWPSS=1 and PSSAMT=0 (has pass and costs nothing)
 or TTWPSS=2 and FARE=0 (does not have pass but fare costs nothing)
 or cash received from passengers etc greater than calculated costs (costs therefore set to zero)
 or ((TTWMOD=1 or 2) and TTWPAY=3) (uses car/motorcycle and pays no costs)

Note: Free travel cards/fares and contributions will have already been calculated: all categories shown here for completeness

Note some cases exist where individual drives to a station but has a free travel pass. These cases still count as zero travel costs.

- 1 If questions in n_Travel have been skipped (questions do not apply to this case)
 or TTWFAR=2 (varies, no usual place of work)
 or TTWMOD=9 (other form of transport: no information collected)

- 2 Any variables missing

TTWMODE

Purpose : Categorical breakdown of mode of transport to work (publication purposes)
Created : 22 February 1999
Database Table : ADULT
Minimum Value : 1
Maximum Value : 7
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : FRS (publication)
Amendments : SB - 9 November 1999 - Security complete, re-define bands
Issue date : 22 February 1999

1 Definition

TTWMODE categorises the mode of transport used to get to work by individuals for publication purposes, and is derived from the variable TTWMOD. It is coded as follows:

- 1 Car / Van
- 2 Motorcycle (includes minibus/works van)
- 3 Bicycle
- 4 Bus (includes coach, private bus)
- 5 Train
- 6 LT underground
- 7 Light rail
- 8 Walk
- 9 Other

2 FRS Specification

For each adult:

<u>Code</u>	<u>Condition</u>
1	IF TTWMOD=1
2	IF TTWMOD=2
3	IF TTWMOD=3
4	IF TTWMOD=4
5	IF TTWMOD=5
6	IF TTWMOD=6
7	IF TTWMOD=7

- 8 IF TTWMOD=8
- 9 IF TTWMOD=9

- 1 Not applicable to this case
- 2 Unable to derive TTWMODE

 --

UGRSPAY

Purpose : To show the total amount of usual earnings received by an adult from each job as an employee, excluding any income from odd jobs..

Created :

Database Table : JOB

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : GROSSPAY

Children :

Parents :

Core variable/user : PSM

Issue date : 3 November 1999

Amendments : JS - 21 February 1996 to allow for skipped values where variables have been imputed

: JS - 21 March 1996 to correct treatment of UGROSS so it is only considered if jobtype=1

: VE - 4 June 1996 - Initial amendments for V32 - New categories of DEDUC and OTHDED

: SG - 6 January 1998 - Initial amendments for V33 - Period codes, replace TAXINC with INCLPAY3, new deductions

: SG - 11 February 1998 - PAYS LIP changes for V33

: EP - 13 August 1998 - No initial V34 update needed

: SB - 3 November 1999 - Security completed

Removal of Charity tax questions and replaced by CHRAMT

1 Definition

This variable is coded as

UGRSPAY The total usual gross earnings before deductions for Income Tax, NI etc from each jobs an adult may have as an employed earner, excluding any income from odd jobs.

-1 Not applicable where an adult does not have any jobs.

-2 Unable to derive where any variables missing

The variable UGRSPAY is derived from a variety of variables held in the ADULT and JOB tables and for each job held by that adult. *It is similar to GROSSPAY but looks at jobs individually and gives usual rather than last pay where appropriate.*

UGRSPAY is derived from the variable GRSWAGE which holds the person's gross earnings before tax, NI etc but only where the payslip has been consulted (where PAYS LIP = 1 or 2).

If the payslip has not been consulted (where PAYS LIP = 3) the amount of net pay is obtained from PAYAMT. This variable holds the total amount of net pay after all deductions have been taken off and these deductions must be added back to PAYAMT to find UGRSPAY.

 --

The amount of income tax deducted is found in PAYE and National Insurance in NATINS these must be found in all cases. Other deductions for example trade union fees, payments to charities etc are also to be added back to PAYAMT but may not be relevant to every case. **Therefore, if CHRAMT > 0 (indicating money was deducted for charities) then this value must be added to PAYAMT.**

If any of OTHDED1 to OTHDED8 = 1 there will be a deduction for pension/superannuation, union fees, friendly societies, sports social clubs, repayment of a loan from employer, private medical insurance and any other deductions with the amount held in the relevant DEDUC variable. DEDUC1 holds the amount for pension/superannuation, DEDUC2 holds the amount additional voluntary contributions, DEDUC3 holds the amount of union fees, DEDUC4 holds amount for friendly societies, DEDUC5 holds amount for sports clubs, DEDUC6 holds amount for repayment of a loan from employer, DEDUC7 holds amount for private medical insurance and DEDOTH holds the amount for any other deduction not already mentioned and any occurrence of these must be added to PAYAMT.

The variables OTHDED1 to OTHDED8, DEDUC1 to DEDUC7 and DEDOTH are database variables collected from the questions OTHDED in the e-main block which asks were there any other deductions from your wage or salary and DEDUC which holds the amount of deduction in each case.

To get a person's gross earnings a check must be made to see if an income tax refund was included in PAYAMT. Therefore, if INCLPAY3=1 the amount held in TAXAMT has to be deducted from PAYAMT to get a true amount of gross earnings. However, if last pay was not usual, then UGRSPAY uses UGROSS.

where payslip has been imputed to yes, GRSWAGE will have been skipped. Similarly where PAYAMT is imputed, PAYPD and other variables may also be skipped. This has to be catered for in the specification.

2 FRS Specification

For each ADULT and each JOB set UGRSPAY to zero.

UGRSPAY From JOB table

IF WORKING=1 or JOBAWAY=1 - process each JOB record for that person where EMPEE=1

If PAYPD equals -1 or 1-10,13,26,52 and

If PAYSZIP = 1 or 2 and GRSWAGE exists, calculate UGRSPAY = amount in GRSWAGE.

Or if PAYSZIP = 3 or ((PAYSZIP=1 or 2) and GRSWAGE=-1), calculate UGRSPAY as follows

UGRSPAY = PAYAMT and

If PAYE exists add PAYE to UGRSPAY

--

If NATINS exists add NATINS to UGRSPAY

If CHRAMT exists add CHRAMT to UGRSPAY

If DEDUC1 exists add DEDUC1 to UGRSPAY

If DEDUC2 exists add DEDUC2 to UGRSPAY

If DEDUC3 exists add DEDUC3 to UGRSPAY

If DEDUC4 exists add DEDUC4 to UGRSPAY

If DEDUC5 exists add DEDUC5 to UGRSPAY

If DEDUC6 exists add DEDUC6 to UGRSPAY

If DEDUC7 exists add DEDUC7 to UGRSPAY

If DEDOTH exists add DEDOTH to UGRSPAY

Then if JOBTYP=1 and INCLPAY3=1 and TAXAMT exists subtract TAXAMT from total UGRSPAY

If JOBTYP=1 and PAYUSL = no and UGROSS exists and UPD= -1 or 1-10,13,26,52 then UGRSPAY=UGROSS

(this should be the same as coding for INDINC. The only difference is that check on UGROSS at the beginning is not necessary because DV is on a job basis and therefore it is possible to overwrite UGRSPAY with UGROSS at the end)

-2 If any of the above variables are missing or PAYPD = 90 or 95 or 97 *(should only be where PAYAMT is missing because for other variables, values are ignored if they do not exist)*

UPERSONA UPERSONC

Purpose : To show the person number within the Benefit Unit
 Created : AJG 10 September 1993
 Database Table : ADULT and CHILD
 Minimum Value : 1
 Maximum Value : 12
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : HBAI
 Amendments : VC - 17 February 1994 Amended to reflect version 30 changes
 : VE - 22 May 1996 - No initial Version 32 update needed
 : SG - 28 October - No initial Version 33 update needed
 : EP - 13 August 1998 - No initial V34 update needed
 : EP - 17 May 1999 - Complete rewrite of DV to accommodate changes to
 person numbering within the household/benefit unit. Also to ensure that head
 of household is the head of their particular benefit unit. For previous DV spec
 see 1996-97 specification.
 : SB - 17 April 2003 - Security completed
 - Spilt code into child and adult levels
 Issued : 17 April 2003

1 Definition

For the benefit unit in which the head of household lies, this variable assigns the value 1 to the head of household, and then increments for each adult and child within that benefit unit. For benefit units that do not contain the head of household, the value 1 is assigned to the first person within the benefit unit. Dependants are numbered in descending order of age.

