
 --

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 : **ISM** PSM (Bold indicates lead user)

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

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 = ~~56~~). 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 = 56 or
If AGE < 60 WORKING = 2, JOBAWAY = 2, LIKEWK = 2 and NOWANT = 56 |
| 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 carers
 Created : 29 April 1996
 Database Table : ADULT
 Minimum Value :
 Maximum Value :
 Units : integer
 Validations :
 Related Variables : CHCARE CHHOUR (variables for children doing care)
 Children :
 Parents :
 Core variable/user : ASD4A
 Amendments : VE - 21 May 1996 - No initial Version 32 update needed
 : VE - 14 May 1997 - To update for Version 33 by amending NEEDPER codes
 Issue date : 5 June 1996

1 Definition

The aim of these derived variables is to provide summary information on adult 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.

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

For each adult

CAREAB total number of adults looked after in the same benefit unit (maximum value of 1, 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 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

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 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 adults in the household looking after that person (x=1-9 - up to 9 adult household members).

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

CAREAB(x)=CAREAB(x)+1

HOURAB(x)=HOURAB(x)+HOUR(x)

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

CAREAH(x)=CAREAH(x)+1

HOURAH(x)=HOURAH(x)+HOUR(x)

If WHOLOO(x) is an adult looking after a **relative outside the household** (NEEDPER=~~21~~**21,22,23 or 24**) then

CARERE(x)=CARERE(x)+1

HOURRE(x)=HOURRE(x)+HOUR(x)

If WHOLOO(x) is an adult looking after a **friend/neighbour outside the household** (NEEDPER=~~22~~**25**) then

CAREFR(x)=CAREFR+1

HOURFR(x)=HOURFR(x)+HOUR(x)

If WHOLOO(x) is an adult looking after a **client of a voluntary organisation** (NEEDPER=~~23~~**26**) then

CARECL(x)=CARECL(x)+1

HOURCL(x)=HOURCL(x)+HOUR(x)

If WHOLOO(x) is an adult looking after **others outside the household** (NEEDPER=~~24~~**27**) then

CAREOT(x)=CAREOT(x)+1

HOUROT(x)=HOUROT(x)+HOUR(x)

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

CARECB(x)=CARECB(x)+1

HOURCB(x)=HOURCB(x)+HOUR(x)

If WHOLOO(x) is an adult 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*, OTBNCT*, OTACCT*, GILTSCT*, UNTRCT*, STSHCT*, NSBOCT*, SAYECT*, PRBOCT*, PEPST*

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

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
 Minimum Value : 0
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ASD4A
 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
 Issue date : 12 May, 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).

The 1994/95 publication looks at assets in terms of different types of account held in the ACCOUNTS table. The mapping of ACCOUNTS to these assets is shown in the attached table (file ~~acctable.xls~~ spreads.xls). To allow checking of the tables in SAS, variables need to be set up at an individual, benefit unit and household level which count the different types of account held. So, for example, if a person held adult NS capital bonds and index-linked certificates, NSBOCTI would equal 2. If their partner held some of the same, NSBOCTB would equal 4. If someone else in the household also held the same, NSBOCTH would equal 6.

--

Note the variables do not count the number of different accounts a person has of a certain type. This can only be gathered for those entering the assets block, questions in the accounts block relate to total interest from particular sources.

To check publication tables, an individual, benefit unit or household will hold a particular type of account if the relevant variable is greater than zero. For example, for National Savings Bonds if NSBOCTI (individual), NSBOCTB (benefit unit) or NSBOCTH (household) is greater than zero.

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, ~~OTBNCTI, OTACCTI~~, GILTSCTI, UNTRCTI, STSHCTI, NSBOCTI, SAYECTI, PRBOCTI, **PEPSCTI**, CURACTB, POACICTB, TESSCTB, OTBSCTB, ~~OTBNCTB, OTACCTB~~, GILTSCTB, UNTRCTB, STSHCTB, NSBOCTB, SAYECTB, PRBOCTB, **PEPSCTB**, CURACTH, POACICHT, TESSCTH, OTBSCTH, ~~OTBNCTH, OTACCTH~~, GILTSCTH, UNTRCTH, STSHCTH, NSBOCTH, SAYECTH, PRBOCTH, **PEPSCTH 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=~~24~~ **20** then POACCTI=POACCTI+1
 If ACCOUNT=~~25~~ **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=~~26~~ **22** then OTBSCTI=OTBSCTI+1
 If ACCOUNT=~~26~~ **23** then OTBSCTI=OTBSCTI+1

~~OTBNCTI If ACCOUNT=7 ~~6~~ then OTBNCTI=OTBNCTI+1
 If ACCOUNT=8 then OTBNCTI=OTBNCTI+1
 If ACCOUNT=~~27~~ **23** then OTBNCTI=OTBNCTI+1~~

~~OTACCTI If ACCOUNT=9 ~~7~~ then OTACCTI=OTACCTI+1
 If ACCOUNT=10 then OTACCTI=OTACCTI+1~~

GILTSCTI If ACCOUNT=~~44~~ **8 6** then GILTSCTI=GILTSCTI+1
 ~~If ACCOUNT=12 then GILTSCTI=GILTSCTI+1~~
 If ACCOUNT=~~28~~ **24** then GILTSCTI=GILTSCTI+1

UNTRCTI If ACCOUNT=~~43~~ **9 7** then UNTRCTI=UNTRCTI+1
 If ACCOUNT=~~29~~ **25** then UNTRCTI=UNTRCTI+1

STSHCTI If ACCOUNT=~~44~~ **10 8** then STSHCTI=STSHCTI+1
 ~~If ACCOUNT=15 then STSHCTI=STSHCTI+1~~
 If ACCOUNT=~~30~~ **26** then STSHCTI=STSHCTI+1

NSBOCTI If ACCOUNT=~~46~~ **10** then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~47~~ **11** then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~48~~ **12** then NSBOCTI=NSBOCTI+1

 --

If ACCOUNT=~~19~~ 13 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~22~~ 16 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~23~~ 17 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~31~~ 27 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~32~~ 28 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~33~~ 29 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~34~~ 30 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~37~~ 32 then NSBOCTI=NSBOCTI+1
 If ACCOUNT=~~38~~ 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

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

PRBOCTI If ACCOUNT=~~24~~ 15 then PRBOCTI=PRBOCTI+1
 If ACCOUNT=~~36~~ 31 then PRBOCTI=PRBOCTI+1

PEPSCTI If ACCOUNT=44 9 then PEPSTI=PEPSCTI+1

For each benefit unit

CURACTB equals all occurrences of CURACTI
 POACCTB equals all occurrences of POACCTI
 TESSCTB equals all occurrences of TESSCTI
 OTBSCTB equals all occurrences of OTBSCTI
~~OTBNCTB equals all occurrences of OTBNCTI~~
~~OTACCTB equals all occurrences of OTACCTI~~
 GILTSCTB equals all occurrences of GILTSCTI
 UNTRCTB equals all occurrences of UNTRCTI
 STSHCTB equals all occurrences of STSCCTI
 NSBOCTB equals all occurrences of NSBOCTI
 SAYECTB equals all occurrences of SAYECTI
 PRBOCTB equals all occurrences of PRBOCTI
PEPSCTB equals all occurrences of PEPSTI

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
~~OTBNCTH equals all occurrences of OTBNCTB~~
~~OTACCTH equals all occurrences of OTACCTB~~
 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
PEPSCTH equals all occurrences of PEPSCTB

AUTHTYPE

Purpose : To indicate in which type of authority the household is situated.
 Created : 17 March 1993
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 6
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : HBM
 Issue date : 12 May 2003
 Amendments : VC - 22 April 1993 To make definition clearer.
 : VC - 18 May 1993 To amend categories as initial information had shown :
 metropolitan districts as non-metropolitan districts and vice versa.
 : VC - 15 June 1993 Wolverhampton moved into Metropolitan category
 : previously shown as Non-metropolitan.
 : VC - 9 February 1994 No version 30 change needed
 : JS - 19 January 1996: coding for Met and Non-Met reversed to maintain consistency
 with HBMIS data; New Towns coding removed (no cases included anyway)
 : VE - 4 June 1996 - No initial amendments needed for V32
 : SG - 7 January 1998 - V33 amendments - LAC code changes

1 Definition

This variable is coded as

- 1 Non-metropolitan England
- 2 Metropolitan England
- 3 London
- 4 Wales
- 5 Scotland

- 1 Not applicable to this case
- 2 Unable to derive as variable LAC is missing

The authority type is derived from the Local Authority Code in the variable LAC which is a unique code only found in FRS. Each household will then be allocated an authority type according to the list provided by Ms Odwell. Authtype is a new variable produced only by FRS for the HBM.

NB - No New Towns are currently included in the specification as there is a problem with FRS codes for them.

Between V32 and V33 the Local Authority Codes changed (resulting in fewer LACs overall and new reference numbers for those remaining). Because the entire lookup has changed the V32 list has been removed below (rather than struck out) and replaced with the V33 version.

2 FRS Specification

Code each authority type according to the attached list - **NB** the type of authority code is the fourth column in the list.

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

-2 Unable to derive as the variable LAC is missing.

NB - These LA codes are unique to FRS.

3 Results

Tabulation to show the number of households in each authority type.

4 Test Cases

None

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

STND REGION	LAC	AUTHORITY NAME	TYPE OF AUTHORITY
1	145	HARTLEPOOL	1
1	146	REDCAR AND CLEVELAND	1
1	237	MIDDLESBROUGH	1
1	238	STOCTON ON TEES	1
1	239	ALLDERDALE	1
1	240	BARROW IN FURNESS	1
1	241	CARLISLE	1
1	242	COPELAND	1
1	243	EDEN	1
1	244	SOUTH LAKELAND	1
1	245	CHESTER LE STREET	1
1	246	DARLINGTON	1
1	337	DERWENTSIDE	1
1	338	DURHAM	1
1	339	EASINGTON	1
1	340	SEDFIELD	1
1	341	TEESDALE	1
1	342	WEAR VALLEY	1
1	343	ALNWICK	1
1	344	BERWICK-U-TWEED	1
1	345	BLYTH VALLEY	1
1	346	CASTLE MORPETH	1
1	437	TYNEDAILE	1
1	438	WANSBECK	1
2	444	KINGSTON-U-HULL	1
2	445	EAST RIDING OF YORKSHIRE	1
2	538	CRAVEN	1
2	539	HAMBLETON	1
2	540	HARROGATE	1
2	541	RICHMONDSHIRE	1
2	542	RYEDAILE	1
2	543	SCARBOROUGH	1
2	544	SELBY	1
2	545	YORK	1
3	172	SOUTH SHROPSHIRE	1
3	375	CHESTER	1
3	376	CONGLETON	1
3	467	CREWE & NANTWICH	1
3	468	ELLESMERE PORT	1
3	469	HALTON	1
3	470	MACCLESFIELD	1
3	471	VALE ROYAL	1
3	472	WARRINGTON	1
3	668	BLACKBURN	1
3	669	BLACKPOOL	1
3	670	BURNLEY	1

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

3	671 CHORLEY	1
3	672 FYLDE	1
3	673 HYNDBURN	1
3	674 LANCASTER	1
3	675 PENDLE	1
3	676 PRESTON	1
3	767 RIBBLE VALLEY	1
3	768 ROSSENDALE	1
3	769 SOUTH RIBBLE	1
3	770 WEST LANCASHIRE	1
3	771 WYRE	1
4	446 NORTH EAST LINCOLNSHIRE	1
4	537 NORTH LINCOLNSHIRE	1
4	566 CORBY	1
4	641 AMBER VALLEY	1
4	642 BOLSOVER	1
4	643 CHESTERFIELD	1
4	644 DERBY	1
4	645 EREWASH	1
4	646 HIGH PEAK	1
4	657 DAVENTRY	1
4	658 EAST NORTHANTS	1
4	659 KETTERING	1
4	660 NORTHAMPTON	1
4	661 SOUTH NORTHANTS	1
4	662 WELLINGBOROUGH	1
4	737 NORTH EAST DERBYS	1
4	738 SOUTH DERBYSHIRE	1
4	739 WEST DERBYSHIRE	1
4	740 BLABY	1
4	741 HINCKLEY/BOSWORTH	1
4	742 CHARNWOOD	1
4	743 HARBOROUGH	1
4	744 LEICESTER	1
4	745 MELTON	1
4	746 NORTH WEST LEICS	1
4	837 OADBY & WIGSTON	1
4	838 RUTLAND	1
4	839 BOSTON	1
4	840 EAST LINDSEY	1
4	841 LINCOLN	1
4	842 NORTH KESTEVEN	1
4	843 SOUTH HOLLAND	1
4	844 SOUTH KESTEVEN	1
4	845 WEST LINDSEY	1
4	846 ASHFIELD	1
4	947 BASSETLAW	1
4	948 BROXTOWE	1
4	949 GEDLING	1
4	950 MANSFIELD	1

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

4	951 NEWARK	1
4	952 NOTTINGHAM	1
4	953 RUSHCLIFFE	1
5	69 BROMSGROVE	1
5	70 HEREFORD	1
5	71 LEOMINSTER	1
5	72 MALVERN HILLS	1
5	73 REDDITCH	1
5	74 SOUTH HEREFORDS	1
5	75 WORCESTER	1
5	76 WYCHAVON	1
5	167 WYRE FOREST	1
5	168 BRIDGNORTH	1
5	169 NORTH SHROPSHIRE	1
5	170 OSWESTRY	1
5	171 SHREWSBURY/ATCHAM	1
5	173 THE WREKIN	1
5	174 CANNOCK CHASE	1
5	175 EAST STAFFS	1
5	176 LICHFIELD	1
5	267 NEWCASTLE-U-LYME	1
5	268 SOUTH STAFFS	1
5	269 STAFFORD	1
5	270 STAFFS MOORLANDS	1
5	271 STOKE ON TRENT	1
5	272 TAMWORTH	1
5	273 NORTH WARWICKS	1
5	274 NUNEATON/BEDWORTH	1
5	275 RUGBY	1
5	276 STRATFORD ON AVON	1
5	367 WARWICK	1
6	48 CAMBRIDGE	1
6	49 EAST CAMBS	1
6	50 FENLAND	1
6	51 HUNTINGDON	1
6	52 PETERBOROUGH	1
6	53 SOUTH CAMBS	1
6	54 BRECKLAND	1
6	55 BROADLAND	1
6	56 GREAT YARMOUTH	1
6	147 NORWICH	1
6	148 NORTH NORFOLK	1
6	149 SOUTH NORFOLK	1
6	150 WEST NORFOLK	1
6	151 BABERGH	1
6	152 FOREST HEATH	1
6	153 IPSWICH	1
6	154 MID SUFFOLK	1
6	155 ST EDMUNDSBURY	1
6	156 SUFFOLK COASTAL	1

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

6	247 WAVENEY	1
8	57 MOLE VALLEY	1
8	58 REIGATE/BANSTEAD	1
8	59 RUNNYMEDE	1
8	60 SPELTHORNE	1
8	61 SURREY HEATH	1
8	62 TANDRIDGE	1
8	63 WAVERLEY	1
8	64 WOKING	1
8	65 ADUR	1
8	66 ARUN	1
8	157 CHICHESTER	1
8	158 CRAWLEY	1
8	159 HORSHAM	1
8	160 MID SUSSEX	1
8	161 WORTHING	1
8	248 NORTH BEDS	1
8	249 LUTON	1
8	250 MID BEDFORDSHIRE	1
8	251 SOUTH BEDS	1
8	252 BROXBOURNE	1
8	253 DACORUM	1
8	254 EAST HERTS	1
8	255 HERTSMERE	1
8	256 NORTH HERTS	1
8	266 BASINGSTOKE/DEANE	1
8	347 ST ALBANS	1
8	348 STEVENAGE	1
8	349 THREE RIVERS	1
8	350 WATFORD	1
8	351 WELWYN HATFIELD	1
8	357 EASTLEIGH	1
8	358 FAREHAM	1
8	359 GOSPORT	1
8	360 HART	1
8	361 HAVANT	1
8	362 NEW FOREST	1
8	363 EAST HAMPSHIRE	1
8	364 PORTSMOUTH	1
8	365 RUSHMOOR	1
8	366 SOUTHAMPTON	1
8	451 BASILDON	1
8	452 BRAINTREE	1
8	453 BRENTWOOD	1
8	454 CASTLE POINT	1
8	455 CHELMSFORD	1
8	456 COLCHESTER	1
8	457 TEST VALLEY	1
8	458 WINCHESTER	1
8	459 ISLE OF WIGHT	1

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

8	465 BRACKNELL	1
8	466 NEWBURY	1
8	547 EPPING FOREST	1
8	548 HARLOW	1
8	549 MALDON	1
8	550 ROCHFORD	1
8	551 SOUTHEND ON SEA	1
8	552 TENDRING	1
8	553 THURROCK	1
8	554 UTTLESFORD	1
8	557 READING	1
8	558 SLOUGH	1
8	559 WINDSOR/MAIDENH'D	1
8	560 WOKINGHAM	1
8	561 AYLESBURY VALE	1
8	562 SOUTH BUCKS	1
8	563 CHILTERN	1
8	564 MILTON KEYNES	1
8	565 WYCOMBE	1
8	663 CHERWELL	1
8	664 OXFORD	1
8	665 VALE WHITE HORSE	1
8	666 SOUTH OXFORDSHIRE	1
8	747 BRIGHTON	1
8	748 EASTBOURNE	1
8	749 HASTINGS	1
8	750 HOVE	1
8	751 LEWES	1
8	752 ROTHER	1
8	753 WEALDEN	1
8	754 ASHFORD	1
8	755 CANTERBURY	1
8	756 DARTFORD	1
8	757 WEST OXFORDSHIRE	1
8	847 DOVER	1
8	848 GILLINGHAM	1
8	849 GRAVESHAM	1
8	850 MAIDSTONE	1
8	851 ROCHESTER	1
8	852 SEVENOAKS	1
8	853 SHEPWAY	1
8	854 SWALE	1
8	855 THANET	1
8	856 TONBRIDGE/MALLING	1
8	957 TUNBRIDGE WELLS	1
8	964 ELMBRIDGE	1
8	965 EPSOM AND EWELL	1
8	966 GUILDFORD	1
9	67 WEST SOMERSET	1
9	68 YEOVIL (aka SOUTH SOMERSET)	1

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

9	258 BOURNEMOUTH	1
9	259 CHRISTCHURCH	1
9	260 NORTH DORSET	1
9	261 POOLE	1
9	262 PURBECK	1
9	263 WEST DORSET	1
9	264 WEYMOUTH/PORTLAND	1
9	265 WIMBORNE (aka EAST DORSET)	1
9	460 KENNET	1
9	461 NORTH WILTSHIRE	1
9	462 SALISBURY	1
9	463 THAMESDOWN	1
9	464 WEST WILTSHIRE	1
9	758 BATH & NORTH EAST SOMERSET	1
9	759 BRISTOL	1
9	760 NORTH SOMERSET	1
9	761 SOUTH GLOUCESTERSHIRE	1
9	762 CARADON	1
9	763 CARRICK	1
9	764 KERRIER	1
9	765 NORTH CORNWALL	1
9	766 PENWITH	1
9	857 RESTORMEL	1
9	858 EAST DEVON	1
9	859 EXETER	1
9	860 NORTH DEVON	1
9	861 PLYMOUTH	1
9	862 SOUTH HAMS	1
9	863 TEIGNBRIDGE	1
9	864 MID DEVON	1
9	865 TORBAY	1
9	866 TORRIDGE	1
9	967 WEST DEVON	1
9	968 CHELTENHAM	1
9	969 COTSWOLD	1
9	970 FOREST OF DEAN	1
9	971 GLOUCESTER	1
9	972 STROUD	1
9	973 TEWKESBURY	1
9	974 MENDIP	1
9	975 SEDGEMOOR	1
9	976 TAUNTON DEANE	1
1	439 GATESHEAD	2
1	440 NEWCASTLE-U-TYNE	2
1	441 NORTH TYNESIDE	2
1	442 SOUTH TYNESIDE	2
1	443 SUNDERLAND	2
2	47 SHEFFIELD	2
2	546 BRADFORD	2
2	637 CALDERDALE	2

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

2	638 KIRKLEES	2
2	639 LEEDS	2
2	640 WAKEFIELD	2
2	954 BARNSELY	2
2	955 DONCASTER	2
2	956 ROTHERHAM	2
3	473 KNOWSLEY	2
3	474 LIVERPOOL	2
3	475 ST HELENS	2
3	476 SEFTON	2
3	567 WIRRAL	2
3	568 BOLTON	2
3	569 BURY	2
3	570 MANCHESTER	2
3	571 OLDHAM	2
3	572 ROCHDALE	2
3	573 SALFORD	2
3	574 STOCKPORT	2
3	575 TAMESIDE	2
3	576 TRAFFORD	2
3	667 WIGAN	2
5	368 BIRMINGHAM	2
5	369 COVENTRY	2
5	370 DUDLEY	2
5	371 SANDWELL	2
5	372 SOLIHULL	2
5	373 WALSALL	2
5	374 WOLVERHAMPTON	2
7	162 CROYDON	3
7	163 KINGSTON-U-THAMES	3
7	164 RICHMOND-U-THAMES	3
7	165 MERTON	3
7	166 SUTTON	3
7	257 WANDSWORTH	3
7	352 BARNET	3
7	353 BRENT	3
7	354 HARROW	3
7	355 EALING	3
7	356 HAMMERSMITH	3
7	447 HOUNSLOW	3
7	448 HILLINGDON	3
7	449 KENSINGTON	3
7	450 WESTMINSTER	3
7	555 BARKING/DAGENHAM	3
7	556 HAVERING	3
7	647 CAMDEN	3
7	648 ISLINGTON	3
7	649 CITY OF LONDON	3
7	650 HACKNEY	3
7	651 NEWHAM	3

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

7	652 TOWER HAMLETS	3
7	653 ENFIELD	3
7	654 HARINGEY	3
7	655 REDBRIDGE	3
7	656 WALTHAM FOREST	3
7	958 BEXLEY	3
7	959 GREENWICH	3
7	960 BROMLEY	3
7	961 LAMBETH	3
7	962 LEWISHAM	3
7	963 SOUTHWARK	3
10	184 ISLE OF ANGLESEY	4
10	185 GWYNEDD	4
10	186 CONWY	4
10	277 DENBIGHSHIRE	4
10	278 FLINTSHIRE	4
10	279 WREXHAM	4
10	280 POWYS	4
10	281 CEREDIGION	4
10	282 PEMBROKESHIRE	4
10	283 CARMARTHENSHIRE	4
10	284 SWANSEA	4
10	285 NEATH PORT TALBOT	4
10	286 BRIDGEND	4
10	377 VALE OF GLAMORGAN	4
10	378 RHONDDA, CYNON, TAFF	4
10	379 MERTHYR TYDFIL	4
10	380 CAERPHILLY	4
10	381 BLAENAU GWENT	4
10	382 TORFAEN	4
10	384 NEWPORT	4
10	385 CARDIFF	4
11	194 ABERDEEN CITY	5
11	195 ABERDEENSHIRE	5
11	196 ANGUS	5
11	287 ARGYLL AND BUTE	5
11	288 SCOTTISH BORDERS	5
11	289 CLACKMANNANSHIRE	5
11	290 WEST DUNBARTONSHIRE	5
11	291 DUMFRIES AND GALLOWAY	5
11	292 DUNDEE CITY	5
11	293 EAST AYRSHIRE	5
11	294 EAST DUNBARTONSHIRE	5
11	295 EAST Lothian	5
11	296 EAST RENFREWSHIRE	5
10	383 MONMOUTHSHIRE	4
11	387 EDINBURGH, CITY OF	5
11	388 FALKIRK	5
11	389 FIFE	5
11	390 GLASGOW, CITY OF	5

11	391 HIGHLAND	5
11	392 INVERCLYDE	5
11	393 MIDLOTHIAN	5
11	394 MORAY	5
11	395 NORTH AYRSHIRE	5
11	396 NORTH LANARKSHIRE	5
11	487 PERTH AND KINROSS	5
11	488 RENFREWSHIRE	5
11	489 SOUTH AYRSHIRE	5
11	490 SOUTH LANARKSHIRE	5
11	491 STIRLING	5
11	492 WEST LOTHIAN	5

 --

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 : **HBM**, ISM (Bold indicates lead user)
 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
 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 CVPD (what period does this cover) = **90 or 95 or 97** (one off/lumpsum 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

Code Condition

BOARDER From ADULT table

 If CONVBL=1 get all variables.

 If CONVBL = 1 (answered boarder), and CVPD equals -1 or ~~4-11~~ **1-52** the amount of BOARDER is the amount in CVPAY (amount of rent paid).

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

-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 = ~~42 or 43~~ **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

 --

BUINC

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 : ASD4A
 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
 Issue date : 12 May 2003

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

Code Condition

 --

BUINC From ADULT table get INEARNs, ~~INSEINC~~ **SEINCAM2**, ININV INRPINC INPENINC, INOTHBEN, INRINC and INDINC for all adults in the benefit unit

 From CHILD table get CHEARNs, CHINV, CHRINC and CHINDV for all children in the benefit unit

 BUINC equals total of all occurrences of INDINC and CHINDV

-2 where any of the components are missing

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 ~~INSEINC~~ **SEINCAM2**

BUIINV 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

 --

BURENT

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.

