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
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

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 = 5). Any other person whose illness or disability has lasted for more than 6 months, ie where injpd = 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 INJPD 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

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

1 From table ADULT

If SEX = 1 and AGE >= 60 or < 65

2 From ADULT table
If AGE < 60 and INJ~~PD~~ **LONG**= 2 or
If AGE < 60 WORKING = 2, JOBAWAY = 2, LIKEWK = 1 and NOLOOK = 5 or
If AGE < 60 WORKING = 2, JOBAWAY = 2, LIKEWK = 2 and NOWANT = 5

3 From table ADULT

If AGE < 60 <u>AND</u> 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:
Amendments

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

CAREABtotal 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)

CAREAHtotal number of adults looked after in the same household but different benefit unit

CARERErelatives outside the household looked after (maximum value of 1, since questionnaire records "relative" as a single response)

CAREFRfriends and neighbours outside the household looked after (maximum value of 1, since questionnaire records "friend/neighbour" as a single response)

CARECLclient of voluntary organisation outside the household looked after (maximum value of 1, since questionnaire records "client of a voluntary organisation" as a single response)

CAREOTothers 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/houshold/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

HOURABtotal number of hours spent caring for adults in the same benefit unit

HOURAHtotal number of hours spent caring for adults in the same household but different benefit unit

HOURREtotal number of hours spent caring for relatives outside the household

HOURFRtotal number of hours spent caring for friends and neighbours outside the household

HOURCLtotal number of hours spent caring for clients of voluntary organisation outside the household

HOUROTtotal number of hours spent caring for others outside the household

HOURCB total number of hours spent caring for children in the same benefit unit

HOURCHtotal 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/houshold/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) 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) 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) 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) 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

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: 21 April 2005
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)

1 Definition

This variable is coded as

~~1Metropolitan England~~
~~2Non-metropolitan England~~
1 Non-metropolitan England
2 Metropolitan England
3London
4Wales
5Scotland
~~6New Towns~~

-1Not applicable to this case
-2Unable 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.

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.

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

-2Unable 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

STND REGION	LAC	AUTHORITY NAME	TYPE OF AUTHORITY
1	776	Gateshead	~~1~~2
1	867	Newcastle upon Tyne	~~1~~2
1	868	North Tyneside	~~1~~2
1	869	South Tyneside	~~1~~2
1	870	Sunderland	~~1~~2
2	076	Barnsley	~~1~~2
2	167	Doncaster	~~1~~2
2	168	Rotherham	~~1~~2
2	169	Sheffield	~~1~~2
2	170	Bradford	~~1~~2
2	171	Calderdale	~~1~~2
2	172	Kirklees	~~1~~2
2	173	Leeds	~~1~~2
2	174	Wakefield	~~1~~2
8	462	Birmingham	~~1~~2
8	463	Coventry	~~1~~2
8	464	Dudley	~~1~~2
8	465	Sandwell	~~1~~2
8	466	Solihull	~~1~~2
8	557	Walsall	~~1~~2
8	558	Wolverhampton	~~1~~2
9	273	Bolton	~~1~~2
9	274	Bury	~~1~~2
9	275	Manchester	~~1~~2
9	276	Oldham	~~1~~2
9	367	Rochdale	~~1~~2
9	368	Salford	~~1~~2
9	369	Stockport	~~1~~2
9	370	Tameside	~~1~~2
9	371	Trafford	~~1~~2
9	372	Wigan	~~1~~2
9	567	Knowsley	~~1~~2
9	568	Liverpool	~~1~~2
9	569	St Helens	~~1~~2
9	570	Sefton	~~1~~2
9	571	Wirral	~~1~~2

1	572	Allerdale	~~2~~1
1	573	Barrow-in-Furness	~~2~~1
1	574	Carlisle	~~2~~1
1	575	Copeland	~~2~~1
1	576	Eden	~~2~~1
1	667	South Lakeland	~~2~~1
1	668	Hartlepool	~~2~~1
1	669	Langbaurgh-on-Tees	~~2~~1
1	670	Middlesbrough	~~2~~1
1	671	Stockton-on-Tees	~~2~~1
1	672	Chester-le-Street	~~2~~1

1	673	Darlington	~~2~~1
1	674	Derwentside	~~2~~1
1	675	Durham	~~2~~1
1	676	Easington	~~2~~1
1	767	Sedgefield	~~2~~1
1	768	Teesdale	~~2~~1
1	769	Wear Valley	~~2~~1
1	770	Alnwick	~~2~~1
1	771	Berwick-upon-Tweed	~~2~~1
1	772	Blyth Valley	~~2~~1
1	773	Castle Morpeth	~~2~~1
1	774	Tynedale	~~2~~1
1	775	Wansbeck	~~2~~1
2	067	Scunthorpe	~~2~~1
2	068	Craven	~~2~~1
2	069	Hambleton	~~2~~1
2	070	Harrogate	~~2~~1
2	071	Richmondshire	~~2~~1
2	072	Ryedale	~~2~~1
2	073	Scarborough	~~2~~1
2	074	Selby	~~2~~1
2	075	York	~~2~~1
2	969	East Yorks Borough of Beverley	~~2~~1
2	970	Boothferry	~~2~~1
2	971	Cleethorpes	~~2~~1
2	972	East Yorkshire	~~2~~1
2	973	Glanford	~~2~~1
2	974	Great Grimsby	~~2~~1
2	975	Holderness	~~2~~1
2	976	Kingston upon Hull	~~2~~1
3	559	Amber Valley	~~2~~1
3	560	Bolsover	~~2~~1
3	561	Chesterfield	~~2~~1
3	562	Derby	~~2~~1
3	563	Erewash	~~2~~1
3	564	High Peak	~~2~~1
3	565	North East Derbyshire	~~2~~1
3	566	South Derbyshire	~~2~~1
3	657	Derbyshire Dales	~~2~~1
3	658	Blaby	~~2~~1
3	659	Charnwood	~~2~~1
3	660	Harborough	~~2~~1
3	661	Hinckley and Bosworth	~~2~~1
3	662	Leicester	~~2~~1
3	663	Melton	~~2~~1
3	664	North West Leicestershire	~~2~~1
3	665	Oadby and Wigston	~~2~~1
3	666	Rutland	~~2~~1
3	757	Boston	~~2~~1
3	758	East Lindsey	~~2~~1
3	759	Lincoln	~~2~~1
3	760	North Kesteven	~~2~~1
3	761	South Holland	~~2~~1
3	762	South Kesteven	~~2~~1

3	763	West Lindsey	~~2~~1
3	764	Corby	~~2~~1
3	765	Daventry	~~2~~1
3	766	East Northamptonshire	~~2~~1
3	857	Kettering	~~2~~1
3	858	Northampton	~~2~~1
3	859	South Northampshire	~~2~~1
3	860	Wellingborough	~~2~~1
3	861	Ashfield	~~2~~1
3	862	Bassetlaw	~~2~~1
3	863	Broxtowe	~~2~~1
3	864	Gedling	~~2~~1
3	865	Mansfield	~~2~~1
3	866	Newark and Sherwood	~~2~~1
3	967	Nottingham	~~2~~1
3	968	Rushcliffe	~~2~~1
4	153	Cambridge	~~2~~1
4	154	East Cambridgeshire	~~2~~1
4	155	Fenland	~~2~~1
4	156	Huntingdonshire	~~2~~1
4	247	Peterborough	~~2~~1
4	248	South Cambridgeshire	~~2~~1
4	249	Breckland	~~2~~1
4	250	Broadland	~~2~~1
4	251	Great Yarmouth	~~2~~1
4	252	Norwich	~~2~~1
4	253	North Norfolk	~~2~~1
4	254	South Norfolk	~~2~~1
4	255	Kings Lynn and West Norfolk	~~2~~1
4	256	Babergh	~~2~~1
4	347	Forest Heath	~~2~~1
4	348	Ipswich	~~2~~1
4	349	Mid Suffolk	~~2~~1
4	350	St Edmundsbury	~~2~~1
4	351	Suffolk Coastal	~~2~~1
4	352	Waveney	~~2~~1
6	047	Guildford	~~2~~1
6	048	Mole Valley	~~2~~1
6	049	Reigate and Banstead	~~2~~1
6	050	Runnymede	~~2~~1
6	051	Spelthorne	~~2~~1
6	052	Surrey Heath	~~2~~1
6	053	Tandridge	~~2~~1
6	054	Waverley	~~2~~1
6	055	Woking	~~2~~1
6	056	Adur	~~2~~1
6	145	Luton	~~2~~1
6	146	Mid Bedfordshire	~~2~~1
6	147	Arun	~~2~~1
6	148	Chichester	~~2~~1
6	149	Crawley	~~2~~1
6	150	Horsham	~~2~~1
6	151	Mid Sussex	~~2~~1
6	152	Worthing	~~2~~1

6	237	North Bedfordshire	21
6	238	South Bedfordshire	21
6	239	Bracknell Forest	21
6	240	Newbury	21
6	241	Reading	21
6	242	Slough	21
6	243	Windsor and Maidenhead	21
6	244	Wokingham	21
6	245	Aylesbury Vale	21
6	246	Chiltern	21
6	337	Milton Keynes	21
6	338	South Buckinghamshire	21
6	339	Wycombe	21
6	340	Brighton	21
6	341	Eastbourne	21
6	342	Hastings	21
6	343	Hove	21
6	344	Lewes	21
6	345	Rother	21
6	346	Wealden	21
6	437	Basildon	21
6	438	Braintree	21
6	439	Brentwood	21
6	440	Castle Point	21
6	441	Chelmsford	21
6	442	Colchester	21
6	443	Epping Forest	21
6	444	Harlow	21
6	445	Maldon	21
6	446	Rochford	21
6	537	Southend-on-Sea	21
6	538	Tendring	21
6	539	Thurrock	21
6	540	Uttlesford	21
6	541	Basingstoke and Deane	21
6	542	East Hampshire	21
6	543	Eastleigh	21
6	544	Fareham	21
6	545	Gosport	21
6	546	Hart	21
6	637	Havant	21
6	638	New Forest	21
6	639	Portsmouth	21
6	640	Rushmoor	21
6	641	Southampton	—21
6	642	Test Valley	21
6	643	Winchester	21
6	644	Broxbourne	21
6	645	Dacorum	21
6	646	East Hertfordshire	21
6	737	Hertsmere	21
6	738	North Hertfordshire	21
6	739	St Albans	21
6	740	Stevenage	21

6	741	Three Rivers	~~2~~1
6	742	Watford	~~2~~1
6	743	Welwyn Hatfield	~~2~~1
6	744	Medina	~~2~~1
6	745	South Wight	~~2~~1
6	746	Ashford	~~2~~1
6	837	Canterbury	~~2~~1
6	838	Dartford	~~2~~1
6	839	Dover	~~2~~1
6	840	Gillingham	~~2~~1
6	841	Gravesham	~~2~~1
6	842	Maidstone	~~2~~1
6	843	Rochester upon Medway	~~2~~1
6	844	Sevenoaks	~~2~~1
6	845	Shepway	~~2~~1
6	846	Swale	~~2~~1
6	947	Thanet	~~2~~1
6	948	Tonbridge and Malling	~~2~~1
6	949	Tunbridge Wells	~~2~~1
6	950	Cherwell	~~2~~1
6	951	Oxford	~~2~~1
6	952	South Oxfordshire	~~2~~1
6	953	Vale Of White Horse	~~2~~1
6	954	West Oxfordshire	~~2~~1
6	955	Elmbridge	~~2~~1
6	956	Epsom and Ewell	~~2~~1
7	057	Cheltenham	~~2~~1
7	058	Cotswold	~~2~~1
7	059	Forest of Dean	~~2~~1
7	060	Gloucester	~~2~~1
7	061	Stroud	~~2~~1
7	062	Tewkesbury	~~2~~1
7	063	Mendip	~~2~~1
7	064	Sedgemoor	~~2~~1
7	065	Taunton Deane	~~2~~1
7	066	West Somerset	~~2~~1
7	158	Kennet	~~2~~1
7	159	North Wiltshire	~~2~~1
7	160	Salisbury	~~2~~1
7	161	Thamesdown	~~2~~1
7	162	West Wiltshire	~~2~~1
7	656	Bath	~~2~~1
7	747	Bristol	~~2~~1
7	748	Kingswood	~~2~~1
7	749	Northavon	~~2~~1
7	750	Wansdyke	~~2~~1
7	751	Woodspring	~~2~~1
7	752	Caradon	~~2~~1
7	753	Carrick	~~2~~1
7	754	Kerrier	~~2~~1
7	755	North Cornwall	~~2~~1
7	756	Penwith	~~2~~1
7	847	Restormel	~~2~~1
7	849	East Devon	~~2~~1

7	850	Exeter	~~2~~1
7	851	Mid Devon	~~2~~1
7	852	North Devon	~~2~~1
7	853	Plymouth	~~2~~1
7	854	South Hams	~~2~~1
7	855	Teignbridge	~~2~~1
7	856	Torbay	~~2~~1
7	957	Torridge	~~2~~1
7	958	West Devon	~~2~~1
7	959	Bournemouth	~~2~~1
7	960	Christchurch	~~2~~1
7	961	North Dorset	~~2~~1
7	962	Poole	~~2~~1
7	963	Purbeck	~~2~~1
7	964	West Dorset	~~2~~1
7	965	Weymouth and Portland	~~2~~1
7	966	East Dorset (aka Wimborne)	~~2~~1
8	157	South Somerset (aka Yeovil)	~~2~~1
8	163	Bromsgrove	~~2~~1
8	164	Hereford	~~2~~1
8	165	Leominster	~~2~~1
8	166	Malvern Hills	~~2~~1
8	257	Redditch	~~2~~1
8	258	South Hertfordshire	~~2~~1
8	259	Worcester	~~2~~1
8	260	Wychavon	~~2~~1
8	261	Wyre Forest	~~2~~1
8	262	Bridgnorth	~~2~~1
8	263	North Shropshire	~~2~~1
8	264	Oswestry	~~2~~1
8	265	Shrewsbury and Atcham	~~2~~1
8	266	South Shropshire	~~2~~1
8	357	The Wrekin	~~2~~1
8	358	Cannock Chase	~~2~~1
8	359	East Staffordshire	~~2~~1
8	360	Lichfield	~~2~~1
8	361	Newcastle-under-Lyme	~~2~~1
8	362	South Staffordshire	~~2~~1
8	363	Stafford	~~2~~1
8	364	Staffordshire Moorlands	~~2~~1
8	365	Stoke-on-Trent	~~2~~1
8	366	Tamworth	~~2~~1
8	457	North Warwickshire	~~2~~1
8	458	Nuneaton & Bedworth	~~2~~1
8	459	Rugby	~~2~~1
8	460	Stratford-on-Avon	~~2~~1
8	461	Warwick	~~2~~1
9	175	Chester	~~2~~1
9	176	Congleton	~~2~~1
9	267	Crewe and Nantwich	~~2~~1
9	268	Ellesmere Port and Neston	~~2~~1
9	269	Halton	~~2~~1
9	270	Macclesfield	~~2~~1
9	271	Vale Royal	~~2~~1

9	272	Warrington	21
9	373	Blackburn	21
9	374	Blackpool	21
9	375	Burnley	21
9	376	Chorley	21
9	467	Fylde	21
9	468	Hyndburn	21
9	469	Lancaster	21
9	470	Pendle	21
9	471	Preston	21
9	472	Ribble Valley	21
9	473	Rossendale	21
9	474	South Ribble	21
9	475	West Lancashire	21
9	476	Wyre	21

5	353	City of London	3
5	354	Camden	3
5	355	Hackney	3
5	356	Hammersmith and Fulham	3
5	447	Haringey	3
5	448	Islington	3
5	449	Kensington and Chelsea	3
5	450	Lambeth	3
5	451	Lewisham	3
5	452	Newham	3
5	453	Southwark	3
5	454	Tower Hamlets	3
5	455	Wandsworth	3
5	456	Westminster, City of	3
5	547	Barking and Dagenham	3
5	548	Barnet	3
5	549	Bexley	3
5	550	Brent	3
5	551	Bromley	3
5	552	Croydon	3
5	553	Ealing	3
5	554	Enfield	3
5	555	Greenwich	3
5	556	Harrow	3
5	647	Havering	3
5	648	Hillingdon	3
5	649	Hounslow	3
5	650	Kingston upon Thames	3
5	651	Merton	3
5	652	Redbridge	3
5	653	Richmond upon Thames	3
5	654	Sutton	3
5	655	Waltham Forest	3

10	077	Torfaen	4
10	078	Aberconwy	4
10	079	Arfon	4
10	080	Dwyfor	4
10	081	Meirionydd	4
10	082	Ynys Mon - Isle of Anglesey	4
10	083	Cynon Valley	4
10	084	Merthyr Tydfil	4
10	085	Ogwr	4
10	086	Rhondda	4
10	177	Rhymney Valley	4
10	178	Taff-Ely	4
10	179	Brecknock	4
10	180	Montgomeryshire	4
10	181	Radnorshire	4
10	182	Cardiff	4
10	183	Vale of Glamorgan	4
10	184	Port Talbot (aka Afan)	4
10	185	Lliw Valley	4
10	186	Neath	4
10	277	Swansea	4
10	871	Alyn and Deeside	4
10	872	Colwyn	4
10	873	Delyn	4
10	874	Glyndwr	4
10	875	Rhuddlan	4
10	876	Wrexham Maelor	4
10	977	Carmarthen	4
10	978	Ceredigion	4
10	979	Dinefwr	4
10	980	Llanelli	4
10	981	Preseli	4
10	982	South Pembrokeshire	4
10	983	Blaenau Gwent	4
10	984	Islwyn	4
10	985	Monmouth	4
10	986	Newport	4

11	278	Berwickshire	5
11	279	Ettrick and Lauderdale	5
11	280	Roxburgh	5
11	281	Tweeddale	5
11	282	Clackmannan	5
11	283	Falkirk	5
11	284	Stirling	5
11	285	Annandale and Eskdale	5
11	286	Nithsdale	5
11	377	Stewartry	5
11	378	Wigtown	5
11	379	Dunfermline	5
11	380	Kirkcaldy	5

11	381	North East Fife	5
11	382	Aberdeen City	5
11	383	Banff and Buchan	5
11	384	Gordon	5
11	385	Kincardine and Deeside	5
11	386	Moray	5
11	477	Badenoch and Strathspey	5
11	478	Caithness	5
11	479	Inverness	5
11	480	Lochaber	5
11	481	Nairn	5
11	482	Ross and Cromarty	5
11	483	Skye and Lochalsh	5
11	484	Sutherland	5
11	485	East Lothian	5
11	486	Edinburgh City	5
11	577	Midlothian	5
11	578	West Lothian	5
11	579	Argyll and Bute	5
11	580	Bearsden & Milngavie	5
11	581	Clydebank / Clydesdale	5
11	582	Cumbernauld and Kilsyth	5
11	583	Cumnock and Doon Valley	5
11	584	Cunninghame	5
11	585	Dumbarton	5
11	586	East Kilbride	5
11	677	Eastwood	5
11	678	Glasgow City	5
11	679	Hamilton	5
11	680	Inverclyde	5
11	681	Kilmarnock and Loudoun	5
11	682	Kyle and Carrick	5
11	683	Lanark (aka Clydesdale)	5
11	684	Monklands	5
11	685	Motherwell	5
11	686	Renfrew	5
11	777	Strathkelvin	5
11	778	Angus	5
11	779	Dundee City	5
11	780	Perth and Kinross	5
11	781	Orkney Islands Area	5
11	782	Shetland Islands Area	5
11	783	Western Isles Islands Area	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?
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)

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

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

-1Not applicable to this case.

-2Unable 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) = 12 or 13 (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

CodeCondition

BOARDERFrom ADULT table

If ~~CVPD = 1 to 11~~ **CONVBL=1** get all variables.

If convbl = 1 (answered boarder), and CVPD~~<>12 or 13~~ **equals -1 or 1-11** 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.