2 FRS Specification

Process each BENUNIT record in the household in turn, incrementing UPERSON as shown.

Code Condition

For each BENUNIT record:

For benefit units which contain the head of household:

1 For the ADULT in the Benefit Unit, who is also the HEAD OF HOUSEHOLD

Then in the following priority:

--

- +1 for the other ADULT in the Benefit Unit
- +1 for each dependent in descending order of age

For benefit units which do not contain the head of household:

- 1 For the FIRST ADULT in the Benefit Unit

Then in the following priority:

- +1 for the second ADULT in the Benefit Unit
- +1 for each dependant in descending order of age

--

WATSEWRT/TUWATSEW

Purpose : To show the total amount of water and sewerage rates paid by each household in England and Wales.

The variable TUWATSEW is the same as WATSEWRT but excludes any adjustment for rent holidays. This variable should be used by ASD3A only.

Created : 2 February 1993
 Database Table : HOUSEHOL
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Period : Weekly
 Related Variables :
 Children :
 Parents :
 Issue Date : 17 April 2003
 Core variable/user : FRS (general)
 Amendments : VC - 4 May 1993 As per Linda Odwell's minute of 22 April 1993 to include sewerage for England and Wales but water only for Scotland.
 : VC - 6 August 1993 - Amended to include Scottish households as imputation
 : will ensure that all spaces are filled.
 : BS - 1 August 1995. Amended to take into account changes to V31.
 : JS - 18 January 1996: amended to take into account routing of renter block: households with an HB statement are sent down a different route than those without one or not on HB
 : JS - 21 February 1996: to allow skipped values where variables have been imputed and to make v30 amendments explicit
 : JS - 8 March 1996: to simplify coding
 : VE - 22 May 1996 - Amended to reflect initial V32 changes - WATERINC SEWERINC replaced by WORSINC. New question SEWSUPP.
 : VE - 26 September 1996 - Final paragraph amended. WSINCAMT replaced by WATSEWRT (error in original form)
 : VE - 10 March 1997 - Amended to bring into line with routing
 : SG - 18 July 1997 - Amend check for whether water/sewerage included in rent
 : SG - 29 December 1997 - No initial V33 update
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 7 October 1999 - Security completed

AND

Deletion of WORSAMT, WORSINC, and STMT,
 replaced by WSINCAMT, WSINC, and RENTDOC

1 Definition

This variable is coded as

 --

WATSEWRT The total weekly amount of water and sewerage rates paid by each household in England and Wales

-1 Not applicable to this case (Scottish Cases)

-2 Unable to derive due to missing values.

Amounts paid for water and sewerage are asked depending on whether paid separately or as part of rent. For renters, if they are on housing benefit and have a statement, amounts are held in WSINCAMT. If they are not on HB but pay charges as part of rent, the amount is held in WSINCAMT. For renters who pay charges separately or owner occupiers, amounts are held in WATAMT, SEWAMT and WSEWAMT.

Previous versions of this program did checks on whether the household was connected to water and/or sewerage mains and calculated each component separately. Since the questions are only asked if this is the case, and there is no requirement to separate water and sewerage payments, these checks have been dropped.

The first step is to set WATSEWRT to zero. Since these questions contain missing values which have been imputed, some of the questions will have been skipped. Instead of setting the DV to -2, no change should be made.

For renters who pay water and sewerage as part of rent, if questions have been skipped WATSEWRT will be imputed using WATER. For households who did not know what they were connected to or whether they paid water and sewerage (WATERPAY or SEWERPAY missing) WATSEWRT will be imputed to zero by default. There are around 1000 zero cases on the 1994/95 database.

2 FRS Specification

For each household, from HOUSEHOL table

Code Condition

WATSEWRT **Set WATSEWRT to zero**

If STDREGN=11 then set WATSEWRT to skipped and exit record

Cases where water and sewerage paid separately

From HOUSEHOL record

If (SEWSEP=1 or (WATERPAY=1 and SEWERPAY=2)) and # WATAMT exists
 WATSEWRT=WATSEWRT+WATAMT else don't change WATSEWRT

If (SEWSEP=1 or (SEWERPAY=1 and WATERPAY=2)) and # SEWAMT exists
 WATSEWRT=WATSEWRT+SEWAMT else don't change WATSEWRT

--

If **SEWSEP=2** and # WSEWAMT exists $WATSEWRT=WATSEWRT+WSEWAMT$ else don't change WATSEWRT

Cases where renting, water and sewerage paid as part of rent, and HB Statement

From the RENTER record

If RENTDOC =1 and WSINCAMT exists $WATSEWRT=WATSEWRT+WSINCAMT$ else don't change WATSEWRT

Cases where renting, water and sewerage paid as part or rent, and no HB statement (including those who didn't know if they had one) or not on HB

From the RENTER record

If WSINC=1 or 2 or 3 (water or sewerage or both included in rent and no HB statement)

If WSINCAMT exists $WATSEWRT=WATSEWRT+WSINCAMT$
Else if WSINCAMT= -1 (value of WORSINC has been imputed but WSINCAMT has been skipped) call WATER, $WATSEWRT=WATSEWRT+WATER$
Else don't change WATSEWRT

-1 Not applicable to this case: Scotland

-2 Not able to derive as any of the above variables are missing

 --

YOUNGCH

Purpose : To show the age of the youngest child in any benefit unit.
 Created : 19 January 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (general)
 Amendments : VC - 3 March 1993. To use MINR function to get age of youngest child
 : VC - 4 May 1993. To expand the definition to explain meaning of database
 : variable used.
 : VC - 17 February 1994 Amended to reflect version 30 changes
 : VE - 21 May 1996 - No initial Version 32 update needed
 : SG - 6 January 1998 - No initial V33 update needed
 : EP - 13 August 1998 - DEPCHLDB a DV for V34
 : JC - 17 September - Security completed, no other changes for V35
 Issue date : 17 April 2003

1 Definition

This variable is coded as

YOUNGCH The age of the youngest child in the benefit unit

-1 Not applicable to this case

-2 Unable to derive because of missing values.

This variable is derived from a comparison of the ages of all children within a benefit unit.

2 FRS Specification

For each benefit unit, get **derived** variable DEPCHILDB (number of dependent child in BU) from BENUNIT table,

CodeCondition

YOUNGCH From CHILD table,

If DEPCHILDB > 0, use MINR function to obtain smallest value of age.