Created : 26 January 1993

Database Table : BENUNIT

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : HHRENT

Children :

Parents :

Core variable/user : **HBM**, TAKE-UP (nb **bold** denotes lead user)

Issue date : 12 May 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

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 - owner-occupiers (this includes shared owner/occupiers where HHSTAT=2), 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. [Check: Jo S For the moment need to add variable CHBAMT to compare with BOARDER/LODGER BURENT to make sure rent is not less than HB.]

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

<u>Code</u>	<u>Condition</u>
BURENT	From HOUSEHOL table, if TENURE = 1 2 or 3 or 4 (renting) or 5 (renting) 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. (Check if can use an HB receipt (CHBAMT) variable here for cross check. Eg in cases where B/L rent less than HB) (In other cases B/L rent may be greater than CHBAMT because rent includes food/fuel which is not eligible for HB. Could set rent to HB here) For nonHB B/L rent may be greater than eligible rent but cant derive later.

If HHSTAT=2

if RENT (of 1ST BU) declared after HB deducted (HBENCHK=2) and RENTPD, HBENPD and SRENTPD <> ~~42 and 43~~ **90, 95 AND 97** then
 TOTRENT=(RENT+HBENAMT (for 1st BU)) + (SRENTAMT+~~SCHBAMT~~
HBOOTHAMT (for all remaining BUs))

(for 1st BU) set BURENT = (RENT+HBENAMT)/TOTRENT * HHRENT
 (for other BUs) set BURENT = (SRENTAMT+~~SCHBAMT~~
HBOOTHAMT)/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+~~SCHBAMT~~ **HBOOTHAMT** (for all remaining BUs)

(for 1st BU) set BURENT = RENT/TOTRENT * HHRENT
 (for other BUs) set BURENT = (SRENTAMT+~~SCHBAMT~~
HBOOTHAMT)/TOTRENT * HHRENT

Above specification assumes that all ~~42 and 43~~ **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 ~~42 and 43~~ **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

 --

CHCARE, CHHOUR

Purpose : To provide summary variables for child carers
 Created : 29 April 1996
 Database Table : CARE
 Minimum Value :
 Maximum Value :
 Units : integer
 Validations :
 Related Variables : CAREAB, CAREAH, CAREERE, CAREFR, CARECL, CAREOT, CARECB,
 CARECH, HOURAB, HOURAH, HOURRE, HOURFR, HOURCL, HOUROT,
 HOURCB, HOURCH(variables for adults caring)
 Children :
 Parents :
 Core variable/user :
 Amendments : VE - 4 June 1996 - No initial V32 update needed
 JS - 29 October 1996 - to reflect actual coding for V31
 VE - 14 May 1997 - No initial update needed for V33
 Issue date : 12 May 2003

1 Definition

The aim of these derived variables is to provide summary information on 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. For adults, a detailed breakdown of whether those being cared for in the household are in the same or different benefit units. This is not possible for children because information is not collected at WHOLOOK for children individually. Summary variables therefore only give information on the total number of individuals looked after by children and how much time they spend caring.

CHCARE Total number of individuals cared for by child/children within the household

CHHOUR Total number of hours spent caring by child/children in the household

0 **For all variables** - not applicable to this case - child/children in the household not look after anybody

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

2 FRS Specification

Set all variables equal to zero.

From CARE record, for each person needing care, If WHOLOO11 (child/children doing caring)
 CHCARE=CHCARE+1

--

$$\text{CHHOUR}=\text{CHHOUR}+\text{HOUR}(11)$$

-2 If any variables are missing

 --

CHINCDV

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 : ASD4A6E
 Amendments : VE - 5 June 1996 - No initial amendments needed for V32 update
 : SG - 31 December 1997 - V33 updates, period codes, account numbers
 Issue date : 12 May 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** ~~chint (block u chint) and chamt (block s chinc) respectively as these question names are duplicated.~~

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	<p>From CHILD table if CHAMTERN exists then get CHAMTERN</p> <p>set CHEARNS to zero If CHAMTERN exists and CHPDERN equal to -1 or 1-44 10, 13,26 or 52 then CHEARNS=CHAMTERN</p> <p>From ACCOUNTS table</p> <p>set CHINV to zero</p> <p>If ACCOUNT = 24 20 get ACCINT and add to CHINV (NSB Post Office - ordinary) If ACCOUNT = 25 21 get ACCINT and add to CHINV (NSB Post Office - investment) If ACCOUNT = 26 22 get ACCINT and add to CHINV (Building Society) If ACCOUNT = 27 23 get ACCINT and add to CHINV (Bank account) If ACCOUNT = 28 24 get ACCINT and add to CHINV (Gilts) If ACCOUNT = 29 25 get ACCINT and add to CHINV (Unit Trusts) If ACCOUNT = 30 26 get ACCINT and add to CHINV (Other stocks/shares)</p> <p>From CHILD table</p> <p>set CHRINC to zero</p> <p>If CHTST = 1 CHEARNS2=1 and CHPDTST equal to -1 or 1-44 10, 13,26 or 52 (child has an income from a trust fund) get CHAMTTST and add to CHRINC</p> <p>If TOTGNTCH exists then add TOTGNTCH/52 to CHRINC</p> <p>CHINCDV=CHEARNS+CHINV+CHRINC</p>
-2	Unable to derive if any components are missing

CWATAMTD

Purpose: To show total amount of council water charge paid by Scottish households
Database Table : HOUSEHOL
Lead User : ASD4A
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
Issue date: : 12 May 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 ~~for cases where council water charge is paid as part of rent and a look up table (used for the 1994/95 FES data supplied from the Scottish Office) for cases where council water charge is paid separately.~~

~~Additional checks are made in the program to pick up the cases which are not asked directly about council tax included in rent. Those who are receiving a combination of discounts or rebates still have a value imputed for CWATAMTD because water charges are not included in CTB. For more information, see minute dated 14/2/95 (file vsctw).~~

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 the SAS code as currently written (by fluke) will interpret this as Council Tax Band D and return the appropriate lookup value of CWATAMTD. This has been left as it seems reasonable assumption. However if correct action is to set CWATAMTD as missing then code needs to be revised.~~ then set CWATAMTD to skipped (-1).

2 FRS Specification

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

Code Condition

CWATAMTD From ~~RENTER~~ HOUSEHOL table

Set DISCOUNT

If CTDISC=1 then

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

~~If COMMINC equals 1 and CWATAMT exists and CWATPD equals 1 or 1-11
then CWATAMTD equals CWATAMT~~

~~Else if COMMINC equals 2 (rent does not include council tax)-
then apply look up table to calculate CWATAMTD~~

~~Else if RENT=0 (COMMINC is not asked in these cases)
and CTBAND=0 (not valued separately) or CTEXREB=1 2 (respondent
has a formal exemption from the council tax) or (CTEXREB=3 and
WHYNOCT=4) (neither of these - these cases are not covered by other
questions so would otherwise be set to unable to derive) set
CWATAMTD to zero~~

~~Else if CTAMT>0 (paying council tax) or CTEXREB=2 1 (combinations of
discounts) or CTEXREB=4 (bill not yet issued) **CTEXREB=3 and
WHYNOCT=1, 2 or 3 (CT not paid because bill not yet received and HH
not previously liable, bill not yet paid and HH not previously liable or
deliberate non-payment)** then apply look up table to calculate CWATAMTD~~

~~Else if CTBAND=0 (not valued separately) or CTEXREB=1 2 (respondent has
a formal exemption from the council tax) or CTEXREB=3 and
WHYNOCT=4 (neither of these - these cases are not covered by other
questions so would otherwise be set to unable to derive) set
CWATAMTD to zero (repeat for cases where RENT was missing at
interview and following questions have therefore been skipped)
Else if CTAMT>0 (paying council tax) or CTEXREB=2 1 (combinations
of discounts) or CTEXREB=4 (bill not yet issued) **CTEXREB=3 and
WHYNOCT=1, 2 or 3 (CT not paid because bill not yet received
and HH not previously liable, bill not yet paid and HH not
previously liable or deliberate non-payment)** then apply look up
table to calculate CWATAMTD (repeat for cases where RENT was
missing at interview and following questions have therefore been
skipped)~~

~~Else if ADDINF table, TENURE equals 1 (not renting)-
then apply look up table.~~

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 (final mop up of any missing CWATAMTD). to calculate
CWATAMTD from LAC and CTBANDA.~~

Apply the calculated DISCOUNT to all CWATAMTD

- 1 Not applicable (see above)
- 2 Unable to derive due to missing values ~~(this should not occur: COMMINC contains no missing values, all cases where COMMINC equals 2 are updated using a look up table, even where TENURE was missing and has been set to owner occupier, values updated using this method)~~

 --

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

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : ASD6E

Issue date : 12 May 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)

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

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 AMTTAXF, AMTOTH, OTHDED3 - OTHDED78, DEDUC3 - DEDUC67 and DEDOTH. AMTTAXF and AMTOTH deal with deductions paid to charities and OTHDED3 - OTHDED78 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 - DEDUC67 and DEDOTH store the amounts paid.

OTHDED3 - OTHDED78 are database variables produced from OTHDED, which when = 1 indicate whether that person has a particular deduction. DEDUC3 - DEDUC67 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 these deductions only. If analysts need information about all charitable deductions the questionnaire will have to be changed.

If PAYPD = ~~12 or 13~~ **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. ~~12 or 13~~ **90, 95 or 97** refer to **less than one week**/one-off/lumpsum deductions or any other period. If the variables cannot be converted the record must be rejected.

Where variables PAYAMT, CHRTAXF and OTHDED1-78 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 OTHDED1-78 skipped) no deductions are made

2 FRS Specification

For each ADULT

<u>Code</u>	<u>Condition</u>
DEDUCTS	From JOB table, for all jobs a person has
	If PAYPD equals -1 or 1 to 11 1 to 52 do the following -

 --

If CHARITY = 1 (deductions for charities) and CHRTAXF = 1 (deductions made under tax free scheme) and AMTAXF exists then DEDUCTS = AMTAXF (amount deducted)

Else if CHARITY = 1 (deductions for charities) and AMTAXF = -1 then DEDUCTS = DEDUCTS + CHRMEAN (from constants table)

If CHROTH = 1 (other deductions for charities) then DEDUCTS = DEDUCTS + AMTOTH

If OTHDED3 = 1 and DEDUC3 >= 0, DEDUCTS = DEDUCTS + DEDUC3 (amount of union fees) else if OTHDED3 = -1 DEDUCTS = DEDUCTS

If OTHDED4 = 1 and DEDUC4 >= 0, DEDUCTS = DEDUCTS + DEDUC4 (amount for friendly societies) else if OTHDED4 = -1 DEDUCTS = DEDUCTS

If OTHDED5 = 1 and DEDUC5 >= 0, DEDUCTS = DEDUCTS + DEDUC5 (amount for sports clubs or specialised pastimes) else if OTHDED5 = -1 DEDUCTS = DEDUCTS

If OTHDED6 = 1 and DEDUC6 >= 0, DEDUCTS = DEDUCTS + DEDUC6 (amount for repayment of a loan from employer) else if OTHDED6 = -1 DEDUCTS = DEDUCTS

If OTHDED7 = 1 and ~~DEDOTH~~ **DEDUC7** >= 0, DEDUCTS = DEDUCTS + ~~DEDOTH~~ **DEDUC7** (amount of any other deductions not included above **amount for private medical insurance**) else if OTHDED7 = -1 DEDUCTS = DEDUCTS

If OTHDED8 = 1 and DEDOTH >= 0, DEDUCTS = DEDUCTS + DEDOTH (amount of any other deductions not included above) else if OTHDED8 = -1 DEDUCTS = DEDUCTS

-1 Not applicable to this case.

-2 Unable to derive as any of the above variables are missing or PAYPD = ~~42 or 43~~ **90, 95 or 97 (less than one week/one-off/lumpsum 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

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.

 --

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 : ISM HBM 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
 Issue date : 12 May 2003

NB This specification has been changed from the original FES specification as the way in which non-dependent deductions are assessed is now done on a different basis. Mr Shiels of ASD4D provided the new categories after consultation with the relevant policy section.

NB2 The specification was amended as Ms Odwell suggested an easier grouping with the models themselves dealing with the earnings bands so that the FRS specification does not have to be changed each year.

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. Depdedcs 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 = ~~47~~ **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 = ~~47~~ **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 = ~~21~~ **1** (TRAIN asks if adult was on any form of govt training scheme, ~~21~~ = 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 : ASD6E
 Amendments :
 Issued : 12 May 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 H BOTHBU = 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 H BOTHBU=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 H BOTHBU=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 H BOTHBU=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 H BOTHBU=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 : HBM **ISM** PSM OTHERS (Bold indicates lead user)
 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 to :
 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
 Issue date : 12 May 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 (BEN1Q1 = 1), BEN2Q01=1 receiving Attendance Allowance (BEN2Q10 = 1) BEN2Q03=1 or where Attendance Allowance has been awarded AA to start at a later date (FUTATT = 1) 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 ~~BEN1Q1 = 1~~ **BEN2Q01=1** or ~~BEN2Q10 = 1~~ **BEN2Q03=1** or ~~FUTATTB2QFUT3~~ **FUTATTB2QFUT3 = 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.

--

DISTEN

Purpose : To indicate the number of households in any specific tenure type for use in the disability survey.
Created : 23 January 1997
Database Table : HOUSEHOL
Minimum Value : 1
Maximum Value : 7
Units : Integer
Validations :
Related Variables : TENTYPE, PTENTYPE
 Children :
 Parents :
Core variable/user : ASD6E
Issue date : 12 May 2003
Amendments :

1 Definition

DISTEN is a simplified version of TENTYPE for use on the disability survey. It is derived using variables TENURE (HOUSEHOL record) and LANDLORD (RENTER record). In keeping with harmonisation, the 'buying with a mortgage' category includes shared owners who own part of the equity and pay part mortgage, part rent.

This variable is coded as

- 1 Rented from council
- 2 Rented from housing association
- 3 Rented privately
- 4 Buying with a mortgage
- 5 Owned outright
- 6 Rent-free
- 7 Squatting
- 2 Unable to derive due to missing values

2 FRS Specification

--

- 1 TENURE = 4 and LANDLORD = 1
- 2 TENURE = 4 and LANDLORD = 2
- 3 TENURE = 4 and (LANDLORD = 3 or LANDLORD = 4 or LANDLORD = 5 or LANDLORD = 6 or
 LANDLORD = 7)
- 4 TENURE = 2 or TENURE = 3
- 5 TENURE = 1
- 6 TENURE = 5
- 7 TENURE = 6

 --

ECOTYPBU

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 : ASD6A **HBAI** ASD4A (**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
 Issued :28 April 1998

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 ~~30~~ **31** hours.

An individual is self-employed if they record they are self-employed EMPSTAT=2 and work ~~30~~ **31** hours or more in their main job (QHRSELF >= ~~29~~ **30 31** and if they are not on a Govt Training Prog (these should all be set as unemployed, except if they're on Enterprise Allowance **Business Start-up schemes**)).

 --

 --

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 ~~30~~ **31** or more hours a week in their main job (QHRS+EMPOVT>= ~~29 30 31~~) ~~and they are not~~ **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 ~~30~~ **31** hours.

EMPSTATC is set as unemployed by the following conditions:

where an individual is in receipt of UB (BEN3Q1=1);
~~or where an individual has looked for work in the last four weeks and can start work in the next two (LOOK4=1 and START=1);~~
~~or where an individual has looked for a Govt Training scheme in the last 4 weeks and can start work in the next two and is under 60 (LKYT4=1 and and AGE<60 START=1); or~~
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
~~or where an individual is on a Govt Training Scheme, except Enterprise Allowance **Business Start-up schemes.**~~

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

 --

- 5 All cases first coded to not working for any other reason and then overwritten when one of the conditions apply
- 4 (from benefit record BEN3Q1=1) or
~~(LOOK4=1 and START=1) or~~
~~(LKYT4=1 and AGE<60 and START=1) or~~
(LOOK=1 and LOOKWK=1 and START=1) or
 WAIT=1 or
 JOBAWAY=3 or
~~or TRAIN = 1,2,3,4,5,6)~~
- 3 ~~(TRAIN <=> = 1,2,3,4,5,6) and or ((EMPSTAT=1 or EMPSTAT=2) and~~
 (ABSPAY<>2,3 and, from first job record, QHRS+EMPOVT<~~30~~ **31**) or (from first job record
 QHRSELF<~~30~~ **31**))
- 2 (TRAIN <=> = 1,2,~~3~~,4,5,6) and or (EMPSTAT=1 and ABSPAY<>2,3 and, from first job record,
 QHRS+EMPOVT>= ~~29~~ ~~30~~ **31**)
- 1 ~~(TRAIN <=> = 1,2,3,4,5,6) and or (EMPSTAT =2 and, from first job record, QHRSELF>=~~ ~~29~~ ~~30~~
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 : 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.
 Issued :12 May 2003

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** as follows. Note that EMPSTATC should also be added to the 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 30 hours.~~

~~An individual is self-employed if they record they are self-employed EMPSTAT=2 and work 30 31 hours or more in their main job (QHRSELF>= 29 30 31) and if they are not on a Govt Training Prog (these should all be set as unemployed, except if they're on Enterprise Allowance **Business Start-up schemes**).~~

~~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 30 31 or more hours a week in their main job (QHRS+EMPOVT>= 29 30 31) and they are not ~~or 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 30 31 hours.~~

~~EMPSTATC is set as unemployed by the following conditions:~~

~~where an individual is in receipt of UB (BEN3Q1=1);~~

~~or where an individual has looked for work in the last four weeks and can start work in the next two (LOOK4=1 and START=1);~~

~~or where an individual has looked for a Govt Training scheme in the last 4 weeks and can start work in the next two and is under 60 (LKYT4=1 and and AGE<60 START=1); or~~

~~**or where an individual is looking for work or training and wants work and can start work in next two weeks or;**~~

~~where an individual is waiting to start a job, either WAIT=1 or JOBAWAY=3~~

~~or where an individual is on a Govt Training Scheme, except Enterprise Allowance **Business Start-up schemes**.~~

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

~~ECSTATBU is coded as:~~

- ~~1 Self employed (benefit units where at least one adult usually works self-employed for 30 31 or more hours a week)~~
- ~~2 Single or couple, all in full time work as an employee (30 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. Total hours worked are based on QHRSSELF (hours worked as self-employed - only asked of first job) and total of QHRS+EMVOVT (contracted hours plus usual overtime) for each job as an employee. This is different from EMPSTATB (although in practice, figures will be very similar) which uses the TOTHOURS DV to distinguish full/part time work. TOTHOURS 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 : 10
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ~~ASD6A~~ ASD4A3E
 Issue date : 12 May, 2003
 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

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 (less than 28 weeks sick)
- 5 Part-time employee temporarily not working (less than 28 weeks sick)
- 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 less than 28 weeks

 --

- 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 lt 65 and SEX = 1 and where AGE lt 60 and SEX = 2
 If WORKING = 2 and JOBAWAY = 3 and LOOKWK = 1, 2 or 3
 If WORKING = 2, JOBAWAY = 2 and LOOK4 = 1
~~If WORKING = 2, JOBAWAY = 2 and LKYT4 = 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 or 6

- 9 Code 9 for the following
 From ADULT table, where AGE ge 65 and SEX = 1 and where AGE ge 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 6** or 7 or 8 or **9**
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 2 and NOWANT = 3 or **4 or 6** 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 = **45** or
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 2, NOWANT = **45** or
 If INJLONG = 1.

- 12 As above from ADULT table,
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 1, NOLOOK = ~~5~~ **6** or
 If WORKING = 2, JOBAWAY = 2, LIKEWK = 2, NOWANT = ~~5~~ **6** or
 If INJLONG = 2

- 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.

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

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	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.
Look4	Looking for work in last 4 weeks. 1 = yes, 2 = no.
Lkyl4	Looking for place on government scheme in last 4 weeks. 1 = yes, 2 = no.
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 nuserly 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 ~~BU~~ ADULT)
 Children :
 Parents :
 Core variable/user : ~~ASD6A~~ ASD4A
 Issue Date : 12 May 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

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 Employee (includes people not at work due to pattern of shifts)~~
- ~~2 Self-employed~~
- ~~3 Employee temporarily not at work (including those less than 28 weeks sick, laid off, on strike or on maternity leave, **paternity leave or compassionate leave**)~~
- ~~4 Work-related Government Training~~
- ~~5 Unemployed (ILO definition)~~

 --

~~6 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. ~~All employees are then asked whether they are going to work today, if the answer is yes (TDAYWRK = 1) the person can be automatically classed as an employee. However, if the person answers no (TDAYWRK = 2 or 3) further investigation must be conducted to find out whether the absence is for less than 3 working days (ABSWK = 2) they are also classed as employees. If ABSWK = 1 the reason for the absence must be found from ABSWHY. If ABSWHY = 1 the absence is due to normal pattern of shifts or ABSWHY = 3 on holiday, the person may be classed an employee).~~

Self-employed people are derived where WORKING = 1 or WORKING = 2 and JOBAWAY = 1 (as above) and EMPSTAT = 2 (self-employed). ~~The question TDAYWRK is not asked of people who declare themselves self-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.

~~There are several ways to indicate that an employee is temporarily absent from their normal employment also using the ABSWHY question. Where ABSWHY = 2 the person is off sick and the period of sickness must be checked in ABS1PD (if it is less than 28 weeks the person falls into code 2 if not they are classed as inactive). If ABSWHY = 4 they are on strike, ABSWHY = 5 they are temporarily laid off or if ABSWHY = 6 they are on maternity leave, **ABSWHY = 7 they are on paternity leave, ABSWHY = 8 they are on compassionate leave.**~~

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 6 indicate a government scheme (~~1 = ET, 2 = Youth Training, 3 = Voluntary Projects Programme, 4 = Community Industry, 5 = Other government programme~~ **1 - Youth Training, 2 - Training for Work, 3 - Business Start-up Scheme, 3 - Work Trial, 4 - Career Development Loans/Youth Credits, 5 - Community Action, 6 - Other training scheme**). ~~People on Business Start-up Schemes are treated as 'Working' and self-employed.~~

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 seek if the person is available for and looking for work. LOOK-4 = 1 indicates that the person is looking for work **or a government scheme** in the last 4 weeks, ~~LKYT4 = 1 indicates that he/she has been looking for a government scheme~~, 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.

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

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

Individuals over state retirement age 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 cannot work because of an injury/disability (INJLONG=1 or 2) or that do not want work or are not looking for work because of sickness/disability.

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

~~Any other person will be caught by the inactive group which acts as a catch all.~~

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 TDAYWRK = 1 or or
~~-----~~
~~-----~~ If WORKING = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWK = 2 or
~~-----~~ If WORKING = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWHY = 1 or 3 (shifts or
 holiday) or
~~-----~~ If JOBAWAY = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWK = 2 or
~~-----~~ If JOBAWAY = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWHY = 1 or 3 (shifts or
 holiday)
 If TRAIN = 1, 2, 3, 3, 4, or 5 or 6 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

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

~~-----~~
~~-----~~ If TRAIN=3

~~3~~ If WORKING = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWHY = 2 and ABS1PD
 <= 28 or (not working today, absent more than 3 days & reason illness/accident) or
~~-----~~
~~-----~~ If WORKING = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWHY = 4 or 5 or 6 or 7
 or 8 or
~~-----~~ If JOBAWAY = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWHY = 2 and ABS1PD
 <= 28 or
~~-----~~ If JOBAWAY = 1 and EMPSTAT = 1 and TDAYWRK = 2 or 3 and ABSWHY = 4 or 5 or 6 or 7
 or 8.

 --

- 4 **If WORKING = 1 and EMPSTAT = 2 and FTPT=2 or
 If JOBAWAY = 1 and EMPSTAT = 2 and FTPT=2**
- 5 If WORKING = 2 and JOBAWAY = 3 and LOOKWK =1 or 2 or 3 or
 If WORKING = 2 and JOBAWAY = 2 and
 If LOOK = 1 or (looking for work) **or**
 If LKYT4 = 1 or (looking for govt scheme)
 If WAIT = 1 or (waiting start job) **or**
 If LIKEWK = 1 and NOLOOK = 1 (would like regular work and reason not looking =
 waiting start new job/business)
 If LIKEWK = 2 and NOWANT = 1 (**does not want work and waiting for result of
 application**)
- 6 **If (AGE>=65 and SEX=1) or (AGE>=60 and SEX=2) or
 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
 If JOBAWAY=2 and LIKEWK=1 and NOWANT=3**
- 9 **If (JOBAWAY=2 and LIKEWK=1 and (NOWANT=6 or NOLOOK=6))
 or INJLONG=2**
- 10 **If (JOBAWAY=2 and LIKEWK=1 and (NOWANT=5 or NOLOOK=5))
 or INJLONG=1**
- 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

3 **Results**

Tabulation required to show the number of adults falling into each category ~~and a breakdown of the cases caught in code 6 (inactive).~~

4 **Test Cases**

None as yet.

 --

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
 Issued : 12 May 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 h:\frs\vars\docs\spread.xls as V32 constants as V33 constants in p:\frs\shared\frs33\metadata\const33.xls**. The relevant values from the following table are accumulated to provide the benefit unit scale.