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

-2Unable to derive due to value of convbl or cvpay missing or where cvpd = 12 or 13

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 : Publication
Amendments :

1 Definition

This variable is coded as

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

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

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

2 FRS Specification

For each benefit unit

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

BUINC From ADULT table get INEARNS, INSEINC,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

BUINV equals total of all occurrences of ININV and CHINV

BURPINC equals total of all occurrences of INRPINC

BPENINC equals total of all occurrences of INPENINC

BUDISBEN equals total of all occurrences of INDISBEN

BUOTHBEN equals total of all occurrences of INOTHBEN

BURINC equals total of all occurrences of INRINC and CHRINC

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

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 : 21 April 2005
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

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).

-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

Code	Condition
BURENT	From HOUSEHOL table, if TENURE = 1 2 or 3 or 4 or 5 (renting) get HHRENT and HHSTAT variables From BENUNIT table get BOARDER and LODGER variables If HHSTAT = 1, then if BENUNIT=1, and not TENURE=1 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 latter.

If HHSTAT=2

if RENT (of 1ST BU) declared after HB deducted (HBENCHK=2) **and RENTPD, HBENPD and SRENTPD<>12 or 13** then
TOTRENT=RENT+HBENAMT (for 1st BU) + SRENTAMT (for all remaining BUs)

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

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

Above specification assumes that all 12 and 13 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 12 or 13

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 : Househol
Minimum Value :
Maximum Value :
Units : integer
Validations :
Related Variables :CAREAB, CAREAH, CARERE, CAREFR, CARECL, CAREOT, CARECB, CARECH, HOURAB, HOURAH, HOURRE, HOURFR, HOURCL, HOUROT, HOURCB, HOURCH(variables for adults caring)
Children :
Parents :
Core variable/user :
Amendments : JS - 29 October 1996 - to reflect actual coding for V31 which gives total numbers cared for/hours caring by children (including for those living outside the HH)

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 ~~in the household~~ looked after by children and how much time they spend caring.

CHCARE Total number of individuals ~~within the household~~ cared for by child/children within the household (note: outside HH categories may include more than one friend/relative etc)

CHHOUR Total number of hours spent caring ~~for household members~~ by child/children in the household household (note: outside HH categories may include more than one friend/relative etc)

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

-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 in the household needing care ~~(NEEDPER=1-20~~), If WHOLOO11 (child/children doing caring)

```
        CHCARE=CHCARE+1
        CHHOUR=CHHOUR+HOUR(11)
```

-2 If any variables are missing

CHBRECHH

Purpose: This variable is the total weekly amount of Child Benefit received within the Household.
Created: VC - 7 September 1993
Database Table: HOUSEHOL
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: ISRECDHH, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 25 February 1994 To exclude any records with period codes 12 or 13
: JS - 20 February 1996 - to allow skipped values of BENPD where BENAMT has been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

CHBRECHHThis is the total weekly amount of Child Benefit received within the household.

0Not applicable as no Child Benefit received.

-2Unable to derive as missing values

NB The normal FRS code of -1 for not applicable is replaced by 0 for this variable.

This variable is derived by adding together the amount of Child Benefit received by any person in the household. The amount of Child Benefit is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 3 the person is receiving Child Benefit and the amount held in BENAMT for this record should be added into CHBRECHH.

The total should include Child Benefit received only, therefore, if the household receives any other form of benefit in addition to CHB, this would not be added into the total. Households with no CHB are not applicable to this case.

If the period over which CHB is paid is stated to be a one-off/lumpsum or other thta record is set to unable to derive as the benefit amount cannot be converted to a weekly amount by the database.

2 FRS Specification

For each HOUSEHOLD, set CHBRECHH to zero

CodeCondition

CHBRECHHProcess all BENEFITS records for household

If BENPD **equal to -1 or 1 to 11** and

If BENEFIT = 3, add BENAMT into CHBRECHH (CHB)

CHBRECHH will then be the total amount of benefit received from this benefit by a particular household regardless of the number of adults.

0Not applicable as does not have any CHB recipients.

-2Unable to derive as above variables are missing or BENPD = 12 or 13

3 Results

Tabulation is required to show the number of households by the total amount of Child Benefit received split into the following weekly bands

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

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	: Publication
Amendments	:
Issue date	: 21 April, 2005

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 CHAMTST (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 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

Code	Condition
CHINCDV	From CHILD table if CHAMTERN exists then get CHAMTERN set CHEARNS to zero If CHAMTERN exists and CHPDERN equal to -1 or 1-11 then CHEARNS=CHAMTERN From ACCOUNTS table set CHINV to zero If ACCOUNT = 24 get ACCINT and add to CHINV (NSB Post Office - ordinary) If ACCOUNT = 25 get ACCINT and add to CHINV (NSB Post Office - investment) If ACCOUNT = 26 get ACCINT and add to CHINV (Building Society) If ACCOUNT = 27 get ACCINT and add to CHINV (Bank account) If ACCOUNT = 28 get ACCINT and add to CHINV (Gilts) If ACCOUNT = 29 get ACCINT and add to CHINV (Unit Trusts) If ACCOUNT = 30 get ACCINT and add to CHINV (Other stocks/shares) From CHILD table **set CHRINC to zero** If CHTST = 1 and CHPDTST equal to -1 or 1-11 (child has an income from a trust fund) get CHAMTTST and add to CHRINC If TOTGNTCH exists then add TOTGNTCH/52 to CHRINC CHINCDV=CHEARNS+CHINV+CHRINC
-2	Unable to derive if any components are missing

CTBINDBU

Purpose: To indicate if the benefit unit has received Council Tax Benefit of more than : 5 pence.
Created: VC - 10 September 1993
Database Table: BENUNIT
Minimum Value: 0
Maximum Value: 1
Units: Integer
Validations:
Related Variables:
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 11 February 1994 To clarify which variable from version 30 is used to : calculate CTBINDBU as did not have documentation earlier
: VC - 25 February 1994 To exclude any records which have a period code of : 12 or 13
: JS - 20 February 1996 to allow skipped values of CTREBPD where CTREBAMT has been imputed
Issued: 21 April 2005

NB - this is for version 30

1 Definition

This variable is coded as

CTBINDBUThis variable indicates whether the head of household benefit unit has received Council Tax Benefit of more than 5p in connection with their payment of Council Tax.

1Head of household benefit unit receiving CTB

0Other benefit units not receiving CTB.

-1Not applicable to this case

-2Unable to derive.

This indicator will only be valid for the head of household benefit unit and all other subsequent benefit units in the household will be coded as 0. The variable will be set to zero for all benefit units which have not received any Council Tax Benefit and 1 when the benefit has been received. If the amount of CTB received by the household is more than 5p, CTBINDBU will be set to 1. If there is no CTB received or the amount is less than 5p the variable is left at zero.

Any records which have a period of lumpsum/one-off or other declared for the amount of CTB - where CTREBPD = 12 or 13.

2 FRS Specification

For each BENEFIT UNIT, set CTBINDBU to zero

CodeCondition

CTBINDBUIf BENUNIT = 1 (head of household), CTREBPD ~~not equal 12 or 13~~ **equals -1, 1-11** and CTREBAMT > 0.05 (from HOUSEHOL record), set CTBINDBU = 1

-1Not applicable to this case - shouldn't be any as all BUs should be 0 or 1.

-2Unable to derive as missing values or CTREBPD = 12 or 13.

3 Results

Tabulation will be required to show the number of benefit units receiving and not receiving CTB.

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

1Definition

CWATAMTDThe 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) 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).

2FRS Specification

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

Code Condition

CWATAMTDFrom RENTER table

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 (respondent has a formal exemption from the council tax) or CTEXREB=3 (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 (combinations of discounts) or CTEXREB=4 (bill not yet issued) then apply look up table to calculate CWATAMTD

Else ifCTBAND=0 (not valued separately) or CTEXREB=1 (respondent has a formal exemption from the council tax) or CTEXREB=3 (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 (combinations of discounts) or CTEXREB=4 (bill not yet issued) 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.

Else apply look up table (final mop up of any missing 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: ASD6A
Issue date: 21 April 2005
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

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

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

-1Not applicable to this case

-2Unable to derive variable

DEDUCTS is derived from the variables AMTTAXF, AMTOTH, OTHDED3 - OTHDED6, DEDUC3 - DEDUC5 and DEDOTH. AMTTAXF and AMTOTH deal with deductions paid to charities and OTHDED3 - OTHDED6 indicate whether payments are made to Trade Unions, Friendly Societies Sports and Social clubs or any others not previously mentioned. DEDUC3 - DEDUC5 and DEDOTH store the amounts paid.

OTHDED3 - OTHDED6 are database variables produced from OTHDED, which when = 1 indicate whether that person has a particular deduction. DEDUC3 - DEDUC5 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 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 refer to one-off/lumpsum deductions or any other period. If the variables cannot be converted the record must be rejected.

Where variables PAYAMT, CHRTAXF and OTHDED1-6 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 over all mean for 1994/95 (this will need to be updated annually)
iiiassume where PAYAMT has been imputed (and CHRTAXF and OTHDED1-6 skipped) no deductions are made

2 FRS Specification

For each ADULT

CodeCondition

DEDUCTSFrom JOB table, for all jobs a person has

If PAYPD ~~not equal 12 or 13~~ **equals -1 or 1 to 11** do the following -

If CHRTAXF = 1 (any deductions made from wage for charities under tax free scheme? ~~get~~ **DEDUCTS=**AMTTAXF (how much deducted) **else if AMTTAXF=skipped DEDUCTS=DEDUCTS+1.27** and if CHROTH = 1 (any other charitable deductions) get AMTOTH (amount of any other deduction)
else if CHRTAXF=-1 DEDUCTS=DEDUCTS (ie no change)
If OTHDED3 = 1~~, get amount of deduction from~~ **DEDUCTS=DEDUCTS+**DEDUC3 (amount of union fees) **else if OTHDED3=-1 DEDUCTS=DEDUCTS**
If OTHDED4 = 1~~, get amount of deduction from~~ **DEDUCTS=DEDUCTS+**DEDUC4 (amount for friendly societies) **else if OTHDED4=-1 DEDUCTS=DEDUCTS**
If OTHDED5 = 1~~, get amount of deduction from~~ **DEDUCTS=DEDUCTS+**DEDUC5 (amount for sports clubs or specialised pastimes) **else if OTHDED5=-1 DEDUCTS=DEDUCTS**
If OTHDED6 = 1, ~~get amount of deduction from~~ **DEDUCTS=DEDUCTS+**DEDOTH (amount of any other deductions not included above excluding tax/NI) **else if OTHDED6=-2 DEDUCTS=DEDUCTS**

~~DEDUCTS will be the sum of any occurrences of AMTTAXF, AMTOTH, DEDUC3, DEDUC4, DEDUC5~~

~~AND DEDOTH.~~

-1Not applicable to this case.

-2Unable to derive as any of the above variables are missing or PAYPD = 12 or 13 (one-off/lumpsum or other

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

DISBENHH

Purpose: This variable is the total weekly amount currently received from disability : benefits by all memebers of the household.
Created: VC - 3 September 1993
Database Table: HOUSEHOL
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: ISRECDHH, FCRECDHH...
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 25 February 1994 To exclude any records where period code is 12 or : 13
: JS - 20 February 1996 to update for V31 and allow skipped period codes where BENAMT has been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

DISBENHHThis is the total weekly amount currently received from disability benefits by each person in the household. Disability benefits include Invalidity Benefit/Pension, Severe Disablement Allowance, Disability Living Allowance (care component), Disability Living Allowance (mobility component) and Attendence Allowance.

0Not applicable to this case - no disability benefits received.

-2Unable to derive as missing values

This variable is derived by adding together the amount of each of the above benefits received by any person in the household. The amount of each benefit is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 1 or 2 or 9 or 11 or 17, the person is receiving one of the disability benefits (IVB, SDA, DLA (CARE), DLA (MOB) and AA) and the amount held in BENAMT for that record should be added into DISBENHH.

~~Some of these benefits will be currently in payment as a result of the way in which the questions are asked, however, others are asked if the person has received it in the last 12 months. If this is the case an additional question is asked (PRES) which asks if the benefit is received at present. This is the case for IVB.~~

Where a benefit may paid in respect of a child (DLA care or mobility), the benefit record is attached to an adult record but indicates which person it is paid for. It is then picked up as another benefit record for that adult and added in as usual.

However, if the period code for the benefit 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 BENPD = 12 or 13 DISBENHH is set to -2.

In cases where BENAMT has been imputed, BENPD will be skipped. These cases need to be allowed for in the specification

2 FRS Specification

For each HOUSEHOLD, set DISBENHH to zero

<u>CodeCondition</u>

DISBENHHProcess all BENEFITS records for household

.If BENPD ~~not equal 12 or 13~~ **equals -1, 1-11** and

If BENEFIT = 1, add BENAMT into DISBENHH (DLA Care)

If BENEFIT = 2, add BENAMT into DISBENHH (DLA Mob)

If BENEFIT = 9, add BENAMT into DISBENHH (SDA)

If BENEFIT = 11, add BENAMT into DISBENHH (AA)

If BENEFIT = 17 ~~and PRES = 1,~~ add BENAMT into DISBENHH (IVB).

DISBENHH will then be the total amount of benefit received from these benefits by a particular household regardless of the number of adults/children.

0Not applicable to this case - any household which does not receive any of the above benefits.

-2Unable to derive due to any of above variables having missing values or BENPD = 12 or 13.

3 Results

Tabulation is required to show the number of households by the total amount of disability benefits received split into the following weekly bands

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 or over.

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
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

1 Definition

This variable is coded as

11 person in benefit unit blind.

22 people in benefit unit blind.

31 person in benefit unit disabled.

42 people in benefit unit disabled.

51 blind person and 1 disabled person in benefit unit.

6No person in benefit unit blind or disabled.

-2Unable 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 ~~FUTATT~~**B2QFUT3** = 1 = dis (temporary variable)

(preset temporary variables to 0)

<u>CodeCondition</u>

1If blind = 1 and dis = 0

2If blind = 2 and dis = 0

3If blind = 0 and dis = 1

4If blind = 0 and dis = 2

5If blind = 1 and dis = 1

6Any other benefit unit not previously coded (where blind = 0 and dis = 0)

-2Unable to derive because any of the above variables have missing values.

3 Results

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

4 Test Cases

To be added at a later date.

ECOTYPBU

Purpose: To derive an HBAI type economic status indicator for each benefit unit.
Created:
Database Table:
Minimum Value: 1
Maximum Value: 8
Units: Integer
Validations:
Related Variables:
Children:
Parents:
Core variable/user: ASD6A HBAI
Amendments:
Issued:

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 (~~this can be a temporary variable and does not need to appear on the flatfile~~ **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 hours.

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

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

EMPSTATC is set as unemployed by the following conditions:

where an individual is in receipt of UB (~~BENEFIT=13 and PRES=1~~ **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
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

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

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

4(from benefit record ~~BENEFIT=13 and PRES=1~~ **BEN3Q1=1**) or (
(LOOK4=1 and START=1) or
(LKYT4=1 and AGE<60 and START=1) or
WAIT=1 or
JOBAWAY=3 or
or TRAIN = 1,2,4,5,6)

3(TRAIN <> 1,2,4,5,6) and (EMPSTAT=1 or EMPSTAT=2) and
(ABSPAY<>2,3 and, from first job record, QHRS+EMPOVT<30) or (from first job record QHRSELF<30)

2(TRAIN <> 1,2,4,5,6) and EMPSTAT=1 and ABSPAY<>2,3 and, from first job record, QHRS+EMPOVT>29

1(TRAIN <> 1,2,4,5,6) and EMPSTAT =2 and, from first job record, QHRSELF>29

ECOTYPBU

For each Benefit Unit:

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

Code

8All 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)

6AGE>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:
Minimum Value: 1
Maximum Value: 9
Units: Integer
Validations:
Related Variables:
Children:
Parents:
Core variable/user: Publication
Amendments:
Issued:

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 follows (~~this can be a temporary variable and does not need to appear on the flatfile~~) **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 hours or more in their main job (QHRSELF>29) and if they are not on a Govt Training Prog (these should all be set as unemployed, except if they're on Enterprise Allowance).

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

EMPSTATC is set as unemployed by the following conditions:

where an individual is in receipt of UB~~ (BENEFIT=13 and PRES=1~~ **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
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

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

ECSTATBU is coded as:

1Self employed (benefit units where at least one adult usually works self-employed for 30 or more hours a week)

2Single or couple, all in full time work *as an employee* (30 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

8If (((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)

6AGE>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).

EMPISM (IN259)

Purpose: To give FES equivalent employment status for use in the ISM
Created: May 1995 as part of parallel run exercise
Database Table: ADULT
Minimum Value: 0
Maximum Value: 8
Units: Integer
Validations:
Related Variables:
    Children:
    Parents: EMPSTATB
Core variable/user: ISM
Amendments:
Issued: 21 April 2005

1 Definition

This variable is coded as

0 not recorded
1employee
2self employed
3unemployed seeking work
4unemployed about to work
5sick
6retired
7 unoccupied
8 government training scheme

2 FRS Specification

For each adult record

<u>CodeCondition</u>

0 EMPSTATB=-1,-2,13

1EMPSTATB=2,3,4,5,6

2EMPSTATB=1

3From ADULT table where age<65 and sex=1 or age<60 and sex=2
    If working=2 and jobaway=3 and lookwk=1,2,3
    If working=2, jobaway=2 and look4=1
    If working=2, jobaway=2 and lkyt4=1
    if working=2, jobaway=2, likewk=1 and nolook=1

4	From ADULT table where age<65 and sex=1 or age<60 and sex=2 If working=2 and jobaway=2 and wait=1
5	EMPSTATB=11,12
6	EMPSTATB=9
7	EMPSTATB=10
8	EMPSTATB=8

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
Issue date : 21 April 2005
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

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
-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).

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 or 8}

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 or 8}

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 = 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 6 or 7 or 8
If Working = 2, Jobaway = 2, Likewk = 2 and Nowant = 3 or 6 or 7 or 8

If Working = 2, Jobaway = 2 and Nolk0002 = 1. (Includes those not working as looking after children)

11 Code 11 for the following - for all adults below pension age
From ADULT table,
If Working = 2, Jobaway = 2, Likewk = 1, Nolook = 4 or
If Working = 2, Jobaway = 2, Likewk = 2, Nowant = 4 or
If Injlong = 1.

12 As above from ADULT table,
If Working = 2, Jobaway = 2, Likewk = 1, Nolook = 5 or
If Working = 2, Jobaway = 2, Likewk = 2, Nowant = 5 or
If Injlong = 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

-1 Not applicable to this case

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

Remaining cases which are set to unable to derive due to missing values (5 cases) should be set to EMPSTATB=10. One case should be set to EMPSTATB=2 SERNUM and BENUNIT numbers are:

sernum	**benunit**	**person**	**empstatb**
1794221	**1**	**1**	**10**
4484181	**1**	**1**	**2**
9344241	**1**	**1**	**10**
16964191	**1**	**1**	**10**
104261	**1**	**2**	**10**
16964191	**1**	**2**	**10**

These need to be hard coded in to the V31 program for EMPSTATB to avoid them being set to unable to derive when the program is rerun.

<u>Key to above coding</u>

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

Jobaway If not doing paid work, do you have a job that away from last 7 days. 1 = yes, 2 = no, 3 = waiting take up new job.

Empstat1 = employed
2 = self-employed (including Enterprise Allowance)

Tdaywrk Are you going to work today. 1 = yes, 2 = no although normal working day, 3 = no not normal working day.

Tothours Number of hours worked by an employee - a derived variable in itself.

Abswk Have you been away from work for more than 3 working days. 1 = yes, 2 = no.

Abswhy Reason for absence. 1 = pattern of shifts, 2 = illness/accident, 3 = holiday, 4 = strike, 5 = Laid off, 6 = maternity leave, 7 = compassionate leave, 8=other.

Injlong How long been unable to work due to illness/injury. 1 = 6 months or less, 2 = more than 6 months.

Look4 Looking for work in last 4 weeks. 1 = yes, 2 = no.