-1 Not applicable to this case - DEPCHILDB = 0

--

-2 Unable to derive due to age variable missing

3 Results

Tabulation to show number of youngest children

4 Test Cases

DEPBAND

Purpose : Shows Household deprivation rank for each Local Authority
Created : 27 October 1998
Database Table : HOUSEHOL
Minimum Value : 1
Maximum Value : 310
Units : Integer
Validations :
Related Variables :
 Children :
 Parents :
Core variable/user : ASD3D
Amendments :

SEE – 28th August – Misleading Deprivation Band Indicator (DEPBAND) removed for non-English Local Authorities. See 2002-03 Changes documentation for full details.

Issue date : 27 October 1998

1 Definition

This is the deprivation rank for the household based on its Local Authority Code. The rank is worked out from Local Authority district scores and values on 12 indicators, and the lower the rank the more deprived the Local Authority.

2 FRS Specification

The amount is read from a lookup table held in CONST33.XLS.

FAMTHBAI

Variable	FAMTHBAI
Purpose:	This is the family type used for HBAI purposes for each benefit unit
Database Table:	Benunit
Variable Type:	Categorical
SAS Code Link	famthbai.sas

Definition

- 1 Pensioner Couple**
- 2 Pensioner Single**
- 3 Couple with children**
- 4 Couple without children**
- 5 Lone parent**
- 6 Single without children**

.D Unable to derive due to missing values

Summary

FAMTHBAI is derived from several variables on the ADULT table (sex and age) and the BENUNIT table (adultb and depchldb) of the FRS database. It uses number of adults and number of children in BU and categorises on a hierarchical bases i.e. If have children and head of BU is over state pension age then classified as a pensioner and not a couple/single with children.

1 Pensioner Couple

A respondent will be classified under this heading if:

- There are two adults in BU (ADULTB = 2) and
- The Male is over pension age (SEX = 1 and AGE => 65) or
- The Female is over pension age ((SEX = 2 and AGE => 60)

2 Pensioner Single

A respondent will be classified under this heading if:

- There is one adult in BU (ADULTB = 1) and
- The adult is over pension age ((SEX = 1 and AGE >= 65) or (SEX = 2 and AGE >= 60))

3 Couple with Children

A respondent will be classified under this heading if:

- There are two adults in BU (ADULTB = 2) and
- There is at least one dependant child in the BU (DEPCHLDB > 0)

INDINC, INEARN, ININV, INRPINC, INPENINC, INDISBEN, INOTHBEN, INRINC

Purpose : To indicate the amount of gross income received by an adult for use in the FRS publication (based on GROSSINC).

Created : 31 January 1996

Database Table : ADULT

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : **GROSSPAY (related to INEARN) – refer to note in main text**

Children :

Parents :

Core variable/user : FRS (general)

Amendments : JS - 21 March to correct INRINC for luncheon vouchers (only asked about first job); INPENINC to allow for skipped tax (where penpay has been imputed or TU pension) and INEARN for questions which are only asked of the first job

: JS - 12 April 1996 - to add in income for contributions to rents/mortgages from those outside the household

: VE - 17 April 1996 - to correct ININV to add in tax for those accounts where interest is received after tax

: VE - 23 May 1996 - Initial amendments for V32 - Still need to look at accounts and benefits to check definitions are the same as for V31

: VE - 16 July 1996 - To amend typo in adjustments for SSP and SMP and to include SSP and SMP in benefits block for cases where the benefit is recorded in the benefits table but not in the job section

: VE - 19 August 1996 - To update ACCOUNTS table definitions in ININV for V32

: VE - 20 August 1996 - to amend benefit definitions for V32

: JS - 12 November 1996 - to amend SSP/SMP adjustment

: VE - 20 November 1996 - to amend INRINC

: VE - 20 November 1996 - to replace INDTAX with ACCTAX

: VE - 28 November 1996 - to amend so that UBONAMT is only subtracted from INEARN when UBONINC=1

: VE - 11 December 1996 - to amend the OJAMT step to take into account the fact that for V32 OJPD=12 or 13 have been weeklyised

: VE - 23 January 1997 - to adjust HBENAMT and SCHBAMT for any rent free holidays as in HHRENT

: VE - 31 January 1997 - to make adjustments to bring into line with HBAI

: SG - 12 November 1997 - V33 changes, also implications of NINDINC (net income version)

: SG - 3 December 1997 - include maintenance even if paid via DSS/CSA

: SG - 30 December 1997 - no period codes for odd jobs

: SG - 11 February 1998 - change in PAYSLIP for V33

: SG - 17 March 1998 - use 20% tax rate

: SG - 23 April 1998 - new self employment DV

: SG - 1 June 1998 - add extended HB/CTB where appropriate

: SG - 8 June 1998 - Social Fund crisis loan should not be included

: EP - 30 November 1998 - V34 - change head of household from person =1 HOH = 1

to : EP - 8 December 1998 - remove references to UB, and replace references to benefit 20 with references to benefits 65 and 66

: EP - 9 December, 1998 - add in Benefit 61 for V34

: EP – 15 December 1998 – include income for students from their parents in INRINC
 : EP – 11 May 1999 – Add in explanation of difference between INEARNs and GROSSPAY
 : SB - 30 November 1999 – Security completed, change to charity tax relief.
 : SB - 5 January 2000 - moved baby-sitting and mail order from ADULT table to ODDJOB table
 : SB - 5 January 2000 - moved ROYAL1-3 down one and added in a new ROYAL1 category
 : SB - 5 January 2000 - remove child income from rent
 : SB - 5 January 2000 - remove deleted benefits
 : SB - 5 January 2000 – SSP/SMP no longer asked on benefits table table
 : AW – 15 February 2000 – checking of dv – changes made to add inearn code bottom of innirben code so that SMP/SSP adjustment is correctly calculated
 :
 Issued : 12 April, 2005

Please Note: For survey years prior to 2003-04 the derivation of INRPINC ,the total amount of benefit income received from state retirement pension plus any income support, MIG or pension credit where the person is over state retirement age, incorrectly assigns income received from state retirement pension or any income support, MIG or pension credit where the person is over state retirement age. The effect of this is minimal; however users of this information would be advised to not use the derived variable provided with the FRS datasets prior to 2003-04.. If you require further advice please contact the FRS team at DWP.

1 Definition

This variable is coded as

INDINC The total amount of gross income received by an adult from all sources.

0 Not applicable as adult does not have any gross income.

-2 Unable to derive due to missing values.

The total amount of gross income is derived from numerous variables from the ADULT, JOB, BENEFITS, ODDJOB and PENSIONS records which when added together form the person's total gross income. It includes gross normal earnings (cf with GROSSPAY which is last pay), self-employed earnings, tax paid on pensions annuities, other income in the form of benefit income, income in kind, royalties, other allowances, income from trust funds and odd jobs etc. Income Support is included gross of any direct payments or social fund repayments. Income from boarders/lodgers has been excluded to avoid double counting at a household level and to simplify definitions (this will need to be discussed in the publication).

Additional amounts for direct expenses from absent partners, regular contributions from household members, education grants and other deductions from pension income have also been included.

Private benefit schemes are included unless they are one off/lump sum payments (these are ignored).

Income from free school milk and meals and free welfare milk are also included (allocated to the head of benefit unit).

Amounts of maintenance are also checked to make sure they are usual.

If the period code for the benefit is 90 or 95 or 97 (lump-sum/one-off or other period) the record must be set to unable to derive as it has not been possible to convert the amount of benefit into a weekly amount during the database conversion process. Therefore, for example, if PAYPD = 90 or 95 or 97 INDINC is set to -2. Additionally, coding has been changed to allow skipped values for period codes: this will occur where the (weekly) amount has been imputed. It has been decided to leave period codes as "skipped" in these cases since this may help flag imputation.