1st adult (head of household)		0.55	HDBU1A
spouse of head		0.45	SPBU1A
heads of subsequent units		0.45	HDBUXA
spouse of any subsequent units		0.45	SPBUXA
each dependent aged	0 - 1	0.07	D01A
	2 - 4	0.18	D24A
	5 - 7	0.24	D57A
	8 - 10	0.23	D810A
	11 - 12	0.26	D1112A
	13 - 15	0.28	D1315A

 --

16 + ~~0.38~~ **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. ~~Is this OK it can be changed if you need anything different.~~

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	0.55	HDBU1A
	If BENUNIT = 1 and SPOUSE = 1 (spouse/partner) add	0.45	SPBU1A
	If BENUNIT >= 2 and HEAD = 1 add	0.45	HDBUXA
	If BENUNIT >= 2 and SPOUSE = 1 add	0.45	SPBUXA
	Process each CHILD record and accumulate the following for each child in the benefit unit.		
	If AGE <= 1 add	0.07	D01A
	If AGE >= 2 and <= 4 add	0.18	D24A
	If AGE >= 5 and <= 7 add	0.24	D57A
	If AGE >= 8 and <= 10 add	0.23	D810A
	If AGE >= 11 and <= 12 add	0.26	D1112A
	If AGE >= 13 and <= 15 add	0.28	D1315A
	If AGE >= 16 add	0.38	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
 Issued : 12 May 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 ~~h:\frs\vars\docs\spread.xls~~ as V32 V33 Constants p:\frs\shared\frs33\metadata\const33.xls.** The relevant values from the following table are accumulated to provide the benefit unit scale.

1st adult (head of household)		0.64	HDBU1B
spouse of head		0.39	SPBU1B
heads of subsequent units		0.46	HDBUXB
spouse of any subsequent units		0.42	SPBUXB
each dependent aged	0 - 1	0.09	D01B
	2 - 4	0.18	D24B
	5 - 7	0.24	D57B
	8 - 10	0.23	D810B
	11 - 12	0.25	D1112B
	13 - 15	0.27	D1315B

 --

16 + ~~0.36~~ **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. ~~Is this OK it can be changed if you need anything different.~~

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	0.64	HDBU1B
	If BENUNIT = 1 and SPOUSE = 1 (spouse/partner) add	0.39	SPBU1B
	If BENUNIT >= 2 and HEAD = 1 add	0.45	HDBUXB
	If BENUNIT >= 2 and SPOUSE = 1 add	0.42	SPBUXB
	Process each CHILD record and accumulate the following for each child in the benefit unit.		
	If AGE <= 1 add	0.09	D01B
	If AGE >= 2 and <= 4 add	0.18	D24B
	If AGE >= 5 and <= 7 add	0.24	D57B
	If AGE >= 8 and <= 10 add	0.23	D810B
	If AGE >= 11 and <= 12 add	0.25	D1112B
	If AGE >= 13 and <= 15 add	0.27	D1315B
	If AGE >= 16 add	0.36	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

Issued : 12 May 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 h:\frs\vars\docs\spread.xls p:\frs\shared\frs33\metadata\const33.xls as V32 Constants.** The relevant values from the following table are accumulated to provide the household scale.

1st adult (head of household)		0.55	AD1A
spouse of head		0.45	AD2A
or other 2nd adult		0.45	AD2A
3rd adult		0.45	AD3A
or any subsequent adult		0.40	SUBSADA
each dependent aged	0 - 1	0.07	DEP01A
	2 - 4	0.18	DEP24A
	5 - 7	0.24	DEP57A
	8 - 10	0.23	DEP810A
	11 - 12	0.26	DEP1112A

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

13 - 15 0.28 **DEP1315A**
 16 + 0.38 **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	0.55	AD1A
	If PERSON = 2 and BENUNIT =1 and ((MS = 2 and SPOUT^=1) OR COHAB=1) 4 or 3 (spouse/partner) add	0.45	AD2A
	Else if PERSON = 2 add	0.45	AD2A
	If PERSON = 3 add	0.45	AD3A
	If PERSON > 3 add for each subsequent adult	0.40	SUBSADA
	Process each CHILD record and accumulate the following for each child in the household		
	If AGE <= 1 add	0.07	DEP01A
	If AGE >= 2 and <= 4 add	0.18	DEP24A
	If AGE >= 5 and <= 7 add	0.24	DEP57A
	If AGE >= 8 and <= 10 add	0.23	DEP810A
	If AGE >= 11 and <= 12 add	0.26	DEP1112A
	If AGE >= 13 and <= 15 add	0.28	DEP1315A
	If AGE >= 16 add	0.38	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

Issued : 12 May 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 h:\frs\vars\docs\spread.xls p:\frs\shared\frs33\metadata\const33.xls as V32 Constants.** The relevant values from the following table are accumulated to provide the household scale.

1st adult (head of household)		0.64	AD1B
spouse of head		0.39	SPHDB
or other 2nd adult		0.46	AD2B
3rd adult		0.42	AD3B
or any subsequent adult		0.36	SUBSADB
each dependent aged	0 - 1	0.09	DEP01B
	2 - 4	0.18	DEP24B
	5 - 7	0.24	DEP57B
	8 - 10	0.23	DEP810B

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

11 - 12	0.25	DEP1112B
13 - 15	0.27	DEP1315B
16 +	0.36	DEP16+B

-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 EQUIVBHC to zero

<u>Code</u>	<u>Condition</u>		
EQUIVBHC	Process each ADULT record and accumulate the following for each adult in the household		
	If PERSON = 1 (head of household) add	0.64	AD1B
	If PERSON = 2 and BENUNIT =1 and ((MS = 2 and SPOUT^=1) OR COHAB=1) 4 or		
	3 (spouse/partner) add	0.39	SPHDB
	Else if PERSON = 2	0.46	AD2B
	If PERSON = 3 add	0.42	AD3B
	If PERSON > 3 add for each subsequent adult	0.36	SUBSADB
	Process each CHILD record and accumulate the following for each child in the household		
	If AGE <= 1 add	0.09	DEP01B
	If AGE >= 2 and <= 4 add	0.18	DEP24B
	If AGE >= 5 and <= 7 add	0.24	DEP57B
	If AGE >= 8 and <= 10 add	0.23	DEP810B
	If AGE >= 11 and <= 12 add	0.25	DEP1112B
	If AGE >= 13 and <= 15 add	0.27	DEP1315B
	If AGE >= 16 add	0.36	DEP16+B

-2 If any of above values are missing

3 Results

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

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 \Rightarrow 65) or
- The Female is over pension age ((SEX = 2 and AGE \Rightarrow 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 \geq 65) or (SEX = 2 and AGE \geq 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)

 --

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 :

Amendments :

Issued : 12 May 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=1,2,3,4,5,6 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)]
3	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=2 and AGE>=60)]
4	If MS= 1,2,3,4,5,6 and SPOUT [^] =1 and (second adult in the benefit unit exists) and depchildb > 0 or SPOUT=1 and MS=2 and depchildb>0
5	If MS=1,2,3,4,5,6 and SPOUT [^] =1 and (second adult in the benefit unit exists) and depchildb=0 or SPOUT=1 and MS=2 and depchildb=0
6	If MS=1,2,3,4,5,6 and SPOUT [^] =1 and only one adult in the benefit unit and depchildb>0
7	If MS=1,2,3,4,5,6 and SPOUT [^] =1 and only one adult in the benefit unit and sex=1 and depchildb=0
8	If MS=1,2,3,4,5,6 and SPOUT [^] =1 and only one adult in the benefit unit and sex=2 and depchildb=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 : ~~HBAI~~ **ASD4A**

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

Issued : 12 May 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 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=1,2,3,4,5,6 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 MS=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 > 0 or SPOUT=1 and MS=2 and depchildb>0
4	If MS=1,2,3,4,5,6 and SPOUT^=1 and (second adult in the benefit unit exists) and depchildb=0 or SPOUT=1 and MS=2 and depchildb=0
5	If MS=1,2,3,4,5,6 and SPOUT^=1 and only one adult in the benefit unit and depchildb>0
6	If MS=1,2,3,4,5,6 and SPOUT^=1 and only one adult in the benefit unit and depchildb=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 : ISM FCM **PSM** (**Bold** indicates lead user)
 Issue date : 12 May 2003
 Amendments : VC - 22 January 1993
 : VC - 23 March 1993 To change the variables used in the coding to make the :
 : derivation much simpler.
 : VC - 7 May 1993 To use income-related benefits definition of pension age ie :
 : 60 and over
 : 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

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 = ~~4~~ 2 ~~and SPOUT=1~~ (married) **or MS=1 or 2 or 3 or 4 or 5 and COHAB=1 (cohabiting)** and, MS = 2 **and SPOUT=1** (married spouse not in household), ~~MS = 3 (cohabiting)~~. Single people will be the rest - MS = 41 (single, never married), MS = 5 (widowed), MS = 63 (separated) and MS = ~~7~~ 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 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=1) 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=1) 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=1) 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=1) 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.

 --

FOSTERBU

Purpose : To indicate the total number of foster children in a benefit unit.
 Created : VC - 3 March 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : INTEGER
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ~~HBAI~~ **ASD4A**
 Amendments : VC - 11 February 1994 Amendments to reflect version 30 changes
 : VE - 6 June 1996 - No amendments needed for initial V32 update
 : VE - 20 June 1996 - To change lead user to ASD4A in response to request
 : SG - 25 June 1997 - No changes needed for V33
 Issue date : 12 May 2003

1 Definition

This variable is coded as

FOSTERBU The total number of foster children in a benefit unit.

-1 Not applicable to this case.

-2 Unable to derive due to missing values.

The variable FOSTERBU is derived from the variable foster which is attached to every child record. FOSTERBU is then the count of each child in the benefit unit where foster = 1. If foster = 2 the child is not a foster child and is not applicable to this case. The variable foster merely asks whether the child is a foster child and is attached to each child record for children up to the age of 15. Dependent children aged 16 or over are not asked the question as they are no longer recognised as being foster children. As a result this group will probably be pulled out as -1 (not applicable).

2 FRS Specification

For each BENEFIT UNIT (BENUNIT record)

<u>Code</u>	<u>Condition</u>
FOSTERBU	From CHILD table, for each child in benefit unit If foster = 1, compute count of foster children for that benefit unit.
-1	Not applicable to this case - where FOSTER = 2

--

-2 Unable to derive due to missing value of FOSTER - probably those dependant children aged 16 - 18.

3 Results

Tabulation is required to show the number of foster children by benefit unit in the following categories - None, 1, 2, 3, 4, 5 and over.

4 Test Cases

None as yet

 --

FSMVAL, FSMBU & FSMHH

Purpose : This specification produces three variables which calculate the value of free school meals for each dependent, benefit unit and household

Created : VC - 2 September 1993

Database Table : CHILD, BENUNIT & HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : 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

Issued : 12 May, 2003

1 Definition

This variable is coded as

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

FSMBU This is the total value of any free school meals received by each benefit unit.

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 ~~FREEITEM~~ and SMLIT variable from the CHILD record. Where SCHMEAL = 1 ~~FREEITEM=4~~ (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 for 1994 to produce the total amount spent each week.

~~The total amount of free school meals for a week (from the TBM) is £4.04 £4.30 per week. This amount is then multiplied by 52/40 to give an average throughout the school year and is then divided by 5 to get a daily amount – £1.0504 £1.118 per day. The daily cost of a free school meal is given by the value COSTM from the V32 Constants table held in h:\frs\vars\docs\spreads.xls V33 constants table held in p:\frs\shared\frs33\metadata\const33.xls.~~

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

2 FRS Specification

<u>Code</u>	<u>Condition</u>
FSMVAL	For each CHILD from CHILD table, Set COST (of free school meals) to £1.0504 £1.118 COSTM (Supplied to FES from Tax Benefit Model for 1994) If SCHMEAL = 1 FREEITEM=4 , calculate the value of free school meals calculate FSMVAL = SMLIT * COST if FREEITEM not eq 4 set FSMVAL=0 For each BENEFIT UNIT sum each occurrence of FSMVAL for each child in benefit unit For each HOUSEHOLD sum each occurrence of FSMVAL for each child in 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

 --

FSMLKVAL, FSMLKBU & FSMLKHH

Purpose : This specification produces three variables which calculates the value of free :
 school milk for each child, benefit unit and household

Created : VC - 3 September 1993

Database Table : CHILD, BENUNIT & HOUSEHOL

Minimum Value : 0

Maximum Value:

Units : Real

Validations :

Related Variables : 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

Issued : 12 May 2003

1 Definition

This variable is coded as

FSMLKVAL This is the total value of any free school milk received by a child.

FSMLKBU This is the total value of any free school milk received by any child in the benefit unit.

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 ~~FREEITEM~~ and SMKIT variables from the CHILD record on the database. Where SCHMILK = 1 ~~FREEITEM=3~~ (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. ~~CSO use the cost of a third of a pint/bottle of milk at 34p 36p per bottle/carton. This gives a cost of a third of a pint to be £0.1133 £0.12~~

 --

The cost of a third of a pint of milk is given by the value COSTMLK from the V33 constants table p:\frs\shared\frs33\metadata\const33.xls.

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>
FSMLKVAL	For each CHILD from CHILD record. If SCHMILK = 1 FREEITEM=3 , calculate the value of free school milk calculate FSMLKVAL = SMKIT x cost of free school milk (0.1133 0.12 COSTMLK) If SCHMILK = 2 FREEITEM not eq 3 , calculate FSMLKVAL = 0.
FSMLKBU	For each BENEFIT UNIT (for BENUNIT record) sum each occurrence of FSMLKVAL for each child in benefit unit.
FSMLKHH	For each HOUSEHOLD (for HOUSEHOL record) sum each occurrence of FSMLKVAL for each child in household.
0	Not applicable - no school milk/no dependents
-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.

 --

FWMLKVAL, FWMLKBU & FWMLKHH

Purpose : This specification produces three variables which calculates the value of free :
 welfare milk for each person (adult & child), benefit unit and household

Created : VC - 3 September 1993

Database Table : ADULT, CHILD, BENUNIT & HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables : FSMVAL, FSMBU, FSMHH, 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

Issued : 12 May 2003

1 Definition

This variable is coded as

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

FWMLKBU This is the total value of any free welfare milk received by any person in the benefit unit.

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 ~~FREEITEM~~ and WMKIT variables from the ADULT and CHILD records on the database. Where WELFMILK = 1 ~~FREEITEM=2~~ (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 (ie consistent with free school milk calculation) rather than the TBM. The cost of a pint of milk for 1994/95 ~~1995-96 is 34p-36p.~~ is given by the value COSTWMK from the ~~V32 Constants table held in~~

 --

h:\frs\dwars\docs\spreads.xls V33Constants table held in
 p:\frs\shared\frs33\metadata\const33.xls.

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	<p>For each ADULT from ADULT record.</p> <p>If WELFMILK = 1 FREEITEM=2, calculate the value of free welfare milk</p> <p>calculate FWMLKVAL = WMKIT x cost of free welfare milk (.34 for 1994/95 0.36 for 1995-96) COSTWMK</p> <p>If WELFMILK = 2 FREEITEM not eq 2, calculate FWMLKVAL = 0.</p> <p>For each CHILD from CHILD record.</p> <p>If WELFMILK = 1 FREEITEM=2, calculate the value of free welfare milk</p> <p>calculate FWMLKVAL = WMKIT x cost of free welfare milk (.34 for 1994/95 0.36 for 1995-96) COSTWMK</p> <p>If WELFMILK = 2 FREEITEM not eq 2, calculate FWMLKVAL = 0.</p>
FWMLKBU	<p>For each BENEFIT UNIT (for BENUNIT record)</p> <p>sum each occurrence of FWMLKVAL for each adult and child in benefit unit.</p>
FWMLKHH	<p>For each HOUSEHOLD (for HOUSEHOL record)</p> <p>sum each occurrence of FWMLKVAL for each adult and child in household.</p>
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

 --

GROSS

Purpose : This variable contains the FRS grossing factors. They allow analysts to weight FRS statistics so that they are representative of the national population.
 Created :
 Database Table : BENUNIT, HOUSEHOL
 Minimum Value :
 Maximum Value :
 Units :
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ASD3A
 Amendments : VE - 3 June 1997 - To correct a mistake in the control total table
 SG - 17 September 1998 - Updated by Ian Davis for V33

1 Definition

This variable contains the FRS grossing factors. They allow analysts to weight FRS statistics so that they are representative of the national population. In doing so they attempt to correct for some of the differential rates of non-response by different types of household. Each household has a different grossing factor and each benefit unit within the household has the same grossing factor. However, **the grossing system can be used for analyses at both levels.**

Examples

To find the number of households with income greater than £x:

- (i) select out the head of households' benefit unit from the flat file
- (ii) count up how many of them have values for the household income variable HHINC greater than £x
- (iii) weight by GROSS~~956~~

To find the number of benefit units with income greater than £x:

- (i) count how many have values for the benefit unit income variable BUINC greater than £x
- (ii) weight by GROSS~~956~~

Units

The grossing factors on the FES were in thousands **but the FRS grossing factors will give actual population numbers.** The reason for changing this is to avoid confusion between grossed and ungrossed statistics in outputs from computer runs. For example, a sample count of 100 might gross up to about 90,000 on the FRS and if this latter is output as "90" it could be mistaken for a sample count. The smaller FES sample did not cause such problems.

 --

In the interests of accuracy, the grossing factors have not been rounded to whole numbers. Hence, grossed counts will not necessarily be integers so that, for example, a count of 2,546.677 households would need to be rounded by the analyst to 2,547. They have been rounded this year as no real gain in accuracy from having a non-rounded figure and therefore it will be less confusing in this form.

Derivation

The grossing system represented by GROSS~~956~~ is just one of the many tested by ASD3A. Alternatives have not been added to the flatfile, to avoid confusion and promote consistency. However, it may be that for some specific tasks, analysts prefer to generate grossing factors which, for example, control to a different combination of variables. ASD3A can then advise on the use of software and on the statistical considerations in choosing a grossing regime.

To derive GROSS~~956~~, the sample is divided into different groups and the grossing factors are the ratio of population estimates to sample counts for those groups. The groups are designed to reflect differences in response rates among different types of households. They have also been chosen with the aims of DSS analyses in mind. The population estimates are based on control variables, with values derived from external data sources.

The control variables and their sources are listed below. The grossing system controls for variables at both household level and benefit unit level. A grossed count of the number of owner occupying households would thus tie in with the DoE figure, whilst the grossed number of single men under 35 would be consistent with the ~~OPCS~~ ONS estimate. Some adjustments have been made to the original control total data sources so that definitions match those in the FRS, eg an adjustment has been made to the demographic data to exclude people not resident in private households.

Control variables used to generate grossing factors		
Variable	Groupings	Source of data
Age/sex/marital status	Single men: <35, 35-59, 60+ Single women: <35, 35-64, 65+ Couples: <65, 65+	OPCS GAD, GAD
Lone parents	Male, female	ASD4E estimates
Families	No. of couples with children	ASD1 Child benefit data
Tenure type	LA renters, private renters, owner occupiers	DoE DETR estimates
Council Tax Band	A, B, C-D, E-H	DoE DETR estimates
Region	London, other	DoE DETR estimates

 --

In order to reconcile control variables at different levels and estimate their joint population software provided by the French national statistics institute INSEE has been used. This program, CALMAR, is a SAS macro and it works by iterating towards a solution. Options within CALMAR that give the solution which minimises the range of grossing factors have been used. This should maximise the potential precision of grossed estimates; if a few cases are associated with very small or very large grossing factors, grossed estimates will have relatively wide confidence intervals.

Careful consideration has been given to the combination of control totals and the way age ranges, CT Bands and so on have been grouped together. The aim has been to strike a balance so that the grossing system will provide, where possible, accurate estimates in different dimensions without significantly increasing variances. Further details are provided in Andrew Ray's Analytical Note 5, copies of which are held in ASD3.

Control totals

The controls below relate to individuals, benefit units or households, depending on the category.

Control Totals for GROSS (000s)	
Single men <35	4,000
Single men 35-59	1,816
Single men 60+	1,163
Single women <35	3,062
Single women 35-64	2,183
Single women 65+	2,912
Couples <65	11,465
Couples 65+	2,630
Male lone parents	126
Female lone parents	1,578
Number of families with children	5,319
LA renters	4,407

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

Private renters	3,363
Owner occupiers	15,712
CT Band A	6,408
CT Band B	4,721
CT Band C-D	8,216
CT Band E-H	4,137
Households in London	2,986
Households in other regions	20,496

 --

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 :

 Children :

 Parents :

Core variable/user : ISM HBM **PSM** FCM (Bold indicates lead user)

Issue date : 12 May 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

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.

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 = ~~2~~ **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 OTHDED~~7~~**8** = 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 OTHDED~~7~~**8**, DEDUC1 to DEDUC~~6~~**7** 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 ~~TAXINC = 1~~ **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 ~~12 or 13~~ **90 or 95 or 97** (lumpsum/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 = ~~12 or 13~~ **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

~~For each ADULT, set GROSSPAY to zero.~~

 --

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 1-11 **13, 26 or 52** and

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

Or if PAYSZIP = ~~2~~ **3** or (PAYSZIP=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 AMTTAXF exists add AMTTAXF to GROSSPAY

If AMTOTH exists add AMTOTH 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 = ~~12 or 13~~ **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 :
 Minimum Value : 1
 Maximum Value : 13
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ASD4A
 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

Issue date : 12 May, 2003

1 Definition

This variable is coded as

1 North East
~~2 North West and Merseyside~~
~~34 Yorkshire and the Humberside~~
 45 East Midlands
 56 West Midlands
 610 South West
 7 Eastern
 8 London
 9 South East
 4011 Wales
 4412 Scotland
 42 North West
 432 Merseyside

-1 Not applicable to this case
 -2 Unable to derive as variable LAC is missing

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). Cumbria,

--

which is included with the North in STDREGN, is in government office region 2 (North West and Merseyside). LACs which are in Bedfordshire, Essex and Hertfordshire are included with the Eastern LACs (GVTREGN=7), whereas they are included with the South East in STDREGN.

Merseyside has been included with North West following concerns that stratification of sample using standard regions leads to problems with analysis by Government Regions.

2 FRS Specification

Code each region according to the attached list.

- 1 Not applicable to this case (shouldn't be any)
- 2 Unable to derive as the variable LAC is missing.

NB - These LA codes are unique to FRS.