Lkyt4 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 = log-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 nusery 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)
Children:
Parents:
Core variable/user: ~~ISM HBM PSM FCM~~ **ASD6A**
Issue Date: 21 April 2005
Amendments: VC 9 June 1993 To put people who are on holiday from their normal place : of work into category 1

1 Definition

This variable is coded as

1Employee (includes people not at work due to pattern of shifts)
2Self-employed
3Employee temporarily not at work (including those less than 28 weeks sick, laid off, on strike or on maternity leave)
4Work-related Government Training
5Unemployed (ILO definition)
6Inactive
-1Not applicable to this case
-2Unable to derive due to missing values

EMPSTATI is derived from several variables in the ADULT table of the FRS database.

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.

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.

The variable TRAIN indicates whether or not a person is on a work-related government training scheme and codes 1 to 5 indicate a government scheme (1 = ET, 2 = Youth Training, 3 = Voluntary Projects Programme, 4 = Community Industry, 5 = Other government programme).

The ILO definition of unemployment is available for and actively seeking work. So 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. LOOK4 = 1 indicates that the person is looking for work 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.

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

<u>CodeCondition</u>

1If WORKING = 1 and EMPSTAT = 1 and TDAYWRK = 1 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)

2If WORKING = 1 and EMPSTAT = 2 or
If JOBAWAY = 1 and EMPSTAT = 2

3If 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
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.

4If TRAIN = 1, 2, 3, 4, or 5

5If WORKING = 2, JOBAWAY = 2 and
If LOOK4 = 1 or (looking for work)
If LKYT4 = 1 or (looking for govt scheme)
If WAIT = 1 or (waiting start job)
If LIKEWK = 1 and NOLOOK = 1 (would like regular work and reason not looking = waiting start new job/business)

6Anyone else not previously coded.

-1Not applicable (should not be relevant to EMPSTATI)

-2Unable 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.
Issued: 21 April 2005

1 Definition

This variable is coded as

EQAHCBUThis 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 relevant values from the following table are accumulated to provide the benefit unit scale.

1st adult (head of household)0.55

spouse of head0.45

heads of subsequent units0.45
spouse of any subsequent units0.45

each dependent aged0 - 10.07
2 - 40.18
5 - 70.21
8 - 100.23
11 - 120.26
13 - 150.28
16 +0.38

-2Unable 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 of 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 Condition</u>

EQAHCBUProcess each ADULT record and accumulate the following for each adult in the benefit unit.

If BENUNIT = 1 and HEAD = 1 (head of household) add0.55
If BENUNIT = 1 and SPOUSE = 1 (spouse/partner) add0.45

If BENUNIT >= 2 and HEAD = 1 add0.45
If BENUNIT >= 2 and SPOUSE = 1 add0.45

Process each CHILD record and accumulate the following for each child in the benefit unit.

If AGE <= 1 add0.07
If AGE >= 2 and <= 4 add0.18
If AGE >= 5 and <= 7 add0.21
If AGE >= 8 and <= 10 add0.23
If AGE >= 11 and <= 12 add0.26
If AGE >= 13 and <= 15 add0.28
If AGE >= 16 add0.38

-2If 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
Issued: 21 April 2005

1 Definition

This variable is coded as

EQBHCBUThis 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 relevant values from the following table are accumulated to provide the benefit unit scale.

1st adult (head of household)0.61

spouse of head0.39

heads of subsequent units0.46
spouse of any subsequent units0.42

each dependent aged0 - 10.09
2 - 40.18
5 - 70.21
8 - 100.23
11 - 120.25
13 - 150.27
16 +0.36

-2Unable 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 of 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

Code Condition

EQBHCBUProcess each ADULT record and accumulate the following for each adult in the benefit unit.

If BENUNIT = 1 and HEAD = 1 (head of household) add0.61
If BENUNIT = 1 and SPOUSE = 1 (spouse/partner) add0.39

If BENUNIT >= 2 and HEAD = 1 add0.45
If BENUNIT >= 2 and SPOUSE = 1 add0.42

Process each CHILD record and accumulate the following for each child in the benefit unit.

If AGE <= 1 add0.09
If AGE >= 2 and <= 4 add0.18
If AGE >= 5 and <= 7 add0.21
If AGE >= 8 and <= 10 add0.23
If AGE >= 11 and <= 12 add0.25
If AGE >= 13 and <= 15 add0.27
If AGE >= 16 add0.36

-2If 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.
Issued: 21 April 2005

1 Definition

This variable is coded as

EQUIVAHCThis 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 relevant values from the following table are accumulated to provide the household scale.

1st adult (head of household)0.55

spouse of head0.45
orother 2nd adult0.45

3rd adult0.45
orany subsequent adult0.40

each dependent aged0 - 10.07
2 - 40.18
5 - 70.21
8 - 100.23
11 - 120.26
13 - 150.28
16 +0.38

-2Unable 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

Code Condition

EQUIVAHCProcess each ADULT record and accumulate the following for each adult in the household

If PERSON = 1 (head of household) add0.55
If PERSON = 2 and MS = 1 or 3 (spouse/partner) add0.45
If PERSON = 2 and MS not equal 1 or 3 add0.45
If PERSON = 3 add0.45
If PERSON > 3 add for each subsequent adult0.40

Process each CHILD record and accumulate the following for each child in the household

If AGE <= 1 add0.07
If AGE >= 2 and <= 4 add0.18
If AGE >= 5 and <= 7 add0.21
If AGE >= 8 and <= 10 add0.23
If AGE >= 11 and <= 12 add0.26
If AGE >= 13 and <= 15 add0.28
If AGE >= 16 add0.38

-2If 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.
Issued: 21 April 2005

1 Definition

This variable is coded as

EQUIVBHCThis 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 relevant values from the following table are accumulated to provide the household scale.

1st adult (head of household)0.61

spouse of head0.39
orother 2nd adult0.46

3rd adult0.42
orany subsequent adult0.36

each dependent aged0 - 10.09
2 - 40.18
5 - 70.21
8 - 100.23
11 - 120.25
13 - 150.27
16 +0.36

-2Unable 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

Code Condition

EQUIVBHCProcess each ADULT record and accumulate the following for each adult in the household

If PERSON = 1 (head of household) add0.61
If PERSON = 2 and MS = 1 or 3 (spouse/partner) add0.39
If PERSON = 2 and MS not equal 1 or 3 add0.46
If PERSON = 3 add0.42
If PERSON > 3 add for each subsequent adult0.36

Process each CHILD record and accumulate the following for each child in the household

If AGE <= 1 add0.09
If AGE >= 2 and <= 4 add0.18
If AGE >= 5 and <= 7 add0.21
If AGE >= 8 and <= 10 add0.23
If AGE >= 11 and <= 12 add0.25
If AGE >= 13 and <= 15 add0.27
If AGE >= 16 add0.36

-2If any of above values are missing

3 Results

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

FAMTHBAI

Purpose : This is the family type used for HBAI purposes for each benefit unit.
Created : VC - 9 September 1993
Database Table : BENUNIT
Minimum Value : 1
Maximum Value : 6
Units : Integer
Validations :
Related Variables :
Children :
Parents :
Core variable/user : HBAI
Amendments :to fit in with HBAI definitions: FAMTHBAI =1 when head of benefit unit over pensionable age; state pension definition
: JS - 13 March 1996 - to tighten up definitions and amend spec in line with ASD6A changes
Issued : 21 April 2005

1 Definition

This variable is coded as

FAMTHBAI This is the definition of family type used by HBAI.

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

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

According to the HBAI publication, pensioner/non-pensioner singles/couples are where they are headed by 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.

This causes problems for cases where 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

Code Condition

1 If MS=1,3,4,5 6 or 7 and (second adult in the benefit unit exists) and man in the benefit unit is over 65 [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=2 and SEX=1 and AGE>=65)]

or 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 MS=married, spouse not in household*

2 If MS=1,3,4,5,6 or 7 and (only one adult in the benefit unit) and person is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60)]

3 If MS= 1,3,4,5,6 or 7 and (second adult in the benefit unit exists) and depchildb > 0
or MS=2 and depchildb>0

4 If MS=1,3,4,5,6 or 7 and (second adult in the benefit unit exists) and depchildb=0
or MS=2 and depchildb=0

5 If MS=1,3,4,5,6 or 7 and only one adult in the benefit unit and depchildb>0

6 If MS=1,3,4,5,6,7 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.

FAMTYPBU

Purpose : This is the family type used for ~~HBAI~~ **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
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
Issued : 21 April 2005

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 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

Code Condition

1 If MS=1,3,4,5 6 or 7 and (second adult in the benefit unit exists) and ~~man in~~ **the head of** the benefit unit is over ~~65~~ **pension age** [ie (UPERSON=1 and SEX=1 and AGE>=65) or ~~(UPERSON=2 and SEX=1 and AGE>=65)~~ **(UPERSON=1 and SEX=2 and AGE>=60)**]

or 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 MS=married, spouse not in household*

2 If MS=1,3,4,5,6 or 7 and (only one adult in the benefit unit) and person is over pension age [ie (UPERSON=1 and SEX=1 and AGE>=65) or (UPERSON=1 and SEX=2 and AGE>=60)]

3 If MS= 1,3,4,5,6 or 7 and (second adult in the benefit unit exists) and depchildb > 0
or MS=2 and depchildb>0

4 If MS=1,3,4,5,6 or 7 and (second adult in the benefit unit exists) and depchildb=0
or MS=2 and depchildb=0

5 If MS=1,3,4,5,6 or 7 and only one adult in the benefit unit and depchildb>0

6 If MS=1,3,4,5,6,7 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
Issue date : 21 April 2005
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

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 = 1 (married spouse in household), MS = 2 (married spouse not in household), MS = 3 (cohabiting). Single people will be the rest - MS = 4 (single, never married), MS = 5 (widowed), MS = 6 (separated) and MS = 7 (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 depchildb, 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 (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

<u>CODE</u> <u>CONDITION</u>

1 If under pension age and DEPCHILDB > 0 and **((second adult record in BU exists) or MS=2 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 **((second adult record in BU exists) or MS=2 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 **((and second adult record in BU exists) or MS=2 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 **((second adult record in BU exists) or MS=2 where single adult)**
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.

FCRECDBU

Purpose: This variable is the total weekly amount currently received from Family Credit : by each person in the benefit unit.
Created: VC - 7 September 1993
Database Table: BENUNIT
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: ISRECDHH, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 11 February 1994 Amendments to reflect version 30 changes
: VC - 25 February 1994 To exclude any records where period code is 12 or : 13
: BS - 4 August 1995. Amended to take into account changes to V31 of the Questionnaire.
: JS - 20 February 1996 to allow for skipped variables where values have been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

FCRECDBUThis is the total weekly amount currently received from Family Credit by each person in the benefit unit.

0Not applicable to this case

-2Unable to derive as missing values

This variable is derived by adding together the amount of Family Credit received by any person in the benefit unit. The amount of Family Credit is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 18 the person is receiving Family Credit and the amount held in BENAMT for that record should be added into FCRECDBU.

~~However, the value in BENAMT may be for a benefit which was received during the last 12 months and is not currently received. An additional variable must be checked to ensure that FC is currently in receipt, where PRES = 1. (This condition is no longer needed in V31 as the question has been changed to ask if FC is currnetly being received.)~~

However, if the period code for the benefit 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 BENPD = 12 or 13 FCRECBU is set to -2.

Where BENAMT has been imputed, BENPD will be set to skipped. These cases have to be

allowed for in the specification

The total should include Family Credit received only, therefore, if the benefit unit receives any other form of benefit in addition to FC, this would not be added into the total. Benefit units with no FC are not applicable to this case.

2 FRS Specification

For each BENEFIT UNIT

<u>CodeCondition</u>

FCRECDBUProcess each BENUNIT record, set FCRECDBU to zero

Process all BENEFITS records for that BENEFIT UNIT (via benunit)

If BENPD ~~not equal 12 or 13~~ **equals -1, 1-11** and

If BENEFIT = 18, add BENAMT into FCRECDBU

FCRECDBU will then be the total amount of benefit received from FC by the benefit unit.

0Not applicable as does not have any FC recipients

-2Unable to derive as above variables are missing or BENPD = 12 or 13

3 Results

Tabulation is required to show the number of benefit units by the total amount of Family Credit received split into the following weekly bands

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 or over

FCRECDHH

Purpose: This variable is the total weekly amount received from Family Credit by : each person in the household.
Created: VC - 7 September 1993
Database Table: HOUSEHOL
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: ISRECDHH, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 25 February 1994 To exclude records where period code is 12 or 13.
: BS - 4 August 1995
: JS - 20 February 1996 to allow for skipped period codes where benamt has been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

FCRECDHHThis is the total weekly amount received from Family Credit by each person in the household.

0Not applicable to this case

-2Unable to derive as missing values

This variable is derived by adding together the amount of Family Credit received by any person in the household. The amount of Family Credit is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 18 the person is receiving Family Credit and the amount held in BENAMT for that record should be added into FCRECDHH.

~~However, the value in BENAMT may be for a benefit which was received during the last 12 months and is not currently received. An additional variable must be checked to ensure that FC is currently in receipt, where PRES = 1. (This condition is no longer needed in V31 as the question has been changed to ask if FC is currnetly being received.)~~

The total should include Family Credit received only, therefore, if the household receives any other form of benefit in addition to FC, this would not be added into the total. Households with no FC are not applicable to this case.

However, if the period code for the benefit 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 BENPD = 12 or 13 FCRECDHH is set to -2.

where BENAMT is imputed, BENPD will be skipped and these values have to be allowed for in the specification

2 FRS Specification

For each HOUSEHOLD, set FCRECDHH to zero

CodeCondition

FCRECDHHProcess all BENEFITS records for household

If BENPD ~~not equal 12 or 13~~ **equals -1 or 1-11** and

If BENEFIT = 18, add BENAMT into FCRECDHH

FCRECDHH will then be the total amount of benefit received from this benefit by a particular household regardless of the number of adults.

0Not applicable as does not have any FC recipients

-2Unable to derive as above variables are missing.

3 Results

Tabulation is required to show the number of households by the total amount of Family Credit received split into the following weekly bands

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 or over

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

1 Definition

This variable is coded as

FOSTERBUThe total number of foster children in a benefit unit.

-1Not applicable to this case.

-2Unable 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 whther 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>CodeCondition</u>

FOSTERBUFrom CHILD table, for each child in benefit unit
If foster = 1, compute count of foster children for that benefit unit.

-1Not applicable to this case - where foster = 2

-2Unable 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
Issued: 21 April 2005

1 Definition

This variable is coded as

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

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

FSMHHThis is the total value of any free school meals received by the household.

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

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

The value of free school meals received by each dependent is derived from the SCHMEAL and SMLIT variable from the CHILD record. Where SCHMEAL = 1 (has some free school meals) the number of free meals is obtained from SMLIT. This amount is then multiplied by the cost of a school meal, which is to be found in the Tax Benefit Model for ~~1993~~ **1994** to produce the total amount spent each week.

The total amount of free school meals for a week (from the TBM) is ~~£3.96~~ **£4.04** 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 = £~~1.03~~ **1.0504** per day.

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

CodeCondition

FSMVALFor each CHILD from CHILD table,

Set COST (of free school meals) to ~~£1.03~~ **1.0504** (supplied to FES from Tax Benefit Model for ~~1993~~ **1994**)

If SCHMEAL = 1, calculate the value of free school meals

calculate FSMVAL = SMLIT * COST

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.

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

-2Unable 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
Issued: 21 April 2005

1 Definition

This variable is coded as

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

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

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

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

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

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

The cost of a bottle of free school milk is calculated using the same method that CSO uses for FES. It may need updating each year so check with FES first before running. CSO use the cost of a third of a pint/bottle of milk at ~~33p~~ **34p** per bottle/carton. This gives a cost of a third of a pint to be ~~.1089p~~ **£0.1133**.

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

CodeCondition

FSMLKVALFor each CHILD from CHILD record.

If SCHMILK = 1, calculate the value of free school milk

calculate FSMLKVAL = SMKIT x cost of free school milk (.~~1089~~ **0.1133**)

If SCHMILK = 2, calculate FSMLKVAL = 0.

FSMLKBUFor each BENEFIT UNIT (for BENUNIT record)

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

FSMLKHHFor each HOUSEHOLD (for HOUSEHOL record)

sum each occurrence of FSMLKVAL for each child in household.

0Not applicable - no school milk/no dependents

-2Unable 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.
Issued: 21 April 2005

1 Definition

This variable is coded as

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

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

FWMLKHHThis is the total value of any free welfare milk received by any person in the household.

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

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

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

~~The cost of free welfare milk produced from TBM is 2.59 per week giving an amount of 0.37 for a pint/carton of welfare milk.~~

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 is 34p.

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

CodeCondition

FWMLKVALFor each ADULT from ADULT record.

If WELFMILK = 1, calculate the value of free welfare milk

calculate FWMLKVAL = WMKIT x cost of free welfare milk (~~0.37 for 1993~~ **.34 for 1994/95**)

If WELFMILK = 2, calculate FWMLKVAL = 0.

For each CHILD from CHILD record.

If WELFMILK = 1, calculate the value of free welfare milk

calculate FWMLKVAL = WMKIT x cost of free welfare milk (~~0.37 for 1993~~**.34 for 1994/95**)

If WELFMILK = 2, calculate FWMLKVAL = 0.

FWMLKBUFor each BENEFIT UNIT (for BENUNIT record)

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

FWMLKHHFor each HOUSEHOLD (for HOUSEHOL record)

sum each occurrence of FWMLKVAL for each adult and child in household.

0Not applicable - no welfare milk

-2Unable 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

4 Test Cases

None produced as yet.

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 : ASD6A
Amendments :

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

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

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.

Derivation

The grossing system represented by GROSS is just one of the many tested by ASD6A. 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. ASD6A can then advise on the use of software and on the statistical considerations in choosing a grossing regime.

To derive GROSS, 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 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
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 estimates
Council Tax Band	A, B, C-D, E-H	DoE estimates
Region	London, other	DoE 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. Otions 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 ASD6.

Control totals

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

Control Totals for GROSS	
Single men <35	4,217,000
Single men 35-59	1,796,000
Single men 60+	1,128,000
Single women <35	3,134,000
Single women 35-64	2,257,000
Single women 65+	2,953,000
Couples <65	11,185,000
Couples 65+	2,625,000
Male lone parents	112,000
Female lone parents	1,433,000
Number of families with children	5,410,000
LA renters	4,540,000
Private renters	3,155,000
Owner occupiers	15,446,000
CT Band A	6,279,000
CT Band B	4,634,000
CT Band C-D	8,103,000
CT Band E-H	4,125,000
Households in London	2,940,000
Households in other regions	20,201,000

GROSSINC

Purpose:To indicate the amount of gross income received by an adult based on the : FES variable used by HBAI.
Created: VC - 1 November 1993
Database Table: ADULT
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables:
Children:
Parents: NETINC
Core variable/user: HBAI
Amendments: VC - 11 February 1994 Amendments to reflect version 30 changes
:VC - 28 February 1994 To exclude any amounts with period code 12 : or 13
:VS - 28 April 1995 to update to reflect the chages which occurred to the : grossinc variable in February/March 1995.
:BS - 1 August 1995 :- amended to take into account changes to V31
Issued:21 April 2005

1 Definition

This variable is coded as

GROSSINCThe total amount of gross income received by an adult from all sources.

0Not applicable as adult does not have any gross income.

-2Unable to derive due to missing values.

The total amount of gross income is derived from numerous variables from the ADULT, JOB, BENEFITS, ODDJOB and PENSIONS records which when added together form the person's total gross income. It includes gross normal earnings (cf with GROSSPAY which is last pay), self-employed earnings, tax paid on pensions annuities, other income in the form of benefit income, income in kind, royalties, other allowances, income from trust funds and odd jobs etc. From this total an adjustment will be made by deducting an amount for SSP/SMP and for any Social Fund Loans a person may have.

However, if the period code for the benefit 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, for example, if PAYPD = 12 or 13 GROSSINC is set to -2.