The addition of HB is adjusted for any rent-free holidays as in the derivation of HHRENT. This adjustment is done for conventional households and for shared households, but not for boarders and lodgers. The reasoning behind this is that boarders and lodgers would probably pay rent every week even if the household had some rent-free weeks.

Note: additional categories are now included for personal pensions

The DV spec is also used to set up components of gross income which are accumulated to obtain BU and HH level variables. These are:

INEARNS	earned income (see note below)
SEINCAM2	self employment income (identical to INCSEO2 but set to zero if not applicable)
ININV	investment income
INRPINC	retirement pension plus any income support
INPENINC	other pensions
INDISBEN	disability benefits
INOTHBEN	other benefits
INRINC	remaining income

For pensioners, any retirement pension is taken together with IS to avoid any issues of misreporting. Analyses of pensioner income by ASD3 also take these two together.

Disability benefits comprise war disablement benefit, DWA, SDA, AA, DLA (mob and care); IIDB and IB.

Although similar to the derived variable GROSSPAY, INEARNNS sometimes differs in value from it. This is because GROSSPAY looks at the *last pay* that the respondent received. In cases where this was not the usual pay, GROSSPAY does not give the best representation of an individual's earnings – this can be obtained from INEARNNS, as this variable includes a check on whether or not the last pay was the usual pay (using the variable UGROSS). GROSSPAY also does not include any bonuses received, which are included in INEARNNS (the variables BONAMT1-6, which are weeklyised when included), and also does not deduct other allowance/refund variables (e.g. HHA1-3, MILEAMT, MOTAMT). If any user is unclear which of these two variables to use, then refer to ASD3E.

2 FRS Specification

For each ADULT

Code Condition

INDINC Gross earnings: INEARNNS

From ADULT record, set INEARNNS to zero

If WORKING = 1 or JOBAWAY = 1 (worked in last 7 days or away from work in the last 7 days) process each JOB record for that person and

If ABSWHY ne 3 (To bring in line with HBAI)

If EMPEE = 1

(Rewording of following section)

(Coding includes a check to see if UGROSS exists, where it doesn't exist, gross pay is calculated dependent on whether a payslip has been consulted, where it does (which can only be JOBTYP=1, PAYUSL=no), UGROSS is used)

If UGROSS does not exist or is missing:

 and If PAYSLIP = 1 or 2 (consulted)

 If GRWAGE exists and PAYPD equals -1 or 1 to 10 or 13 or 26 or 52, add it into INEARNNS

 If it is missing set INDINC and INEARNNS to -2

 else if PAYSLIP = 3 or 4 (or PAYSLIP=1 or 2 and GRSWAGE=-1) and PAYPD equals -1 or 1 to 10 or 13 or 26 or 52 ,

 If PAYAMT exists, add it into INEARNNS.

 If it is missing set INDINC and INEARNNS to -2

 If PAYE exists, add it into INEARNNS.

 If it is missing do not change INEARNNS

 If NATINS exists, add it into INEARNNS.

 If it is missing do not change INEARNNS

 If CHRAMT exists, add it into INEARNNS.

 If it is missing do not change INEARNNS

 If OTHDED1 = 1 add DEDUC1 to INEARNNS

 If OTHDED2 = 1 add DEDUC2 to INEARNNS

 If OTHDED3 = 1 add DEDUC3 to INEARNNS

 If OTHDED4 = 1 add DEDUC4 to INEARNNS

 If OTHDED5 = 1 add DEDUC5 to INEARNNS

 If OTHDED6 = 1 add DEDUC6 to INEARNNS

 If OTHDED7 = 1 add DEDUC7 to INEARNNS

 If OTHDED* = 1 add DEDOTH to INEARNNS

Else if UGROSS exists

 If UGROSS exists add UGROSS to INEARNNS

Adjustments to gross earnings for HBAI consistency:

deduction of income tax refunds, mileage and motoring allowances, refunds for items of household expenditure

if INEARNNS ne-2 (other conditions relating to PAYAMT and PAYPD will have been met by this point if INEARNNS has not been set to -2)

and UGROSS does not exist (ie all jobtypes except jobtype=1 where pay not usual)

and JOBTYP=1 and TAXAMT exists $INEARNNS=INEARNNS-TAXAMT$
(TAXAMT only asked for first job)

and MILEAMT exists $INEARNNS=INEARNNS-MILEAMT$

and MOTAMT exists $INEARNNS=INEARNNS-MOTAMT$

and HHA1 exists $INEARNNS=INEARNNS-HHA1$

and HHA2 exists $INEARNNS=INEARNNS-HHA2$

and HHA3 exists $INEARNNS=INEARNNS-HHA3$

Adjustments to gross earnings for HBAI consistency:

addition of bonuses received in last 12 months divided by 52

for up to 6 bonuses $i=1-6$:

If BONAMT(i) exists and BONTAX(i) (before tax)=1
 $INEARNNS=INEARNNS+((BONAMT(i)/52))$

If BONAMT(i) exists and BONTAX(i)=2 or -1 (after tax or skipped where BONAMT imputed)
 $INEARNNS=INEARNNS+((BONAMT(i)/52)/0.77)$

{questions about bonuses are asked regardless of whether pay usual or not, however, if UGROSS has been taken, have to make sure that bonus is not double counted. If bonus is after tax then need to make adjustment to add tax back in because looking at gross pay, before tax*0.77=net tax therefore net tax/0.77=before tax.}

If UGROSS exists
and UBONINC=1 and UBONAMT exists
 $INEARNNS=INEARNNS-(UBONAMT/0.77)$

(UBONAMT is on a net basis, assume that if included in net pay ie UBONINC = yes. then it will also have been in usual gross pay; UBONAMT is only asked if UBONINC = yes.

Adjustment for possible receipt of SSP or SMP

NOTE : The section of code which actually works out the adjustment is contained within innirben code. The inearn code just takes of the adjustment which has been earlier on.

calculated

Set ADJUST (temporary variable) and INOTHBEN to zero

For JOBTYP=1 only (SSP/SMP questions only asked once)

If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYS LIP = 1 or 2
Calculate ADJUST = SSPAMT

If JOBAWAY = 1 and ABSWHY = 6 and (SSPSMP = 1 or 3) and PAYS LIP = 1 or 2
Calculate ADJUST = ADJUST + SMPAMT

If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYS LIP= 3 and
SSPRATE=1 or 2 Calculate ADJUST =52.50

If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYS LIP= 3 and
SMPRATE=1 or 2 Calculate ADJUST=ADJUST+52.50

If ADJUST>INEARNS, then leave INEARN. This is in line with HBAI assumptions that where SSP is bigger than the pay it's supposedly in then the respondent has made a mistake and has actually quoted their pay after SSP/SMP.

Else INEARN=INEARN-ADJUST.

Add ADJUST to INOTHBEN

This adjustment is to check whether any SMP/SSP is included in gross earnings and to switch it to benefit income. Previous specifications of GROSSINC reduced earnings in respect of SSP/SMP where a payslip was consulted but did not include amounts with benefits because the benefits table did not record them They therefore undercounted income by the rates of SSP/SMP in appropriate cases. Within the pay block, if a payslip is consulted (payslip=1 or 2), respondents are asked for the amount included (SSPAMT/SMPAMT) otherwise, they are asked what rate was in payment (SSPRATE/SMPRATE). These are standard rates –one standard rate for SSP and two rates for SMP (but higher SMP is used as this is in line with HBAI). These standard rates are input into DVMETA.XLS which contains constants – so that they can be easily updated year on year Earned income equals amount calculated at INEARN less the estimated SMP/SSP (the adjustment). ADJUST is added to INOTHBEN.