3 Results

Tabulation to show the number of households in each standard region

4 Test Cases

None

 --

Note - Between V32 and V33 LAC code numbers have changed so much that normal convention of marking up all changes has not been used below

GOVERNMENT OFFICE REGION	LAC	LOCAL AUTHORITY
1	145	HARTLEPOOL UA
1	146	REDCAR AND CLEVELAND UA
1	237	MIDDLESBROUGH UA
1	238	STOCTON ON TEES UA
1	245	CHESTER LE STREET CD
1	246	DARLINGTON CD
1	337	DERWENTSIDE CD
1	338	DURHAM CD
1	339	EASINGTON CD
1	340	SEDFIELD CD
1	341	TEESDALE CD
1	342	WEAR VALLEY CD
1	343	ALNWICK CD
1	344	BERWICK-U-TWEED CD
1	345	BLYTH VALLEY CD
1	346	CASTLE MORPETH CD
1	437	TYNEDALE CD
1	438	WANSBECK CD
1	439	GATESHEAD MD
1	440	NEWCASTLE-U-TYNE MD
1	441	NORTH TYNESIDE MD
1	442	SOUTH TYNESIDE MD
1	443	SUNDERLAND MD
2	239	ALLDERDALE CD
2	240	BARROW IN FURNESS CD
2	241	CARLISLE CD
2	242	COPELAND CD
2	243	EDEN CD
2	244	SOUTH LAKELAND CD
2	375	CHESTER CD
2	376	CONGLETON CD
2	467	CREWE & NANTWICH CD
2	468	ELLESMERE PORT CD
2	469	HALTON CD
2	470	MACCLESFIELD CD
2	471	VALE ROYAL CD
2	472	WARRINGTON CD
2	568	BOLTON MD
2	569	BURY MD

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

2	570	MANCHESTER MD
2	571	OLDHAM MD
2	572	ROCHDALE MD
2	573	SALFORD MD
2	574	STOCKPORT MD
2	575	TAMESIDE MD
2	576	TRAFFORD MD
2	667	WIGAN MD
2	668	BLACKBURN CD
2	669	BLACKPOOL CD
2	670	BURNLEY CD
2	671	CHORLEY CD
2	672	FYLDE CD
2	673	HYNDBURN CD
2	674	LANCASTER CD
2	675	PENDLE CD
2	676	PRESTON CD
2	767	RIBBLE VALLEY CD
2	768	ROSSENDALE CD
2	769	SOUTH RIBBLE CD
2	770	WEST LANCASHIRE CD
2	771	WYRE CD

32	473	KNOWSLEY MD
32	474	LIVERPOOL MD
32	475	ST HELENS MD
32	476	SEFTON MD
32	567	WIRRAL MD

4	47	SHEFFIELD MD
4	444	KINGSTON-U-HULL UA
4	445	EAST RIDING OF YORKSHIRE UA
4	446	NORTH EAST LINCOLNSHIRE UA
4	537	NORTH LINCOLNSHIRE UA
4	538	CRAVEN CD
4	539	HAMBLETON CD
4	540	HARROGATE CD
4	541	RICHMONDSHIRE CD
4	542	RYEDAILE CD
4	543	SCARBOROUGH CD
4	544	SELBY CD
4	545	YORK UA
4	546	BRADFORD MD
4	637	CALDERDALE MD
4	638	KIRKLEES MD
4	639	LEEDS MD
4	640	WAKEFIELD MD
4	954	BARNESLEY MD

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

4	955	DONCASTER MD
4	956	ROTHERHAM MD
5	566	CORBY CD
5	641	AMBER VALLEY CD
5	642	BOLSOVER CD
5	643	CHESTERFIELD CD
5	644	DERBY CD
5	645	EREWASH CD
5	646	HIGH PEAK CD
5	657	DAVENTRY CD
5	658	EAST NORTHANTS CD
5	659	KETTERING CD
5	660	NORTHAMPTON CD
5	661	SOUTH NORTHANTS CD
5	662	WELLINGBOROUGH CD
5	737	NORTH EAST DERBYS CD
5	738	SOUTH DERBYSHIRE CD
5	739	WEST DERBYSHIRE CD
5	740	BLABY CD
5	741	HINCKLEY/BOSWORTH CD
5	742	CHARNWOOD CD
5	743	HARBOROUGH CD
5	744	LEICESTER CD
5	745	MELTON CD
5	746	NORTH WEST LEICS CD
5	837	OADBY & WIGSTON CD
5	838	RUTLAND CD
5	839	BOSTON CD
5	840	EAST LINDSEY CD
5	841	LINCOLN CD
5	842	NORTH KESTEVEN CD
5	843	SOUTH HOLLAND CD
5	844	SOUTH KESTEVEN CD
5	845	WEST LINDSEY CD
5	846	ASHFIELD CD
5	947	BASSETLAW CD
5	948	BROXTOWE CD
5	949	GEDLING CD
5	950	MANSFIELD CD
5	951	NEWARK CD
5	952	NOTTINGHAM CD
5	953	RUSHCLIFFE CD
6	69	BROMSGROVE CD
6	70	HEREFORD CD
6	71	LEOMINSTER CD
6	72	MALVERN HILLS CD

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

6	73	REDDITCH CD
6	74	SOUTH HEREFORDS CD
6	75	WORCESTER CD
6	76	WYCHAVON CD
6	167	WYRE FOREST CD
6	168	BRIDGNORTH CD
6	169	NORTH SHROPSHIRE CD
6	170	OSWESTRY CD
6	171	SHREWSBURY/ATCHAM CD
6	172	SOUTH SHROPSHIRE CD
6	173	THE WREKIN CD
6	174	CANNOCK CHASE CD
6	175	EAST STAFFS CD
6	176	LICHFIELD CD
6	267	NEWCASTLE-U-LYME CD
6	268	SOUTH STAFFS CD
6	269	STAFFORD CD
6	270	STAFFS MOORLANDS CD
6	271	STOKE ON TRENT CD
6	272	TAMWORTH CD
6	273	NORTH WARWICKS CD
6	274	NUNEATON/BEDWORTH CD
6	275	RUGBY CD
6	276	STRATFORD ON AVON CD
6	367	WARWICK CD
6	368	BIRMINGHAM MD
6	369	COVENTRY MD
6	370	DUDLEY MD
6	371	SANDWELL MD
6	372	SOLIHULL MD
6	373	WALSALL MD
6	374	WOLVERHAMPTON MD
7	48	CAMBRIDGE CD
7	49	EAST CAMBS CD
7	50	FENLAND CD
7	51	HUNTINGDON CD
7	52	PETERBOROUGH CD
7	53	SOUTH CAMBS CD
7	54	BRECKLAND CD
7	55	BROADLAND CD
7	56	GREAT YARMOUTH CD
7	147	NORWICH CD
7	148	NORTH NORFOLK CD
7	149	SOUTH NORFOLK CD
7	150	WEST NORFOLK CD
7	151	BABERGH CD
7	152	FOREST HEATH CD

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

7	153	IPSWICH CD
7	154	MID SUFFOLK CD
7	155	ST EDMUNDSBURY CD
7	156	SUFFOLK COASTAL CD
7	247	WAVENEY CD
7	248	NORTH BEDS CD
7	249	LUTON CD
7	250	MID BEDFORDSHIRE CD
7	251	SOUTH BEDS CD
7	252	BROXBOURNE CD
7	253	DACORUM CD
7	254	EAST HERTS CD
7	255	HERTSMERE CD
7	256	NORTH HERTS CD
7	347	ST ALBANS CD
7	348	STEVENAGE CD
7	349	THREE RIVERS CD
7	350	WATFORD CD
7	351	WELWYN HATFIELD CD
7	451	BASILDON CD
7	452	BRAINTREE CD
7	453	BRENTWOOD CD
7	454	CASTLE POINT CD
7	455	CHELMSFORD CD
7	456	COLCHESTER CD
7	547	EPPING FOREST CD
7	548	HARLOW CD
7	549	MALDON CD
7	550	ROCHFORD CD
7	551	SOUTHEND ON SEA CD
7	552	TENDRING CD
7	553	THURROCK CD
7	554	UTTLESFORD CD
8	162	CROYDON LB
8	163	KINGSTON-U-THAMES LB
8	164	RICHMOND-U-THAMES LB
8	165	MERTON LB
8	166	SUTTON LB
8	257	WANDSWORTH LB
8	352	BARNET LB
8	353	BRENT LB
8	354	HARROW LB
8	355	EALING LB
8	356	HAMMERSMITH LB
8	447	HOUNSLOW LB
8	448	HILLINGDON LB
8	449	KENSINGTON LB

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

8	450	WESTMINSTER LB
8	555	BARKING/DAGENHAM LB
8	556	HAVERING LB
8	647	CAMDEN LB
8	648	ISLINGTON LB
8	649	CITY OF LONDON LB
8	650	HACKNEY LB
8	651	NEWHAM LB
8	652	TOWER HAMLETS LB
8	653	ENFIELD LB
8	654	HARINGEY LB
8	655	REDBRIDGE LB
8	656	WALTHAM FOREST LB
8	958	BEXLEY LB
8	959	GREENWICH LB
8	960	BROMLEY LB
8	961	LAMBETH LB
8	962	LEWISHAM LB
8	963	SOUTHWARK LB
9	57	MOLE VALLEY CD
9	58	REIGATE/BANSTEAD CD
9	59	RUNNYMEDE CD
9	60	SPELTHORNE CD
9	61	SURREY HEATH CD
9	62	TANDRIDGE CD
9	63	WAVERLEY CD
9	64	WOKING CD
9	65	ADUR CD
9	66	ARUN CD
9	157	CHICHESTER CD
9	158	CRAWLEY CD
9	159	HORSHAM CD
9	160	MID SUSSEX CD
9	161	WORTHING CD
9	266	BASINGSTOKE/DEANE CD
9	357	EASTLEIGH CD
9	358	FAREHAM CD
9	359	GOSPORT CD
9	360	HART CD
9	361	HAVANT CD
9	362	NEW FOREST CD
9	363	EAST HAMPSHIRE CD
9	364	PORTSMOUTH CD
9	365	RUSHMOOR CD
9	366	SOUTHAMPTON CD
9	457	TEST VALLEY CD
9	458	WINCHESTER CD

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

9	459	ISLE OF WIGHT UA
9	465	BRACKNELL CD
9	466	NEWBURY CD
9	557	READING CD
9	558	SLOUGH CD
9	559	WINDSOR/MAIDENH'D CD
9	560	WOKINGHAM CD
9	561	AYLESBURY VALE CD
9	562	SOUTH BUCKS CD
9	563	CHILTERN CD
9	564	MILTON KEYNES CD
9	565	WYCOMBE CD
9	663	CHERWELL CD
9	664	OXFORD CD
9	665	VALE WHITE HORSE CD
9	666	SOUTH OXFORDSHIRE CD
9	747	BRIGHTON CD
9	748	EASTBOURNE CD
9	749	HASTINGS CD
9	750	HOVE CD
9	751	LEWES CD
9	752	ROTHER CD
9	753	WEALDEN CD
9	754	ASHFORD CD
9	755	CANTERBURY CD
9	756	DARTFORD CD
9	757	WEST OXFORDSHIRE CD
9	847	DOVER CD
9	848	GILLINGHAM CD
9	849	GRAVESHAM CD
9	850	MAIDSTONE CD
9	851	ROCHESTER CD
9	852	SEVENOAKS CD
9	853	SHEPWAY CD
9	854	SWALE CD
9	855	THANET CD
9	856	TONBRIDGE/MALLING CD
9	957	TUNBRIDGE WELLS CD
9	964	ELMBRIDGE CD
9	965	EPSOM AND EWELL CD
9	966	GUILDFORD CD
10	67	WEST SOMERSET CD
10	68	YEOVIL CD (aka SOUTH SOMERSET)
10	258	BOURNEMOUTH CD
10	259	CHRISTCHURCH CD
10	260	NORTH DORSET CD
10	261	POOLE CD

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

10	262	PURBECK CD
10	263	WEST DORSET CD
10	264	WEYMOUTH/PORTLAND CD
10	265	WIMBORNE CD (aka EAST DORSET)
10	460	KENNET CD
10	461	NORTH WILTSHIRE CD
10	462	SALISBURY CD
10	463	THAMESDOWN CD
10	464	WEST WILTSHIRE CD
10	758	BATH & NORTH EAST SOMERSET UA
10	759	BRISTOL UA
10	760	NORTH SOMERSET UA
10	761	SOUTH GLOUCESTERSHIRE UA
10	762	CARADON CD
10	763	CARRICK CD
10	764	KERRIER CD
10	765	NORTH CORNWALL CD
10	766	PENWITH CD
10	857	RESTORMEL CD
10	858	EAST DEVON CD
10	859	EXETER CD
10	860	NORTH DEVON CD
10	861	PLYMOUTH CD
10	862	SOUTH HAMS CD
10	863	TEIGNBRIDGE CD
10	864	MID DEVON CD
10	865	TORBAY CD
10	866	TORRIDGE CD
10	967	WEST DEVON CD
10	968	CHELTENHAM CD
10	969	COTSWOLD CD
10	970	FOREST OF DEAN CD
10	971	GLOUCESTER CD
10	972	STROUD CD
10	973	TEWKESBURY CD
10	974	MENDIP CD
10	975	SEDGEMOOR CD
10	976	TAUNTON DEANE CD
11	184	ISLE OF ANGLESEY UA
11	185	GWYNEDD UA
11	186	CONWY UA
11	277	DENBIGHSHIRE UA
11	278	FLINTSHIRE UA
11	279	WREXHAM UA
11	280	POWYS UA
11	281	CEREDIGION UA
11	282	PEMBROKESHIRE UA

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

11	283	CARMARTHENSHIRE UA
11	284	SWANSEA UA
11	285	NEATH PORT TALBOT UA
11	286	BRIDGEND UA
11	377	VALE OF GLAMORGAN UA
11	378	RHONDDA, CYNON, TAFF UA
11	379	MERTHYR TYDFIL UA
11	380	CAERPHILLY UA
11	381	BLAENAU GWENT UA
11	382	TORFAEN UA
11	383	MONMOUTHSHIRE UA
11	384	NEWPORT UA
11	385	CARDIFF UA
12	194	ABERDEEN CITY UA
12	195	ABERDEENSHIRE UA
12	196	ANGUS UA
12	287	ARGYLL AND BUTE UA
12	288	SCOTTISH BORDERS UA
12	289	CLACKMANNANSHIRE UA
12	290	WEST DUNBARTONSHIRE UA
12	291	DUMFRIES AND GALLOWAY UA
12	292	DUNDEE CITY UA
12	293	EAST AYRSHIRE UA
12	294	EAST DUNBARTONSHIRE UA
12	295	EAST LoTHIAN UA
12	296	EAST RENFREWSHIRE UA
12	387	EDINBURGH, CITY OF UA
12	388	FALKIRK UA
12	389	FIFE UA
12	390	GLASGOW, CITY OF UA
12	391	HIGHLAND UA
12	392	INVERCLYDE UA
12	393	MIDLoTHIAN UA
12	394	MORAY UA
12	395	NORTH AYRSHIRE UA
12	396	NORTH LANARKSHIRE UA
12	487	PERTH AND KINROSS UA
12	488	RENFREWSHIRE UA
12	489	SOUTH AYRSHIRE UA
12	490	SOUTH LANARKSHIRE UA
12	491	STIRLING UA
12	492	WEST LoTHIAN UA

 --

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 : ASD6A

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

Issued : 12 May 2003

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)

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

- 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~~ **four** 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)**

~~CVHB=1 (adult record: lodgers in BUs>1 in receipt of HB) where HHSTAT=1 (conventional household)~~
- iii ~~SCVHB=1 (adult record: adults in BUs>1 in receipt of HB) where HHSTAT=2 (shared household)~~

IS receipt is identified where any person in the benefit unit answers "yes" to BEN3Q02 (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 BENUNIT>1, for each adult where CONVBL=1 or 2, CVHB=2 (HB not in receipt for boarders or lodgers)-~~
 ~~or if BENUNIT>1 and HHSTAT=2, for each adult, SCVHB=2 (HB not in receipt for adults in shared household benefit units)~~
 or if HHSTAT=2 or CVPAY>0 and for each adult H BOTHBU=2

 --

and for each adult in benefit unit, BEN3Q02<>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 BENUNIT>1, for each adult where CONVBL=1 or 2, CVHB=2 (HB not in receipt for boarders or lodgers)~~
 - ~~or if BENUNIT>1 and HHSTAT=2, for each adult, SCVHB=2 (HB not in receipt for adults in shared household benefit units)~~
 - or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2**
 and for each adult in benefit unit, BEN3Q02<>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 BENUNIT>1, for at least one adult where CONVBL=1 or 2, CVHB=1 (HB in receipt for boarders or lodgers)~~
 - ~~or if BENUNIT>1 and HHSTAT=2, for at least one adult SCVHB=1 (HB in receipt for adults in shared household benefit units)~~
 - or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1**
 and for each adult in benefit unit, BEN3Q02<>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 BENUNIT>1, for each adult where CONVBL=1 or 2, CVHB=2 (HB not in receipt for boarders or lodgers)~~
 - ~~or if BENUNIT>1 and HHSTAT=2, for each adult, SCVHB=2 (HB not in receipt for adults in shared household benefit units)~~
 - or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2**
 and for at least one adult in benefit unit, BEN3Q02=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 BENUNIT>1, for at least one adult where CONVBL=1 or 2, CVHB=1 (HB in receipt for boarders or lodgers)~~
 - ~~or if BENUNIT>1 and HHSTAT=2, for at least one adult SCVHB=1 (HB in receipt for adults in shared household benefit units)~~
 - or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1**
 and for each adult in benefit unit, BEN3Q02<>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 BENUNIT>1, for at least one adult where CONVBL=1 or 2, CVHB=1 (HB in receipt for boarders or lodgers)~~
 - ~~or if BENUNIT>1 and HHSTAT=2, for at least one adult SCVHB=1 (HB in receipt for adults in shared household benefit units)~~
 - or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1**
 and for at least one adult in benefit unit, BEN3Q02=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 BENUNIT>1, for each adult where CONVBL=1 or 2, CVHB=2 (HB not in receipt for boarders or lodgers)~~
~~or if BENUNIT>1 and HHSTAT=2, for each adult, SCVHB=2 (HB not in receipt for adults in shared household benefit units)~~
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
 and for at least one adult in benefit unit, BEN3Q02=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 HBOTHBU=1)
 and for at least one adult in benefit unit, BEN3Q02=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 : ISRECDHH FCRECDHH

Children :

Parents :

Core variable/user : ASD6A

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

Issued : 12 May 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 ~~four~~ **two** separate questions depending on the type of household:

- i HBENEFIT = 1 (has received HB in connection with last rent payment) (household record).
 - ~~ii CVHB=1 (adult record: lodgers in BUs>1 in receipt of HB) where HHSTAT=1 (conventional household)~~
 - ~~iii SCVHB=1 (adult record: adults in BUs>1 in receipt of HB) where HHSTAT=2 (shared household)~~
 - 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 BEN3Q02 (ADULT record - are you at present receiving IS)

2 FRS Specification

Code Condition

HBINDHH

- 0 If CTREB=2
and HBENEFIT=2
~~or for each relevant adult in the household (in BUs>1 where CONVBL=1 or 2), CVHB=2~~
~~or for each relevant adult in the household (in BUs>1 and HHSTAT=2), SCVHB=2~~
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
and for each adult in household, BEN3Q02<>1
- 1 If CTREB=1
and HBENEFIT=2
~~or for each relevant adult in the household (in BUs>1 where CONVBL=1 or 2), CVHB=2~~
~~or for each relevant adult in the household (in BUs>1 and HHSTAT=2), SCVHB=2~~
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
and for each adult in household, BEN3Q02<>1
- 2 If CTREB=2
and HBENEFIT=1

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

~~or for at least one relevant adult in the household (in BUs>1 where CONVBL=1 or 2),
 CVHB=1~~
~~or for at least one relevant adult in the household (in BUs>1 and HHSTAT=2) where
 SCVHB=1~~
or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 and for each adult in household, BEN3Q02<>1

3 If CTREB=2
 and HBENEFIT=2
~~or for each relevant adult in the household (in BUs>1 where CONVBL=1 or 2), CVHB=2~~
~~or for each relevant adult in the household (in BUs>1 and HHSTAT=2), SCVHB=2~~
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
 and for at least one adult in household, BEN3Q02=1

4 If CTREB=1
 and HBENEFIT=1
~~or for at least one relevant adult in the household (in BUs>1 where CONVBL=1 or 2),
 CVHB=1~~
~~or for at least one relevant adult in the household (in BUs>1 and HHSTAT=2) where
 SCVHB=1~~
or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 and for each adult in household, BEN3Q02<>1

5 If CTREB=2
 and HBENEFIT=1
~~or for at least one relevant adult in the household (in BUs>1 where CONVBL=1 or 2),
 CVHB=1~~
~~or for at least one relevant adult in the household (in BUs>1 and HHSTAT=2) where
 SCVHB=1~~
or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 and for at least one adult in household, BEN3Q02=1

6 If CTREB=1
 and HBENEFIT=2
~~or for each relevant adult in the household (in BUs>1 where CONVBL=1 or 2), CVHB=2~~
~~or for each relevant adult in the household (in BUs>1 and HHSTAT=2), SCVHB=2~~
or if HHSTAT=2 or CVPAY>0 and for each adult HBOTHBU=2
 and for at least one adult in household, BEN3Q02=1

7 If CTREB=1
 and HBENEFIT=1
~~or for at least one relevant adult in the household (in BUs>1 where CONVBL=1 or 2),
 CVHB=1~~
~~or for at least one relevant adult in the household (in BUs>1 and HHSTAT=2) where
 SCVHB=1~~
or if HHSTAT=2 or CVPAY>0 and for at least one adult HBOTHBU=1
 and for at least one adult in household, BEN3Q02=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,superann

Children :

Parents :

Core variable/user : HBM

Issue date : 12 May 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

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 42 or 43 (less than a week,**
 lumpsum/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 42 or 43** 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- 14 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 12 OR 13 .

 --

HDAGE

Purpose :To create a variable for use in hotdecking 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 : Hotdecking
 Amendments : SG - 27 June 1997 - no updates required for V33
 Issue date :12 May 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 \geq 16 and AGE \leq 24
- 2 From ADULT table
- If AGE \geq 25 and AGE \leq 34
- 3 From ADULT table
- If AGE \geq 35 and AGE \leq 44
- 4 From ADULT table
- If AGE \geq 45 and AGE \leq 54
- 5 From ADULT table
- If AGE \geq 55 and AGE \leq 64
- 6 From ADULT table
- If AGE \geq 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 hotdecking 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 : Hotdecking
 Amendments : SG - 27 June 1997 - no updates required for V33
 Issue date :12 May 2003

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 hotdecking.

2 FRS Specification

For each child

<u>Code</u>	<u>Condition</u>
1	From CHILD table If AGE<=4
2	From CHILD table

--

- 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 hotdecking 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 : Hotdecking
 Amendments : SG - 27th June - No updates required for V33
 Issue date : 12 May, 2003

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

<u>Code</u>	<u>Condition</u>
1	From ADULT table If INIRBEN>=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 : Hotdecking
Amendments
Issue date :12 May 2003

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

 --

HDPAY

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 : Hotdecking
 Amendments :
 Issue date :12 May 2003

1 Definition

This variable 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.

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

 --

HDPEN

Purpose :To create a variable for use in hotdecking 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 : Hotdecking
 Amendments : : SG - 17 February 1998 - no updates required for V33
 Issue date :12 May 2003

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

Code Condition

- | | |
|----|---------------------------------------------------------|
| 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

 --

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
 Minimum Value : 0
 Maximum Value: 14
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : Hotdecking
 Amendments :
 Issue date :12 May 2003

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 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

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

--

- 9 From MORTGAGE table
If PURCAMT \geq 80000 and PURCAMT \leq 89999.99
- 10 From MORTGAGE table
If PURCAMT \geq 90000 and PURCAMT \leq 99999.99
- 11 From MORTGAGE table
If PURCAMT \geq 100000 and PURCAMT \leq 124999.99
- 12 From MORTGAGE table
If PURCAMT \geq 125000 and PURCAMT \leq 149999.99
- 13 From MORTGAGE table
If PURCAMT \geq 150000 and PURCAMT \leq 199999.99
- 14 From MORTGAGE table
If PURCAMT \geq 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 hotdecking 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 : Hotdecking
 Amendments :
 Issue date :12 May 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

<u>Code</u>	<u>Condition</u>
1	From RENTER table
	If RENT=0

--

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

- 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 : Publication
Issue date : 12 May 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

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 variables DEPCHLDH (number of dependent children in HH), ADULTH (number of adults in HH), 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 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=0

- 1 ADULTH=1 and (SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)
- 2 ADULTH=1 and (SEX=1 and AGE<65) or (SEX=2 and AGE<60)
- 3 ADULTH=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=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=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>=3
- 7 ADULTH=1 and DEPCHLDH=1
- 8 ADULTH=1 and DEPCHLDH=2
- 9 ADULTH=1 and DEPCHLDH >=3
- 10 ADULTH=2 and DEPCHLDH=1
- 11 ADULTH=2 and DEPCHLDH=2
- 12 ADULTH=2 and DEPCHLDH>=3
- 13 ADULTH>=3 and DEPCHLDH=1
- 14 ADULTH>=3 and DEPCHLDH=2
- 15 ADULTH>=3 and DEPCHLDH >=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 HHCOMP 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 : Publication
 Issue date : 12 May 2003
 Amendments :

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 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, HHCOMPS 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

HHCOMPS is derived by looking at variables DEPCHLDH (number of dependent children in HH), ADULTH (number of adults in HH), 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 HHCOMPS 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 1-6 all where DEPCHLDH=0

- 1 ADULTH=1 and (SEX=1 and AGE>=65)
- 2 ADULTH=1 and (SEX=2 and AGE>=60)
- 3 ADULTH=1 and (SEX=1 and AGE<65)
- 4 ADULTH=1 and (SEX=2 and AGE<60)
- 5 ADULTH=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=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=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>=3
- 9 ADULTH=1 and DEPCHLDH=1
- 10 ADULTH=1 and DEPCHLDH=2
- 11 ADULTH=1 and DEPCHLDH >=3
- 12 ADULTH=2 and DEPCHLDH=1
- 13 ADULTH=2 and DEPCHLDH=2
- 14 ADULTH=2 and DEPCHLDH>=3
- 15 ADULTH>=3 and DEPCHLDH=1
- 16 ADULTH>=3 and DEPCHLDH=2
- 17 ADULTH>=3 and DEPCHLDH >=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)

 --

HHINC

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 : BUINC
 Core variable/user : ASD4A
 Amendments : VE - 4 June 1996 - No initial amendments needed for V32 update
 : SG - 27 June 1997 - No initial amendments needed for V33 update

1 Definition

This variable is coded as

HHINC The total amount of income received each week by all members of the benefit unit.
 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

 --

HHRENT

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.

Created : 26 January 1993

Database Table : HOUSEHOL

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables :

 Children : BURENT, **WATSEWRT**

 Parents :

Core variable/user : **HBM PSM (BOLD indicates lead user)**

Issue date : 12 May, 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

 --

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), ~~CWATAMT (amount of community water charge), COMMINC (any council tax included in rent y/n), COMMANT (amount of council tax included),~~ 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 SERVAMT (amount of services).

First omit owner-occupiers, boarders **and** lodgers ~~and rent free~~. 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 ~~council tax, community water rates and~~ 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 ~~12 or 13~~ **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 RENTPD, COMPD, HBENPD, ACCPD or ~~CWATAMT = 12 or 13~~ **90, 95 or 97** HHRENT is set to -2. Since 94/5 all code 12 are edited so this problem should be reduced.

2 FRS Specification

For each household

<u>Code</u>	<u>Condition</u>
HHRENT	<p>From HOUSEHOL table, if TENURE = 2 or 3 or 4 or 5 or 6 (renting or rent free) get HHRENT and HHSTAT variables (Drop tenure=6 rent free have non-zero)</p> <p>From RENTER table, if RENTPD equals -1 or 1-11, 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 1-11, 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 = -2 1 (2 or -1) and ACCPD equals -1 or 1-11, 13, 26 or 52 add ACCAMT to rent to get HHRENT (Someone outside of HH pays part of rent and original rent given after 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.</p> <p>If CWATAMTD exists and RENTHOL=1 and WEEKHOL exists adjust CWATAMTD as above and deduct adjusted CWATAMTD from HHRENT to get HHRENT (community water charge). Else if CWATAMTD exists and no rent holiday deduct from HHRENT if not, do not change HHRENT.</p> <p>If COMMINC = 1 and COMPD equals -1 or 1-11, and RENTHOL=1 and WEEKHOL exists adjust COMMAMT as above and deduct adjusted COMMAMT to get HHRENT.</p>

 --

~~(council tax). Else if COMMINC=1 and COMPD equals -1 or 1-11 and no rent holiday deduct COMMAMT to get HHRENT. If COMMINC = 2, do not change HHRENT.~~

If SERINC = 1 and RENTHOL=1 and WEEKHOL exists adjust SERVAMT as above and , deduct adjusted SERVAMT to get HHRENT (other services). Else if SERINC=1 and no rent holiday, deduct SERVAMT to get HHRENT. If SERINC = 2, do not change HHRENT.