2 FRS Specification

For each ADULT

CodeCondition

GROSSINC**Gross earnings**

From ADULT record, set EARNS to zero

If WORKING = 1 or JOBAWAY = 1 - process each JOB record for that person and

If EMPEE = 1

If PAYUSL = 1 or 3 and PAYSLIP = 1
If GRWAGE exists and PAYPD not equal 12 and 13, add it into EARNS
If it is missing set grossinc -2

If PAYUSL = 1 or 3 and PAYSLIP = 2 and PAYPD not equal 12 and 13
If PAYAMT exists, add it into EARNS.
If it is missing set grossinc -2

If PAYE exists, add it into EARNS.
If it is missing set grossinc -2

If NATINS exists, add it into EARNS.
If it is missing set grossinc -2

Then if CHARITY = 1 and CHRTAXF = 1
if AMTTAXF exists add it into EARNS
else set grossinc -2
if CHARITY = 1 and CHROTH = 1
if AMTOTH exists add it into EARNS
else set grossinc -2

If OTHDED1 = 1 add DEDUC1 to EARNS
If OTHDED2 = 1 add DEDUC2 to EARNS
If OTHDED3 = 1 add DEDUC3 to EARNS
If OTHDED4 = 1 add DEDUC4 to EARNS
If OTHDED5 = 1 add DEDUC5 to EARNS
If OTHDED6 = 1 add DEDOTH to EARNS

If PAYUSL = 2 and PAYSLIP = 1 or 2
If exists UGROSS, add it into EARNS
If it is missing set grossinc to -2

If PAYUSL not exist and it is missing don't know or refusal set grossinc to -2.

Tax

For each ADULT, set TAX to zero and then process PENSION record

If PAYPD not equal 12 or 13 and PENTAX = 1 and PTINC = 2, calculate TAX = PTAMT.

2. Adjustment for possible receipt of SSP or SMP

Set ADJUST to zero (temporary variable)

If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYSLIP = 1

Calculate ADJUST = SSPAMT

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

If ADJUST >= to employment income calculated in EARNS, reset ADJUST to zero.

3. Self - employment income

Set SEINC to zero
If EMPEE = 2, calculate SEINC = INCSE1

Any cases with no self-employed earnings set SEINC to zero.

4. Other income

Set OTHINC to zero

Income as a baby-sitter

From ADULT record, if BABY1 = 1 add BABPAY into OTHINC

Income as a mail order agent

From ADULT record, if BABY2 = 1 add BABPAY into OTHINC

Allowance from absent spouse

From ADULT record, if ABSPAR = 1 and APPD not equal 12 and 13, add APAMT to OTHINC.

Allowances from spouse in forces, friends other relatives etc

From ADULT record, if ALLOW1 = 1 and APPD not equal 12 and 13, add ALLPAY1 to OTHINC.

Allowance from an organisation

From ADULT record, if ALLOW2 = 1 and ALLPD1 not equal 12 or 13, add ALLPAY2 to OTHINC.

Allowance from a Local Authority for a foster child

From ADULT record, if ALLOW3 = 1 and ALLPD2 not equal 12 or 13 add ALLPAY3 to OTHINC.

Allowance from a Local Authority for an adopted child.

From ADULT record, if ALLOW4 = 1 and ALLPD3 not equal 12 or 13, add ALLPAY4 to OTHINC.

Income from boarders/lodgers

From ADULT record, if CVPAY exists for any person in household and CVPD not equal 12 or 13, attribute income to head of household - assume this is PERSON = 1.

Income in kind

From JOB record, if LUNCHV = 1 and LV7DY = 1, add amount in LVAMT into OTHINC (luncheon vouchers)

From ADULT record, if FCASH = 1 and FCAMTPD not equal 12 or 13, add amount in FCAMT into OTHINC (cash in lieu of concessionary coal)

Royalties

From ADULT record, if ROYAL1 = 1 add ROYYR1 into OTHINC.

Income as a sleeping partner

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

Pension from an overseas Government

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

Maintenance

From ADULT record, if MNTREC = 1 and MNTPD not equal 12 or 13, add MNTAMT into OTHINC.

Council Tax Benefit

If CTREB = 1, add CTREBAMT into OTHINC only for PERSON = 1.

Odd jobs

From ADULT Rrecord if ODDJOB =1 and if OJPD not equal 12 and 13, add OJAMT into OTHINC.

Then if ODDJOB = 1 from ODDJOB record for X = 1 to 3 if OJOTH!X = 1 and exists(OJAMT) = 1 and if OJPD does not equal 12 and 13, add all occurrences of OJAMT into OTHINC.

Income from property

If PROPRENT exists add to OTHINC.

Income from sub-tenants
If SUBLET = 1, add amount held in SUBRENT into OTHINC for PERSON = 1 (head of household).

Interest/income from savings accounts or investments

From ACCOUNTS record, if ACCOUNT = 1 to 15, add amount in ACCINT to OTHINC (see attached list for definition of ACCOUNT = 1 to 15).

Occupational pensions

From PENSIONS record, if PENTYPE = 1 (occupational pension) anf PENPD not equal 12 and 13, get amount from PENPAY. If PTINC = 1 deduct PTAMT then add result into OTHINC

Trade union Friendly society pensions

From PENSIONS record, if PENTYPE = 2 and PENPD not equal 12 or 13, get amount from PENPAY and add to OTHINC. If PTINC = 1 deduct PTAMT then add result into OTHINC

Annuity/personal pension

From PENSIONS record, if PENTYPE = 3 and PENPD not equal 12 or 13, get amount from PENPAY. If PTINC = 1 deduct PTAMT then add result to OTHINC.

Trust/covenant

From PENSIONS record, if PENTYPE = 4 and PENPD not equal 12 or 13, get amount from PENPAY. If PTINC = 1 deduct PTAMT then add result to OTHINC.

Housing Benefit

From RENTER record, if BENUNIT = 1 and HBENPD not equal 12 or 13 and HBENEFIT = 1 or REBATE = 1 add in HBENAMT to OTHINC for PERSON = 1.

From HOUSEHOL record, if HHSTAT = 2 then from ADULT record if CVHB = 1 and CVPD <> 12 or 13 and exists(CHBAMT) = 1 then add CHBAMT to OTHINC.

Income from benefits

From BENEFITS record, if BENPD not equal 12 and 13 and

If BENEFIT = 1 add BENAMT to OTHINC (DLA Care)

If BENEFIT = 2 add BENAMT to OTHINC (DLA Mob)

If BENEFIT = 3 add BENAMT to OTHINC (CHB)

If BENEFIT = 4 add BENAMT to OTHINC (OPB)

If BENEFIT = 5 add BENAMT to OTHINC (RP)

If BENEFIT = 6 add BENAMT to OTHINC (OAP)

If BENEFIT = 7 add BENAMT to SSBENHH (Widows Pension)

If BENEFIT = 8 add BENAMT to SSBENHH (War Disablement Pension)

If BENEFIT = 9 add BENAMT to SSBENHH (SDA)

If BENEFIT = 10 add BENAMT to SSBENHH (DWA)

If BENEFIT = 11 add BENAMT to OTHINC (AA)

If BENEFIT = 12 add BENAMT to OTHINC (Invalid Care Allowance)

If BENEFIT = 13 add BENAMT to OTHINC (UB)

If BENEFIT = 14 add BENAMT to OTHINC(Industrial Injuries)

If BENEFIT = 16 add BENAMT to OTHINC (Sickness Benefit)

If BENEFIT = 17 add BENAMT to OTHINC (IVB)

If BENEFIT = 18 add BENAMT to OTHINC (FC)

If BENEFIT = 19 add BENAMT to OTHINC (IS)

If BENEFIT = 21 add BENAMT to OTHINC (Maternity Benefit)

If BENEFIT = 26 and PRES = 1 add BENAMT to OTHINC (Any other DSS benefits)

If BENEFIT = 27 and PRES = 1 add BENAMT to OTHINC (Trade Union sick)

If BENEFIT = 28 and PRES = 1 add BENAMT to OTHINC (Friendly sick)

If BENEFIT = 29 and PRES = 1 add BENAMT to OTHINC (Private sick)

If BENEFIT = 30 and PRES = 1 add BENAMT to OTHINC (Accident)

If BENEFIT = 31 and PRES = 1 add BENAMT to OTHINC (Hospital savings)

If BENEFIT = 32 and VAR1=1 or VAR1=2 or VAR1 = 3, add BENAMT to OTHINC (Training)

If BENEFIT = 33, add BENAMT to OTHINC (Guardians Allowance)

NB - Benefit = 15 and 23 were removed as they do not have a benamt in the benefits tables (March 1995)

6. Social Fund loan repayments

From BENEFITS record, if BENEFIT = 34 calculate SOCFUND = BENAMT.

GROSSINC will then be calculated as follows for each ADULT -

EARNS + TAX - ADJUST + SEINC + OTHINC - SOCFUND.

-2If any of above variables are missing or if a period code is 12 or 13

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
Issue date: 21 April 2005
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 JOBTYPE=1

1 Definition

This variable is coded as

GROSSPAYThe 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.

-1Not applicable where an adult does not have any jobs.

-2Unable 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 GRSWAGE which holds the person's gross earnings before tax, NI etc but only where the payslip has been consulted (where PAYSLIP = 1).

If the payslip has not been consulted (where PAYSLIP = 2) 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 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 OTHDED6 = 1 there will be a deduction for pension/superannuation, union fees, friendly societies, sports social clubs 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 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 OTHDED6, DEDUC1 to DEDUC5 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 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 (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 PAYPD = 12 or 13 GROSSPAY is set to -2.

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 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 PAYPD ~~not equal 12 or 13~~ **equals -1 or 1-11** and

If PAYSLIP = 1, **and GRSWAGE exists** calculate GROSSPAY = amount in GRSWAGE.

Or if PAYSLIP = 2 **or (PAYSLIP=1 and GRSWAGE=-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 ~~CHRTAXF = 1~~ **AMTTAXF exists** add AMTTAXF to GROSSPAY

If ~~CHROTH = 1~~ **AMTOTH exists** add AMTOTH to GROSSPAY

If ~~OTHDED1~~ **DEDUC1 exists** add DEDUC1 to GROSSPAY

If ~~OTHDED2~~ **DEDUC2 exists** add DEDUC2 to GROSSPAY

If ~~OTHDED3~~ **DEDUC3 exists** add DEDUC3 to GROSSPAY

If ~~OTHDED4~~ **DEDUC4 exists** add DEDUC4 to GROSSPAY

If ~~OTHDED5~~ **DEDUC5 exists** add DEDUC5 to GROSSPAY

If ~~OTHDED6 = 1~~ **DEDOTH exists add** DEDOTH to GROSSPAY

Then if **JOBTYPE=1 and** ~~TAXINC = 1,~~ **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)

-2If any of the above variables are missing or PAYPD = 12 or 13

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

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 Qustionnaire.
: JS - 19 January 1996: to include additional codes for IS receipt
: JS - 25 January: to make clear that codes for IS refr to benefit units where any adult is in receipt of IS
Issued : 21 April 2005

Assumptions

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

1 Definition

This variable is coded as

HBINDBU Indicates that someone in the benefit unit is receiving Income Support, Housing Benefit or Council Tax Benefit as follows,

0 No HB/CTB/IS

1 If receives CTB only (no IS or HB)

2 If receives HB only (no IS or CTB)

3 If receives IS only (no HB or CTB)

4 If receives both HB and CTB (no IS)

5 If receives both HB and IS (no CTB)

6 If receives both CTB and IS (no HB)

7 If receives HB, CTB and IS

-1 Not applicable to this case: these should not occur since questions are asked of all households

-2 Unable to derive because of missing values

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

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

Receipt of Housing Benefit is identified from 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 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)
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)
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)

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)
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)
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)
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)
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)
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.

For 1994/95 there were 29 cases where HBINDBU was missing, 10 were the same as for HBINDHH, in other cases, HBINDHH had been calculated. To ensure these cases were not set to -2, values for HBINDBU were hard coded into the program:

sernum	**benunit**	**hhbindbu**	**missing hhbindhh**
574141	**1**	**0**	**yes**
904021	**1**	**0**	**yes**
1234221	**1**	**0**	**yes**
2084021	**1**	**5**	**no**
2714041	**1**	**5**	**no**
3444051	**1**	**5**	**no**
3444051	**2**	**3**	**no**
3444051	**3**	**0**	**no**
3894051	**1**	**0**	**yes**
4944071	**1**	**2**	**no**
5544061	**1**	**5**	**no**
5824221	**1**	**2**	**no**
6214131	**1**	**4**	**no**
7464051	**1**	**5**	**no**
8914121	**1**	**5**	**no**
9384071	**1**	**5**	**no**
9434162	**1**	**5**	**no**
9764171	**1**	**0**	**yes**
10354101	**1**	**0**	**yes**
10354102	**1**	**0**	**yes**
10874081	**1**	**5**	**no**
11344081	**1**	**0**	**yes**
11344201	**1**	**0**	**yes**
13484141	**1**	**5**	**no**
14104192	**1**	**0**	**yes**
14374111	**1**	**5**	**no**
15804091	**1**	**5**	**no**
16064081	**1**	**5**	**no**
16114221	**1**	**5**	**no**

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 Qustionnaire.
: 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
Issued: 21 April 2005

1 Definition

This variable is coded as

HBINDHHIndicates that someone in the household is receiving Income Support, Housing Benefit or Council Tax Benefit as follows,

0No HB/CTB/IS

1If receives CTB only (no IS or HB)

2If receives HB only (no IS or CTB)

3 If receives IS only (no HB or CTB)

4If 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

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

-2Unable 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 separate questions depending on the type of household:

i HBENEFIT = 1 (has received HB in connection with last rent payment) (household record). iiCVHB=1 (adult record: lodgers in BUs>1 in receipt of HB) where HHSTAT=1 (conventional household)
iiiSCVHB=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 household answers "yes" to BEN3Q02 (ADULT record - are you at present receiving IS)

2 FRS Specification

CodeCondition

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
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
and for each adult in household, BEN3Q02<>1

2If 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
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
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) SCVHB=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) SCVHB=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
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) SCVHB=1
and for at least one adult in household, BEN3Q02=1

-1Not applicable to this case

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

For 1994/95 there were 10 cases where HBINDHH was set to unable to derive due to missing values; all such cases were set to HBINDHH=0.

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: 21 April 2005
Amendments : JS - 21 February 1996 to allow for skipped values when variables have been imputed

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

1 Definition

This variable is coded as

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

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

-2Unable 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 repsonse 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 (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 HBSUPRAN is set to -2.

2 FRS Specification

For each ADULT with record Job for all jobs

CodeCondition

HBSUPRANIf PAYPD ~~not equal 12 or 13~~ **equals -1 or 1-11** and -

If OTHDED1 = 1 **and DEDUC1 exists**, get the amount of the superannuation or pension contribution deducted from variable DEDUC1 **else leave as skipped**.

-1Not applicable in this case - where OTHDED1 = 2 or 3 or OTHDED1 does not exist or there are no job records

-2If variable cannot be defined because of missing data - where there is a job record but no values or if PAYPD = 12 OR 13.

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 : 21 April 2005
Amendments : JS - 14 March to collapse some of the households with children categories

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 revised coding of HHCOMP removes age combinations for those household with children and combines categories for three and four or more children in to three or more only.

The coding for HHCOMP is:		**The revised coding is:**	
1	One adult, no children over pension age	**1**	**as before**
2	One adult, no children, under pension age	**2**	**as before**
3	Two adults, no children, both over pension age	**3**	**as before**
4	Two adults, no children, one over pension age	**4**	**as before**
5	Two adults, no children, both under pension age	**5**	**as before**

6	Three or more adults, no children	**6**	**as before**
7	One adult, with children, over pension age	7	(depending on number of children:no age breakdown) One adult, one child
8	One adult, one child, under pension age	**7**	**One adult, one child**
9	One adult, two children, under pension age	**8**	**One adult, two children**
10	One adult, three children, under pension age	**9**	**One adult three or more children**
11	One adult, four or more children, under pension age	9	
12	Two adults, with children, both over pension age	10	(depending on number of children: no age breakdown) Two adults, one child
13	Two adults, with children, one over pension age	**10**	**Two adults, one child**
14	Two adults, one child, both under pension age	10	
15	Two adults, two children, both under pension age	**11**	**Two adults, two children**
16	Two adults, three children, both under pension age	**12**	**Two adults, three or more children**
17	Two adults, four + children, both under pension age	12	
18	Three or more adults, one child	**13**	**as before**
19	Three or more adults, two children	**14**	**as before**
20	Three or more adults, three children	**15**	**three or more adults, three or more children**
21	Three or more adults, four or more children	15	

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

Households with children, codes 7-21 ***all*** *where* DEPCHLDH>0

7 ADULTH=1 and ~~(SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)~~ **and DEPCHLDH=1**

8 ADULTH=1 and ~~((SEX=1 and AGE<65) or (SEX=2 and AGE<60))~~ and DEPCHLDH=~~1~~ **2**

9 ADULTH=1 and ~~((SEX=1 and AGE<65) or (SEX=2 and AGE<60))~~ and DEPCHLDH=~~2~~ **>=3**

~~10 ADULTH=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)) and DEPCHLDH=3~~

~~11 ADULTH=1 and ((SEX=1 and AGE<65) or (SEX=2 and AGE<60)) and DEPCHLDH>=4~~

~~12~~ **10** 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)))~~
and DEPCHLDH=1

~~13~~ **11** 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)))~~

and DEPCHLDH=2

~~14~~ **12** 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)))~~
and DEPCHLDH=~~1~~ **>=3**

~~15 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)))~~
~~and DEPCHLDH=2~~

~~16 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)))~~
~~and DEPCHLDH=3~~

~~17 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)))~~
~~and DEPCHLDH>=4~~

~~18~~ **13** ADULTH>=3 and DEPCHLDH=1

~~19~~ **14** ADULTH>=3 and DEPCHLDH=2

~~20~~ **15** ADULTH>=3 and DEPCHLDH~~=3~~ **>=3**

~~21 ADULTH>=3 and DEPCHLDH>=4~~

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)

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 : Publication
Amendments :

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

Code	Condition
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:

HEARNS 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 : 21 April 2005
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

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), commamt (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, 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)

(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 (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 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	From HOUSEHOL table, if TENURE = 2 or 3 or 4 or 5 (renting) get HHRENT and HHSTAT variables (Drop tenure=6 rent free have non-zero)
	From RENTER table, if RENTPD equals -1 or 1-11 get RENT variable and calculate HHRENT = RENT
	From HOUSEHOL record if HHSTAT = 2 then from ADULT record for all BUs > 1 if SRENTPD equals -1 or 1-11 and SRENTAMT exists add SRENTAMT to HHRENT.

_

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.

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)

From RENTCONT record if ACCNONHH = 1 and ACCCHK = 2 and ACCPD equals -1 or 1-11, add ACCAMT to rent to get HHRENT Someone outside of HH pays part of rent and original rent given after this deducted). If ACCNONHH = 1 and ACCCHK = 1 or ACCNONHH = 2, do not change amount in HHRENT.

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.

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 (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 = 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 WATERINC=2 and SEWERINC=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.

Note: in 1994/95 there were four non-conventional households (HHSTAT=2) recorded as rebate cases where srentamt and schvamt were positive (sernums 2494141, 9804251, 9964181,15554221).

Adding these in to rent was considered but this would have given rise to improbable levels of rent. Instead it may have been that respondents were confused: giving the total rent which was shared between them.

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

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

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:
Issued: 21 April 2005

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.

CodeCondition

1If BENUNIT = 1 and PERSON = 1
+1In the following priority:
+1If BENUNIT = 1, and PERSON = 2,
+1If 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: ~~HBAI~~ **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
Issued: 21 April 2005

1 Definition

This variable is coded as

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

0Not applicable as have no housing costs

-2Unable 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.

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.

The amount of community water charge should also be added into HSCOSTHH and is found in ~~CWATAMT~~ **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 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.

However, if the period code for sharepayers 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 SHAREPD = 12 or 13 HSCOSTBU 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 HSCOSTBU are derived using previously derived variables the period codes need not be looked at for these variables as they are alredy excluded.

2 FRS Specification

For each HOUSEHOLD, set HSCOSTHH to zero.