Other sources of income: INRINC

Set INRINC to zero

Income as occasional work, professional advice

From ODDJOB record, if ODDTYPE=3 and OJNOW=1 then add OJAMT to INRINC

Income as a baby-sitter

From ODDJOB record, if ODDTYPE=1 and OJNOW=1 then add OJAMT to INRINC
~~From ADULT record, if BABY1 = 1 and BABNOW=1 (doing work as a baby sitter currently) add BABPAY into INRINC~~

Income as a mail order agent

From ODDJOB record, if ODDTYPE=2 and OJNOW=1 then add OJAMT to INRINC
~~From ADULT record, if BABY2 = 1 and BABNOW=1 add BABPAY into INRINC~~

Allowance from absent spouse

From ADULT record, if ABSPAR = 1 and APPD equals -1 or 1 to 10 or 13 or 26 or 52, add APAMT to INRINC.

If APDIR=1 and APDPD equals -1 or 1 to 10 or 13 or 26 or 52 add APDAMT to INRINC

Allowances from spouse in forces, friends other relatives etc

From ADULT record, if ALLOW1 = 1 and ALLPD1 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY1 to INRINC.

Allowance from an organisation

From ADULT record, if ALLOW2 = 1 and ALLPD2 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY2 to INRINC.

Allowance from a Local Authority for a foster child

From ADULT record, if ALLOW3 = 1 and ALLPD3 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY3 to INRINC.

Allowance from a Local Authority for an adopted child.

From ADULT record, if ALLOW4 = 1 and ALLPD4 equals -1 or 1 to 10 or 13 or 26 or 52, add ALLPAY4 to INRINC.

Income in kind

From JOB record, if JOBTYP=1 and LVAMT exists add amount in LVAMT into INRINC (luncheon vouchers) else do not change INRINC (*lvamt only asked if lunchv=1 and lv7dy=1, and only asked of first job*)

Rent from any property before tax

From ADULT record, if ROYAL1 = 1 add ROYYR1 into INRINC

Royalties

From ADULT record, if ROYAL2 = 1 add ROYYR2 into INRINC

Income as a sleeping partner

From ADULT record, if ROYAL3 = 1 add ROYYR3 into INRINC.

Pension from an overseas Government

From ADULT record, if ROYAL4 = 1 add ROYYR4 into INRINC.

Maintenance

From ADULT record, if MNTREC = 1 and MNTPD1 equals -1 or 1 to 10 or 13 or 26 or 52, add MNTAMT1 into INRINC.

From ADULT record, if MNTREC = 1 and MNTPD2 equals -1 or 1 to 10 or 13 or 26 or 52, add MNTAMT2 into INRINC.

Odd jobs

From ADULT record if ~~ODDJOB=1~~
for all occurrences of OJAMT in ODDJOB record
(note that OJAMT converted to weekly amount in conversion process)

Income from property

If PROPARENT exists add PROPARENT to INRINC.

Income from sub-tenants

If SUBLET = 1, add SUBRENT into INRINC for HOH = 1 (head of household).

Income from those outside the household paying towards rents/mortgages

For rented property, all contributions are included excluding contributions from DSS, consistent with HBAI. The income is assumed to be for the head of household
From RENTCONT record if ACCNONHH =1 and ACCPAY <> 1 and ACCAMT exists and RENTHOL=2 and ACCPD=-1 or 1-10 then INRINC=INRINC+ACCAMT for HOH = 1 only (head of household). If ACCNONHH=1 and ACCPAY <> 1 and ACCAMT exists and ACCPD=-1 or 1-11 and WEEKHOL exists multiply ACCAMT by the number of weeks rent actually paid (52-WEEKHOL) and divide the result by 52. Add this to INRINC for **HOH = 1** only

For those buying their house with a mortgage, contributions from outside are included for all cases except the DSS. This is to avoid double counting of direct payments which are already included as part of benefit income. Amounts are included for all types of mortgage (having been explicitly added back for endowment mortgages where appropriate and implicit in repayment mortgages calculation)

From MORTCONT record if OUTSMORT=1 and OUTSPAY=2,3,4 or 5 (ie not equal to 1) OUTSAMT exists and OUTSPD=-1 or 1-11 then INRINC=INRINC+OUTSAMT for HOH = 1 only.

Income from education grants

If TOTGRANT exists add TOTGRANT/52 to INRINC

Income from free welfare milk, free school meals and free school milk

(these are assumed to be income of the head of benefit unit - largely for convenience, individual amounts are not held on the data base)

If from BENUNIT record FWMLKBU exists add FWMLKBU to INRINC *for head of benefit unit only*

If from BENUNIT record FSMBU exists add FSMBU to INRINC *for head of benefit unit only*

If from BENUNIT record FSMLKBU exists add FSMLKBU to INRINC *for head of benefit unit only*

Income from student loans

If from ADULT record TUBORR exists add TUBORR/52 to INRINC

Interest/income from savings accounts or investments: ININV

set ININV to zero

From ACCOUNTS record,

if ACCOUNT = 1 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 1 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 2 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 2 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 3 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 3 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 4, add amount in ACCINT to ININV
 if ACCOUNT = 5 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 5 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 6 and ACCTAX=1, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 6 and ACCTAX<>1, add amount in ACCINT to ININV
 if ACCOUNT = 7, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 8, add (5/4 x amount in ACCINT) to ININV
 if ACCOUNT = 9, add amount in ACCINT to ININV

Note: Accounts 7 (Unit Trusts) and 8 (Stocks & Shares) are assumed net of tax, so we do need to add the tax back in. There is no ACCTAX check for these accounts.

Income from parents to students

If from ADULT record PareAmt exists,

add PareAmt / 52 to INRINC if ParePd in (95,97)
else add PareAmt to INRINC

Personal pensions: INPENINC

set INPENINC to zero

Occupational pensions

From PENSIONS record, if PENTYPE = 1 (occupational pension) and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC). If PENOTH exists and equal to 1 and POINC exists and =2 (other deductions from pension not included in figure at PENPAY) and POAMT exists add POAMT to INPENINC otherwise, do not change INPENINC.

(this is the same approach as taken in OCCPEN to deal with skipped values)

Widow's employee pension

From PENSIONS record, if PENTYPE = 2 (widow's employee pension) and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=INPENINC+PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC). If PENOTH exists and equal to 1 and POINC exists and =2 (other deductions from pension not included in figure at PENPAY) and POAMT exists add POAMT to INPENINC otherwise, do not change INPENINC.

Personal pension

From PENSIONS record, if PENTYPE = 3 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=INPENINC+PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC)

Trade union Friendly society pensions

From PENSIONS record, if PENTYPE = 4 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

get amount from PENPAY and add to INPENINC If PTINC = 2 add PTAMT then add result into INPENINC

Annuity pension

From PENSIONS record, if PENTYPE = 5 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

INPENINC=INPENINC+PENPAY. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC)

Trust/covenant

From PENSIONS record, if PENTYPE = 6 and PENPD equals -1 or 1 to 10 or 13 or 26 or 52,

get amount from PENPAY and add to INPENINC. If PTINC exists and = 2 and PTAMT exists add PTAMT then add result into INPENINC (otherwise, do not change INPENINC)

Income from benefits: INRPINC, INDISBEN, INOTHBEN (plus bits of INRINC held in BENEFITS record)

set INRPINC, INDISBEN to zero

Housing Benefit

if BENUNIT = 1 and PERSON = 1: From RENTER record, if RENTHOL = 1 and WEEKHOL exists and HBENEFIT = 1 and HBENPD equals -1 or 1 to 11 or 13 or 26 or 52, multiply HBENAMT by number of weeks rent actually paid (52 - WEEKHOL) and divide the result by 52. Add to INOTHBEN. If RENTHOL ne 1 and HBENEFIT = 1 and HBENPD equals -1 or 1 to 10 or 13 or 26 or 52, add HBENAMT to INOTHBEN.