~~If WATERINC = 1 or SEWERINC~~ **WORSINC = 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 WATERINC=2 and SEWERINC~~ **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=-1 and HBENPD equals -1 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 -1 or 1-11, 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 ~~SCHBPD~~ **HBOTHPD** equals -1 or 1-11 or 13, 26 or 52 and ~~SCHBAMT~~ **HBOTHAMT** exists add ~~SCHBAMT~~ **HBOTHAMT** to HHRENT.

~~If HBENEFIT = 1 and HBSTMT = 1 and WORSINC = 1 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.~~

If HBENEFIT = 1 and HBSTMT = 1 and ~~HBSERAMT~~ **HBSERA01>0** or ~~HBSERA02~~ **HBSERA02>0** or ~~HBSERA03~~ **HBSERA03>0** or ~~HBSERA04~~ **HBSERA04>0** or ~~HBSERA05~~ **HBSERA05>0** or ~~HBSERA06~~ **HBSERA06>0** or ~~HBSERA07~~ **HBSERA07>0** or ~~HBSERA08~~ **HBSERA08>0** or ~~HBSERA09~~ **HBSERA09>0** or ~~HBSERA10~~ **HBSERA10>0** or ~~HBSERA11~~ **HBSERA11>0** or ~~HBSOAMT~~ **HBSOAMT >0** and RENTHOL = 1 and WEEKHOL exists, adjust ~~HBSERAMT~~ **HBSERAMT** and ~~HBSOAMT~~ **HBSOAMT** as above and deduct ~~HBSERAMT~~ **HBSERAMT** and ~~HBSOAMT~~ **HBSOAMT** from HHRENT; else if no rent holiday deduct unadjusted ~~HBSERAMT~~ **HBSERAMT** and ~~HBSOAMT~~ **HBSOAMT** from HHRENT.

 --

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 : ISM
 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
 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 for the disability trailer
 Created : 6 February 1997
 Database Table : ADULT
 Minimum Value :
 Maximum Value :
 Units : integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ASD6B2
 Amendments :
 Issue date : 12 May, 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. HOUR01-14 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=00-14).

$$\text{HOURCARE} = \text{HOUR01} + \text{HOUR01} + \dots + \text{HOUR13} + \text{HOUR14}$$

- 2 If any variables are missing

 --

HPERSON

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
 Issued : 12 May 2003

1 Definition

This variable assigns the value 1 to the first person in the first Benefit Unit in the Household and increments by one for each adult and each child, by Benefit Unit.

2 FRS Specification

Process each Benefit Unit in the household in turn, incrementing HPERSON as shown.

Code Condition

- 1 If BENUNIT = 1 and PERSON = 1
- +1 In the following priority:
- +1 If BENUNIT = 1, and PERSON = 2,
- +1 If BENUNIT = 1, 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 : Publication
 Amendments : VC - 12 October 1993 To emphasize 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
 Issued : 12 May 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 ~~be adjusted for rent free holidays and~~ also be added into HSCOSTHH and is found in the DV CWATAMTD.

Finally any other housing costs have to 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, CHAMT2, CHAMT3, CHAMT4, CHAMT5, CHAMT6 and CHAMT7 and **CHAMT8**.

However, if the period code for **charges** is ~~42 or 43~~ **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 ~~CHARGE PD1 = 42 or 43~~ **CHARGE PD1 = 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 ~~42 or 43~~ 90,95 or 97.

 --

For each HOUSEHOLD, set HSCOSTHH to zero.

Code Condition

HSCOSTHH From HOUSEHOL **and RENTER** records,

 If HHRENT exists and does not equal -1 or -2 (not applicable or unable to derive) add the amount in HHRENT into HSCOSTHH.

 If **RENTHOL=1 and WEEKHOL exists and WATSEWRT exists** and 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.

 If MORTINT exists and does not equal -1 or -2 (as above) add the amount in MORTINT into HSCOSTHH.

 From HOUSEHOL record get STRCOV, STRAMT1, **STRAMT2**, STRPD1, **STRPD2**, STROTHS, **COVOTHS**, TENURE

set STRINTH equal to zero

If STRCOV=1 ~~or STRCOV=3~~ and STRAMT1 exists (structural insurance only ~~or combined with furniture and contents as part of mortgage payment~~) and STRPD1=-1 or ~~4-44 1-10,13,26,52~~ STRINTH=STRAMT1

If STRCOV=3 and STRAMT1 exists (structural insurance combined with furniture and contents as part of mortgage payment) and STRPD1=-1 or 4-44 1-10,13,26,52 STRINTH=STRAMT1*0.6666 (to bring in line with HBAI)

If STROTHS=1 ~~and TENURE=1~~ and **COVOTHS=2 and STRAMT2** exists (insurance premium paid on structure of accommodation separately from any mortgage payments, ~~owner occupiers only~~) and STRPD2=-1 or ~~4-44 1-10,13,26,52~~ STRINTH=STRAMT2*0.6666

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 ~~4-44 1-10,13,26,52~~ STRINTH=STRAMT2

Add STRINTH to HSCOSTHH

CHPD1-78 From ~~owner~~ **HOUSEHOL** record get CHARGE1-~~78~~, CHAMT1-~~78~~, ~~CHARGE2~~

If CHARGE1=1 and ~~CHARGE2~~ **CHPD1=-1**, or ~~4-44 1-10,13,26,52~~ add CHAMT1 to HSCOSTHH

If CHARGE2=1 and ~~CHARGE2~~ **CHPD2=-1**, or ~~4-44 1-10,13,26,52~~ add CHAMT2 to HSCOSTHH

 --

If CHARGE3=1 and ~~CHARGE=~~ ~~CHPD3=-1~~, or ~~1-11~~ **1-10,13,26,52** add
 CHAMT3 to HSCOSTHH

If CHARGE4=1 and ~~CHARGE=~~ ~~CHPD4=-1~~, or ~~1-11~~ **1-10,13,26,52** add
 CHAMT4 to HSCOSTHH

If CHARGE5=1 and ~~CHARGE=~~ ~~CHPD5=-1~~, or ~~1-11~~ **1-10,13,26,52** add
 CHAMT5 to HSCOSTHH

If CHARGE6=1 and ~~CHARGE=~~ ~~CHPD6=-1~~, or ~~1-11~~ **1-10,13,26,52** add
 CHAMT6 to HSCOSTHH

If CHARGE7=1 and ~~CHARGE=~~ ~~CHPD7=-1~~, or ~~1-11~~ **1-10,13,26,52** add
 CHAMT7 to HSCOSTHH

**If CHARGE8=1 and ~~CHARGE=~~ ~~CHPD8=-1~~, or ~~1-11~~ 1-10,13,26,52 add
 CHAMT8 to HSCOSTHH**

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

If CWATAMTD exists ~~and RENTHOL=1 and WEEKHOL exists adjust CWATAMTD
 as above and add adjusted CWATAMTD to HSCOSTHH. Else 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

 --

INCSE1, INSEINC

Purpose : To calculate the total gross income/earnings from self-employment
 Created : JS - 10 January 1997
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value:
 Units : Real
 Validations :
 Related Variables : INCSE2
 Children :
 Parents :
 Core variable/user : **ASD6E**
 Amendments : SCG 14 November 1997 - add INSEINC
 Issued : 12 May 2003

1 Definition

This variable is coded as

INCSE1 This is the total amount of income received from self-employed earnings gross of tax and national insurance payments.

-1 Not applicable to this case

-2 Unable to derive due to missing values

INSEINC This is same as **INCSE1** except set to zero for non self employed, rather than -1 (for use in **INDINC**)

-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. The definition of **INCSE1** has therefore moved away from the emphasis on profits to look at income coming in to the household.

Drawings from business accounts are assumed to be net of tax and NI. For those quoting income from a job/business, unless tax or NI have been deducted at source, the value quoted is assumed to be gross of NI and tax.

NI and tax payments are estimated using quoted figures on payments made during the last 12 months. This is consistent with the previous definition of the related variable **INCSE2**, but may or may not

 --

actually relate to the income quoted and therefore assumes that the level has not changed over the previous 12 months.

For those who are part of a partnership, Interviewers make clear at the start of the questions that they refer only to the respondent's share.

Although the questions refer to various time periods, all amounts will have been converted to weekly figures as part of the conversion process. (Similarly, all one off/lump sum payments will be weeklised, although the code to set these cases to unable to derive is still included as a safeguard).

The variable 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).

Then for those with separate business accounts (WORKACC=1) and who draw money for non-business purposes (OWNSUM=1), INCSE1 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 INCSE1. NI and tax are then added back.

For those who do not keep business accounts, or without separate business/personal accounts or who have separate accounts but do not draw money from them, INCSE1 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 INCSE1. *(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 INCSE1. Similarly, any tax and NI lump sum payments are also added back.

2 FRS Specification

For each ADULT, get ADULT and JOB records.

Code Condition

INCSE1

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 two temporary variables to zero

xincse1b=0 (for cases with separate business accounts)

xincse1i=0 (for other cases)

for those with separate business accounts who draw money from them

If OWNSUM=1 and OWNAMT exists then $xincse1b = xincse1b + OWNAMT$

If OWNOTHER=1 and OWNOTAMT exists then $xincse1b = xincse1b + OWNOTAMT$

If (SENIREG=1 and SENIRAMT exists and SENIRPD<>90,95,97) then $xincse1b = xincse1b + SENIRAMT$

If SETAX=1 and SETAXAMT exists then $xincse1b = xincse1b + SETAXAMT$

If SENILUMP=1 and SENILAMT exists then $xincse1b = xincse1b + SENILAMT$

for all other cases {possible that with more than one self-employed job, some have separate business accounts whilst others don't, so running total continues for each job record. However, no single job record should have an amount in both OWNAMT and SEINCAMT - this will need to be checked as part of editing/imputation}

If SEINCAMT exists then $xincse1i = xincse1i + SEINCAMT$

If (CHECKTAX=1 and CHKINCOM=2 and TAXDAMT exists and TAXDPD<>90,95,97) then $xincse1i = xincse1i + TAXDAMT$

If (CHECKTAX=2 and CHKINCOM=2 and NIDAMT exists and NIDPD<>90,95,97) then $xincse1i = xincse1i + NIDAMT$

If last.jobtype

output INCSE1=sum(xincse1b,xincse1i) (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

INSEINC

Initialise INSEINC =0.

If working and INCSE1 ^= -1 THEN INSEINC=INCSE1

 --

INCSE2

Purpose : To calculate the total net income/earnings from self-employment
 Created : JS - 10 January 1997
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value:
 Units : Real
 Validations :
 Related Variables : INCSE1
 Children :
 Parents :
 Core variable/user : **HBAI**
 Amendments :
 Issued : 12 May 2003

1 Definition

This variable is coded as

INCSE2 This is the total amount of income received from self-employed earnings net of tax and national insurance payments. For those keeping separate business accounts, this is calculated as the amount drawn from work accounts for personal use, together with any additional cash used. For the remainder, it is average income less tax and NI.

-1 Not applicable to this case

-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. The definition of INCSE2 has therefore moved away from the emphasis on profits to look at income coming in to the household.

Drawings from business accounts are assumed to be net of tax and NI. For those quoting income from a job/business, unless tax or NI have been deducted at source, the value quoted is assumed to be gross of NI and tax.

NI and tax payments are estimated using quoted figures on payments made during the last 12 months. This is consistent with the previous definition of INCSE2, but may or may not actually relate to the income quoted and therefore assumes that the level has not changed over the previous 12 months.

For those who are part of a partnership, Interviewers make clear at the start of the questions that they refer only to the respondent's share.

 --

Although the questions refer to various time periods, all amounts will have been converted to weekly figures as part of the conversion process. (Similarly, all one off/lump sum payments will be weeklised, although the code to set these cases to unable to derive is still included as a safeguard).

The variable 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).

Then for those with separate business accounts (WORKACC=1) and who draw money for non-business purposes (OWNSUM=1), INCSE2 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 INCSE2.

For those who do not keep business accounts, or without separate business/personal accounts or who have separate accounts but do not draw money from them, INCSE2 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 deducted from INCSE2. *(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.)*

If no NI has been deducted at source, respondents are asked whether they make any regular contribution (SENIREG), if this is yes, the amount at SENIRAMT is deducted from INCSE2. Respondents are then asked at SETAX whether they make any other tax payments (even if tax has been deducted at source). If yes, the amount at SETAXAMT is also deducted. This may also include some Class 4 NI contribution (SENIINC=1), but where it does not and lump sum contributions have been made (SENILUMP=1) , these (SENILAMT) are also deducted from INCSE2.

2 FRS Specification

For each ADULT, get ADULT and JOB records.

Code Condition

INCSE2

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

If first.jobtype

 If EMPEE = 2 (self-employed)

 --

To avoid subtracting tax and NI payments twice from drawings from business cases, set two temporary variables to zero:

xincse2b=0 (for cases with separate business accounts)
 xincse2i=0 (for other cases)

for those with separate business accounts who draw money from them

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

for all other cases {possible that with more than one self-employed job, some have separate business accounts whilst others don't, so running total continues for each job record. However, no single job record should have an amount in both OWNAMT and SEINCAMT - this will need to be checked as part of editing/imputation}

If SEINCAMT exists then xincse2i=xincse2i+SEINCAMT

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 SETAX=1 and SETAXAMT exists then xincse2i=xincse2i-SETAXAMT

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

If last.jobtype

output INCSE2=sum(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

 --

INCSEO1, SEINCAM1, NINCSEO1, NINSEIN1

Purpose : To calculate the total income/earnings from self-employment - Option 1 based on profit where available and then income/drawings, GROSS and NET versions
 Created : SG - 31 March 1998
 Database Table : ADULT
 Minimum Value : 0
 Maximum Value :
 Units : Real
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : **ASD6E**
 Amendments :
 Issued : 12 May 2003

1 Definition

This variable is coded as

INCSEO1 This is the total amount of income received from self-employment GROSS of tax and national insurance payments, based on profits where available, or on estimated earnings/drawings otherwise.

-1 Not applicable to this case

-2 Unable to derive due to missing values

SEINCAM1 This is same as INCSEO1 except set to zero for non self employed, rather than -1 (for use in INDINC)

-2 Unable to derive due to missing values

NINCSEO1 This is the total amount of income received from self-employment NET of tax and national insurance payments, based on profits where available, or on estimated earnings/drawings otherwise.

-1 Not applicable to this case

-2 Unable to derive due to missing values

NINSEIN1 This is same as NINCSEO1 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. 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).

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 profit data

The profit data is used where it is available. 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.

No Profits data

 --

Where self-employed respondents have not provided information on profits then SEINCAMT (income from job/business after paying for any materials, equipment or goods used for work) is used if available. 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). To obtain gross profit, where appropriate amounts in TAXDAMT and NIDAMT are added back in to INCSEO1. *(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 INCSEO1. Similarly, any tax and NI lump sum payments are also added back.

For those who do not answer the profit or income questions information on drawings from business accounts for non-business purposes will be used. INCSEO1 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 INCSEO1. 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

 xincse1p=0 {fpr profit cases}

 xincse1b=0 (for cases with separate business accounts)

 xincse1i=0 (for other cases)

 If PROFIT1 exists and PROFIT2=1 then

 if PROFIT2=2 (a loss) then UPROFIT= -PROFIT1

 else UPROFIT = PROFIT1

 {{ FOR GROSS VERSION }}

 --

if PRBEFORE>0 then UPROFIT=PRBEFORE
 }

Calculate uprating factor.

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 1988 then SE2=March 1988, SE1=April 1987

Calculate PERIOD referred to by SE1 and SE2 = (SE2 - SE1)/30.416666 and then round to the nearest month (using RND function). 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 SENILAMT 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 xincse1p = UPROFIT + xincse1p

Else If SEINCAMT exists then

 --

xincse1i=xincse1i+SEINCAMT

{{ FOR GROSS VERSION }

If (CHECKTAX=1 and CHKINCOM=2 and TAXDAMT exists and TAXDPD<>90,95,97)
 then xincse1i=xincse1i+TAXDAMT

If (CHECKTAX=2 and CHKINCOM=2 and NIDAMT exists and NIDPD<>90,95,97)
 then xincse1i=xincse1i+NIDAMT

}

{{ FOR NET VERSION }

If (CHECKTAX=1 and CHKINCOM=1 and TAXDAMT exists and TAXDPD<>90,95,97)
 then xincse1i=xincse1i-TAXDAMT

If (CHECKTAX=2 and CHKINCOM=1 and NIDAMT exists and NIDPD<>90,95,97)
 then xincse1i=xincse1i-NIDAMT

If (SENIREG=1 and SENIRAMT exists and SENIRPD<>90,95,97) then
 xincse1i=xincse1i-SENIRAMT

If (SENILUMP=1 and SENILAMT exists) then xincse1i=xincse1i-SENILAMT

If (SETAX=1 and SETAXAMT exists) then xincse1i=xincse1i-SETAXAMT

}

Else {seincamt not collected}

If OWNSUM=1 and OWNAMT exists then xincse1b=xincse1b+OWNAMT

If OWNOTHER=1 and OWNOTAMT exists then xincse1b=xincse1b+OWNOTAMT

{{ FOR GROSS VERSION }

If (SENIREG=1 and SENIRAMT exists and SENIRPD<>90,95,97) then
 xincse1b=xincse1b+SENIRAMT

If SETAX=1 and SETAXAMT exists then xincse1b=xincse1b+SETAXAMT

If SENILUMP=1 and SENILAMT exists then xincse1b=xincse1b+SENILAMT

}

If last.jobtype

output INCSEO2=sum(xincse1p,xincse1b,xincse1i) (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

SEINCAM1

Initialise SEINCAM1 =0.

If working and INCSEO1 ^= -1 THEN SEINCAM1=INCSEO1

NINSEIN1

--

Initialise NINSEIN1 =0.

If working and NINCSEO1 ^= -1 THEN NINSEIN1=INCSEO1

 --

INCSEO2, SEINCAM2, NINCSEO2, NINSEINO2

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 : **ASD6E**
 Amendments :
 Issued : 12 May 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 non self employed, 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)

If PROFIT1 exists and JOBBUS=2 or 3 then

If PROFIT1 exists and PROFIT2=1 then

if PROFIT2=2 (a loss) then UPROFIT= -PROFIT1
 else UPROFIT = 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 1988 then SE2=March 1988, SE1=April 1987

Calculate PERIOD referred to by SE1 and SE2 = (SE2 - SE1)/30.416666 and then round to the nearest month (using RND function). 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 xincse1i=xincse1i+TAXDAMT
        If (CHECKTAX=2 and CHKINCOM=2 and NIDAMT exists and NIDPD<>90,95,97)
        then xincse1i=xincse1i+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 OWNNOTAMT exists then xincse2b=xincse2b+OWNNOTAMT

{{ 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 EMPPEE<>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=INCSEO2

 --

INDINC

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 :

 Children :

 Parents :

Core variable/user :ASD4A

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 INEARNNS 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 INEARNNS 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 – 10 December 1998 – Subtract Soc. Fund Loan: repayment from
 INOTHBEN
Issued : 12 May, 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.

For V32, the specification for INDINC has been adjusted to bring it in line with HBAI definitions.

A new DV, OLDINC, has been set up to maintain the old definition of INDINC to enable comparisons to be made between V31 and V32.

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. Unlike V30 GROSSINC the adjustment for SSP/SMP is only used to avoid double counting. **For V32, the SMP/SSP adjustment causes problems for one case (15325171) where SMP is received on the first job where the income is less than £52.50, but adding on income received from the second job gives a value greater than SMP. Since the adjustment is made to INEARNNS, SMP is deducted for this case, but it should not be. The programmes need to be amended for V33 to ensure the adjustment is done on a job by job basis.** 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** (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, for example, if PAYPD = **90 or 95 or 97** INDINC is set to -2. ~~However, for odd jobs, since by definition they should be irregular, lump sum payments are allowed but are divided by 4 to produce a weekly amount.~~ 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 (some grouped together on V30)

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
- SEINCAM2** ~~INSEINC~~ self employment income (identical to INCSEO2 4—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 IVB

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 - process each JOB record for that person and

If ABSWHY ne 3 (To bring in line with HBAI)

If EMPEE = 1

(PAYUSL, and therefore UGROSS if PAYUSL equals "no" is only asked for JOBTYP=1, therefore coding has been changed so that a check is made to see if UGROSS exists. Where it doesn't, gross pay is calculated dependent on whether a

 --

payslip has been consulted, where it does - which can only be JOBTYP=1 - UGROSS is used)

If UGROSS does not exist

and If PAYSLIP = 1 or 2

If GRWAGE exists and PAYPD equals -1 or 1 to 11 or 13 or 26 or 52, add it into INEARNNS

If it is missing set INDINC and INEARNNS -2

else if PAYSLIP = 3 (or PAYSLIP=1 or 2 and GRSWAGE=-1) and PAYPD equals -1 or 1 to 11 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

(NATINS only asked for JOBTYP=1, do avoid variable falling over at this point, only include PAYE and NATINS if they exist, else do not change INEARNNS)

Then (AMTTAXF only asked if CHARITY=1 and CHRTAXF=1 and only for first job: to avoid setting cases to -2, only add in if value exists as passing through jobtypes 1-3)

if AMTTAXF exists add it into INEARNNS

if AMTOTH exists add it into INEARNNS

(as for AMTTAXF)

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 DEDOTH to INEARNNS

Else if UGROSS exists

If UGROSS exists add UGROSS to INEARNNS

If it is missing do not change INEARNNS (ie use PAYAMT calculation if it exists)

Adjustments to gross earnings for HBAI consistency:

--

income tax refunds, mileage and motoring allowances, refunds for items of household expenditure

if INEARNNS<->-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

else if UGROSS exists

and U2MOT exists INEARNNS=INEARNNS-U2MOT

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)/0.75)

(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
INEARNNS=INEARNNS-(UBONAMT/0.75)

 --

(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**)

Adjustment for possible receipt of SSP or SMP

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 PAYSZIP = 1 **or 2**
 Calculate ADJUST = SSPAMT

If JOBAWAY = 1 and ABSWHY = 6 and (SSPSMP = 1 or 3) and PAYSZIP = 1 **or 2**
 Calculate ADJUST = ADJUST + SMPAMT

If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYSZIP=~~2~~ **3** and
 SSPRATE=1 **or 2** Calculate ADJUST =52.50

~~If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYSZIP=2 and
 SSPRATE=2 Calculate ADJUST=47.80~~

If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSZIP=~~2~~ **3** and
~~SSPRATE=1~~ **SMPRATE=1 or 2** Calculate ADJUST=ADJUST+52.50

~~If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSZIP=2 and
 SSPRATE=2~~ **SMPRATE=2** Calculate ADJUST=ADJUST+48.80

~~If ADJUST >= to employment income calculated in INEARNNS, reset ADJUST to
 INEARNNS~~

If ADJUST>INEARNNS, 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 (amounts are only recorded there if respondents have not answered questions as part of pay). 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). These are standard rates (apart from higher SMP where an estimate has been taken - this will not affect overall income). SSPSMP is coded 1=both SSP and SMP, 2=SSP only, 3=SMP only. Earned income equals amount calculated at INEARN less the estimated SMP/SSP (the adjustment). ~~Where the total adjustment is greater than recorded income, all income is assumed to be from benefit and the adjustment is reset to INEARN (ie INEARN=0).~~ ADJUST is added to INOTHBEN.

Self - employment income: INSEINC

Set INSEINC=zero
 If WORKING = 1 or JOBAWAY = 1 INSEINC=INCSE1

Other sources of income: INRINC

Set INRINC to zero

Income as a baby-sitter

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 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 11 **or 13 or 26 or 52**, add APAMT to INRINC.

If APDIR=1 and APDPD equals -1 or 1 to 11 **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 11 **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 11 **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 11 **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 11 **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*)

~~From ADULT record, if FCASH = 1 **COALCOKE=2** and FCAMTPD equals -1 or 1 to 11 **or 13 or 26 or 52** add amount in FCAMT into INRINC (cash in lieu of concessionary coal)~~

Royalties

From ADULT record, if ROYAL1 = 1 add ROYYR1 into INRINC

Income as a sleeping partner

From ADULT record, if ROYAL2 = 1 add ROYYR2 into INRINC.

Pension from an overseas Government

From ADULT record, if ROYAL3 = 1 add ROYYR3 into INRINC.

Maintenance

From ADULT record, if MNTREC = 1 **and MNTDSS=1 MNTDSS1=1** and MNTPD1 equals -1 or 1 to 11 **or 13 or 26 or 52** ~~and MNTUS1=4~~, add MNTAMT1 into INRINC.