CodeCondition

HSCOSTHHFrom HOUSEHOL record,

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

If WATSEWRT exists and does not equal -1 or -2 (as above) 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 OWNER record if STRPD and CHARGP not equal 12 or 13,~~

~~If STRINTH exists add the amount in STRINTH into HSCOSTHH.~~

From househol record get STRCOV, STRAMT, STRPD, STROTHS, TENURE

set STRINTH equal to zero

If STRCOV=1 or STRCOV=3 **and STRAMT exists** *(structural insurance only or combined with furniture and contents as part of mortgage payment)* and

STRPD=-1 or 1-11 STRINTH=STRAMT

If STROTHS=1 and TENURE=1 **and STRAMT exists** *(insurance premium paid on structure of accommodation separately from any mortgage payments, owner occupiers only)* and STRPD=-1 or 1-11 STRINTH=STRAMT

Add STRINTH to HSCOSTHH

From owner record get CHARGE1-7, CHAMT1-7, CHARGEP1-7

If CHARGE1=1 and **CHARGP1=-1, or 1-11** add CHAMT1 to HSCOSTHH
If CHARGE2=1 and **CHARGP2=-1, or 1-11** add CHAMT2 to HSCOSTHH
If CHARGE3=1 and **CHARGP3=-1, or 1-11** add CHAMT3 to HSCOSTHH
If CHARGE4=1 and **CHARGP4=-1, or 1-11** add CHAMT4 to HSCOSTHH
If CHARGE5=1 and **CHARGP5=-1, or 1-11** add CHAMT5 to HSCOSTHH
If CHARGE6=1 and **CHARGP6=-1, or 1-11** add CHAMT6 to HSCOSTHH
If CHARGE7=1 and **CHARGP7=-1, or 1-11** add CHAMT7 to HSCOSTHH

From RENTER record,

If **CWATAMTD exists** ~~CWATAMT exists and CWATPD not equal 12 or 13~~ ~~**equals -1 or 1-11**~~, add the amount in ~~CWATAMT~~ **CWATAMTD** into HSCOSTHH. **(this change was made for 1993/94 but was not made in the spec)**

0Not applicable as household has no housing costs.

-2Unable 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

Purpose : To calculate the total amount of income from self-employment using the : traditional CSO/FES approach
Created : VC - 17 August 1993
Database Table : ADULT
Minimum Value : 0
Maximum Value :
Units : Real
Validations :
Related Variables : INCSE2 LOSS
Children :
Parents :
Core variable/user : publication
Amendments : VC - 14 September 1993 To amend period used to calculate uprating factor : as may not be full 12 months.
: VC - 1 March 1994 To exclude period codes 12 and 13
: JS - 21 February 1996 to allow for skipped values where variables have been imputed and to take on changes made by ASD6C
: VE - 25 November 1996 - to allow for cases where PARINC has been imputed and PARAMT remains skipped.
Issued : 21 April 2005

1 Definition

This variable is coded as

INCSE1This is the total amount of income received from self-employed earnings which has been calculated using the traditional CSO/FES approach to the definition of earnings. The CSO/FES definition includes drawings from a business where a person has declared that they have made a loss or nil profit from their business. The amount will be uprated to a common time period using uprating factors provided by Mr Nicholls.

-1 Not applicable to this case

-2 Unable to derive due to missing values

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).

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).

Where a profit has been made the full amount of profit may only be used if the person does not have a business partner or if he/she does have a business partner the amount in PROFIT1 has been declared after the partner's share has been deducted. Therefore, if a person has a business partner (SOLE = 2) and if the partner's share has not been deducted (PARINC = 1) the amount held in PARAMT must be deducted.

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 provided by Mr Nicolls 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 12. 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.

If PROFIT2 = 2 indicating a loss, the amount used for INCSE1 will be the amount of drawings which is established from OWNSUM = 1 (has drawn sums from business for own use) and the amount withdrawn is to be found in OWNAMT. This should also be used if the amount declared in PROFIT1 is zero.

However, if the period code for ownamt 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 OWNFRQ = 12 or 13 INCSE1 is set to -2.

Where variables have been imputed, questions following will have been skipped. This means allowances have to be made in the spec as follows:

i For OWNAMT, OWNPD=-1 is allowed (imputed amounts will be weekly)

ii For PROFIT1, have to assume that the amount is weekly because SE1 and SE2 will have been skipped (this is acceptable), that it is a profit, because PROFIT2 will have been skipped (this is ropey: although PROFIT2 was taken in to account during imputation using hotdecking, there may be some losses figures included which cannot be identified) and that they are working on their own. These cases will not be uprated.

2 FRS Specification

For each ADULT, convert SE1, SE2 and INTDATE into month year only date variables.

Code Condition

INCSE1Where WORKING = 1 or JOBAWAY = 1, set INCSE1 to 0

For each JOB record where EMPEE = 2 (self-employed)

If PROFIT2 = 1 and SOLE = 1 **(**(no partner) or **SOLE=2 and PARINC=2)** calculate uprating factor.

If PROFIT~~1~~ **2**= 1 and SOLE = 2 (has partner) and PARINC = 1 (partner's share included in profit1) **and PARAMT exists** subtract PARAMT from PROFIT1 then calculate uprating factor using new amount for PROFIT1. **Else if PARAMT=-1 calculate uprating factor from PROFIT1.**

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 ~~12 months~~ **(PERIOD-1) months** prior to date of interview (INTDATE) divided by ~~12~~ **PERIOD**.

Calculate uprated profit UPROFIT = PROFIT1 * (SEINDEX1/SEINDEX2)

Calculate INCSE1 = UPROFIT + INCSE1

If (PROFIT2 = 2 or PROFIT1 = 0) and OWNSUM = 1 and **OWNPD=-1 or 1-11** ~~OWNFRQ not equal 12 or 13.~~

Compute INCSE1 = OWNAMT + INCSE1

Else if PROFIT1 exists INCSE1=PROFIT1

-1 Not applicable - people who are not working or are employees

-2 Underivable - where any of above values are missing or OWNPD = 12 or 13.

3 Results

Tabulation is required to show the number of adults falling into the following bands of weekly earnings from self-employment

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 - £350
£350 or over

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 : Publication
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 INEARNS 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
Issued : 21 April 2005

1 Definition

This variable is coded as

INDINC The total amount of gross income received by an adult from all sources.

0	Not applicable as adult does not have any gross income.
-2	Unable to derive due to missing values.

The total amount of gross income is derived from numerous variables from the ADULT, JOB, BENEFITS, ODDJOB and PENSIONS records which when added together form the person's total gross income. It includes gross normal earnings (cf with GROSSPAY which is last pay), self-employed earnings, tax paid on pensions annuities, other income in the form of benefit income, income in kind, royalties, other allowances, income from trust funds and odd jobs etc. Unlike V30 GROSSINC the adjustment for SSP/SMP is only used to avoid double counting. 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 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, for example, if PAYPD = 12 or 13 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.

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
INSEINC	self employment income (identical to INCSE1 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

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

INDINC **Gross earnings: INEARNS**

From ADULT record, set INEARNS to zero

If WORKING = 1 or JOBAWAY = 1 - process each JOB record for that person and

If EMPEE = 1

(PAYUSL, and therefore UGROSS if PAYUSL equals "no" is only asked for JOBTYPE=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 JOBTYPE=1 - UGROSS is used)

If UGROSS does not exist

and If PAYSLIP = 1

If GRWAGE exists and PAYPD equals -1 or 1 to 11, add it into **INEARNS**

If it is missing set **INDINC and INEARNS** -2

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

If PAYAMT exists, add it into **INEARNS.**

If it is missing set **INDINC and INEARNS to** -2

If PAYE exists, add it into **INEARNS.**

If it is missing do not change INEARNS

If NATINS exists, add it into **INEARNS**.

If it is missing do not change INEARNS

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

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 **INEARNS**

if AMTOTH exists add it into **INEARNS**

(as for AMTTAXF)

If OTHDED1 = 1 add DEDUC1 to **INEARNS**

If OTHDED2 = 1 add DEDUC2 to **INEARNS**

If OTHDED3 = 1 add DEDUC3 to **INEARNS**

If OTHDED4 = 1 add DEDUC4 to **INEARNS**

If OTHDED5 = 1 add DEDUC5 to **INEARNS**

If OTHDED6 = 1 add DEDOTH to **INEARNS**

Else if UGROSS exists

If UGROSS exists add UGROSS to **INEARNS**

If it is missing do not change INEARNS (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 INEARNS<>-2 *(other conditions relating to PAYAMT and PAYPD will have been met by this point if INEARNS has not been set to -2)*

and UGROSS does not exist (ie all jobtypes except jobtype=1 where pay not usual)

```
            and JOBTYPE=1 and TAXAMT exists INEARNS=INEARNS-TAXAMT (TAXAMT only asked for first job)

            and MILEAMT exists INEARNS=INEARNS-MILEAMT

            and MOTAMT exists INEARNS=INEARNS-MOTAMT

            and HHA1 exists INEARNS=INEARNS-HHA1

            and HHA2 exists INEARNS=INEARNS-HHA2

            and HHA3 exists INEARNS=INEARNS-HHA3


        else if  UGROSS exists

            and U2MOT exists INEARNS=INEARNS-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
    INEARNS=INEARNS+((BONAMT(i)/52))


    If BONAMT(i) exists and BONTAX(i)=2 or -1 (after tax or skipped where BONAMT imputed)
    INEARNS=INEARNS+((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 UBONAMT exists
    INEARNS=INEARNS-(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)

Adjustment for possible receipt of SSP or SMP

Set ADJUST (temporary variable) **and INOTHBEN** to zero

For JOBTYPE=1 only *(SSP/SMP questions only asked once)*

If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYSLIP = 1
Calculate ADJUST = SSPAMT

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

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

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

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

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

If ADJUST >= to employment income calculated in INEARNS, reset ADJUST to **INEARNS**

INEARNS=INEARNS-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), 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 INEARNS 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 INEARNS (ie INEARNS=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, add APAMT to **INRINC**.

If **APDIR**=1 and APDPD equals -1 or 1 to 11 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 , add ALLPAY1 to **INRINC**.

Allowance from an organisation

From ADULT record, if ALLOW2 = 1 and ALLPD2 equals -1 or 1 to 11 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 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, add ALLPAY4 to **INRINC**.

Income in kind

From JOB record, if JOBTYPE=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 and FCAMTPD equals -1 or 1 to 11 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 MNTPD equals -1 or 1 to 11 and MNTUS=1, add MNTAMT into **INRINC**.

Else if MNTREC=1 and MNTUS=2 and MNTUSPD equals -1 or 1 to 1 add MNTUSAMT 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=12 or 13 add OJAMT/4 into INRINC

Income from property

If PROPRENT exists add PROPRENT 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). The income is assumed to be for the head of household
From RENTCONT record if ACCNONHH =1 and ACCAMT exists and ACCPD=-1 or 1-11 then INRINC=INRINC+ACCAMT for PERSON=1 only (head of household)

For those buying their house with a mortgage, contributions from outside are included for all cases except the DSS. This is to avoid double counting of direct payments which are already included as part of benefit income. Amounts are included for all types of mortgage (having been explicitly added back for endowment mortgages where appropriate and implicit in repayment mortgages calculation)
From MORTCONT record if OUTSMORT=1 and OUTSPAY=2,3,4 or 5 (ie not equal to 1) OUTSAMT exists and OUTSPD=-1 or 1-11 then INRINC=INRINC+OUTSAMT for 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*

Interest/income from savings accounts or investments: ININV

set ININV to zero

From ACCOUNTS record,

if ACCOUNT = 1, add amount in ACCINT to **ININV**
if ACCOUNT = 2, add amount in ACCINT to **ININV**
if ACCOUNT = 3, add amount in ACCINT to **ININV**
if ACCOUNT = 4, add amount in ACCINT to **ININV**
if ACCOUNT = 5, add (4/3 x amount in ACCINT) to **ININV**
if ACCOUNT = 6, add amount in ACCINT to **ININV**
if ACCOUNT = 7, add (4/3 x amount in ACCINT) to **ININV**
if ACCOUNT = 8, add amount in ACCINT to **ININV**
if ACCOUNT = 9, add (4/3 x amount in ACCINT) to **ININV**
if ACCOUNT = 10, add 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.**

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,

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,

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, 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, 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, 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, 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

From RENTER record, if BENUNIT = 1 and HBENPD equals -1 or 1 to 11, and HBENEFIT = 1 add in HBENAMT to **INOTHBEN** 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 **INOTHBEN**.

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

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to 11 add CTREBAMT into **INOTHBEN** only for PERSON = 1.

Income from benefits

From BENEFITS record, if BENPD equals -1 or 1 to 11, 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 = 7 add BENAMT to **INOTHBEN**(Widows Pension)

If BENEFIT = 8 add BENAMT to **INDISBEN** (War Disablement Pension)

If BENEFIT = 9 add BENAMT to **INDISBEN** (SDA)

If BENEFIT = 10 add BENAMT to **INDISBEN** (DWA)

If BENEFIT = 11 add BENAMT to **INDISBEN** (AA)

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

If BENEFIT = 13 add BENAMT to **INOTHBEN**(UB)

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

If BENEFIT = 16 add BENAMT to **INOTHBEN**(Sickness Benefit)

If BENEFIT = 17 add BENAMT to **INDISBEN** (IVB)

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 Benefit)

If BENEFIT = 26 and PRES = 1 add BENAMT to **INOTHBEN** (Any other DSS benefits)

If BENEFIT = 27 and PRES = 1 and BENPD=-1 or 1-11 add BENAMT to **INRINC** (Trade Union sick) else if PRES=1 and BENPD=12 or 13 do not change INRINC

If BENEFIT = 28 and PRES = 1 and BENPD=-1 or 1-11 add BENAMT to **INRINC** (Friendly sick) else if PRES=1 and BENPD=12 or 13 do not change INRINC

If BENEFIT = 29 and PRES = 1 add BENAMT to **INRINC** (Private sick) else if PRES=1 and BENPD=12 or 13 do not change INRINC

If BENEFIT = 30 and PRES = 1 add BENAMT to **INRINC** (Accident) else if PRES=1 and BENPD=12 or 13 do not change INRINC

If BENEFIT = 31 and PRES = 1 add BENAMT to **INRINC** (Hospital savings) else if PRES=1 and BENPD=12 or 13 do not change INRINC

If BENEFIT = 32 add BENAMT to (Training) **INRINC**

If BENEFIT = 33, add BENAMT to **INOTHBEN** (Guardians Allowance)

If BENEFIT=34 and VAR2=2
    and ((SEX=1 and AGE>=65) or (SEX=2 and AGE>=60)) add BENAMT to INRPINC
else if BENEFIT=34 and VAR2=2 add BENAMT to INOTHBEN
(adjustment for IS quoted after social fund repayments: pensioners income must be included in INRPINC whilst others are in INOTHBEN)

INDINC will then be calculated as follows for each ADULT -

INEARNS+INSEINC+ININV+INRPINC+INPENINC+INDISBEN+INOTHBEN+INRINC

-2 If any of above variables are missing or if a period code is 12 or 13 (skipped is OK) also applied to individual component variables ININV, INRPINC, INPENINC, INDISBEN, INOTHBEN and INRINC which are calculated in this spec (missing INEARNS components are already documented above).

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	:

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

Non-income related benefits are: SSP, SMP, DLA, CHB, OPB, RP, Widows' pension, War disablement pension, SDA, AA, ICA, UB, Industrial Injuries, Sickness benefit, IVB, Maternity benefit

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

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

Set INIRBEN and INNIRBEN to zero and get INEARNS

SSP or SMP

set ADJUST (temporary variable) to zero

If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYSLIP = 1
Calculate ADJUST = SSPAMT

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

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

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

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

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

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

Add ADJUST to **INNIRBEN**

Housing Benefit

From RENTER record, if BENUNIT = 1 and HBENPD equals -1 or 1 to 11, 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 then from ADULT record if SCVHB=1 and SCHBPD equals -1 or 1 to 11 then add SCHBAMT to **INIRBEN.**

Council Tax Benefit

If CTREB = 1, and CTREBPD equals -1 or 1 to 11 add CTREBAMT to **INIRBEN** for PERSON = 1 only.

Other benefits

From BENEFITS record, if BENPD equals -1 or 1 to 11 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 add BENAMT to **INNIRBEN** (Widows Pension)

If BENEFIT = 8 add BENAMT to **INNIRBEN** (War Disablement Pension)

If BENEFIT = 9 add BENAMT to **INNIRBEN** (SDA)

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

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

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

If BENEFIT = 13 add BENAMT to **INNIRBEN** (UB)

If BENEFIT = 14 add BENAMT to **INNIRBEN** (Industrial Injuries)

If BENEFIT = 16 add BENAMT to **INNIRBEN** (Sickness Benefit)

If BENEFIT = 17 add BENAMT to **INNIRBEN** (IVB)

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)

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

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

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

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 occurences 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 occurences of BUIRBEN

        HHNIRBEN equals total occurences of BUNIRBEN

-2      if any components are missing
```

ISMNDD01 - ISMNDD10 (IN201-IN210)

Purpose: To provide FES equivalent non-dependant deduction variables for ISM model
Created: May 1995 as part of parallel run exercise
Database Table: BENUNIT
Minimum Value: 0
Maximum Value:
Units:
Validations:
Related Variables:
Children:
Parents:
Core variable/user: ISM
Amendments: VE - 16 April 1996 - To give fuller definitions
Issued: 21 April 2005

1 Definition

These variables hold codes for how non-householder benefit units would be classified if it were a non-dependant. Up to 10 non-HoH benefit units can be assessed (ISMND01-ISMNDD10) in any given household.

Variables are consistent with FES variables IN201 to IN210

This variable is coded as

1 Indicating a member of the householder BU is in receipt of AA or DLA (care component).

2 Indicating an adult aged 18+ in full-time work in a non-HoH BU with a combined income of 130 or over.

3 Indicating an adult aged 18+ in full-time work in a non-HoH BU with a combined income of 100 or over but less than 130.

4 Indicating an adult aged 18+ in full-time work in a non-HoH BU with a combined income of 65 or over but less than 100.

5 Indicating an adult aged 18+ in full-time work in a non-HoH BU with a combined income of less than 65.

6 Indicating an adult aged 18+ in a non-HoH BU who is not a student, nor on a YTS scheme, nor under the age of 25 and in receipt of IS.

7 Indicating an adult aged 18+ in a non-HoH BU who is a full-time student or on a YTS scheme.

8 Indicating an adult aged 16 or 17 in a non-HoH BU.

From the ADULT and JOB tables, a person is classified as being in full-time work if EMPSTATB=1 and QHRSSELF>=16 (self-employed), or if EMPSTATB=2 and TOTHOURS>=16 (full-time employee), or if ABSPAY=1 (absent from work but being paid full pay).

Combined income is taken to be the sum of GROSSINC (from the ADULT table) for all members of the benefit unit

2 FRS Specification

Using the definitions of full-time work (FTW=1 implies an individual in full-time work) and combined income (CI) as above, for each non-HoH BU

Code Condition

1 From table BENEFIT for HoH BU

If BENEFIT=1 or BENEFIT=11

2 From ADULT and JOB tables for non-HoH BU

If AGE>=18 and FTW=1 and CI>=130

3 From ADULT and JOB tables for non-HoH BUs

If AGE>=18 and FTW=1 and CI>=100 and <130

4 From ADULT and JOB tables for non-HoH BUs

If AGE>=18 and FTW=1 and CI>=65 and <100

5 From ADULT and JOB tables for non-HoH BUs

If AGE>=18 and FTW=1 and CI<65

6 From ADULT and BENEFIT tables for non-HoH BUs

If AGE>=18 and FTED=2 and (TRAIN not equal 1) and not (AGE<25 and BENEFIT=19)

7 From ADULT and BENEFIT tables for non-HoH BUs

If AGE>=18 and AGE<=24 and BENEFIT=19

8 From ADULT table for non-HoH BUs

If AGE>=18 and FTED=1 or TRAIN=1

9 From ADULT table for non-HoH BUs

If AGE=16 or AGE=17

ISRECDBU

Purpose: This variable is the total weekly amount currently received from Income : Support by each person in the benefit unit.
Created: VC - 6 September 1993
Database Table: BENUNIT
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: FCRECDBU, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 17 February 1994 Amended to reflect version 30 changes
: VC - 1 March 1994 to exclude period codes 12 or 13
: BS - 3 August 1995. To exclude PRES as it is no longer needed in V31
: JS - 21 February 1996 to allow for skipped BENPD when BENAMT has been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

ISRECDBUThis is the total weekly amount currently received from Income Support by each person in the benefit unit.