From HOUSEHOL record, (if HHSTAT = 2 or CVPAY>0 (boarders/lodgers paying rent)) then from ADULT record if HBOTHBU=1 (individual qualifies for HB) and CVPD equals -1 or 1 to 10 or 13 or 26 or 52, then add HBOTHAMT to INOTHBEN.

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to 10 or 13 or 26 or 52 add CTREBAMT into INOTHBEN only for PERSON = 1.

Income from benefits

From BENEFITS record, if BENPD equals -1 or 1 to 10 or 13 or 26 or 52, and

If BENEFIT = 1 add BENAMT to INDISBEN (DLA Care)

If BENEFIT = 2 add BENAMT to INDISBEN (DLA Mob)

If BENEFIT = 3 add BENAMT to INOTHBEN (CHB)

If BENEFIT = 5 add BENAMT to INRPINC (RP)

If BENEFIT = 6 add BENAMT to INOTHBEN (Widows Pension)

If BENEFIT = 7 add BENAMT to INOTHBEN (Widowed Mothers allowance)

If BENEFIT = 8 add BENAMT to INDISBEN (War Disablement Pension)

If BENEFIT = 9 add BENAMT to INOTHBEN (War Widows Pension)

If BENEFIT = 10 add BENAMT to INDISBEN (SDA)

If BENEFIT = 11 add BENAMT to INDISBEN (DWA)

If BENEFIT = 12 add BENAMT to INDISBEN (AA)

If BENEFIT = 13 add BENAMT to INOTHBEN (Invalid Care Allowance)

If BENEFIT = ~~13~~ 14 add BENAMT to INOTHBEN JSA

If BENEFIT = 15 add BENAMT to INDISBEN (Industrial Injuries)

~~If BENEFIT = 16 add BENAMT to INOTHBEN (SSP)~~

If BENEFIT = 17 add BENAMT to INDISBEN (Incapacity Benefit)

If BENEFIT = 18 add BENAMT to INOTHBEN (FC)

If BENEFIT = 60 add BENAMT / 52 to INOTHBEN (widow's payment)

If BENEFIT = 19 and (SEX=1 and AGE>=65) or (SEX=2 and AGE>=60) add BENAMT to INRPINC (IS)
 else add BENAMT to INOTHBEN

If BENEFIT = 65 and VAR2=2
 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)) add BENAMT to INRPINC
 else if BENEFIT= 65 and VAR2=2 add BENAMT to INOTHBEN
(adjustment for IS quoted after any direct payments: pensioners income must be included in INRPINC whilst others are in INOTHBEN)

If BENEFIT = 66 and VAR2=2 add BENAMT to INOTHBEN
 If BENEFIT = 21 add BENAMT to INOTHBEN (Maternity Allowance)

~~If BENEFIT = 23 add BENAMT to INOTHBEN (SMP)~~

~~If BENEFIT = 28 and HBINDBU in (0,1,2,3,5,6) and HOH = 1 add BENAMT / 4 to INOTHBEN (Extended HB or extended HB and CTB)~~

~~If BENEFIT = 29 and HBINDBU in (0,2,3,5) and HOH = 1 add BENAMT / 4 to INOTHBEN (Extended CTB)~~

If BENEFIT = 78,80 and HBINDBU = 0←3,5,6 and UPERSON=1 add BENAMT / 4

If BENEFIT = 79 and HBINDBU = 0,2,3,5 and HOH =1 add BENAMT / 4

If BENEFIT = 30 add BENAMT to INOTHBEN (Any other DSS benefits)

If BENEFIT = 37 add BENAMT to INOTHBEN (Guardians Allowance)

If BENEFIT = 69 and var2=1 subtract BENAMT from INOTHBEN (social fund repayment - IS)

If BENEFIT = 70 subtract BENAMT from INOTHBEN (social fund repayment – JSA)
 (for v34 the question Sflnc, “was the amount of IS/JSA you said earlier before or after taking off your Social Fund Loan repayments?” was only asked if they didn't know or

refused to say how much they were repaying. Anyone not asked this question is assumed to have quoted an amount that was after any repayments have been taken off)

If BENEFIT = 22 and BENAMT/26 to INOTHBEN (Maternity Grant)

If BENEFIT = 24 and BENAMT/26 to INOTHBEN (Funeral Grant)

If BENEFIT = 25 and BENAMT/26 to INOTHBEN (Comm Care Grant)

If BENEFIT = 41 and BENAMT/26 to INOTHBEN (FC Lump Sum)

If BENEFIT = 26 and BENAMT/26 to INOTHBEN (Back to work bonus)

Note: following benefits are included in INRINC

If BENEFIT = 31 and PRES = 1 and BENPD=-1 or 1-10 or 13 or 26 or 52 add BENAMT to INRINC (Trade Union sick) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 32 and PRES = 1 and BENPD=-1 or 1-10 or 13 or 26 or 52 add BENAMT to INRINC (Friendly sick) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 33 and PRES = 1 add BENAMT to INRINC (Private sick) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT =34 and PRES = 1 add BENAMT to INRINC (Accident) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 35 and PRES = 1 add BENAMT to INRINC (Hospital savings) else if PRES=1 and BENPD=90, 95 or 97 do not change INRINC

If BENEFIT = 36 add BENAMT to (Training) INRINC

If BENEFIT = **61** and PRES = 1 and BENPD=-1 or 1-11 or **13 or 26 or 52** add BENAMT to INRINC (**Unemployment/Redundancy Insurance**) else if PRES=1 and BENPD=**90, 95 or 97** do not change INRINC

INDINC will then be calculated as follows for each ADULT -

$$\text{INEARNS} + \text{SEINCAM2} + \text{INSEINC} + \text{ININV} + \text{INRPINC} + \text{INPENINC} + \text{INDISBEN} + \text{INOTHBEN} + \text{INRINC}$$

-2 If any of above variables are missing or if a period code is 90 or 95 or 97 (skipped is OK) also applied to individual component variables ININV, INRPINC, INPENINC, INDISBEN, INOTHBEN and INRINC which are calculated in this spec (missing INEARNs components are already documented above).

PTENTYPE

Purpose	: To indicate the number of households in any specific tenure type for use in the FRS publication.
Created	: 29 January 1996 (although coding existed for 1993/94 publication)
Database Table	: HOUSEHOL
Minimum Value	: 1
Maximum Value	: 5
Units	: Integer
Validations	:
Related Variables	: TENTYPE
Children	:
Parents	:
Core variable/user	: FRS (publication)
Issue date	: 12 April 2005
Amendments	: JS - 21 February 1996 - to allow for missing values where variables have been imputed
	: JS - 15 March 1996 to bring in line with SEH categories (at request of ASD3F)
	: VE - 19 August 1996 - To amend so that part owner/part renter cases are now included with cases buying with a mortgage
	: VE - 10 October 1996 - To bring in line with Harmonisation of Outputs categories
	: SG - 31 October 1997 - V33 mods - changes to TENTYPE coding
	: EP - 13 August 1998 - No initial V34 update needed
	: SB - 21 October 1999 - Security completed, no other changes needed for V35

Please Note: For survey years prior to 2003-04 the derivation of unfurnished and furnished rented property assigned 'partially unfurnished' as 'furnished'. This allocation is out of line with the National Statistics harmonised guidance and so users are advised to not use the derived variable provided with FRS datasets prior to 2003-04. If you require further advice please contact the FRS team at DWP.