~~Else if MNTREC=1 **and MNTDSS=1 MNTDSS1=1** and MNTUS1=2 and MNTUSPD1 equals -1 or 1 to 1 add MNTUSAMT1 into INRINC~~

Odd jobs

From ADULT record if ODDJOB =1
 for all occurrences of OJAMT in ODDJOB record
~~if OJPD = 1 or 1-11~~, add OJAMT into INRINC
~~else if OJPD=90 or 95 or 97~~ add OJAMT/4 into INRINC
 (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 PERSON = 1 (head of household).

Income from those outside the household paying towards rents/mortgages

For rented property, all contributions are included ~~(including any from DSS other than HB, this is consistent with HHRENT~~ **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 1-11 then INRINC=INRINC+ACCAMT for PERSON=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 PERSON=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) OUTSMT exists and OUTSPD=-1 or 1-11 then INRINC=INRINC+OUTSMT for PERSON=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 ~~(4/3- 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 ~~(4/3- 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 ~~(4/3- 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 ~~(4/3- 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 ~~(4/3- 5/4 x amount in ACCINT)~~ to ININV
if ACCOUNT = 6 and ACCTAX<>1, add amount in ACCINT to ININV
~~if ACCOUNT = 7 and ACCTAX=1, add (4/3 x amount in ACCINT) to ININV~~
~~if ACCOUNT = 7 and ACCTAX<>1, add amount in ACCINT to ININV~~
 if ACCOUNT = 7 and ~~INVTAXACCTAX=1~~, add ~~(4/3- 5/4 x amount in ACCINT)~~ to

ININV

~~if ACCOUNT = 8 and INVTAXACCTAX<>1, add amount in ACCINT to ININV~~
 if ACCOUNT = 8, add ~~(4/3 x 4/3- 5/4 x amount in ACCINT)~~ to ININV
if ACCOUNT = 9, add amount in ACCINT to ININV
~~if ACCOUNT = 10, add (4/3 x amount in ACCINT) to ININV~~
~~if ACCOUNT = 11, add (4/3 x amount in ACCINT) to ININV~~
~~if ACCOUNT = 12, add amount in ACCINT to ININV~~
~~if ACCOUNT = 13, add amount in ACCINT to ININV~~
~~if ACCOUNT = 14, add (4/3 x amount in ACCINT) to ININV~~
~~if ACCOUNT = 15, add amount in ACCINT to ININV.~~

Note : Accounts 7 (Stocks & Shares) and 8 (Unit Trusts) are assumed net of tax, so we do need to add the tax back in. There is no ACCTAX check for these accounts.

Personal pensions: INPENINC

set INPENINC to zero

Occupational pensions

From PENSIONS record, if PENTYPE = 1 (occupational pension) and PENPD equals -1 or 1 to 11 **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 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 11 **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 11 **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 11 **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 11 **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 11 **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 11 or 13 or 26 or 52, add HBENAMT to INOTHBEN.

~~From HOUSEHOL record, if HHSTAT = 1 then from ADULT record if CVHB = 1 and CVPD equals -1 or 1 to 11, and exists(CHBAMT) = 1 then add CHBAMT 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 11 or 13 or 26 or 52, then add HBOTHAMT to INOTHBEN.

~~From HOUSEHOL record and RENTER record, if HHSTAT=2 and RENTHOL = 1 then from ADULT record if SCVHB=1 and SCHBPD equals -1 or 1 to 11 then add SCHBAMT to INOTHBEN.~~

~~From HOUSEHOL record and RENTER record, if HHSTAT = 2 and RENTHOL = 1 and WEEKHOL exists and SCVHB = 1 and SCHBPD equals -1 or 1 to 11, multiply SCHBAMT by (52 - WEEKHOL) and divide the result by 52. Add to INOTHBEN.~~

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to 11 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 11 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 = 4 add BENAMT to INOTHBEN(OPB)

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 = ~~9~~ **10** add BENAMT to INDISBEN (SDA)

If BENEFIT = ~~10-11~~ add BENAMT to INDISBEN (DWA)

 --

If BENEFIT = ~~11~~12 add BENAMT to INDISBEN (AA)

If BENEFIT = ~~12~~ 13 add BENAMT to INOTHBEN(Invalid Care Allowance)

If BENEFIT = ~~13~~ 14 add BENAMT to INOTHBEN(UB/JSA)

If BENEFIT = ~~14~~ 15 add BENAMT to INDISBEN (Industrial Injuries)

If BENEFIT = ~~15~~ 16 add BENAMT to INOTHBEN (SSP)

If BENEFIT = 17 add BENAMT to INDISBEN (~~IVB~~ Incapacity Benefit)

If BENEFIT = 18 add BENAMT to INOTHBEN(FC)

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 = 20 and VAR2=2

and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)) add BENAMT to INRPINC

else if BENEFIT=20 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 = 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 PERSON=1 add BENAMT / 4 to INOTHBEN (Extended HB or extended HB and CTB)

If BENEFIT = 29 and HBINDBU in (0,2,3,5) and PERSON=1 add BENAMT / 4 to INOTHBEN (Extended CTB)

If BENEFIT = 30 add BENAMT to INOTHBEN (Any other DSS benefits)

If BENEFIT = 37 add BENAMT to INOTHBEN (Guardians Allowance)

If BENEFIT = 38 add subtract BENAMT to from INOTHBEN (Social Fund repayment)

(for v33 the question Sflnc, "was the amount 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 = 39 and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)) add BENAMT to INRPINC~~

 --

~~add BENAMT to INOTHBEN (Social Fund-budget)~~

~~If BENEFIT = 40 add BENAMT/26 to INOTHBEN (Social Fund-crisis)~~

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 = ~~35~~ 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 = ~~27~~ 31 and PRES = 1 and BENPD=-1 or 1-11 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 = ~~28~~ 32 and PRES = 1 and BENPD=-1 or 1-11 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 = ~~29~~ 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 = ~~30~~ 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 = ~~34~~ 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

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 INEARNNS components are already documented above).

 --

INIRBEN, INNIRBEN, BUIRBEN, BUNIRBEN, HHIRBEN, HHNIRBEN

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 : 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 - 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 - 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

 : EP - 10 December 1998 - correct benefit 38 so that it is subtracted from INIRBEN

Issued : 12 May, 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**

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 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=~~2~~ **3** and
 SSPRATE=1 Calculate ADJUST =52.50

If JOBAWAY=1 and ABSWHY=2 and (SSPSMP=1 or 2) and PAYS LIP= ~~2~~ **3** and
 SSPRATE=2 Calculate ADJUST=47.80

 --

If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSLIP= ~~2~~ 3 and SSPRATE=1 Calculate ADJUST=ADJUST+52.50

If JOBAWAY=1 and ABSWHY=6 and (SSPSMP=1 or 3) and PAYSLIP= ~~2~~ 3 and SSPRATE=2 Calculate ADJUST=ADJUST+48.80

If ADJUST >= to employment income calculated in INEARNNS, reset ADJUST to ~~zero~~ INEARNNS

Add ADJUST to **INNIRBEN**

Housing Benefit

From RENTER record, if BENUNIT = 1 and HBENPD equals -1 or 1 to ~~44~~ 52, and HBENEFIT = 1 add in HBENAMT to **INIRBEN** for PERSON = 1.

~~From HOUSEHOL record, if HHSTAT = 1 then from ADULT record if CVHB = 1 and CVPD equals -1 or 1 to 11, and exists(CHBAMT) = 1 then add CHBAMT to **INIRBEN**.~~

From HOUSEHOL record, if HHSTAT=2 or **CVPAY>0** then from ADULT record if ~~SCVHB~~**HBOTHBU=1** and ~~SCHB~~**HBOTHBP** equals -1 or 1 to ~~44~~ 52 then add ~~SCHB~~**HBOTHAMT** to **INIRBEN**.

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to ~~44~~ 52 add CTREBAMT to **INIRBEN** for PERSON = 1 only.

Other benefits

From BENEFITS record, if BENPD equals -1 or 1 to ~~44~~ 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 = ~~7~~ 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 = ~~9~~ 10 add BENAMT to **INNIRBEN** (SDA)

If BENEFIT = ~~10~~ 11 add BENAMT to **INIRBEN** (DWA)

If BENEFIT = ~~11~~ 12 add BENAMT to **INNIRBEN** (AA)

If BENEFIT = ~~12~~ 13 add BENAMT to **INNIRBEN** (Invalid Care Allowance)

If BENEFIT = ~~13~~ 14 add BENAMT to **INNIRBEN** (UB/Jobseeker's Allowance)

If BENEFIT = ~~14~~ 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 ~~Benefit~~ 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 = ~~26~~ 30 and PRES = 1 add BENAMT to **INNIRBEN** (Any other DSS benefits)

If BENEFIT = 37 add BENAMT to INNIRBEN (Guardians Allowance)

If BENEFIT=~~34~~ 38 and VAR2=~~2~~ 1 add **subtract** BENAMT to **INIRBEN** (Social Fund loan repayments included in quoted IS)

(for v33 the question Sflnc, "was the amount 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=40 add BENAMT/26 to INIRBEN (Social Fund - Crisis)~~

If BENEFIT=35 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)**

-2 variables set to unable to derive if any components are missing or period codes equal ~~12 or 13~~ **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

 --

KID04

Purpose : To indicate the total number of children in the benefit unit aged 0 to 4 inclusive. :

Created : VC - 3 March 1993

Database Table : BENUNIT

Minimum Value : 0

Maximum Value:

Units : integer

Validations :

Related Variables :

Children :

Parents :

Core variable/user : ~~ASD6A~~ **ASD4A**

Amendments : VC - 11 May 1993 To expand and clarify the definition
 : 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

1 Definition

This variable is coded as

KID04 This is the number of children in each benefit unit aged 0 to 4 years inclusive.

-1 Not applicable to this case

-2 Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

<u>Code</u>	<u>Condition</u>
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.
-1	Not applicable to this case - this should not occur for this variable.
-2	Unable to derive as component information is missing.

3 Results

Tabulation to show the total number of children aged 0 to 4 by benefit unit.

--

4 Test Cases

None as yet

 --

KID510

Purpose : To indicate the total number of children in the benefit unit aged 5 to 10 inclusive. :

Created : VC - 3 March 1993

Database Table : BENUNIT

Minimum Value : 0

Maximum Value:

Units : integer

Validations :

Related Variables :

Children :

Parents :

Core variable/user : ~~ASD6A~~ **ASD4A**

Issue date : 12 May 2003

Amendments : 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

1 Definition

This variable is coded as

KID510 This is the number of children in each benefit unit aged 5 to 10 inclusive.

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

2 FRS Specification

For each BENUNIT record

<u>Code</u>	<u>Condition</u>
-------------	------------------

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.
--------	-------------------------------------------------------------------------------------------------------------------------------------------

- 1 Not applicable to this case - should not occur for this variable.
- 2 Unable to derive as missing data.

3 Results

Tabulation to show the total number of children aged 5 to 10 by benefit unit.

--

4 Test Cases

None as yet

 --

KID1115

Purpose : To indicate the total number of children in the benefit unit aged 11 to 15 inclusive.
 Created : VC - 3 March 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value:
 Units : integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ~~ASD6A~~ **ASD4A**
 Amendments : VC - 11 May 1993 Amended to show coding
 : VC - 17 February 1994 Amended to reflect changes to version 30
 : VE - 22 May 1996 - No initial Version 32 update needed
 : SG - 12 November 1997 - No initial Version 33 update needed

1 Definition

This variable is coded as

KID1115 This is the number of children in each benefit unit aged 11 to 15 inclusive.
 -1 Not applicable to this case.
 -2 Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

<u>Code</u>	<u>Condition</u>
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.
-1	Not applicable to this case - should never occur for this variable.
-2	Unable to derive due to missing information.

3 Results

Tabulation to show the total number of children aged 11 to 15 inclusive by benefit unit.

--

4 Test Cases

None as yet

 --

KID1618

Purpose : To indicate the total number of dependents in the benefit unit aged 16 to 18 inclusive
 Created : VC - 4 March 1993
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value :
 Units : integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ~~ASD6A~~ **ASD4A**
 Amendments : VC - 11 May 1993 amended to show coding
 : VC - 17 February 1994 Amended to reflect version 30 changed
 : VE - 22 May 1996 - No initial Version 32 update needed
 : SG - 12 November 1997 - No initial Version 33 update needed

1 Definition

This variable is coded as

KID1618 This is the number of dependents in each benefit unit aged 16 to 18 inclusive.
 -1 Not applicable to this case
 -2 Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

<u>Code</u>	<u>Condition</u>
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 this variable.
-2	Unable to derive due to missing values.

3 Results

--

Tabulation to show the total number of dependents aged 16 to 18 inclusive by benefit unit.

4 Test Cases

None as yet

 --

KIDS0BU, KIDS1BU....KIDS18BU

Purpose : Total number of dependents aged under 1, aged from 1 to 2 years, ..
 Created : NM - 3 November 1992
 Database Table : BENUNIT
 Minimum Value : 0
 Maximum Value:
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ~~ASD6A~~ ASD4A
 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
 Issue date : 12 May 2003

1 Definition

This variable is coded as

KIDS0BU	Number of dependents under age 1
KIDS1BU	Number of dependents between age 1 and 2
KIDS2BU	Number of dependents between age 2 and 3
KIDS3BU	Number of dependents between age 3 and 4
KIDS4BU	Number of dependents between age 4 and 5
KIDS5BU	Number of dependents between age 5 and 6
KIDS6BU	Number of dependents between age 6 and 7
KIDS7BU	Number of dependents between age 7 and 8
KIDS8BU	Number of dependents between age 8 and 9
KIDS9BU	Number of dependents between age 9 and 10
KIDS10BU	Number of dependents between age 10 and 11
KIDS11BU	Number of dependents between age 11 and 12
KIDS12BU	Number of dependents between age 12 and 13
KIDS13BU	Number of dependents between age 13 and 14
KIDS14BU	Number of dependents between age 14 and 15
KIDS15BU	Number of dependents between age 15 and 16
KIDS16BU	Number of dependents between age 16 and 17
KIDS17BU	Number of dependents between age 17 and 18
KIDS18BU	Number of dependents between age 18 and 19

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 dependents in the benefit unit where age < 1
KIDS1BU	The sum of all dependents in the benefit unit where age >= 1 and < 2
KIDS2BU	The sum of all dependents in the benefit unit where age >= 2 and < 3
KIDS3BU	The sum of all dependents in the benefit unit where age >= 3 and < 4
KIDS4BU	The sum of all dependents in the benefit unit where age >= 4 and < 5
KIDS5BU	The sum of all dependents in the benefit unit where age >= 5 and < 6
KIDS6BU	The sum of all dependents in the benefit unit where age >= 6 and < 7
KIDS7BU	The sum of all dependents in the benefit unit where age >= 7 and < 8
KIDS8BU	The sum of all dependents in the benefit unit where age >= 8 and < 9
KIDS9BU	The sum of all dependents in the benefit unit where age >= 9 and < 10
KIDS10BU	The sum of all dependents in the benefit unit where age >= 10 and < 11
KIDS11BU	The sum of all dependents in the benefit unit where age >= 11 and < 12
KIDS12BU	The sum of all dependents in the benefit unit where age >= 12 and < 13
KIDS13BU	The sum of all dependents in the benefit unit where age >= 13 and < 14
KIDS14BU	The sum of all dependents in the benefit unit where age >= 14 and < 15
KIDS15BU	The sum of all dependents in the benefit unit where age >= 15 and < 16
KIDS16BU	The sum of all dependents in the benefit unit where age >= 16 and < 17
KIDS17BU	The sum of all dependents in the benefit unit where age >= 17 and < 18
KIDS18BU	The sum of all dependents in the benefit unit where age >= 18 and < 19

3 Results

Tabulation is required to show the number of dependents in each benefit unit by age of the dependent.

4 Test Cases

None as yet

 --

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 : HBM
 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
 Issue date : 12 May 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.

However, if the period code for the the amount of rent paid by the lodger is **90, 95 or 97** ~~12 or 13~~ (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 CVPD = **90, 95 or 97** ~~12 or 13~~ 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 4-11 1-10,13,26 or 52 , the amount of LODGER is the amount in CVPAY.
	If there is more than one adult in the benefit unit, the amount of LODGER is the total amount paid by both adults.
-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 12 or 13 .

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

--

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 :
 Amendments
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

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 : Publication/Regional Trends ASD6A **ASD4A** (**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 INCMAMP and INCMPPD to INCMAMP1, 2 and 3 and INCMPPD1, 2 and 3
 : SG - 29 October 1997 - V33 changes, tenure codes, period codes
 Issue date : 12 May 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 (~~ownhow=1~~ **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**

<u>Code</u>	<u>Condition</u>
-------------	------------------

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 ~~4-44~~ **1-10,13,26,52** for all mortgages then MORTPAY=MORTPAY+INCMAMP1+**INCMAMP2+INCMAMP3**

Else if ~~OWNHOW=1~~ **TENURE=1** (owned outright) MORTPAY=0

Endowment premiums

From ENDOWMNT table get MENPOLAM, MENPOLPD

Set ENDOWPAY equal to zero

For each mortgage, if ~~OWNHOW=2~~ **TENURE=2 or 3** and MENPOLPD equal to -1 or ~~4-44~~ **1-10,13,26,52** then ENDOWPAY=total of MENPOLAM else do not change ENDOWPAY

 --

(if ~~OWNHOW=1~~ **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 ~~4-44~~ **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 ~~4-44~~ **1-10,13,26,52** STRUINS=**STRUINS**+STRAMT2
 else do not change STRUINS

Service payments

From OWNER record get CHARGE1-7, CHAMT1-7, CHARGE1-7

set SERVPAY=0

If CHARGE1=1 and CHARGE1=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT1 to
 SERVPAY else do not change SERVPAY

If CHARGE2=1 and CHARGE2=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT2 to
 SERVPAY else do not change SERVPAY

If CHARGE3=1 and CHARGE3=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT3 to
 SERVPAY else do not change SERVPAY

If CHARGE4=1 and CHARGE4=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT4 to
 SERVPAY else do not change SERVPAY

If CHARGE5=1 and CHARGE5=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT5 to
 SERVPAY else do not change SERVPAY

If CHARGE6=1 and CHARGE6=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT6 to
 SERVPAY else do not change SERVPAY

If CHARGE7=1 and CHARGE7=-1, or ~~4-44~~ **1-10,13,26,52** add CHAMT7 to
 SERVPAY else do not change SERVPAY

MORTCOST=MORTPAY+ENDOWPAY+STRUINS+SERVPAY

-1 not applicable to this case - TENURE ne 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 HBM ASD4A **ISM** (**Bold** indicates lead user)
 Issued : 12 May, 2003
 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

~~NB - This variable does not include information about top-up loans for repairing the home.~~

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 ~~OWNHOW in the OWNER table indicates~~ 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 ~~as the variable INTL12M only holds the average amount of interest paid over the 12 months that the mortgagee holds information for. As a result, 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 ~~h:\frs\shared\docs\spread.xls p:\frs\shared\frs33\metadata\const33.xls~~ as V32Constants. 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** ~~12 or 13~~ (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** ~~12 or 13~~ MORTINT is set to -2. Cases where period codes have been skipped (where amount has been imputed) are allowed.

Top-up loans

Amounts borrowed for mortgage top-up loans for purposes that are unrelated to housing costs (TOPPUR=2,3,4,5 or 7) should be subtracted from the amount of mortgage interest paid by a household. For repayment mortgages, the proportion of the total amount borrowed (TOP/BORRAMT+TOP) which is made up of the top-up is calculated. The amount of the mortgage still left to pay is taken to be (1-this proportion) multiplied by MORTLEFT, assuming that the two loans are paid in tandem. For other types of mortgage, the interest on the top-up loan is calculated and subtracted from MORINPAY.

MIRAS

Allowance also has to be made for MIRAS which is deducted at ~~20%~~ **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.

As of the release of FRS 956b, the MIRAS adjustment for second mortgages is incorrect. We have assumed that tax relief of up to £30,000 is allowed on each mortgage, when in fact from 1 August 1988 the allowance is up to £30,000 per residence. Only three cases are affected (9045071, 12595041 and 4225191) by this ie. the total value of MORTLEFT for the first and second mortgages adds to more than £30,000. We are awaiting action from PSM before amending the spec.

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 (**SECM PUR=2,3,4,5 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.

~~Without sufficient information on the reason for these second loans,~~ It has been agreed that:

 --

- i the MIRAS adjustment will only be applied to the first mortgage **and to the second mortgage where SECMPUR=1 or 6**
- 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

Code Condition

Set TOP (temporary variable) to ZERO. TOP calculates the total amount of top-up loans borrowed for purposes that are unrelated to housing costs.

- If TOPSEQ = 1 and TOPUP = 1 and TOPPUR = 2, 3, 4, 5 or 7 then TOP = TOPAMT
- If TOPSEQ = 2 and TOPUP = 1 and TOPPUR = 2, 3, 4, 5 or 7 then TOP = TOP + TOPAMT
- If TOPSEQ = 3 and TOPUP = 1 and TOPPUR = 2, 3, 4, 5 or 7 then TOP = TOP + TOPAMT

Set INT (temporary variable) to ZERO. INT calculates the weekly interest paid on top-up loans borrowed for purposes that are unrelated to housing costs.

INT = (TOP * INTRATE) / 52

MORTINT From HOUSEHOL table if TENURE = ~~1 (owns/is buying), 2 (co-ownership scheme), 3 (shared ownership) or 4 (part own part rent)~~, get variable OWNHOW from OWNER record. - **2 (buy with mortgage) or 3 (part own/rent)**

~~If OWNHOW = 2 (bought with mortgage or loan) process MORTGAGE record~~

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 (April January 94-95)	INTRATE= 7.64% 7.99%	INTM1
If MNTHCODE=2 (May February 94-95)	INTRATE= 7.63% 7.97%	INTM2
If MNTHCODE=3 (June March 94-95)	INTRATE= 7.61% 7.95%	INTM3
If MNTHCODE=4 (July April 94-95)	INTRATE= 7.59% 7.94%	INTM4
If MNTHCODE=5 (August May 94-95)	INTRATE= 7.56% 7.92%	INTM5
If MNTHCODE=6 (September June 94-95)	INTRATE= 7.57% 7.90%	INTM6
If MNTHCODE=7 (October July 94-95)	INTRATE= 7.85% 7.56%	INTM7
If MNTHCODE=8 (November August 94-95)	INTRATE= 7.83% 7.51%	INTM8
If MNTHCODE=9 (December September 94-95)	INTRATE= 7.84% 7.48%	INTM9
If MNTHCODE=10 (January October 94-95)	INTRATE= 7.84% 7.28%	INTM10
If MNTHCODE=11 (February November 94-95)	INTRATE= 7.98% 7.08%	INTM11

 --

If MNTHCODE=12 (~~March~~ **December** 94-95) INTRATE=~~8.00%~~ **7.03%** 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 ~~12 or 13~~ **90,95 or 97** (but may equal -1) **and MORINPAY not equal -1**

Calculate MORTINT = MORINPAY - INT (how much interest did you pay last time - **interest payable on any top-up loans that are unrelated to housing costs**).

If MORTINT<0 set MORTINT=-2

If MORTPROT = 1 and INCMP1 = 1 and INCMPPD1 not equal ~~12 or 13~~ **90,95 or 97** (but may equal -1), **or INCMP2 = 1 and INCMPPD2 not equal 12 or 13 90,95 or 97 (but may equal -1) or INCMP3 = 1 and INCMPPD3 not equal 12 or 13 90,95 or 97 (but may equal -1)** calculate MORTINT = MORTINT - (INCMPAM1+INCMPAM2+INCMPAM3). (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 ~~12 or 13~~ **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 ~~4-11~~ **1-10,13,26,52** then MORTINT=MORTINT+OUTSAMT

If STRMORT=1 and STRPD1=-1, ~~4-11~~ **1-10,13,26,52** and STRAMT1<MORTINT then MORTINT=MORTINT-STRAMT1

Else if STRMORT=1 and STRPD1=-1, ~~4-11~~ **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 SECMPUR=1 or 6:**

If MORTSEQ=1 **or (MORTSEQ=2 and SECMPUR=1 or 6)** and

 --

If TAXRELF=2 (interest quoted does not include MIRAS arrangements)
 and MORTLEFT>30,000
 then $MORTINT = MORTINT - ((30,000 * INTRATE * 0.20.15) / 52)$

Else if TAXRELF=2 and MORTLEFT<=30,000
 then $MORTINT = MORTINT - ((MORTLEFT * INTRATE * 0.20.15) / 52)$

(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)

Set PROP (temporary variable) to ZERO. PROP calculates the
 proportion of the total amount borrowed which is made up of the top-up
 loan.

Set REMAINS (temporary variable) to ZERO. REMAINS calculates the
 amount remaining to pay on the mortgage multiplied by (1-PROP).