0Not applicable to this case

-2Unable to derive as missing values

This variable is derived by adding together the amount of Income Support received by any person in the benefit unit. The amount of Income Support is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 19 the person is receiving Income Support and the amount held in BENAMT for that record should be added into ISRECDBU.

The total should include Income Support received only, therefore, if the benefit unit receives any other form of benefit in addition to IS, this would not be added into the total. Benefit units with no IS are not applicable to this case. The variable PRES indicates whether IS is currently in receipt rather than received within the last 12 months. In V31 PRES is no longer asked as the question now asks if the respondent is currently receiving IS.

However, if the period code for the benefit 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 BENPD = 12 or 13 ISRECBU is set to -2.

2 FRS Specification

For each BENEFIT UNIT

CodeCondition

ISRECDBUProcess all benefit unit records, set ISRECDBU to zero

Process all BENEFITS records for that benefit unit (via benunit)

If BENEFIT = 19 and BENPD ~~not equal 12 or 13~~ **equals -1 or 1-11**, add BENAMT into ISRECDBU

ISRECDBU will then be the total amount of benefit received from IS by the benefit unit.

0Not applicable as does not have any IS recipients

-2Unable to derive as above variables are missing or BENPD = 12 or 13.

3 Results

Tabulation is required to show the number of benefit units by the total amount of Income Support received split into the following weekly bands

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 or over.

ISRECDHH

Purpose: This variable is the total weekly amount currently received from Income : Support by each person in the household.
Created: VC - 6 September 1993
Database Table: HOUSEHOL
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: FCRECDBU, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 1 March 1994 To exclude period codes 12 or 13
: BS - 3 August 1995. To exclude PRES from the DV as the question in V31 now asks if the respondent is currently receiving IS.
: JS - 21 February 1996 to allow skipped BENPD when BENAMT has been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

ISRECDHHThis is the total weekly amount curently received from Income Support by each person in the household.

0Not applicable to this case

-2Unable to derive as missing values

This variable is derived by adding together the amount of Income Support received by any person in the household. The amount of Income Support is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 19 the person is receiving Income Support and the amount held in BENAMT for that record should be added into ISRECDHH. The variable PRES indicated whether Income Support is currently in paymet or only received during last 12 months in V30. In V31 PRES is no longer asked as the question now asks if the respondent is currently receiving IS.

The total should include Income Support received only, therefore, if the household receives any other form of benefit in addition to IS, this would not be added into the total. Households with no IS are not applicable to this case.

However, if the period code for the benefit 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 BENPD = 12 or 13 ISRECHH is set to -2.

2 FRS Specification

For each HOUSEHOLD, set ISRECDHH to zero

CodeCondition

ISRECDHHProcess all BENEFITS records for that household

If BENEFIT = 19 and BENPD ~~not equal 12 or 13~~ **equals -1 or 1-11**, add BENAMT into ISRECDHH

ISRECDHH will then be the total amount of benefit received from IS by the household.

0Not applicable as does not have any IS recipients

-2Unable to derive as above variables are missing or BENPD = 12 or 13.

3 Results

Tabulation is required to show the number of households by the total amount of Income Support received split into the following weekly bands

Under £50
£50 - £100
£100 - £150
£150 - £200
£200 - £250
£250 - £300
£300 or over.

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**
Amendments: VC - 11 May 1993 To expand and clarify the definition
: VC - 17 February 1994 Amended to reflect version 30 changes

1 Definition

This variable is coded as

KID04This is the number of children in each benefit unit aged 0 to 4 years inclusive.

-1Not applicable to this case

-2Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

CodeCondition

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.

-1Not applicable to this case - this should not occur for this variable.

-2Unable 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**

Amendments: VC - 11 May 1993 amended to show coding
: VC - 17 February 1994 Amended to reflect version 30 changes

1 Definition

This variable is coded as

KID510This is the number of children in each benefit unit aged 5 to 10 inclusive.

-1Not applicable to this case.

-2Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

<u>CodeCondition</u>

KID510For each child in benefit unit, from CHILD table.
If age >= 5 and <= 10, count total number of children where age falls in this range.

-1Not applicable to this case - should not occur for this variable.

-2Unable 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**
Amendments: VC - 11 May 1993 Amended to show coding
: VC - 17 February 1994 Amended to reflect changes to version 30

1 Definition

This variable is coded as

KID1115This is the number of children in each benefit unit aged 11 to 15 inclusive.

-1Not applicable to this case.

-2Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

<u>CodeCondition</u>

KID1115For each child in benefit unit, from CHILD table
If age >= 11 and <= 15, count number of children where age falls in this range.

-1Not applicable to this case - should never occur for this variable.

-2Unable 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**
Amendments: VC - 11 May 1993 amended to show coding
: VC - 17 February 1994 Amended to reflect version 30 changed

1 Definition

This variable is coded as

KID1618This is the number of dependents in each benefit unit aged 16 to 18 inclusive.

-1Not applicable to this case

-2Unable to derive due to missing values.

2 FRS Specification

For each BENUNIT record

CodeCondition

KID1618For 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.

-1Not applicable to this case - should not occur for this variable.

-2Unable 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**
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

1 Definition

This variable is coded as

KIDS0BUNumber of dependents under age 1
KIDS1BUNumber of dependents between age 1 and 2
KIDS2BUNumber of dependents between age 2 and 3
KIDS3BUNumber of dependents between age 3 and 4
KIDS4BUNumber of dependents between age 4 and 5
KIDS5BUNumber of dependents between age 5 and 6
KIDS6BUNumber of dependents between age 6 and 7
KIDS7BUNumber of dependents between age 7 and 8
KIDS8BUNumber of dependents between age 8 and 9
KIDS9BUNumber of dependents between age 9 and 10
KIDS10BUNumber of dependents between age 10 and 11
KIDS11BUNumber of dependents between age 11 and 12
KIDS12BUNumber of dependents between age 12 and 13
KIDS13BUNumber of dependents between age 13 and 14
KIDS14BUNumber of dependents between age 14 and 15
KIDS15BUNumber of dependents between age 15 and 16
KIDS16BUNumber of dependents between age 16 and 17
KIDS17BUNumber of dependents between age 17 and 18
KIDS18BUNumber 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

<u>CodeCondition</u>

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.

KIDS0BUThe sum of all dependents in the benefit unit where age < 1
KIDS1BUThe sum of all dependents in the benefit unit where age >= 1 and < 2
KIDS2BUThe sum of all dependents in the benefit unit where age >= 2 and < 3
KIDS3BUThe sum of all dependents in the benefit unit where age >= 3 and < 4
KIDS4BUThe sum of all dependents in the benefit unit where age >= 4 and < 5
KIDS5BUThe sum of all dependents in the benefit unit where age >= 5 and < 6
KIDS6BUThe sum of all dependents in the benefit unit where age >= 6 and < 7
KIDS7BUThe sum of all dependents in the benefit unit where age >= 7 and < 8
KIDS8BUThe sum of all dependents in the benefit unit where age >= 8 and < 9
KIDS9BUThe sum of all dependents in the benefit unit where age >= 9 and < 10
KIDS10BUThe sum of all dependents in the benefit unit where age >= 10 and < 11
KIDS11BUThe sum of all dependents in the benefit unit where age >= 11 and < 12
KIDS12BUThe sum of all dependents in the benefit unit where age >= 12 and < 13
KIDS13BUThe sum of all dependents in the benefit unit where age >= 13 and < 14
KIDS14BUThe sum of all dependents in the benefit unit where age >= 14 and < 15
KIDS15BUThe sum of all dependents in the benefit unit where age >= 15 and < 16
KIDS16BUThe sum of all dependents in the benefit unit where age >= 16 and < 17
KIDS17BUThe sum of all dependents in the benefit unit where age >= 17 and < 18
KIDS18BUThe 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

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

1 Definition

This variable is coded as

LODGERThe total weekly amount paid by a benefit unit classed as a lodger to the householder for a room but no food.

-1Not applicable to this case.

-2Unable to derive due to missing values.

LODGER is derived by benefit unit from the variable convbl which asks whther the person is a borarder 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 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 = 12 or 13 LODGER is set to -2.

2 FRS Specification

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

CodeCondition

LODGERFrom ADULT table

If CONVBL = 2 and CVPD ~~not equal 12 or 13~~ **equals -1 or 1-11**, 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.

-1Not applicable to this case - where convbl = 1 or 3 or is missing as the question has not been asked.

-2Unable to derive due to value of cvpay missing or CVPD = 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

MORTCOST

Purpose : To weekly show 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**
Amendments : To correct identification of mortgage protection policies

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 are included as part of MORTCOST.

If the property is owned outright (ownhow=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

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

MORTCOST

Mortgage payments

From househol record get MORTINT
From mortgage get INCMPAMT, INCMPPD

Set MORTPAY equal to MORTINT

If MORTINT <>-2 and MORTPROT=1 and INCMPAMT exists and INCMPPD equal to -1 or 1-11 for all mortgages
then MORTPAY=MORTPAY+INCMPAMT

Else if ownhow=1 (owned outright) MORTPAY=0

Endowment premiums

From ENDOWMNT table get MENPOLAM, MENPOLPD

Set ENDOWPAY equal to zero

For each mortgage, if ownhow=2 and MENPOLPD equal to -1 or 1-11
then ENDOWPAY=total of MENPOLAM else do not change ENDOWPAY

(if ownhow=1, endowpay remains set as zero)

Structural insurance

From househol record get STRCOV, STRAMT, STRPD, 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 STRPD=-1 or 1-11 STRUINS=STRAMT else do not change STRUINS

If STROTHS=1 and TENURE=1 *(insurance premium paid on structure of accommodation separately from any mortgage payments, owner occupiers only)* and STRPD=-1 or 1-11 STRUINS=STRAMT else do not change STRUINS

Service payments

From owner record get CHARGE1-7, CHAMT1-7, CHARGEP1-7 set SERVPAY=0

If CHARGE1=1 and CHARGEP1=-1, or 1-11 add CHAMT1 to SERVPAY else do not change SERVPAY
If CHARGE2=1 and CHARGEP2=-1, or 1-11 add CHAMT2 to SERVPAY else do not change SERVPAY
If CHARGE3=1 and CHARGEP3=-1, or 1-11 add CHAMT3 to SERVPAY else do not change SERVPAY
If CHARGE4=1 and CHARGEP4=-1, or 1-11 add CHAMT4 to SERVPAY else do not change SERVPAY
If CHARGE5=1 and CHARGEP5=-1, or 1-11 add CHAMT5 to SERVPAY else do not change SERVPAY
If CHARGE6=1 and CHARGEP6=-1, or 1-11 add CHAMT6 to SERVPAY else do not change SERVPAY
If CHARGE7=1 and CHARGEP7=-1, or 1-11 add CHAMT7 to SERVPAY else do not change SERVPAY

MORTCOST=MORTPAY+ENDOWPAY+STRUINS+SERVPAY

-1 not applicable to this case - TENURE ne 1.

-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
Issued : 21 April 2005
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

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 uf 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 STRAMT should be removed. The value of any structural insurance is added back in HSCOSTHH.

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, 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 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 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 = 12 or 13 MORTINT is set to -2. Cases where period codes have been skipped (where amount has been imputed) are allowed.

MIRAS

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

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. 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

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
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. 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 94~~ **January 1995**) INTRATE=~~7.64%~~ **7.84%**
If MNTHCODE=2 (~~May 94~~ **February 1995**) INTRATE=~~7.63%~~ **7.98%**
If MNTHCODE=3 (~~June 94~~ **March 1995**) INTRATE=~~7.61%~~ **8.00%**
If MNTHCODE=4 (~~July 94~~ **April 1994**) INTRATE=~~7.59%~~ **7.64%**
If MNTHCODE=5 (~~August 94~~ **May 1994**) INTRATE=~~7.56%~~ **7.63%**
If MNTHCODE=6 (~~September 94~~ **June 1994**) INTRATE=~~7.57%~~ **7.61%**
If MNTHCODE=7 (~~October 94~~ **July 1994**) INTRATE=~~7.85%~~ **7.59%**
If MNTHCODE=8 (~~November 94~~ **August 1994**) INTRATE=~~7.83%~~ **7.56%**
If MNTHCODE=9 (~~December 94~~ **September 1994**) INTRATE=~~7.84%~~ **7.57%**
If MNTHCODE=10 (~~January 95~~ **October 1994**) INTRATE=~~7.84%~~ **7.85%**
If MNTHCODE=11 (~~February 95~~ **November 1994**) INTRATE=~~7.98%~~ **7.83%**
If MNTHCODE=12 (~~March~~ **December 1994**) INTRATE=~~8.00%~~ **7.84%**

From MORTGAGE table, for each mortgage get all variables

If MORTTYPE = 1 or 3 or 4 (endowment, pension mortgage or other type of mortgage) and MORINPD not equal 12 or 13 (but may equal -1)

Calculate MORTINT = MORINPAY (how much interest did you pay last time).

If MORTPROT = 1 and INCMP = 1 and INCMPPD not equal 12 or 13 (but may equal -1), calculate MORTINT = MORTINT - INCMPAMT. (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 (but may equal -1), MORTINT = MORTINT - MENPOLAM

From MORTCONT table

If OUTSINCL=2 (amount of contribution to interest not included in amount mentioned earlier) and OUTSPD=-1 or 1-11 then MORTINT=MORTINT+OUTSAMT

If STRMORT=1 and STRPD=-1, 1-11 and STRAMT<MORTINT then MORTINT=MORTINT-STRAMT
Else if STRMORT=1 and STRPD=-1, 1-11 and STRAMT>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 STRAMT 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:

If MORTSEQ=1 and

If TAXRELF=2 (interest quoted does not include MIRAS arrangements)
and MORTLEFT>30,000
then MORTINT=MORTINT-((30,000*INTRATE*0.2)/52)

Else if TAXRELF=2 and MORTLEFT<=30,000
then MORTINT=MORTINT-((MORTLEFT*INTRATE*0.2)/52)

(morinpay will be held as a weekly amount, but the miras adjustment also has to be converted)

If MORTTYPE = 2 (repayment mortgage),

Calculate MORTINT = (MORTLEFT * INTRATE)/52

MIRAS adjustment to first mortgage:

If MORTSEQ=1 and

```
If MORTLEFT>30,000
then MORTINT=MORTINT-((30,000*INTRATE*0.2)/52)

Else if MORTLEFT<=30,000
then MORTINT=MORTINT-((MORTLEFT*INTRATE*0.2)/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 = 12 or 13 (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.

MORTISM

Purpose	: The amount of mortgage interest paid by each household using definition for ISM.
Created	: 30 April 1996
Database Table	: HOUSEHOL
Minimum Value	: 0
Maximum Value	:
Units	: Real
Validations	:
Related Variables	:
Children	:
Parents :	
Core variable/user	: ISM
Issued	: 21 April 2005
Amendments	: JS - 16 May to set DV to zero in cases where adjustment for outside payments or mortgage interest result in negative values

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

1 Definition

This variable is coded as

MORTISM	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.

This variable is coded in exactly the same way as MORTINT except that it excludes regular payments made by others outside the household. Therefore, for endowment mortgages, the adjustment which exists in the MORTINT calculation is amended. For repayment mortgages, if the total amount paid by those outside is greater than the calculated mortgage interest, it is assumed that all interest is covered by external payments and MORTISM is set to zero.

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 before 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 STRAMT should be removed. The value of any structural insurance is added back in HSCOSTHH.

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, 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 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 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 = 12 or 13 MORTISM is set to -2. Cases where period codes have been skipped (where amount has been imputed) are allowed.

MIRAS

Allowance also has to be made for MIRAS which is deducted at 20% of the interest rate on the first £30,000 of mortgages taken out for the express purpose of buying a house. All payments in MORTISM 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.

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. 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

but ii that MORTISM 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 MORTISM.

2 FRS Specification

For each HOUSEHOLD

Code	Condition
MORTISM	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.

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 94~~ **January 1995**) INTRATE=~~7.64%~~ **7.84%**
If MNTHCODE=2 (~~May 94~~ **February 1995**) INTRATE=~~7.63%~~ **7.98%**
If MNTHCODE=3 (~~June 94~~ **March 1995**) INTRATE=~~7.61%~~ **8.00%**
If MNTHCODE=4 (~~July 94~~ **April 1994**) INTRATE=~~7.59%~~ **7.64%**
If MNTHCODE=5 (~~August 94~~ **May 1994**) INTRATE=~~7.56%~~ **7.63%**
If MNTHCODE=6 (~~September 94~~ **June 1994**) INTRATE=~~7.57%~~ **7.61%**
If MNTHCODE=7 (~~October 94~~ **July 1994**) INTRATE=~~7.85%~~ **7.59%**
If MNTHCODE=8 (~~November 94~~ **August 1994**) INTRATE=~~7.83%~~ **7.56%**
If MNTHCODE=9 (~~December 94~~ **September 1994**) INTRATE=~~7.84%~~ **7.57%**
If MNTHCODE=10 (~~January 95~~ **October 1994**) INTRATE=~~7.84%~~ **7.85%**
If MNTHCODE=11 (~~February 95~~ **November 1994**) INTRATE=~~7.98%~~ **7.83%**
If MNTHCODE=12 (~~March~~ **December 1994**) INTRATE=~~8.00%~~ **7.84%**

From MORTGAGE table, for each mortgage get all variables

If MORTTYPE = 1 or 3 or 4 (endowment, pension mortgage or other type of mortgage) and MORINPD not equal 12 or 13 (but may equal -1)

Calculate MORTISM = MORINPAY (how much interest did you pay last time).

If MORTPROT = 1 and INCMP = 1 and INCMPPD not equal 12 or 13 (but may equal -1), calculate MORTISM = MORTISM - INCMPAMT. (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 (but may equal -1), MORTISM = MORTISM - MENPOLAM

From MORTCONT table

If OUTSINCL=1 (amount of contribution to interest included in amount mentioned earlier) and OUTSPD=-1 or 1-11 **and MORTISM>sum of OUTSAMT then** MORTISM=MORTISM-OUTSAMT (MORTISM less OUTSAMT) **else if calculation makes MORTISM negative, set MORTISM to zero**

If STRMORT=1 and STRPD=-1, 1-11 and STRAMT<MORTISM then MORTISM=MORTISM-STRAMT
Else if STRMORT=1 and STRPD=-1, 1-11 and STRAMT>MORTISM then do not change MORTISM
(this should only be applied to endowment mortgages because repayment mortgages are calculated and do not use respondents' answers. If, however, the amount in STRAMT is greater than MORTISM, assume that an error has been made and do not change MORTISM)

MIRAS adjustment to first mortgage:

If MORTSEQ=1 and

If TAXRELF=2 (interest quoted does not include MIRAS arrangements)
and MORTLEFT>30,000
then MORTISM=MORTISM-((30,000*INTRATE*0.2)/52)

Else if TAXRELF=2 and MORTLEFT<=30,000
then MORTISM=MORTISM-((MORTLEFT*INTRATE*0.2)/52)

(morinpay will be held as a weekly amount, but the miras adjustment also has to be converted)

If adjustment for MIRAS makes MORTISM negative, reset MORTISM to zero

If MORTTYPE = 2 (repayment mortgage),

Calculate MORTISM = (MORTLEFT * INTRATE)/52

MIRAS adjustment to first mortgage:

If MORTSEQ=1 and

If MORTLEFT>30,000
then MORTISM=MORTISM-((30,000*INTRATE*0.2)/52)

Else if MORTLEFT<=30,000

then MORTISM=MORTISM-((MORTLEFT*INTRATE*0.2)/52)

Adjustment for payments made by those outside the household

From MORTCONT table

Set up a temporary variable OUTS which adds up all occurrences of OUTSAMT:

If **OUTSAMT exists** (amount of contribution to interest ~~not~~ included in amount mentioned earlier) and OUTSPD=-1 or 1-11 then OUTS=OUTS+OUTSAMT

OUTS then needs to be subtracted from MORTISM. However, if OUTS is greater than MORTISM, MORTISM is set to zero:

If OUTS<MORTISM then MORTISM=MORTISM-OUTS
Else if OUTS>MORTISM then MORTISM=0;

-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 = 12 or 13 (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 ~~GROSSINC~~ **INDINC** - both are derived variables
Core variable/user: ~~HBAI~~ **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
Issued: 21 April 2005

1 Definition

This variable is coded as

NDDCTBThis 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).

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 + 1.15

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

If ~~GROSSINC~~ **INDINC** 108, calculate NDDCTB = NDDCTB + 1.15

Else calculate NDDCTB = NDDCTB + 2.30

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 ~~GROSSINC~~ **INDINC**- both are derived variables
Core variable/user: ~~HBAI~~ **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
Issued: 21 April 2005

1 Definition

This variable is coded as

NDDISHCThis 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).

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

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

If ~~GROSSINC~~ **INDINC** < 72, calculate NDDISHC = NDDISHC + 5
Else if ~~GROSSINC~~ **INDINC** < 108, calculate NDDISHC = NDDISHC + 9
Else if ~~GROSSINC~~ **INDINC** < 139, calculate NDDISHC = NDDISHC + 13
Else if ~~GROSSINC~~ **INDINC** >=139, calculate NDDISHC = NDDISHC + 25

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 ~~GROSSINC~~ **INDINC** - both are derived variables
Core variable/user: ~~HBAI~~ **ASD6A**
Amendments: VC - 28 February 1994 To reflect changes to version 30
VE - 19 April 1996 - to replace GROSSINC with INDINC
Issued: 21 April 2005

1 Definition

This variable is coded as

NDDRENTRThis 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).

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

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

If ~~GROSSINC~~ **INDINC**<72, calculate NDDRENTR = NDDRENTR+5
Else if ~~GROSSINC~~ **INDINC**<108, calculate NDDRENTR= NDDRENTR+9
Else if ~~GROSSINC~~ **INDINC**<139, calculate NDDRENTR=NDDRENTR +13
Else if ~~GROSSINC~~ **INDINC**>=139, calculate NDDRENTR= NDDRENTR+25.

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

NETINC

Purpose:To calculate the total amount of net income received by an adult following the : FES specification for IN407.

Created:VC - 20 September 1993
Database Table:ADULT
Minimum Value:0
Maximum Value:
Units:Real
Validations:
Related Variables:
Children:HBAI variables income before/after housing costs
Parents:OCCUPPEN INCSE1
Core variable/user:HBAI
Amendments:VC - 11 October 1993
:VC - 17 February 1994 Amended to refect version 30 changes
:VC - 1 March 1994 To exclude period codes 12 and 13
:VS - 28 April 1995 to reflect the changes made to the derived variable
:program in March 1995
:BS - 1 August 1995 Spec revised to take into account the changes in V31
Issued:21 April 2005

1 Definition

This variable is coded as

NETINCThis is the total amount of net income received by an adult from all sources.

0Not applicable - adults who have zero net income

-2Underivable due to missing values.

This variable is derived from a large variety of base data which bring together all forms of income. It includes income from
Employment as an employed earner
Self-employment
PAYE income tax refunds
Baby sitting - mail order agent
Interest from all types of savings and investments
Occupational pensions
Trade union/Friendly Society pensions
Annuities
Trust funds or covenants
Allowances from absent spouse
Allowances from family in forces friends etc
Regular allowances from organisations
Allowances from a Local Authority for a foster child
Allowances from a Local Authority for an adopted child
Income from property
Social Security benefits

Educational grants and scholarships
Cash in lieu of concessionary coal
Earnings from odd jobs
Royalties
Income from being a sleeping partner
Pension from an overseas Government
Maintenance
Any Housing Benefit paid to head of household benefit unit

Any Social Fund loan repayments are deducted as are any Income Tax refunds paid as part of an employed earner's salary.

2 FRS Specification

For each ADULT

<u>CodeCondition</u>

NETINC**1. Employed earners**

From ADULT record, if WORKING = 1 or JOBAWAY = 1
set EARNS to zero (temporary variable to calculate net earnings for a person) then process each JOB record for that person. This variable uses usual earnings wherever possible to be consistent with the other HBAI variables.

If EMPEE = 1 and PAYUSL = 1 or 3 and PAYPD not equal 12 and 13,
if PAYAMT exists add it into EARNS
else set netinc to -2

If EMPEE = 1 and PAYUSL = 2 and UPD not equal 12 and 13,
if UNETT exists add it into EARNS
else set netinc to -2

If EMPEE = 1 and PAYUSL is missing
if PAYUSL missing don't know/refusal set netinc -2

Then if CHARITY = 1 and CHRTAXF = 1
if AMTTAXF exists add it into EARNS
else set netinc -2
if CHARITY = 1 and CHROTH = 1
if AMTOTH exists add it into EARNS
else set netinc -2

If OTHDED1 = 1 add DEDUC1 to EARNS
If OTHDED2 = 1 add DEDUC2 to EARNS
If OTHDED3 = 1 add DEDUC3 to EARNS
If OTHDED4 = 1 add DEDUC4 to EARNS
If OTHDED5 = 1 add DEDUC5 to EARNS
If OTHDED6 = 1 add DEDOTH to EARNS

If TAXINC = 1 subtract TAXAMT from EARNS (this applies to both of the above).

Check - at this point if EARNS is less than zero, set it back to zero.

Any cases with no earnings should have EARNS set to zero.

2. Adjustment for possible receipt of SSP or SMP

Set ADJUST to zero (temporary variable)

If JOBAWAY = 1 and ABSWHY = 2 and (SSPSMP = 1 or 2) and PAYSLIP = 1
Calculate ADJUST = SSPAMT

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

If ADJUST >= to employment income calculated in EARNS, reset ADJUST to zero.

3. Self - employment income

Set SEINC to zero
If EMPEE = 2, calculate SEINC = INCSE1

Any cases with no self-employed earnings set SEINC to zero.

4. Other income

Set OTHINC to zero

Income as a baby-sitter

From ADULT record, if BABY1 = 1 add BABPAY into OTHINC

Income as a mail order agent

From ADULT record, if BABY2 = 1 add BABPAY into OTHINC

Allowance from absent spouse

From ADULT record, if ABSPAR = 1 and APPD not equal 12 and 13, add APAMT to OTHINC.

Allowances from spouse in forces, friends other relatives etc

From ADULT record, if ALLOW1 = 1 and APPD not equal 12 and 13, add ALLPAY1 to OTHINC.

Allowance from an organisation

From ADULT record, if ALLOW2 = 1 and ALLPD1 not equal 12 or 13, add ALLPAY2 to OTHINC.

Allowance from a Local Authority for a foster child

From ADULT record, if ALLOW3 = 1 and ALLPD2 not equal 12 or 13 add ALLPAY3 to OTHINC.

Allowance from a Local Authority for an adopted child.

From ADULT record, if ALLOW4 = 1 and ALLPD3 not equal 12 or 13, add ALLPAY4 to OTHINC.

Income from boarders/lodgers

From ADULT record, if CVPAY exists for any person in household and CVPD not equal 12 or 13, attribute income to head of household - assume this is PERSON = 1.

Income in kind

From JOB record, if LUNCHV = 1 and LV7DY = 1, add amount in LVAMT into OTHINC (luncheon vouchers)

From ADULT record, if FCASH = 1 and FCAMTPD not equal 12 or 13, add amount in FCAMT into OTHINC (cash in lieu of concessionary coal)

Royalties

From ADULT record, if ROYAL1 = 1 add ROYYR1 into OTHINC.

Income as a sleeping partner

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

Pension from an overseas Government

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

Maintenance

From ADULT record, if MNTREC = 1 and MNTPD not equal 12 or 13, add MNTAMT into OTHINC.

Council Tax Benefit

If CTREB = 1, add CTREBAMT into OTHINC only for PERSON = 1.

Odd jobs

From ADULT Rrecord if ODDJOB =1 and if OJPD not equal 12 and 13, add OJAMT into OTHINC.

Then if ODDJOB = 1 from ODDJOB record for X = 1 to 3 if OJOTH!X = 1 and exists(OJAMT) = 1 and if OJPD does not equal 12 and 13, add all occurrences of OJAMT into OTHINC.

Income from property

If PROPRENT exists add to OTHINC.

Income from sub-tenants

If SUBLET = 1, add amount held in SUBRENT into OTHINC for PERSON = 1 (head of household).

Interest/income from savings accounts or investments

From ACCOUNTS record, if ACCOUNT = 1 to 15, add amount in ACCINT to OTHINC (see attached list for definition of ACCOUNT = 1 to 15).

Occupational pensions

From PENSIONS record, if PENTYPE = 1 (occupational pension) anf PENPD not equal 12 and 13, get amount from PENPAY. If PTINC = 1 deduct PTAMT then add result into OTHINC

Trade union Friendly society pensions

From PENSIONS record, if PENTYPE = 2 and PENPD not equal 12 or 13, get amount from PENPAY and add to OTHINC. If PTINC = 1 deduct PTAMT then add result into OTHINC

Annuity/personal pension

From PENSIONS record, if PENTYPE = 3 and PENPD not equal 12 or 13, get amount from PENPAY. If PTINC = 1 deduct PTAMT then add result to OTHINC.

Trust/covenant

From PENSIONS record, if PENTYPE = 4 and PENPD not equal 12 or 13, get amount from PENPAY. If PTINC = 1 deduct PTAMT then add result to OTHINC.

Housing Benefit

From RENTER record, if BENUNIT = 1 and HBENPD not equal 12 or 13 and HBENEFIT = 1 or REBATE = 1 add in HBENAMT to OTHINC for PERSON = 1

From HOUSEHOL record, if HHSTAT = 2 then from ADULT record if CVHB = 1 and CVPD <> 12 or 13 and exists(CHBAMT) = 1 then add CHBAMT to OTHINC.

Income from benefits

From BENEFITS record, if BENPD not equal 12 and 13 and

If BENEFIT = 1 add BENAMT to OTHINC (DLA Care)

If BENEFIT = 2 add BENAMT to OTHINC (DLA Mob)

If BENEFIT = 3 add BENAMT to OTHINC (CHB)

If BENEFIT = 4 add BENAMT to OTHINC (OPB)

If BENEFIT = 5 add BENAMT to OTHINC (RP)

If BENEFIT = 6 add BENAMT to OTHINC (OAP)

If BENEFIT = 7 add BENAMT to SSBENHH (Widows Pension)

If BENEFIT = 8 add BENAMT to SSBENHH (War Disablement Pension)

If BENEFIT = 9 add BENAMT to SSBENHH (SDA)

If BENEFIT = 10 add BENAMT to SSBENHH (DWA)

If BENEFIT = 11 add BENAMT to OTHINC (AA)

If BENEFIT = 12 add BENAMT to OTHINC (Invalid Care Allowance)

If BENEFIT = 13 add BENAMT to OTHINC (UB)

If BENEFIT = 14 add BENAMT to OTHINC(Industrial Injuries)

If BENEFIT = 16 add BENAMT to OTHINC (Sickness Benefit)

If BENEFIT = 17 add BENAMT to OTHINC (IVB)

If BENEFIT = 18 add BENAMT to OTHINC (FC)

If BENEFIT = 19 add BENAMT to OTHINC (IS)

If BENEFIT = 21 add BENAMT to OTHINC (Maternity Benefit)

If BENEFIT = 26 and PRES = 1 add BENAMT to OTHINC (Any other DSS benefits)

If BENEFIT = 27 and PRES = 1 add BENAMT to OTHINC (Trade Union sick)

If BENEFIT = 28 and PRES = 1 add BENAMT to OTHINC (Friendly sick)

If BENEFIT = 29 and PRES = 1 add BENAMT to OTHINC (Private sick)

If BENEFIT = 30 and PRES = 1 add BENAMT to OTHINC (Accident)

If BENEFIT = 31 and PRES = 1 add BENAMT to OTHINC (Hospital savings)

If BENEFIT = 32 and VAR1=1 or VAR1=2 or VAR1 = 3, add BENAMT to OTHINC (Training)

If BENEFIT = 33, add BENAMT to OTHINC (Guardians Allowance)

NB - Benefit = 15 and 23 were removed as they do not have a benamt in the benefits tables (March 1995)

6. Social Fund loan repayments

From BENEFITS record, if BENEFIT = 34 calculate SOCFUND = BENAMT.

NETINC is then calculated by combining the components for each adult on the database as follows -

EARNS
- ADJUST
+ SEINC
+ TAX
+ OTHINC
- SOCFUND
-2Where any of the above variables are missing or where a period code is 12 or 13

3 Results

No results required as used as part of other derived variables

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

1 Definition

This variable is coded as

NETOCPENThis 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).

-1Not applicable to this case - adults who do not have occupational pensions

-2Unable 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>CodeCondition</u>

NETOCPENFrom PENSION table, for each pension **calculate a temporary variable OCCUP**

If PENTYPE = 1 (indicating an occupational pension is being received) and PENPD **equal to -1 or 1-11** ~~not equal 12 or 13~~, get variables PENPAY, PENTAX, PTAMT, PTINC, PENOTH, POAMT and POINC.

Compute OCCUP = PENPAY.

If PENTAX = **equal to 1 or has been skipped** (tax deducted at source) and PTINC = 2 (original amount declared <u>after</u> amount of tax deducted) do not change OCCUP

If PENTAX=1 (tax deducted at source) and PTINC=1 **and PTAMT exists** (original amount before amount of tax deducted) subtract PTAMT from OCCUP **else do not change OCCUP**

If PENOTH = 1 (other deductions) and POINC = 2 (original amount declared <u>after</u> deduction), add POAMT to OCCUP.

From ADULT table

If ROYAL3 = 1 (pension from an overseas government) get amount from ROYYR3 and add to OCCUP.

NETOCPEN will then be the sum of all occurrences of OCCUP as each adult is able to have up to 5 occupational pensions.

-1Not applicable to this case -

-2Unable to derive because any of the above variables are missing or PENPD = 12 or 13.

NSCB16, INSC17, FNSC18, PGIB19, SAYE20, PB21, NSIB22, NSDB23

Purpose: To indicate ownership of NSB assets, SAYE and premium bonds for which no interest information is collected.
Created:
Database Table: Adult
Minimum Value: 0
Maximum Value:
Units: Integer
Validations:
Related Variables: NSBOCTI, SAYECTI, PRBOCTI (publication DVs)
Children:
Parents:
Core variable/user: PSM
Amendments

1 Definition

These variables are coded as

NSCB16, INSC17, FNSC18, PGIB19, SAYE20, PB21, NSIB22, NSDB23

0 No such accounts

-1Not applicable to this case (no cases should exist - either individuals, benefit units or households have or don't have accounts).

-2Unable to derive due to missing values (not applicable to this specification: variables are a simple count of records).

2 FRS Specification

CodeCondition

Process ACCOUNTS table via ADULT table

Set NSCB16, INSC17, FNSC18, PGIB19, SAYE20, PB21, NSIB22, NSDB23 to zero

NSCB16 If ACCOUNT=16 then NSCB16=NSCB16+1 (National Savings Capital Bonds)

INSC17 If ACCOUNT=17 then INSC17=INSC17+1 (Index linked National Savings Certificates)

FNSC18 If ACCOUNT=18 then FNSC18=FNSC18+1 (Fixed Interest National Savings Certificates)

PGIB19 If ACCOUNT=19 then PGIB19=PGIB19+1 (Pensioner's Guaranteed Income Bonds)

SAYE20 If ACCOUNT=20 then SAYE20=SAYE20+1 (SAYE)

PB21	If ACCOUNT=21 then PB21=PB21+1 (Premium bonds)
NSIB22	If ACCOUNT=22 then NSIB22=NSIB22+1 (National Savings Income Bonds)
NSDB23	If ACCOUNT=23 then NSDB23=NSDB23+1 (National Savings Deposit Bonds)

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: ~~ASD3A~~ **PSM**
Amendments: VC - 17 February 1994 Amended to reflect version 30 changes

1 Definition

This variable is coded as

OCCUPNUMThe total number of occupational pensions a person receives from both a former employer or from any pensions from overseas governments or companies.

-1Not 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

CodeCondition

OCCUPNUMSet 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

1 Definition

This variable is coded as

OCCUPPENThis 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).

-1Not applicable to this case - adults who do not have occupational pensions

-2Unable 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 deal 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 (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 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:

iskipped values of PENPD and PENTAX where PENPAY has been imputed (PENTAX assumed to be equal to 2 for these cases)
iimissing PTAMT where PENTAX has been imputed to yes
iiimissing PTINC where PTAMT has been skipped (assumed to be after)

2 FRS Specification

For each ADULT with a pension record

CodeCondition

OCCUPPENFrom PENSION table, for each pension

If PENTYPE = 1 (indicating an occupational pension is being received) and PENPD ~~not equal 12 or 13~~ **equal to -1 or 1-11**, 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 after amount of tax deducted), **and PTAMT exists** add PTAMT to OCCUP **(otherwise, do not change OCCUP)**.

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

From ADULT table

If ROYAL3 = 1 (pension from an overseas government) get amount from ROYYR3 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.

-1Not applicable to this case -

-2Unable to derive because any of the above variables are missing or PENPD = 12 or 13.

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

OLDGROSS

Purpose :The grossing factors in OLDGROSS are **not meant to be an alternative to GROSS**. They are on the flatfile because they represent 1994/95 equivalents of the old FES grossing factors.
Created :
Database Table : BENUNIT, HOUSEHOL
Minimum Value :
Maximum Value :
Units :
Validations :
Related Variables :
Children :
Parents :
Core variable/user : ASD6A
Amendments :

1 Definition

The grossing factors in OLDGROSS are **not meant to be an alternative to GROSS**. They are on the flatfile because they represent 1994/95 equivalents of the old FES grossing factors. Analysts can therefore use OLDGROSS to see the effect of moving to the improved grossing system represented by GROSS.

Units

OLDGROSS is consistent with GROSS. It gives actual numbers of benefit units.

Derivation

OLDGROSS has been produced in the traditional manner and has different factors for each of 17 family types. Unlike GROSS it is applicable at ***benefit unit level only***. For household level analyses, the analyst has to use the first benefit unit's grossing factor, or an average across the household's benefit units of their grossing factors. The control totals for the family types were found by applying the iterative proportional scaling technique to the available demographic information.

The OLDGROSS weights can be used to see the effect of moving to the improved GROSS factors. The control totals used for OLDGROSS have also been used to generate 1994/5 FES grossing factors. Hence to study the effect of moving from the 1993/4 FES to the 1994/5 FRS, analysts could calculate four sets of statistics:

(i) using the 93/4 FES
(ii) using the 94/5 FES
(iii) using the 94/5 FRS with OLDGROSS
(iv) using the 94/5 FRS with GROSS.

Apart from a change in year for the FES and control totals, the move from (i) to (ii) also involves two changes in information source for the controls. A change to GAD estimates for marital status breakdown has the effect of increasing the number of single people and decreasing the number of couples. A move from the John Haskey lone parent estimate to an estimate by Howard Redway has a minimal effect (but should be an improvement in future year). The difference between (ii) and (iii) should solely reflect the difference between FES and FRS. The move from (iii) to (iv) then represents the effect of improving the grossing system.

For further information on GROSS see the GROSS specification. For further information on grossing up in general, see Andrew Ray's Analytical Note 5, copies of which are held in ASD6.

Control totals

The following figures relate to benefit units in the population.

Control totals for OLDGROSS	
Pensioner couples, head aged 65-74	1,805,000
Pensioner couples, head aged 75+	819,000
Pensioner singles, men	893,000
Pensioner singles, women aged 60-74	1,688,000
Pensioner singles, women aged 75+	1,689,000
Couples with no children	5,781,000
Couples with 1 child	2,095,000
Couples with 2 children	2,302,000
Couples with 3 children	1,006,000
Male lone parents	121,000
Female lone parents	1,420,000
Single men aged <30	3,450,000
Single men aged 30-54	2,194,000
Single men aged 55-64	482,000
Single women aged <20	591,000
Single women aged 20-39	1,872,000
Single women aged 40-59	1,083,000

OPBRECHH

Purpose: This variable is the total weekly amount of One Parent Benefit within the household.
Created: VC - 7 September 1993
Database Table: HOUSEHOL
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: ISRECDHH, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 1 March 1994 To exclude any period codes 12 or 13
: JS - 21 February 1996 to allow for missing BENPD where BENAMT has been imputed
Issued: 21 April 2005

1 Definition

This variable is coded as

OPBRECHHThis is the total weekly amount received of One Parent Benefit within the household.

0Not applicable to this case

-2Unable to derive as missing values

NB There will be no code -1, not applicable as 0 is used instead for this variable.

This variable is derived by adding together the amount of One Parent Benefit received by any person in the household. The amount of One Parent Benefit is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 4 the person is receiving One Parent Benefit and the amount held in BENAMT for this record should be added into OPBRECHH.

The total should include One Parent Benefit received only, therefore, if the household receives any other form of benefit in addition to OPB, this would not be added into the total. Households with no OPB are not applicable to this case.

However, if the period code for the benefit 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 BENPD = 12 or 13 OPBRECHH is set to -2.

2 FRS Specification

For each HOUSEHOLD, set OPBRECHH to zero

CodeCondition

OPBRECHHProcess all BENEFITS records for household

If BENEFIT = 4 and BENPD ~~not equal 12 or 13~~ **equals -1 or 1-11**, add BENAMT into OPBRECHH (OPB)

OPBRECHH will then be the total amount of benefit received from this benefit by a particular household regardless of the number of adults.

0Not applicable as does not have any OPB recipients.

-2Unable to derive as above variables are missing or BENPD = 12 or 13.

3 Results

Tabulation is required to show the number of households by the total amount of One Parent Benefit received split into the following weekly bands

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

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: 21 April 2005
Amendments:

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

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 : **ASD4A**
Issue date : 21 April 2005
Amendments : JS - 21 February 1996 - to allow for missing values where variables have been imputed
: JS - 15 March 1996 to bring in line with SEH categories (at request of ASD3F)

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** (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**

4 **Rented privately - furnished**

~~4~~ **5** Owned with mortgage (including co- and shared ownership)

~~5~~ **6** Owned outright

-2 unable to derive due to missing values

-1Not applicable to this case where BENUNIT > 1 or SUBLET = 2

-2Unable 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 21 April 2005
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: 21 April 2005
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 emphasisethat 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

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

1 Definition

This variable is coded as

SUPERANThe total amount of superannuation or pension contributions deducted from a person's earnings from all jobs also includes contributions to additional voluntary pension.

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

-2Unable 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 repsonse 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 (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 SUPERAN is set to -2.

2 FRS Specification

For each ADULT with record Job for all jobs

<u>CodeCondition</u>

SUPERANIf PAYPD ~~not equal 12 or 13~~ **equals -1 or 1-11** 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.

-1Not applicable in this case - where OTHDED1 = 2 or 3, OTHDED2 = 2 or 3 or OTHDED1/OTHDED2 not exist or there are no job records

-2If variable cannot be defined because of missing data - where there is a job record but no values or if PAYPD = 12 OR 13.

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: 21 April 2005
Amendments: VC - 28 January 1993
: VC - 15 March 1993 Amendments to insert the part-own part-renter category : into the appropriate renter category, to put tenure = 2 or 3 into the owner : categories and to exclude tenure = 5 as these tenants are not eligible for : Housing Benefit.
: VC - 18 May 1993. Amended to re-insert tenure = 5 as a separate : category and also to add another category for co-ownership schemes : as both of these groups are not eligible for HB.
: VC - 9 June 1993. Amended to insert extra category for shared ownership.
: BS - 3 August 1995. Amended to include changes to V31 of the questionnaire. New var FURNISH asked if TENURE equals new codes 6, 7 or 8.
:JS - to clarify tentype=3 category and to allow skipped values of LANDLORD, FURNISH and OWNHOW (in cases where LANDLORD or TENURE have been imputed)
:JS - to note categories are actually slightly different from the name suggested!

1 Definition

This variable is coded as

1LA rented ~~unfurnished~~
2Housing Association
3Other **private** rented unfurnished
4Other **private** rented furnished
5Owned with mortgage
6Owned outright
7Rent free
8Other Crown Estates/Government Departments.
9Co-ownership schemes
10 Shared ownership

-1Not 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.

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	1) Owns/is buying	**LANDLORD**	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
	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		

For each household, process HOUSEHOL record

CodeCondition

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

1If TENURE = 4 or 5 and LANDLORD = 1 or 2 (council/New Town Corporation)

2If TENURE = 4 or 5 and LANDLORD = 3 (Housing Association)

3If TENURE = 4 or 5 and **((**LANDLORD = 4**) or (LANDLORD=** 6, 7 or 8 and FURNISH = 2**))** (Crown Estates/other unfurnished)

4If TENURE = 4 or 5 and **((**LANDLORD = 6, 7 or 8 and **(**FURNISH = 1 **or FURNISH= -1)) or LANDLORD=-1)** (other furnished)

8If TENURE = 4 or 5 and LANDLORD = 5 (other Crown Estates/Govt Dept)

Where TENURE = 1, process OWNER record for details of how the property is owned

5If TENURE = 1 and OWNHOW = 2 (owns/bought with mortgage or loan)

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

Where TENURE = 6

7If TENURE = 6 (rent free).

Where TENURE = 2

9If TENURE = 2 (co-ownership)

Where TENURE = 3

10If TENURE = 3 (shared ownership)

-1Not applicable to this case
-2Unable 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

ASingle person living in Local Authority housing.
BSingle person living in Housing Association accommodation.
CSingle person living in an unfurnished flat rented from a private landlord.
DSingle person living in a furnished flat rented from a private landlord.

ESingle person living in his/her own home with a mortgage outstanding on the property.
FSingle person living in his/her own home with no outstanding mortgage.
GSingle 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 : **HBM**
Amendments: : updated to V31: totsav now held on BU record

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 assetype 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 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.

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 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%.

The coding of Totsav was changed at the beginning of July ~~1995~~ **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 From BENUNIT record, if TOTSAV = 2 or (TOTSAV=3 and SAMPLQTR=2,3 or 4)

From ASSETS record, for each asset held for each adult

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 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).

TOTCAPBU is then the total of any occurrences of the above

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)

Set the interest rates for each type of account
Then calculate amounts as follows -

From ACCOUNTS record

If ACCOUNTS = 1, calculate CAP = ACCINT/r1%
If ACCOUNTS = 2, calculate CAP = ACCINT/r2% or r3%
If ACCOUNTS = 3, calculate CAP = ACCINT/r4% or r5% or r6%
If ACCOUNTS = 4, calculate CAP = ACCINT/r7%
If ACCOUNTS = 5, calculate CAP = ACCINT/r8%
If ACCOUNTS = 6, calculate CAP = ACCINT/r9%
If ACCOUNTS = 7, calculate CAP = ACCINT/r10%
If ACCOUNTS = 8, calculate CAP = ACCINT/r11%
If ACCOUNTS = 9, calculate CAP = ACCINT/r12%
If ACCOUNTS = 10, calculate CAP = ACCINT/r13%
If ACCOUNTS = 11, calculate CAP = ACCINT/r14%
If ACCOUNTS = 12, calculate CAP = ACCINT/r15%
If ACCOUNTS = 13, calculate CAP = ACCINT/r16%
If ACCOUNTS = 14, calculate CAP = ACCINT/r17%
If ACCOUNTS = 15, calculate CAP = ACCINT/r18%

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 then TOTCAP=TOTCAP*1.20

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	:
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 - 8 November 1996 - To take into account the fact that TOTSAVE categories changed during the 1994/95 sample year

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 assetype indicates whether or not a child has a particular asset so if this variable equals 3, 5, 7, 10, 12 or 14 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) 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). 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 ret to £1499 and if the amount calculated is under £2000 and TOTSAVE = 3, TOTCAP is set to £20001.

The coding of Totsav was changed at the beginning of July ~~1995~~ 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 TOTCAPCH

2 FRS Specification

For each CHILD

Code Condition

TOTCAPCH From CHILD table, if TOTSAVE = 2 **or (TOTSAV=3 and SAMPLQTR=2,3 or 4)**

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 HOWMUCH.
If assetype = 7, get 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 HOWMUCH.

TOTCAPCH is then the sum of the above where TOTSAVE = 2 **or (TOTSAV=3 and SAMPLQTR=2,3 or 4)**

.

Else if (TOTSAV=1,3,4 or missing and SAMPLQTR=1) or (TOTSAV=1,4 or 5 or missing and SAMPLQTR=2,3 or 4)

~~If TOTSAVE = 1, 3 or 4~~

~~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, calculate CAP = ACCINT/r2% or r3%
If ACCOUNTS = 25, calculate CAP = ACCINT/r4%
If ACCOUNTS = 26, calculate CAP = ACCINT/r5%
If ACCOUNTS = 27, calculate CAP = ACCINT/r5%
If ACCOUNTS = 28, calculate CAP = ACCINT/r7%
If ACCOUNTS = 29, calculate CAP = ACCINT/r8%
If ACCOUNTS = 30, calculate CAP = ACCINT/r8%

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 **-** 1 April 1997: add in where 3 grants recorded at GRTNUM
Issue date : 21 April, 2005

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,~~ GRTAMT1, GRTVAL1~~, GRTSCE2 **GRTDIR1**,~~ GRTAMT2 and GRTVAL2~~ **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)

Code Condition

TOTGNTCH From CHILD table,

If GRTNUM = 1 (Number of grants = one) and GRTSCE1 = 1 (source is state), get amount from ~~GRTAMT1~~ **GRTDIR1** (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 ~~GRTVAL1~~ **GRTDIR1** (amount of grant **paid directly to student**).

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 ~~GRTAMT2~~ **GRTDIR2**.
If GRTSCE2 = 2 or 3 (source is private or overseas), get amount from ~~GRTVAL2~~ **GRTDIR2**.

TOTGNTCH will then be the sum of the two grants.
This is equivalent to summing all occurrences of GRTDIR1 and GRTDIR2 for each child in full time education

(NB - this does not include top-up loans as they are not payable to children)

-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 - 1 April 1997 - to add in where 3 or more grants recorded

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, ~~Grtamt1, Grtval1~~ **GrtDir1**, Grtsce2, ~~Grtamt2~~ and ~~Grtval2~~ **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=96** and ~~School~~ **TYPEED**= 7 (university, polytechnic, further education)

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

TOTGRANT From ADULT table,

If Grtnum = 1 (Number of grants = one) and Grtsce1 = 1 (source is state), get amount from ~~Grtamt1~~ **GrtDir1** (amount of grant).
If Grtnum = 1 (Number of grants = one) and Grtsce1 = 2 or 3 (source is private or overseas) get amount from ~~Grtval1~~ **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 ~~Grtamt2~~ **GrtDir2**.
If Grtsce2 = 2 or 3 (source is private or overseas) get amount from ~~Grtval2~~ **GrtDir2**

Changes have the effect of totalling all occurrences of GrtDir1 and GrtDir2 for each adult!

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

TOTHOURS

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.

NB - This replaces seftpt as this was only produced for self-employed people - tothours also includes employees .

1 Definition

This variable is coded as

TOTHOURS	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

TOTHOURS 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 TOTHOURS to zero

Code	Condition
TOTHOURS	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) an 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:

1 Definition

This variable is coded as

1 Less than £1,500

2Over £1,500 and up to £20,000

3 Over £20,000

4Does 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)

-2Unable 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

2Over £1,500 and up to £8,000

3 Over £8,000 and up to £20,000

4 Over £20,000

5Does 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

Code	Condition
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 : **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 - 9 December 1996 - To amend for change in data structure regarding TTWCODE1 and TTWCODE2

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 may be coded as having up to 6 modes of transport and costs are collected on all types except walk/bicycle or "other" (TTWMOD=1 or 6). 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.

Questionnaire asks about total distance travelled to usual place of work. Where respondents use both car/motorcycle and train/bus/tube, original specification will double count costs. Therefore have to assume that most of the journey is by train/bus/tube and that the car/van/motorcycle is just to get individuals to the station. For want of anything better at the moment (although potential sources of better information are being pursued), assume average distance travelled by car/motorcycle is 2 miles.

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=4 (bus/train/tube) or TTWMOD=5 (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=2 and (TTWMOD=4 or TTWMOD=5)
then TTWCOSTS=TTWCOSTS+(4*TTWFRQ*carrate)

Else if only car/van used (or used in combination with walking/bicycle) calculate total costs per mile per week (CARCOST - based on midpoints of TTWFAR categories),

If TTWMOD=2 (car or van)
and TTWFAR=3 then CARCOST=0.5*2*TTWFRQ*carrate
and TTWFAR=4 then CARCOST=2*2*TTWFRQ*carrate
and TTWFAR=5 then CARCOST=4*2*TTWFRQ*carrate
and TTWFAR=6 then CARCOST=7.5*2*TTWFRQ*carrate
and TTWFAR=7 then CARCOST=17.5*2*TTWFRQ*carrate
and TTWFAR=8 then CARCOST=32.5*2*TTWFRQ*carrate

1994/95 carrate=42p per mile

Then calculate TTWCOSTS, taking into account any contributions

If TTWMOD=2
If TTWPAY=1 (all) then TTWCOSTS=TTWCOSTS+CARCOST
else If TTWPAY=2 (some) and TTWCODE**1**=1 **or TTWCODE2=1** then TTWCOSTS=TTWCOSTS+TTWCOST
else If TTWPAY=2 and TTWCODE**1**=2 **or TTWCODE2=2**
then TTWCOSTS=TTWCOSTS+(CARCOST-TTWCOST)
(if CARCOST<TTWCOST then set TTWCOSTS to zero)
~~else if TTWCOST or TTWPAY has been skipped then do not change TTWCOSTS~~

else if TTWCOST or TTWPAY has been skipped then TTWCOSTS=TTWCOSTS+CARCOST

Else if respondent drives a motorcycle, use similar approach, first checking whether motorcycle used in combination with bus/train/tube or works bus/company transport