1 Definition

PTENTYPE is derived using variables TENURE (household record), FURNISH and LANDLORD (renter). Codes are effectively a re-code of the TENTYPE breakdown. (*All cases where TENTYPE=7 and LANDLORD=1, 2 or 3 will be edited to rebate cases.*) FURNISH is not asked for TENTYPE=8 so assumed to be rented furnished.

This variable is coded as

- 1 Rented from council
- 2 Rented from housing association
- 3 Rented privately unfurnished
- 4 Rented privately - furnished
- 5 Owned with mortgage (including part rent/part own)

6 Owned outright

-2 unable to derive due to missing values

2 FRS Specification

1 TENTYPE=1

2 TENTYPE=2

3 TENTYPE=3 or (TENTYPE=7 and FURNISH=3) or (TENTYPE=8 and FURNISH=3)

4 TENTYPE=4 or ((TENTYPE=8 or TENTYPE=7) and (FURNISH=1 or FURNISH=-2 or FURNISH=.A))

5 If TENTYPE=5

6 If TENTYPE=6

TENTYPE

Purpose	: To indicate the number of households in any specific tenure type.
Created	: 14 December 1992
Database Table	: HOUSEHOL
Minimum Value	: 1
Maximum Value	: 7
Units	: Integer
Validations	:
Related Variables	:
Children	:
Parents	:
Core variable/user	: FRS (publication)
Issue date	: 12 April 2005
Amendments	: VC - 28 January 1993
	: VC - 15 March 1993 Amendments to insert the part-own part-renter category into the appropriate renter category, to put tenure = 2 or 3 into the owner categories and to exclude tenure = 5 as these tenants are not eligible for Housing Benefit.
	: VC - 18 May 1993. Amended to re-insert tenure = 5 as a separate category and also to add another category for co-ownership schemes as both of these groups are not eligible for HB.
	: VC - 9 June 1993. Amended to insert extra category for shared ownership.
	: BS - 3 August 1995. Amended to include changes to V31 of the questionnaire. New var FURNISH asked if TENURE equals new codes 6, 7 or 8.
	:JS - to clarify tentype=3 category and to allow skipped values of LANDLORD, FURNISH and OWNHOW (in cases where LANDLORD or TENURE have been imputed)
	:JS - to note categories are actually slightly different from the name suggested!
	: VE - 22 May 1996 - No initial Version 32 update needed
	: VE - 10 June 1996 - To bring specification in line with Harmonisation of Survey Outputs: part owner/part renter cases are now included with cases buying with a mortgage and lettings which go with the job of someone in the household are now classified as rented privately.
	: SG - 30 October 1997 - V33 changes - values of Tenure, Landlord, Othway
	: EP - 13 August 1998 - V34 update, Tenancy agreement question (SHORT) split up into SHORT1 and SHORT2
	: JC - 20 September 1999 - Security completed, no other changes for V35

Please Note: For survey years prior to 2003-04 the derivation of unfurnished and furnished rented property assigned 'partially unfurnished' as 'furnished'. This allocation is out of line with the National Statistics harmonised guidance and so users are advised to not use the derived variable provided with FRS datasets prior to 2003-04. If you require further advice please contact the FRS team at DWP.

1 Definition

This variable is coded as

-
- | | |
|----|--|
| 1 | LA / New Town / Council rented |
| 2 | Housing Association / Co-Op / Trust rented |
| 3 | Other private rented unfurnished |
| 4 | Other private rented furnished |
| 5 | Owned with mortgage (Includes part rent/part own) |
| 6 | Owned outright |
| 7 | Rent-free |
| 8 | Squats |
| | |
| -1 | Not applicable to this case |
| -2 | Unable to derive variable |

Tenure type is derived from the variable tenure in the HOUSEHOL record which when combined with a second variable from either the OWNER (furnish) or the RENTER (landlord) records will indicate to which category of tenure that particular household belongs.

All tenants whose accommodation goes with the job of someone in the household are allocated to rented privately, even if landlord is a local authority or housing association or Trust, or if rent free.

FURNISH is only asked where landlord=6-8, coding for TENTYPE=3 needs to reflect this

2 FRS Specification

For each household, process HOUSEHOL record

Code Condition

Where TENURE = 4, process RENTER record for details of type of landlord

- | | |
|---|---|
| 1 | If TENURE = 4 and LANDLORD = 1 (council/New Town Corporation) and ACCJOB=2 |
| 2 | If TENURE = 4 and LANDLORD = 2 (Housing Association/Co-Op/Trust) and ACCJOB=2 |
| 3 | If TENURE = 4 and (LANDLORD= 3,4,5,6,or 7 and FURNISH = 2) or (ACCJOB=1 and FURNISH=2) (other unfurnished) |
| 4 | If TENURE = 4 and ((LANDLORD = 3,4,5,6,or 7 and (FURNISH = 1 or FURNISH= -1)) or LANDLORD=-1) or (ACCJOB=1 and FURNISH=1) (other furnished) |

Where TENURE = 1 or 3, process OWNER record for details of how the property is owned

- | | |
|---|--|
| 5 | If TENURE = 2 or 3 (owns/bought with mortgage or loan/part own-rent) |
| 6 | If TENURE = 1 (owns/outright) (<i>this closes down the route</i>) |

Where TENURE = 5

- | | |
|---|----------------------------|
| 7 | If TENURE = 5 (rent-free). |
|---|----------------------------|

Where TENURE = 6 (squatting).

8 If TENURE = 6 (squatting)

-1 Not applicable to this case

-2 Unable to derive - if any of the above variables are missing.

3 Results

Tabulation is required to show the numbers of households which fall into each category.

4 Test Cases

A Single person living in Local Authority housing.

B Single person living in Housing Association accommodation.

C Single person living in an unfurnished flat rented from a private landlord.

D Single person living in a furnished flat rented from a private landlord.

E Single person living in his/her own home with a mortgage outstanding on the property.

F Single person living in his/her own home with no outstanding mortgage.

G Single person living rent-free.

Repeat all of the above for couples.

TTWCOSTS

Purpose : To show weekly travel to work costs for each adult
 Created :
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : FRS (publication)
 Amendments : JS: for V31, to change calculation of weekly travel pass costs
 : JS 18/12/95 to take on changes made to V30 (additional methods of transport,
 coding of variable work costs cases)
 : JS - 21 February 1996 to allow for skipped values where variables have been
 imputed
 : JS - 15 March 1996, to stop weeklyising of pass within program (already
 weekly on the data base)
 : JS - 17 April 1996 to amend cases where TTWCOST or TTWPAY has been
 skipped to use calculated amount for car/bike (currently no change made)
 : VE - 5 June 1996 - to include new category of TTWCODE for V32
 : VE - 14 June 1996 - To amend for 1995-96 rates for cost per mile
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : VE - 29 October 1996 - to include TTWREC
 : VE - 9 December 1996 - To amend spec for change in data structure
 regarding TTWCODE1 and TTWCODE2
 : VE - 1 April 1997 - To tidy up spec to bring in line with SAS code for V32
 : SG - 8 January 1998 - V33 updates (CONST33.XLS location)
 : EP - 13 August 1998 - No initial V34 update needed
 : SB - 17 August 1999 - Change of source of mileage rates data. Motorbikes
 changed from 9p to 25.3p, See Dvmeta35
 : SB - 28 October 1999 - Security completed, TTWMOD1-6 replaced by
 TTWMOD
 : EP - 14 August 2001- DV reinstated for 2000/01 and inserted new mode of
 transport, TAXI
 :ND - 8 May 2003 - Two rates used for CARRATE.