If TOP>0, $PROP = TOP / (BORRAMT + TOP)$

If TOPUP = 2 then REMAINS = MORTLEFT
 Else if TOPUP = 1 then REMAINS = MORTLEFT * (1-PROP)

Calculate $MORTINT = (MORTLEFT REMAINS * INTRATE) / 52$

MIRAS adjustment to first mortgage and second mortgage where
 SECMPUR=1 or 6:

If MORTSEQ=1 or (MORTSEQ=2 and SECMPUR=1 or 6) and

If ~~MORTLEFT~~ REMAINS >30,000
 then $MORTINT = MORTINT - ((30,000 * INTRATE * 0.20.15) / 52)$

Else if ~~MORTLEFT~~ REMAINS <=30,000
 then $MORTINT = MORTINT - ((MORTLEFT REMAINS * INTRATE * 0.20.15) / 52)$

- 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 = 42 or 43 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

Purpose : To calculate total amount of non-dependent deductions for each household :
 using HBAI specifications - for Council Tax Benefit
 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 : ASD6A
 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 withINDINC
 : 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)
 Issued : 12 May 2003

1 Definition

This variable is coded as

NDDCTB This is the total amount of non-dependent deductions for each household using the HBAI specification. This specification has been created from Andrew Ray's specification dated 12 November 1993 (see attached). **The constants and income levels are held in the table V32Constants in h:\frs\vars\docs\spread.xls p:\frs\shared\frs33\metadata\const33.xls.**

2 FRS Specification

For each HOUSEHOL table - set NDDCTB to zero, then

Process each BENUNIT table,

If BENUNIT greater than 1 (ie not head of household BU), get DEPDEDS derived variable.

If DEPDEDS = 6 or 8 (over 25 in receipt of IS, or any other person over age of 18), calculate NDDCTB = NDDCTB + ~~4.15~~ **C1CTB**

Else if DEPDEDS = 3 (18+ working over 16 hours), process each ADULT table

--

If ~~INDINC < 408~~ ~~411~~ **INCLEV**, calculate $NDDCTB = NDDCTB + 4.15$
C2CTB

Else calculate $NDDCTB = NDDCTB + 2.30$ **C3CTB**

Else if BENUNIT = 1,

Process each ADULT table, reset NDDCTB back to zero if any of the following conditions are met for either adult in the BU

Process each BENEFITS table for each adult in BU 1

If BENEFIT = 1 or 11 (receiving DLA care or AA) calculate
 $NDDCTB = 0$

If SPCREG1 = 1 (is blind), calculate $NDDCTB = 0$

If SUBLTAMT > 0 (receives a sublet income) AND NOUNITS = 2 (number of BUs in a household), calculate $NDDCTB = 0$

3 Results

To show the number of households falling into the following set of categories

Up to £4
£4 - £8
£8 - £12
£12 - £16
£16 - £20
£20 - £24
£24 - £28
£32 - £36
£36 - £40
Over £40

4 Test Cases

 --

NDDISHC

Purpose : To calculate total amount of non-dependent deductions for each household :
 using HBAI specifications - for IS housing costs
 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 : ASD6A
 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 and constants
 : VE - 1 July 1996 - Amended for constants being held in a separate table
 : SG - 28 October 1997 - V33, Updated constants table, no change to spec
 Issued : 12 May 2003

1 Definition

This variable is coded as

NDDISHC This is the total amount of non-dependent deductions for each household using the HBAI specification. This specification has been created from Andrew Ray's specification dated 12 November 1993 (see attached). **The constants and income levels are held in the table V32Constants in h:\frs\vars\docs\spread.xls**
~~p:\frs\shared\frs33\metadata\const33.xls.~~

2 FRS Specification

For each HOUSEHOL table - set NDDISHC to zero, then

Process each BENUNIT table,

If BENUNIT greater than 1 (ie not head of household BU), get DEPDEDS derived variable.

If DEPDEDS = 6 or 8 (over 25 in receipt of IS, or any other person over age of 18), calculate NDDISHC = NDDISHC + ~~5~~ **C1IS**

Else if DEPDEDS = 3 (18+ working over 16 hours), process each ADULT table

 --

If ~~INDINC < 72-74 INCL1IS~~, calculate ~~NDDISHC = NDDISHC + 5 C1IS~~
 Else if ~~INDINC < 108-111 INCL2IS~~, calculate ~~NDDISHC = NDDISHC + 9-10 C2IS~~
 Else if ~~INDINC < 139-145 INCL3IS~~, calculate ~~NDDISHC = NDDISHC + 13-14 C3IS~~
 Else if ~~INDINC >= 139-145 INCL3IS~~, calculate ~~NDDISHC = NDDISHC + 25-30 C4IS~~

Else if BENUNIT = 1,

Process each ADULT table, reset NDDISHC back to zero if any of the following conditions are met for either adult in the BU

Process each BENEFITS table for each adult in BU 1

If BENEFIT = 1 or 11 (receiving DLA care or AA) calculate NDDISHC = 0

If SPCREG1 = 1 (is blind), calculate NDDISHC = 0

If SUBLTAMT > 0 (receives a sublet income) and NOUNITS = 2 (number of BUs in household from HOUSEHOL table), calculate NDDISHC = 0

3 Results

To show the number of households falling into the following set of categories

Up to £4
 £4 - £8
 £8 - £12
 £12 - £16
 £16 - £20
 £20 - £24
 £24 - £28
 £32 - £36
 £36 - £40
 Over £40

4 Test Cases

 --

NDDRENTR

Purpose : To calculate total amount of non-dependent deductions for each household :
 using HBAI specifications - for rent rebate.
 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 : ASD6A
 Amendments : VC - 28 February 1994 To reflect changes to version 30
 VE - 19 April 1996 - to replace GROSSINC with INDINC
 VE - 4 June 1996 - No amendments needed for initial V32 update
 VE - 27 June 1996 - To update income levels and constants
 VE - 1 July 1996 - Amended for constants being held in a separate table.
 SG - 28 October 1997 - Updated constants table, no change to spec
 Issued : 12 May 2003

1 Definition

This variable is coded as

NDDRENTR This is the total amount of non-dependent deductions for each household using the HBAI specification. This specification has been created from Andrew Ray's specification dated 12 November 1993 (see attached). **The constants and income levels are held in the table V32Constants in h:\frs\vars\docs\spreads.xls, p:\frs\shared\frs33\metadata\const33.xls.**

2 FRS Specification

For each HOUSEHOL table - set NDDRENTR to zero, then

Process each BENUNIT table,

If BENUNIT greater than 1 (ie not head of household BU), get DEPDEDS derived variable.

If DEPDEDS = 6 or 8 (over 25 in receipt of IS, or any other person over age of 18), calculate $NDDRENTR = NDDRENTR + 5 \cdot C1RT$

Else if DEPDEDS = 3 (18+ working over 16 hours), process each ADULT table

If $INDINC < 72 - 74 \cdot INCL1RT$, calculate $NDDRENTR = NDDRENTR + 5 \cdot C1RT$

 --

Else if ~~INDINC < 108~~ ~~111~~ **INCL2RT**, calculate $NDDRENTR = NDDRENTR + 9 - 10 - C2RT$

Else if ~~INDINC < 139~~ ~~145~~ **INCL3RT**, calculate $NDDRENTR = NDDRENTR + 13 - 14 - C3RT$

Else if ~~INDINC >= 139~~ ~~145~~ **INCL3RT**, calculate $NDDRENTR = NDDRENTR + 25 - 30 - C4RT$.

Else if BENUNIT = 1,

Process each ADULT table, reset NDDRENTR back to zero if any of the following conditions are met for either adult in the BU

Process each BENEFITS table for each adult in BU 1

If BENEFIT = 1 or 11 (receiving DLA care or AA) calculate $NDDRENTR = 0$

If SPCREG1 = 1 (is blind), calculate $NDDRENTR = 0$

If SUBLTAMT > 0 (receives a sublet income) and NOUNITS = 2 (number of BUs in a household from HOUSEHOL table), calculate $NDDRENTR = 0$

3 Results

To show the number of households falling into the following set of categories

Up to £4
 £4 - £8
 £8 - £12
 £12 - £16
 £16 - £20
 £20 - £24
 £24 - £28
 £32 - £36
 £36 - £40
 Over £40

4 Test Cases

 --

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 : OCCUPPEN

Children :

Parents :

Core variable/user : HBM

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

Issue date : 6 January 1998

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, ROYAL3 AND ROYYR3. 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.

 --

ROYAL3 and ROYYR3 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.

ROYAL3 and ROYYR3 are also database variables which hold the information obtained from questions royal and royyr (NB - royal1 = royalties, royal2 = sleeping partners and royal3 = occ pen from o/s govt). However, if the period code for the pension is 12 or 13 (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 PENPD = 12 or 13 NETOCPEN is set to -2.

2 FRS Specification

For each ADULT with a pension record

<u>Code</u>	<u>Condition</u>
NETOCPEN	<p>From PENSION table, for each pension calculate a temporary variable OCCUP</p> <p>If PENTYPE = 1 (indicating an occupational pension is being received) and PENPD equal to -1 or 1-11, 13,26 or 52, get variables PENPAY, PENTAX, PTAMT, PTINC, PENOTH, POAMT and POINC.</p> <p>Compute OCCUP = PENPAY.</p> <p>If PENTAX = equal to 1 or has been skipped (tax deducted at source) and PTINC = 2 (original amount declared <u>after</u> amount of tax deducted) do not change OCCUP</p> <p>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</p> <p>If PENOTH = 1 (other deductions) and POINC = 2 (original amount declared <u>after</u> deduction), add POAMT to OCCUP.</p> <p>From ADULT table</p> <p>If ROYAL3 = 1 (pension from an overseas government) get amount from ROYYR3 and add to OCCUP.</p> <p>NETOCPEN will then be the sum of all occurrences of OCCUP as each adult is able to have up to 5 occupational pensions.</p>
-1	Not applicable to this case -

--

-2 Unable to derive because any of the above variables are missing or PENPD = ~~12 or 13~~
90, 95 or 97.

 --

NINDINC

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 :ASD4A
 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
 Issued : 12 May, 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 NINEARNNS, 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 (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, for example, if PAYPD = 90 or 95 or 97 NINDINC is set to -2. ~~However, for odd jobs, since by definition they should be irregular, lump sum payments are allowed but are divided by 4 to produce a weekly amount.~~ 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
NINPENINC	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 tax-payer 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 :

~~INCSE2~~ 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

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 ne 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

Then (AMTTAXF only asked if CHARITY=1 and CHRTAXF=1 and only for first job: to avoid setting cases to -2, only add in if value exists as passing through jobtypes 1-3)

if AMTTAXF exists add it into NINEARNS

if AMTOTH exists add it into NINEARNS
 (as for AMTTAXF)

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 (ie 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 (ie 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)
 $NINEARNS = NINEARNS + ((BONAMT(i)/52)/0.75)$

 --

(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 and ACCTAX=2, add (3/4 x amount in ACCINT) to NININV~~
 if ACCOUNT = 1 and ACCTAX=2, add amount in ACCINT to NININV
~~if ACCOUNT = 2 and ACCTAX=2, add (3/4 x amount in ACCINT) to NININV~~
 if ACCOUNT = 2 and ACCTAX=2, add amount in ACCINT to NININV
~~if ACCOUNT = 3 and ACCTAX=2, add (3/4 x amount in ACCINT) to NININV~~
 if ACCOUNT = 3 and ACCTAX=2, add amount in ACCINT to NININV
 if ACCOUNT = 4, add amount in ACCINT to NININV
~~if ACCOUNT = 5 and ACCTAX=2, add (3/4 x amount in ACCINT) to NININV~~
 if ACCOUNT = 5 and ACCTAX=2, add amount in ACCINT to NININV
~~if ACCOUNT = 6 and ACCTAX=2, add (3/4 x amount in ACCINT) to NININV~~
 if ACCOUNT = 6 and ACCTAX=2, 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 11 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 11 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 11 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 11 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 11 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 11 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 -

$INCSE2 + NININV + INRPINC + NINPENIN + INDISBEN + INOTHBEN + INRINC + NINEARNNS$

-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, NINEARNNS, INOTHBEN and INRINC which are calculated in this spec (missing components are already documented above).

 --

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 : OCCUPPEN - 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
 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 ROYAL3 = 1. ROYAL3 is a database variable produced from the question ROYAL where royal1 = royalties, royal2 = sleeping partners and royal3 = 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 ROYAL3 = 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.

 --

OCCUPPEN

Purpose : To show the amount of income received from all forms of occupational pensions from former employers :

Created : 21 January 1993

Database Table : ADULT

Minimum Value : 0

Maximum Value :

Units : Real

Validations :

Related Variables :

 Children :

 Parents :

Core variable/user : ISM HBM FCM PSM

Amendments : VC - 2 March 1993 change to multi response questions

 : VC - 27 April 1993. To expand definition to include descriptions of all questions/database variables used for the derived variable. :

 : VC - 13 July 1993 Amended to add back any tax deducted at source if the amount declared in penpay is net of tax. This is now a gross income variable although the amount collected by royrr2 cannot be said to be either gross or net of tax. :

 : VC - 17 February 1994 Amended to reflect version 30 changes

 : VC - 1 March 1994 To exclude any period codes 12 or 13.

 : JS - 21 February 1996 to allow for skipped values where amounts have been imputed

 : VE - 23 May 1996 - No amendments needed for initial V32 update

 : VE - 24 February 1997 - To amend for V33 HDS where ROYYR3 and PENOTH are not included in the database

 : VE - 25 February 1997 - To amend for changes in period codes for V33

Issue date : 12 May, 2003

1 Definition

This variable is coded as

OCCUPPEN 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

OCCUPPEN will be derived from variables PENTYPE, PENPAY, ~~ROYAL3~~ AND ROYYR3. 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 OCCUPPEN must be the total of all pensions.

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), ~~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.

~~ROYAL3 and ROYYR3~~ 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 ROYYR3

~~ROYAL3 and ROYYR3 are also database variables which hold the information obtained from questions royal and royyr (NB - royal1 = royalties, royal2 = sleeping partners and royal3 = occ pen from o/s govt.~~ However, if the period code for the pension is ~~12 or 13~~ **90 or 95 or 97 (less than one week/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 PENPD = ~~12 or 13~~ **90 or 95 or 97** OCCUPPEN 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>
OCCUPPEN	From PENSION table, for each pension If PENTYPE = 1 (indicating an occupational pension is being received) and PENPD equal to -1 or 1- 1152 , get variables PENPAY, PENTAX, PTAMT, PTINC, PENOTH , POAMT and POINC. Compute OCCUP = PENPAY. 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 OCCUP (otherwise, do not change OCCUP).

 --

If ~~PENOTH = 1~~ **POAMT >= 0** (other deductions) and POINC= 2 (original amount declared after deduction), add POAMT to OCCUP.

From ADULT table

If ~~ROYAL3 = 1~~ **ROYR3 >= 0** (pension from an overseas government) get amount from ROYR3 and add to OCCUP.

OCCUPPEN 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 = ~~12 or 13~~ **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

--

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 : ASD4A
Issue date : 12 May 2003
Amendments : VE - 5 June 1996 - No initial amendments needed for V32 update
 SG - 6 January 1998 - V33 update

1 Definition

ACCTYPE 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/maisonnette (including part of house/converted flat etc)
- 5 Other

2 FRS Specification

- 1 ACCTYPE=1
- 2 ACCTYPE=2
- 3 ACCTYPE=3
- 4 ACCTYPE=4 or 5
- 5 ACCTYPE=6 or 7

 --

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 : Publication
 Issue date : 12 May 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

1 Definition

PTENTYPE is derived using variables TENURE (household record), FURNISH and **LANDLORD (renter) OWNHOW (owner)**. Codes are effectively a recode of the TENTYPE breakdown. ~~Rent free category (TENTYPE=7) is now put in with renters.~~ (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 ~~unfurnished from local authority council (including part owners/part renters)~~
- 2 ~~Rented furnished (including part own/part rent)~~ **Rented from housing association**
- 3 ~~Rent free (no cases are expected)~~ **Rented privately unfurnished ~~rent free~~**
- 4 **Rented privately - furnished ~~rent paid~~**
- 4 5 Owned with mortgage (including ~~co- and shared ownership~~ **part rent/part own**)
- 5 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=2) or (TENTYPE=8 and FURNISH=2)

4 TENTYPE=4 or ~~TENTYPE=3~~ or ((TENTYPE=8 or TENTYPE=7) and (FURNISH=1 or FURNISH=-1))

~~4 5 If ((TENURE=1 or TENURE=2 or TENURE=3 or TENURE=4) and OWNHOW=2)

 TENTYPE=5,9,10- TENTYPE=5~~

~~5 6 If ((TENURE=1 or TENURE=2 or TENURE=3 or TENURE=4) and (OWNHOW=1 or
 OWNHOW=-1))
 TENTYPE=6~~

 --

STDREGN

Purpose : To indicate in which standard region the interviewee lives.
 Created : 18 January 1993
 Database Table : HOUSEHOL
 Minimum Value : 1
 Maximum Value : 12
 Units : Integer
 Validations :
 Related Variables :
 Children :
 Parents :
 Core variable/user : ISM HBM PSM **ASD6A** (**Bold** indicates lead user)
 Amendments : VC - 2 March 1993. Revised list of local authority codes received.
 : VC - 26 April 1993. South Somerset was in incorrect region moved to region :
 7. Amended description of region one to North (including Cumbria).
 : VC - 27 April 1993. to extend the definition to include meanings of each :
 question/database variable.
 : VC - 13 July 1993. Amended to same order produced by FES
 : VC - 6 August 1993. Moved Bromsgrove from region 5 to region 8, corrected :
 South Hertfordshire to South Herefordshire
 : VE - 4 June 1996 - No amendments needed for initial V32 update
 : VE - 12 March 1997 - New LAC codes and names for V33
 Issue date : 17 July, 1996

NB - There are currently no codes for the Isles of Scilly or the New Towns

1 Definition

This variable is coded as

- 1 North (including Cumbria)
 - 2 Yorkshire and Humberside
 - 3 North West
 - 4 East Midlands
 - 5 West Midlands
 - 6 East Anglia
 - 7 Greater London
 - 8 South East - excluding London
 - 9 South West
 - 10 Wales
 - 11 Scotland
- 1 Not applicable to this case

--

-2 Unable to derive as variable LAC is missing

The standard region is derived from the Local Authority Code in the variable LAC. The standard regions have been provided by Ms Odwell and are slightly different to the original FES definition as Northern Ireland is not included. The order in which the regions are coded are also different.

2 FRS Specification

Code each region according to the attached list.

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

-2 Unable to derive as the variable LAC is missing.

NB - These LA codes are unique to FRS.

3 Results

Tabulation to show the number of households in each standard region

4 Test Cases

None

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

1 HARTLEPOOL UA	145
1 REDCAR AND CLEVELAND UA	146
1 MIDDLESBROUGH UA	237
1 STOCTON ON TEES UA	238
1 ALLDERDALE CD	239
1 BARROW IN FURNESS CD	240
1 CARLISLE CD	241
1 COPELAND CD	242
1 EDEN CD	243
1 SOUTH LAKELAND CD	244
1 CHESTER LE STREET CD	245
1 DARLINGTON CD	246
1 DERWENTSIDE CD	337
1 DURHAM CD	338
1 EASINGTON CD	339
1 SEDGEFIELD CD	340
1 TEESDALE CD	341
1 WEAR VALLEY CD	342
1 ALNWICK CD	343
1 BERWICK-U-TWEED CD	344
1 BLYTH VALLEY CD	345
1 CASTLE MORPETH CD	346
1 TYNEDALE CD	437
1 WANSBECK CD	438
1 GATESHEAD MD	439
1 NEWCASTLE-U-TYNE MD	440
1 NORTH TYNESIDE MD	441
1 SOUTH TYNESIDE MD	442
1 SUNDERLAND MD	443
2 SHEFFIELD MD	47
2 KINGSTON-U-HULL UA	444
2 EAST RIDING OF YORKSHIRE UA	445
2 CRAVEN CD	538
2 HAMBLETON CD	539
2 HARROGATE CD	540
2 RICHMONDSHIRE CD	541
2 RYEDAILE CD	542
2 SCARBOROUGH CD	543
2 SELBY CD	544
2 YORK UA	545
2 BRADFORD MD	546
2 CALDERDALE MD	637
2 KIRKLEES MD	638

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

2 LEEDS MD	639
2 WAKEFIELD MD	640
2 BARNESLEY MD	954
2 DONCASTER MD	955
2 ROTHERHAM MD	956
3 SOUTH SHROPSHIRE CD	172
3 CHESTER CD	375
3 CONGLETON CD	376
3 CREWE & NANTWICH CD	467
3 ELLESMERE PORT CD	468
3 HALTON CD	469
3 MACCLESFIELD CD	470
3 VALE ROYAL CD	471
3 WARRINGTON CD	472
3 KNOWSLEY MD	473
3 LIVERPOOL MD	474
3 ST HELENS MD	475
3 SEFTON MD	476
3 WIRRAL MD	567
3 BOLTON MD	568
3 BURY MD	569
3 MANCHESTER MD	570
3 OLDHAM MD	571
3 ROCHDALE MD	572
3 SALFORD MD	573
3 STOCKPORT MD	574
3 TAMESIDE MD	575
3 TRAFFORD MD	576
3 WIGAN MD	667
3 BLACKBURN CD	668
3 BLACKPOOL CD	669
3 BURNLEY CD	670
3 CHORLEY CD	671
3 FYLDE CD	672
3 HYNDBURN CD	673
3 LANCASTER CD	674
3 PENDLE CD	675
3 PRESTON CD	676
3 RIBBLE VALLEY CD	767
3 ROSSENDALE CD	768
3 SOUTH RIBBLE CD	769
3 WEST LANCASHIRE CD	770
3 WYRE CD	771
4 NORTH EAST LINCOLNSHIRE UA	446
4 NORTH LINCOLNSHIRE UA	537

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

4 CORBY CD	566
4 AMBER VALLEY CD	641
4 BOLSOVER CD	642
4 CHESTERFIELD CD	643
4 DERBY CD	644
4 EREWASH CD	645
4 HIGH PEAK CD	646
4 DAVENTRY CD	657
4 EAST NORTHANTS CD	658
4 KETTERING CD	659
4 NORTHAMPTON CD	660
4 SOUTH NORTHANTS CD	661
4 WELLINGBOROUGH CD	662
4 NORTH EAST DERBYS CD	737
4 SOUTH DERBYSHIRE CD	738
4 WEST DERBYSHIRE CD	739
4 BLABY CD	740
4 HINCKLEY/BOSWORTH CD	741
4 CHARNWOOD CD	742
4 HARBOROUGH CD	743
4 LEICESTER CD	744
4 MELTON CD	745
4 NORTH WEST LEICS CD	746
4 OADBY & WIGSTON CD	837
4 RUTLAND CD	838
4 BOSTON CD	839
4 EAST LINDSEY CD	840
4 LINCOLN CD	841
4 NORTH KESTEVEN CD	842
4 SOUTH HOLLAND CD	843
4 SOUTH KESTEVEN CD	844
4 WEST LINDSEY CD	845
4 ASHFIELD CD	846
4 BASSETLAW CD	947
4 BROXTOWE CD	948
4 GEDLING CD	949
4 MANSFIELD CD	950
4 NEWARK CD	951
4 NOTTINGHAM CD	952
4 RUSHCLIFFE CD	953
5 BROMSGROVE CD	69
5 HEREFORD CD	70
5 LEOMINSTER CD	71
5 MALVERN HILLS CD	72
5 REDDITCH CD	73

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

5 SOUTH HEREFORDS CD	74
5 WORCESTER CD	75
5 WYCHAVON CD	76
5 WYRE FOREST CD	167
5 BRIDGNORTH CD	168
5 NORTH SHROPSHIRE CD	169
5 OSWESTRY CD	170
5 SHREWSBURY/ATCHAM CD	171
5 THE WREKIN CD	173
5 CANNOCK CHASE CD	174
5 EAST STAFFS CD	175
5 LICHFIELD CD	176
5 NEWCASTLE-U-LYME CD	267
5 SOUTH STAFFS CD	268
5 STAFFORD CD	269
5 STAFFS MOORLANDS CD	270
5 STOKE ON TRENT CD	271
5 TAMWORTH CD	272
5 NORTH WARWICKS CD	273
5 NUNEATON/BEDWORTH CD	274
5 RUGBY CD	275
5 STRATFORD ON AVON CD	276
5 WARWICK CD	367
5 BIRMINGHAM MD	368
5 COVENTRY MD	369
5 DUDLEY MD	370
5 SANDWELL MD	371
5 SOLIHULL MD	372
5 WALSALL MD	373
5 WOLVERHAMPTON MD	374
6 CAMBRIDGE CD	48
6 EAST CAMBS CD	49
6 FENLAND CD	50
6 HUNTINGDON CD	51
6 PETERBOROUGH CD	52
6 SOUTH CAMBS CD	53
6 BRECKLAND CD	54
6 BROADLAND CD	55
6 GREAT YARMOUTH CD	56
6 NORWICH CD	147
6 NORTH NORFOLK CD	148
6 SOUTH NORFOLK CD	149
6 WEST NORFOLK CD	150
6 BABERGH CD	151
6 FOREST HEATH CD	152