 Issue date : 28 October 1999

1 Definition

This variable is coded as

TTWCOSTS Total weekly travel to work costs paid by adult
 0 Free travel

-1 Not applicable to this case - including adults with no usual place of work, or coded "other" form of transport (no information collected)

-2 Unable to derive due to missing values

Travel to work costs are based on the number of round trips per week (TTWFRQ). Adults are only asked about their main method of transport and costs are collected on all methods of transport except walk/cycle or "other" (TTWMOD). Public transport costs are calculated using the costs of bus passes or season tickets, recorded fare, contributions to drivers of shared cars, or on costs per mile of journeys in cars or on motorcycle. DSS rates for costs per mile have been used. **The rates (CARRATE and BIKERATE) are held on the table in p:\frs\shared\frs35\metadata\Dvmeta35.xls**

Questionnaire asks about total distance travelled to usual place of work. Where respondents use both car/motorcycle and train/bus/tube, only the main method is recorded. Therefore we assume most of the journey is by one method of transport.

Where PSSAMT has been imputed, PSSDATE1/2 will have been skipped. Amount held in PSSAMT is taken as weekly. Where TTWCODE has been imputed, TTWCOST is skipped: do not change TTWCOSTS in these cases

2 FRS Specification

TTWCOSTS From adult record, add up costs for each mode of transport TTWMOD 2 - 5

TTWMOD in (4,5,6,7) (bus/train/tube/light rail) or TTWMOD=1 (works bus/company transport)

If TTWPSS=1 (Yes) and PSSAMT exists TTWCOSTS = PSSAMT
Else if PSSDATE1/2 is skipped then cost of pass equals PSSAMT

Else if respondent does not have a season ticket, calculate cost of round trip and multiply by number of trips each week

If TTWPSS=2 (No)

If ONEWAY=1 then TTWCOSTS=FARE*2*TTWFRQ
else If ONEWAY=2 then TTWCOSTS=FARE*TTWFRQ

Else if travels in a car/van, first check whether car used in combination with bus/train/tube or works bus/company transport, if yes, assume average journey is 2 miles

If TTWMOD=1 and TTWPAY ≠ 3 (pays all/some of costs of taking car/van to work)
then TTWCOSTS=TTWCOSTS+(4*TTWFRQ*CARRATE)

Else if only car/van used (or used in combination with walking/bicycle) calculate total costs per mile per week (CARCOST - based on midpoints of TTWFAR categories),

- CARCOST is based on RATE1 or RATE2 depending on whether the annual distance is less than (Rate 1) or more then (Rate 2) a set mileage (6000 miles in 2002-03). Total miles calculated based on midpoints of TTWFAR categories.

If TTWMOD=1 (car or van) and TOTMILES < 6000
 and TTWFAR=3 then CARCOST=0.5*2*TTWFRQ*CARRATE
 and TTWFAR=4 then CARCOST=2*2*TTWFRQ*CARRATE
 and TTWFAR=5 then CARCOST=4*2*TTWFRQ*CARRATE
 and TTWFAR=6 then CARCOST=7.5*2*TTWFRQ*CARRATE
 and TTWFAR=7 then CARCOST=17.5*2*TTWFRQ*CARRATE
 and TTWFAR=8 then CARCOST=32.5*2*TTWFRQ*CARRATE

If TTWMOD=1 (car or van) and TOTMILES < 6000
 and TTWFAR=3 then CARCOST=0.5*2*TTWFRQ*CARRATE
 and TTWFAR=4 then CARCOST=2*2*TTWFRQ*CARRATE
 and TTWFAR=5 then CARCOST=4*2*TTWFRQ*CARRATE
 and TTWFAR=6 then CARCOST=7.5*2*TTWFRQ*CARRATE
 and TTWFAR=7 then CARCOST=17.5*2*TTWFRQ*CARRATE
 and TTWFAR=8 then CARCOST=32.5*2*TTWFRQ*CARRATE

Then calculate TTWCOSTS, taking into account any contributions

If TTWMOD=1
 If TTWPAY=1 (all) then TTWCOSTS=TTWCOSTS+CARCOST
 If TTWPAY=2 (some) and TTWCODE1=1 then TTWCOSTS=TTWCOSTS+TTWCOST
 If TTWPAY=2 and TTWCODE2=1
 then TTWCOSTS=TTWCOSTS+(CARCOST - TTWREC)
 (if CARCOST < TTWREC then set TTWCOSTS to zero)

else if TTWCOST and TTWREC have been skipped (where TTWCODE1/TTWCODE2 are set) or TTWPAY has been skipped or if TTWCODE1=3 and TTWCODE2=3 then
 TTWCOSTS=TTWCOSTS+CARCOST

Else if respondent drives a motorcycle, use similar approach.

If TTWMOD3=1 and (TTWMOD4=1 or TTWMOD5=1) and TTWPAY ≠ 3 (pays all/some of costs of taking motorcycle to work)
 then TTWCOSTS=TTWCOSTS+(4*TTWFRQ*BIKERATE)

Else if only motorbike used (or in combination with walking/bicycle) calculating variable BIKECOST

If TTWMOD=2 (motorcycle)
 and TTWFAR=3 then BIKECOST=0.5*2*TTWFRQ*BIKERATE
 and TTWFAR=4 then BIKECOST=2*2*TTWFRQ*BIKERATE
 and TTWFAR=5 then BIKECOST=4*2*TTWFRQ*BIKERATE
 and TTWFAR=6 then BIKECOST=7.5*2*TTWFRQ*BIKERATE
 and TTWFAR=7 then BIKECOST=17.5*2*TTWFRQ*BIKERATE
 and TTWFAR=8 then BIKECOST=32.5*2*TTWFRQ*BIKERATE

Then calculate TTWCOSTS, taking into account any contributions

If TTWMOD=2
 If TTWPAY=1 (all) then TTWCOSTS=TTWCOSTS+BIKECOST
 If TTWPAY=2 (some) and TTWCODE1=1, then TTWCOSTS=TTWCOSTS+TTWCOST

If TTWPAY=2 and TTWCODE2=1
then TTWCOSTS=TTWCOSTS+(BIKECOST - TTWREC)
(if BIKECOST < TTWREC then set TTWCOSTS to zero)

else if TTWCOST or TTWREC have been skipped (where TTWCODE1/TTWCODE2 are set) or
TTWPAY has been skipped or if TTWCODE1=3 or TTWCODE2=3 then
TTWCOSTS=TTWCOSTS+BIKECOST

- 0 TTWFAR=1 (work at home, live at work, no work journey)
TTWMOD=(3 or 8) (Walks or cycles to work)
If TTWPSS=1 and PSSAMT=0 (has pass and costs nothing)
or TTWPSS=2 and FARE=0 (does not have pass but fare costs nothing)
or cash received from passengers etc greater than calculated costs (costs therefore set to zero)
or ((TTWMOD=1 or 2) and TTWPAY=3) (uses car/motorcycle and pays no costs)

Note: Free travel cards/fares and contributions will have already been calculated: all categories
shown here for completeness

Note some cases exist where individual drives to a station but has a free travel pass. These
cases still count as zero travel costs.

- 1 If questions in n_Travel have been skipped (questions do not apply to this case)
or TTWFAR=2 (varies, no usual place of work)
or TTWMOD=(9, 10)(Taxi or other form of transport: no information collected)
- 2 Any variables missing