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

6 IPSWICH CD	153
6 MID SUFFOLK CD	154
6 ST EDMUNDSBURY CD	155
6 SUFFOLK COASTAL CD	156
6 WAVENEY CD	247
7 CROYDON LB	162
7 KINGSTON-U-THAMES LB	163
7 RICHMOND-U-THAMES LB	164
7 MERTON LB	165
7 SUTTON LB	166
7 WANDSWORTH LB	257
7 BARNET LB	352
7 BRENT LB	353
7 HARROW LB	354
7 EALING LB	355
7 HAMMERSMITH LB	356
7 HOUNSLOW LB	447
7 HILLINGDON LB	448
7 KENSINGTON LB	449
7 WESTMINSTER LB	450
7 BARKING/DAGENHAM LB	555
7 HAVERING LB	556
7 CAMDEN LB	647
7 ISLINGTON LB	648
7 CITY OF LONDON LB	649
7 HACKNEY LB	650
7 NEWHAM LB	651
7 TOWER HAMLETS LB	652
7 ENFIELD LB	653
7 HARINGEY LB	654
7 REDBRIDGE LB	655
7 WALTHAM FOREST LB	656
7 BEXLEY LB	958
7 GREENWICH LB	959
7 BROMLEY LB	960
7 LAMBETH LB	961
7 LEWISHAM LB	962
7 SOUTHWARK LB	963
8 MOLE VALLEY CD	57
8 REIGATE/BANSTEAD CD	58
8 RUNNYMEDE CD	59
8 SPELTHORNE CD	60
8 SURREY HEATH CD	61
8 TANDRIDGE CD	62
8 WAVERLEY CD	63

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

8 WOKING CD	64
8 ADUR CD	65
8 ARUN CD	66
8 CHICHESTER CD	157
8 CRAWLEY CD	158
8 HORSHAM CD	159
8 MID SUSSEX CD	160
8 WORTHING CD	161
8 NORTH BEDS CD	248
8 LUTON CD	249
8 MID BEDFORDSHIRE CD	250
8 SOUTH BEDS CD	251
8 BROXBOURNE CD	252
8 DACORUM CD	253
8 EAST HERTS CD	254
8 HERTSMERE CD	255
8 NORTH HERTS CD	256
8 BASINGSTOKE/DEANE CD	266
8 ST ALBANS CD	347
8 STEVENAGE CD	348
8 THREE RIVERS CD	349
8 WATFORD CD	350
8 WELWYN HATFIELD CD	351
8 EASTLEIGH CD	357
8 FAREHAM CD	358
8 GOSPORT CD	359
8 HART CD	360
8 HAVANT CD	361
8 NEW FOREST CD	362
8 EAST HAMPSHIRE CD	363
8 PORTSMOUTH CD	364
8 RUSHMOOR CD	365
8 SOUTHAMPTON CD	366
8 BASILDON CD	451
8 BRAINTREE CD	452
8 BRENTWOOD CD	453
8 CASTLE POINT CD	454
8 CHELMSFORD CD	455
8 COLCHESTER CD	456
8 TEST VALLEY CD	457
8 WINCHESTER CD	458
8 ISLE OF WIGHT UA	459
8 BRACKNELL CD	465
8 NEWBURY CD	466
8 EPPING FOREST CD	547

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

8 HARLOW CD	548
8 MALDON CD	549
8 ROCHFORD CD	550
8 SOUTHEND ON SEA CD	551
8 TENDRING CD	552
8 THURROCK CD	553
8 UTTLESFORD CD	554
8 READING CD	557
8 SLOUGH CD	558
8 WINDSOR/MAIDENH'D CD	559
8 WOKINGHAM CD	560
8 AYLESBURY VALE CD	561
8 SOUTH BUCKS CD	562
8 CHILTERN CD	563
8 MILTON KEYNES CD	564
8 WYCOMBE CD	565
8 CHERWELL CD	663
8 OXFORD CD	664
8 VALE WHITE HORSE CD	665
8 SOUTH OXFORDSHIRE CD	666
8 BRIGHTON CD	747
8 EASTBOURNE CD	748
8 HASTINGS CD	749
8 HOVE CD	750
8 LEWES CD	751
8 ROTHER CD	752
8 WEALDEN CD	753
8 ASHFORD CD	754
8 CANTERBURY CD	755
8 DARTFORD CD	756
8 WEST OXFORDSHIRE CD	757
8 DOVER CD	847
8 GILLINGHAM CD	848
8 GRAVESHAM CD	849
8 MAIDSTONE CD	850
8 ROCHESTER CD	851
8 SEVENOAKS CD	852
8 SHEPWAY CD	853
8 SWALE CD	854
8 THANET CD	855
8 TONBRIDGE/MALLING CD	856
8 TUNBRIDGE WELLS CD	957
8 ELMBRIDGE CD	964
8 EPSOM AND EWELL CD	965
8 GUILDFORD CD	966

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

9 WEST SOMERSET CD	67
9 YEOVIL CD	68
9 BOURNEMOUTH CD	258
9 CHRISTCHURCH CD	259
9 NORTH DORSET CD	260
9 POOLE CD	261
9 PURBECK CD	262
9 WEST DORSET CD	263
9 WEYMOUTH/PORTLAND CD	264
9 WIMBORNE CD	265
9 KENNET CD	460
9 NORTH WILTSHIRE CD	461
9 SALISBURY CD	462
9 THAMESDOWN CD	463
9 WEST WILTSHIRE CD	464
9 BATH & NORTH EAST SOMERSET UA	758
9 BRISTOL UA	759
9 NORTH SOMERSET UA	760
9 SOUTH GLOUCESTERSHIRE UA	761
9 CARADON CD	762
9 CARRICK CD	763
9 KERRIER CD	764
9 NORTH CORNWALL CD	765
9 PENWITH CD	766
9 RESTORMEL CD	857
9 EAST DEVON CD	858
9 EXETER CD	859
9 NORTH DEVON CD	860
9 PLYMOUTH CD	861
9 SOUTH HAMS CD	862
9 TEIGNBRIDGE CD	863
9 MID DEVON CD	864
9 TORBAY CD	865
9 TORRIDGE CD	866
9 WEST DEVON CD	967
9 CHELTENHAM CD	968
9 COTSWOLD CD	969
9 FOREST OF DEAN CD	970
9 GLOUCESTER CD	971
9 STROUD CD	972
9 TEWKESBURY CD	973
9 MENDIP CD	974
9 SEDGEMOOR CD	975
9 TAUNTON DEANE CD	976
10 ISLE OF ANGLESEY UA	184

FAMILY RESOURCES SURVEY**DERIVED VARIABLE SPECIFICATION**

--

10 GWYNEDD UA	185
10 CONWY UA	186
10 DENBIGHSHIRE UA	277
10 FLINTSHIRE UA	278
10 WREXHAM UA	279
10 POWYS UA	280
10 CEREDIGION UA	281
10 PEMBROKESHIRE UA	282
10 CARMARTHENSHIRE UA	283
10 SWANSEA UA	284
10 NEATH PORT TALBOT UA	285
10 BRIDGEND UA	286
10 VALE OF GLAMORGAN UA	377
10 RHONDDA, CYNON, TAFF UA	378
10 MERTHYR TYDFIL UA	379
10 CAERPHILLY UA	380
10 BLAENAU GWENT UA	381
10 TORFAEN UA	382
10 MONMOUTHSHIRE UA	383
10 NEWPORT UA	384
10 CARDIFF UA	385
11 ABERDEEN CITY UA	194
11 ABERDEENSHIRE UA	195
11 ANGUS UA	196
11 ARGYLL AND BUTE UA	287
11 SCOTTISH BORDERS UA	288
11 CLACKMANNANSHIRE UA	289
11 WEST DUNBARTONSHIRE UA	290
11 DUMFRIES AND GALLOWAY UA	291
11 DUNDEE CITY UA	292
11 EAST AYRSHIRE UA	293
11 EAST DUNBARTONSHIRE UA	294
11 EAST LoTHIAN UA	295
11 EAST RENFREWSHIRE UA	296
11 EDINBURGH, CITY OF UA	387
11 FALKIRK UA	388
11 FIFE UA	389
11 GLASGOW,CITY OF UA	390
11 HIGHLAND UA	391
11 INVERCLYDE UA	392
11 MIDLoTHIAN UA	393
11 MORAY UA	394
11 NORTH AYRSHIRE UA	395
11 NORTH LANARKSHIRE UA	396
11 PERTH AND KINROSS UA	487

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

11 RENFREWSHIRE UA	488
11 SOUTH AYRSHIRE UA	489
11 SOUTH LANARKSHIRE UA	490
11 STIRLING UA	491
11 WEST LOTHIAN UA	492

 --

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 : ISM **HBM** (**Bold** indicates lead user)
 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
 Issue date : 12 May 2003

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 has to 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.

--

<u>Code</u>	<u>Condition</u>
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 : ISM HBM FCM PSM

Issue date : 12 May 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

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 ~~12 or 13~~ **90 or 95 Or 97** (lumpsum/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 = ~~12 or 13~~ **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 1-11 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 = 12 OR 13 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

 --

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 : **ASD6A**
 Issue date : 12 May 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

1 Definition

This variable is coded as

- 1 ~~LA rented~~
- 2 ~~Housing Association~~
- 3 ~~Other private rented unfurnished~~
- 4 ~~Other private rented furnished~~
- 5 ~~Owned with mortgage (Includes part rent/part own)~~
- 6 ~~Owned outright (Includes part rent/part own)~~
- 7 ~~Rent free~~

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

- ~~8 Other Crown Estates/Government Departments.~~
- ~~9 Co-ownership schemes.~~
- ~~10 Shared ownership~~
- ~~-1 Not applicable to this case~~
- ~~-2 Unable to derive variable~~

- 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 or the renter records will indicate to which category of tenure that particular household belongs.

All tenants whose accomodation 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

IT SHOULD BE NOTED THAT THIS SPEC IS ONLY A BASIC REWORKING OF *TENTYPE* (V30). MORE DETAILED INFORMATION CAN BE OBTAINED IF *TENURE* = 6, 7 OR 8. THEN *FURNISH* IS ASKED TOGETHER WITH *SHORT* AND *OTHWAY* (*RENTER* TABLE). THESE QUESTIONS OBTAIN DATA RELATING TO HOW THE PROPERTY IS LET. AT PRESENT (3 AUGUST 1995) *TENTYPE* DOES NOT USE THIS INFORMATION.

NB/

TENURE	LANDLORD
1) Owns/is buying	1) Council
2) Co-ownership scheme	2) New Town Corporation
3) Shared ownership	3) Housing Association
4) Part own/Part rent	4) Crown Estates Commissioners

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

--

	5) Rented		5) Other Crown/Government Depts
	6) Rent free		6) Friend or relative
			7) Other organisation
			8) Other individual
FURNISH	1) Furnished	SHORT	1) An assured shorthold
	2) Unfurnished		2) A shorthold, but not assured
			3) Or it does not say it is a shorthold-at all.
			4) A shorthold, but not sure if assured or not.
OTHWAY	1) Company licence		
	2) College licence		
	3) Non-exclusive occupancy agreement		
	4) Holiday let		
	5) Low season let		

NB/

TENURE	1) Owns outright	LANDLORD	1) LA\Council\New Town corp
	2) Buying with mortgage		2) HA\Co-Op\Trust
	3) Part own/rent		3) Employer of HH member
	4) Rent		4) Other organisation
	5) Rent-free		5) Relative or friend of HH member
	6) Squat		6) Employer (individual)
			7) Other individual private landlord

FAMILY RESOURCES SURVEY

DERIVED VARIABLE SPECIFICATION

 --

FURNISH	1) Furnished	SHORT	1) An assured shorthold
	2) Unfurnished		2) A shorthold, but not assured
			3) Or it does not say it is a shorthold at all.
			4) A shorthold, but not sure if assured or not.
OTHWAY	1) Company licence		
	2) College licence		
	3) Non-exclusive occupancy agreement		
	4) Holiday let		
	5) Low season let		
	6) None of these		

2 FRS Specification

For each household, process HOUSEHOL record

Code Condition

Where TENURE = ~~4 or 5~~ 4, process RENTER record for details of type of landlord

- 1 If TENURE = ~~4 or 5~~ 4 and LANDLORD = 1 ~~or 2~~ (council/New Town Corporation) **and ACCJOB=2**
- 2 If TENURE = ~~4 or 5~~ 4 and LANDLORD = ~~3-2~~ (Housing Association/**Co-Op/Trust**) **and ACCJOB=2**
- 3 If TENURE = ~~4 or 5~~ 4 and ((~~LANDLORD = 4~~) or (LANDLORD= **3,4,5,6,or 7 6, 7 or 8** and FURNISH = 2)) **or (ACCJOB=1 and FURNISH=2)** (~~Crown Estates~~/other unfurnished)
- 4 If TENURE = ~~4 or 5~~ 4 and ((LANDLORD = **3,4,5,6,or 7 6, 7 or 8** and (FURNISH = 1 or FURNISH= -1)) or LANDLORD=-1) **or (ACCJOB=1 and FURNISH=1)** (other furnished)

 --

Where TENURE = 1 ~~or 4~~ 2 or 3, process OWNER record for details of how the property is owned

5 If TENURE = 1 ~~or 4~~ 2 or 3 and OWNHOW = 2 (owns/bought with mortgage or loan/**part own-
rent**)

6 If TENURE = 1 and OWNHOW = 1 or OWNHOW = 4 (owns/outright) (*this closes down the
route*)

Where TENURE = ~~6~~ 5

7 If TENURE = 6 5 (rent free).

Where TENURE = 6 (squatting).

8 If TENURE = 6 (squatting)

~~Where TENURE = 2~~

~~9 If TENURE = 2 (co-ownership)~~

~~Where TENURE = 3~~

~~10 If TENURE = 3 (shared ownership)~~

-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 (amended TOTCAP)

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 : ASD3A
 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
 Issue date: :12 May, 2003

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 which have information about the number and type of different assets/accounts. The variable assettype indicates whether or not a person has a particular asset so if this variable is coded 1,2,4,6,8,9,11 or 13, 17,18 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 4, 6, 8, and 13 (gilts, stocks and shares, national savings certificates, SAYE, national savings 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 £1500 and £20000. For benefit units which have answered that they have assets of less than £1500 or over £20000 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 Mr Ray 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 table V32Constants in the spreadsheet h:\frs\vars\docs\spreads.xls p:\frs\shared\frs33\metadata\const33.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. ~~In these cases totcap should be increased by 20%. 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 mid-points are read as informat from CONST33.XLS. Note that the top band is £30,000+ so a mid-pont 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 1993/94 were split into 5 categories:

was	from	
	July 95 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

 --

<u>Code</u>	<u>Condition</u>
TOTCAPBU	<p>From BENUNIT record, if TOTSAV = 2 or (TOTSAV=3 and SAMPLQTR=2,3 or 4)</p> <p>From ASSETS record, for each asset held for each adult</p> <p>If assetype = 1, get amount of capital in current accounts from HOWMUCH. If assetype = 2, get amount of capital in savings accounts from HOWMUCH. If assetype = 4, get HOWMUCHE if it exists else use HOWMUCH (gilts/trusts). If assetype = 6, get HOWMUCHE if it exists else use the issue value of National Savings Certs from ISSVAL. If assetype = 8, get HOWMUCHE if it exists else use HOWMUCH (SAYE) If assetype = 9, get amount of capital in premium bonds from HOWMUCH. If assetype = 11, get amount of capital in Nat Sav Income Bonds from HOWMUCH. If assetype = 13, get HOWMUCHE if it exists else use HOWMUCH (Nat Sav Capital Bonds). If assetype = 17, get HOWMUCH (FIRST Option Bonds). If assetype = 18, get HOWMUCH (Adult Yearly Plan).</p> <p>TOTCAPBU is then the total of any occurrences of the above</p> <p>Else if from BENUNIT record, (TOTSAV=1,3,4 or missing and SAMPLQTR=1) or (TOTSAV=1,4 or 5 or missing and SAMPLQTR=2,3 or 4)</p> <p>Set the interest rates for each type of account Then calculate amounts as follows -</p> <p>From ACCOUNTS record</p> <p>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)</p> <p>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)</p> <p>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)</p> <p>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)</p>

--

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% (BUILDING SOC)

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)

~~If ACCOUNTS = 10 and TOTSAV = 1 calculate CAP = ACCINT/r10a%
Else if ACCOUNTS = 10 and TOTSAV = 4 calculate CAP = ACCINT/r10b%
Else if ACCOUNTS = 10 and TOTSAV = 5 or missing calculate CAP =
ACCINT/r10c%~~

~~If ACCOUNTS = 11 and TOTSAV = 1 calculate CAP = ACCINT/r11a%
Else if ACCOUNTS = 11 and TOTSAV = 4 calculate CAP = ACCINT/r11b%
Else if ACCOUNTS = 11 and TOTSAV = 5 or missing calculate CAP =
ACCINT/r11c%~~

~~If ACCOUNTS = 12 and TOTSAV = 1 calculate CAP = ACCINT/r12a%
Else if ACCOUNTS = 12 and TOTSAV = 4 calculate CAP = ACCINT/r12b%
Else if ACCOUNTS = 12 and TOTSAV = 5 or missing calculate CAP =
ACCINT/r12c%~~

~~If ACCOUNTS = 13 and TOTSAV = 1 calculate CAP = ACCINT/r13a%
Else if ACCOUNTS = 13 and TOTSAV = 4 calculate CAP = ACCINT/r13b%
Else if ACCOUNTS = 13 and TOTSAV = 5 or missing calculate CAP =
ACCINT/r13c%~~

~~If ACCOUNTS = 14 and TOTSAV = 1 calculate CAP = ACCINT/r14a%
Else if ACCOUNTS = 14 and TOTSAV = 4 calculate CAP = ACCINT/r14b%~~

 --

~~Else if ACCOUNTS = 14 and TOTSAV = 5 or missing calculate CAP =
 ----- ACCINT/r14c%~~

~~----- If ACCOUNTS = 15 and TOTSAV = 1 calculate CAP = ACCINT/r15a%
 Else if ACCOUNTS = 15 and TOTSAV = 4 calculate CAP = ACCINT/r15b%
 Else if ACCOUNTS = 15 and TOTSAV = 5 or missing calculate CAP =
 ----- ACCINT/r15c%~~

~~Note: r* is for demonstration only: actual values included in program may be specified differently~~

TOTCAPBU is then the total of each occurrence of CAP

If ACCOUNT=16,17,18,20,21,22,23 10,11,12,13,14,15,16,17,18,19 then
~~TOTCAP=TOTCAP*1.20~~ 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 : ASD6A
 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
 Issue date : 12 May, 2003

~~NB - Andrew Ray will provide updated interest rates when they have been finalised for 1993/94.~~

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, ~~3 or 4~~ **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 (see ~~attached minute V32 Constants held in h:\frs\vars\docs\spreads.xls in the spreadsheet p:\frs\shared\frs33\metadata\const33.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.

~~However, there must also be a check built into the programme so that if this calculation produces an amount which does not agree with the original answer to TOTSAVE, it is assumed that TOTSAVE is more accurate and a standard amount is used instead. Therefore, if the amount is over £1500 and TOTSAVE = 1, TOTCAP is set to £1499 and if the amount calculated is under £20000 and TOTSAVE = 3-4, TOTCAP is set to £20001.~~

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, 3 or 4 4 or 5 or missing

~~Set the following rates for 1992/93 (standard table?)~~

~~r1 = 3.5~~

~~r2 = 5.0 } Use r2 if ACCINT > £500, use r3 if <= £500.~~

 --

~~r3 = 2.5}~~
~~r4 = 7.0~~
~~r5 = 4.0~~
~~r6 = sum of r1 to r5 and r7 to r8/7~~
~~r7 = 7.0~~
~~r8 = 4.5~~

If INTDATE (interview date) is on or after 1/1/93 use the following rates for 1992/93

~~_____ r1 = 3.5~~
~~_____ r2 = 3.75 } Use r2 if ACCINT > £500, use r3 if <= £500.~~
~~_____ r3 = 3.75 }~~
~~_____ r4 = 7.0~~
~~_____ r5 = 4.0~~
~~_____ r6 = sum of r1 to r5 and r7 to r8/7~~
~~_____ r7 = 7.0~~
~~_____ r8 = 4.5~~

Then calculate amount as follows -

From ACCOUNTS record

If ACCOUNTS = ~~24~~ 20 and TOTSAV = 1, calculate CAP = ACCINT/~~r2% or r3%~~ **r02a%**
 Else if ACCOUNTS = ~~24~~ 20 and TOTSAV = 4, calculate CAP = ACCINT/**r02b%**
 Else if ACCOUNTS = ~~24~~ 20 and TOTSAV = 5 or missing, calculate CAP =
 ACCINT/**r02c%**

If ACCOUNTS = ~~25~~ 21 and TOTSAV = 1, calculate CAP = ACCINT/~~r4%~~ **r03a%**
 Else if ACCOUNTS = ~~25~~ 21 and TOTSAV = 4, calculate CAP = ACCINT/**r03b%**
 Else if ACCOUNTS = ~~25~~ 21 and TOTSAV = 5 or missing , calculate CAP =
 ACCINT/**r03c%**

If ACCOUNTS = ~~26~~ 22 or ACCOUNTS = ~~27~~ 23 and TOTSAV = 1, calculate CAP =
 ACCINT/~~r5%~~ **r01a%**

CAP

Else if ACCOUNTS = ~~26~~ 22 or ACCOUNTS = ~~27~~ 23 and TOTSAV = 4, calculate
 = ACCINT/**r01b%**

Else if ACCOUNTS = ~~26~~ 22 or ACCOUNTS = ~~27~~ 23 and TOTSAV = 5 or missing,
 calculate CAP = ACCINT/**r01c%**

~~If ACCOUNTS = 27, calculate CAP = ACCINT/r5%~~

If ACCOUNTS = ~~28~~ 24, calculate CAP = ACCINT/~~r7%~~ **r11a%**

If ACCOUNTS = ~~29~~ 25, calculate CAP = ACCINT/~~r8%~~ **r13a%**

If ACCOUNTS = ~~30~~ 26, calculate CAP = ACCINT/~~r8%~~ **r14a%**

TOTCAP is then the total of each occurrence of CAP.

~~If TOTCAP >= 1500 and TOTSAVE = 1, reset TOTCAP to £1499~~

~~_____ If TOTCAP <= 20000 and TOTSAVE = 3, reset TOTCAP to £20001.~~

--

- 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 : ~~ADULT~~ **CHILD**
 Minimum Value : 0
 Maximum Value :
 Units : Integer
 Validations :
 Related Variables : TOTGRANT - total grant received by an adult
 Children :
 Parents :
 Core variable/user : ASD6A
 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
 Issue date : 12 May 2003

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 : ASD6A
 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
 Issue date : 12 May, 2003

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 ~~where FTED = 1 or TEA = 06 and TYPEED = 7 (university, polytechnic, further education)~~

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 : ISM
 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
 Issue date : 12 November 1997

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. 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

Code Condition

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

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 ~~for the whole of 1994/95~~

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 : Publication

Amendments : VE - 5 June 1996 - To tidy up spec for V32

: SG - 12 November 1997 - No initial updates for V33

Issue date : 12 November 1997

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 (~~SAMPLQTR=2~~), 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 data base in TOTSAV. ~~To make available a single variable which can be used across the whole year, TOTSAVBU condenses the 5 codes for the second half of the year using variable SAMPLQTR (common variable for sample quarter).~~

2 FRS specification

From table BENUNIT

<u>Code</u>	<u>Condition</u>
1	TOTSAV=1
2	TOTSAV=2 or (SAMPLQTR= 2,3 or 4 and TOTSAV=3)
3	(SAMPLQTR=1 and TOTSAV=3) or (SAMPLQTR= 2,3 or 4 and TOTSAV=4)
4	(SAMPLQTR=1 and TOTSAV=4) or 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
 : 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 $TTWPAY \neq 3$ (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

 --

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 : ISM HBM PSM **FCM** (**Bold** indicates lead user)

Issue date : 12 May 2003

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

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 = ~~2~~**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 CHRTAXF = 1 (indicating that the person has a deduction for charities) (see deducts for reson 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 OTHDED~~678~~ = 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 OTHDED~~678~~, DEDUC1 to DEDUC~~567~~ 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 ~~TAXINC=1~~ **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 ~~1-11~~ **1-10,13,26,52** and

If PAYSZIP = 1 **or 2** and GRSWAGE exists, calculate UGRSPAY = amount in GRSWAGE.

Or if PAYSZIP = ~~2~~ **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 AMTTAXF exists add AMTTAXF to UGRSPAY

If AMTOTH exists add AMTOTH 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 ~~1-11~~ **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 = ~~12 or 13~~ **90 or 95 or 97** (should only be where PAYAMT is missing because for other variables, values are ignored if they do not exist)

 --

UPERSON

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
 Issued : 12 May 2003

1 Definition

This variable assigns the value 1 to the first person in each Benefit Unit increments by one for each adult and each child, within that Benefit Unit Benefit Unit. Dependents are number 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:

- 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 dependent in descending order of age,

3 Results

What tabulation should be produced to check the results?

4 Test Cases

 --

WATSEWRT

Purpose :To show the total amount of water and sewerage rates paid by each household in England and Wales.

Created :2 February 1993

Database Table :HOUSEHOL

Minimum Value :0

Maximum Value :

Units :Real

Validations :

Related Variables :

 Children :

 Parents :

Issue Date :12 May 2003

Core variable/user :HBM

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.

AND : 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

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 HBWSAMT. 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 data base.

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 HBSTMT ~~exists~~ ~~and~~ =1 and HBWSAMT exists
 WATSEWRT=WATSEWRT+HBWSAMT 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 WORSINC=1 or 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 ~~WATERINC exists and -1 or SEWERINC exists and -1~~ and WSINCAMT=-1
 (values of waterinc or sewerinc or **WORSINC** has have been imputed but WSINCAMT
 has been skipped) call WATER ~~WSINCAMT-WSINCAMT+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 : ~~ASD6A~~ **ASD4A**
 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
 Issue date : 12 May 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 variable DEPCHILDB (number of dependent child in BU) from BENUNIT table,

Code Condition

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