```
If TTWMOD=3 and (TTWMOD=4 or TTWMOD=5)
        then TTWCOSTS=TTWCOSTS+(4*TTWFRQ*bikerate)
```

Else if only motorbike used (or in combination with walking/bicycle) calculating variable BIKECOST

```
If TTWMOD=3 (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
```

1994/95 bikerate=9p per mile

Then calculate TTWCOSTS, taking into account any contributions

```
     If TTWMOD=3
     If TTWPAY=1 (all) then TTWCOSTS=TTWCOSTS+BIKECOST
else If TTWPAY=2 (some) and TTWCODE1=1 or TTWCODE2=1 then
     TTWCOSTS=TTWCOSTS+TTWCOST
else If TTWPAY=2 and TTWCODE1=2 or TTWCODE2=2
     then TTWCOSTS=TTWCOSTS+(BIKECOST-TTWCOST)
                (if BIKECOST<TTWCOST then set TTWCOSTS to zero)
```

~~else if TTWCOST or TTWPAY has been skipped then do not change TTWCOSTS~~

else if TTWCOST or TTWPAY has been skipped then TTWCOSTS=TTWCOSTS+BIKECOST

0 TTWFAR=1 (work at home, live at work, no work journey)
If TTWPSS=1 and PSSAMT=0 (has pass and costs nothing)
or TTWPSS=2 and FARE=0 (does not have pass but fare costs nothing)
or cash received from passengers etc greater than calculated costs (costs therefore set to zero)
or ((TTWMOD=2 or TTWMOD=3) 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: will also apply to individuals who walk to work)
or TTWFAR=2 (varies, no usual place of work)

or TTWMOD=6 (other form of transport: no information collected)

-2 Any variables missing

NB: For presentational purposes, -1 cases should be excluded on basis of TTWFAR=2 or TTWMOD=6. However, will be left with -1 cases referring to those who walk/bicycle to work; these should be included as part of "no costs" cases.

UBRECDHH

Purpose: This variable is the total weekly currently amount received from : Unemployment Benefit by each person in the household.
Created: VC - 7 September 1993
Database Table: HOUSEHOL
Minimum Value: 0
Maximum Value:
Units: Real
Validations:
Related Variables: ISRECDHH, DISBENHH....
Children:
Parents:
Core variable/user: HBAI
Amendments: VC - 1 March 1994 To exclude any period codes 12 or 13
: JS - 21 February 1996 to allow skipped BENPD where BENAMT has been imputed and update for V31
Issued: 21 April 2005

1 Definition

This variable is coded as

UBRECDHHThis is the total weekly amount currently received from Unemployment Benefit by each person in the household.

0Not applicable to this case

-2Unable to derive as missing values

This variable is derived by adding together the amount of Unemployment Benefit received by any person in the household. The amount of Unemployment Benefit is held in the variable BENAMT. When an adult receives a particular benefit, a record is created in the database which is identified by the BENEFIT variable. Therefore, where BENEFIT = 13 the person is receiving Unemployment Benefit and the amount held in BENAMT for that record should be added into UBRECDHH. ~~The variable PRES indicates whether UB is currently in receipt or received only with last 12 months.~~

The total should include Unemployment Benefit received only, therefore, if the household receives any other form of benefit in addition to UB, this would not be added into the total. Households with no UB are not applicable to this case.

However, if the period code for the benefit 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 BENPD = 12 or 13 UBRECHH is set to -2.

2 FRS Specification

For each HOUSEHOLD, set UBRECDHH to zero

CodeCondition

UBRECDHHProcess all BENEFITS records for household

If BENEFIT = 13 ~~and PRES = 1~~and BENPD ~~not equal 12 or 13,~~ **equals -1 or 1-11** add BENAMT into UBRECDHH

UBRECDHH will then be the total amount of benefit received from this benefit by a particular household regardless of the number of adults.

0Not applicable as does not have any UB recipients

-2Unable to derive as above variables are missing or BENPD = 12 or 13.

3 Results

Tabulation is required to show the number of households by the total amount of Unemployment Benefit received split into the following weekly bands

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

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**
Issue date:
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

1 Definition

This variable is coded as

UGRSPAYThe 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.
-1Not applicable where an adult does not have any jobs.

-2Unable 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 PAYSLIP = 1).

If the payslip has not been consulted (where PAYSLIP = 2) 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 OTHDED6 = 1 there will be a deduction for pension/superannuation, union fees, friendly societies, sports social clubs 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 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 OTHDED6, DEDUC1 to DEDUC5 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 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.

UGRSPAYFrom JOB table

IF WORKING=1 or JOBAWAY=1 - process each JOB record for that person where EMPEE=1

If PAYPD ~~not equal 12 or 13~~ **equals -1 or 1-11** and

If PAYSLIP = 1 **and GRSWAGE exists**, calculate UGRSPAY = amount in GRSWAGE.

Or if PAYSLIP = 2 **or (PAYSLIP=1 and GRSWAGE=-1)**, calculate UGRSPAY as follows

UGRSPAY = ~~sum of UGRSPAY,~~ PAYAMT and

If PAYE **exists** add PAYE to UGRSPAY

If NATINS **exists** add NATINS to UGRSPAY

If ~~CHRTAXF = 1~~ AMTTAXF **exists** add AMTTAXF to UGRSPAY

If ~~CHROTH = 1 add~~ AMTOTH **exists** add AMTOTH to UGRSPAY

If ~~OTHDED1 add~~ DEDUC1 **exists** add DEDUC1 to UGRSPAY

If ~~OTHDED2 add~~ DEDUC2 **exists add DEDUC2** to UGRSPAY

If ~~OTHDED3 add~~ DEDUC3 **exists add DEDUC3** to UGRSPAY

If ~~OTHDED4 add~~ DEDUC4 **exists add DEDUC4** to UGRSPAY

If ~~OTHDED5 add~~ DEDUC5 **exists add DEDUC5** to UGRSPAY

If ~~OTHDED6 = 1~~ DEDOTH **exists add DEDOTH** to UGRSPAY

Then if **JOBTYPE=1 and** ~~TAXINC = 1, subtract~~ **TAXAMT exists subtract** TAXAMT from total UGRSPAY

If **JOBTYPE=1 and** PAYUSL=no **and UGROSS exists** and UPD=-1 or 1-11 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)

-2If any of the above variables are missing or PAYPD = 12 or 13 *(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
Issued: 21 April 2005

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.

CodeCondition

For each BENUNIT record:

1For 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:21 April 2005
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

1 Definition

This variable is coded as

WATSEWRTThe total weekly amount of water and sewerage rates paid by each household in England and Wales

-1Not applicable to this case (Scottish Cases)

-2Unable 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.

CodeCondition

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 WATAMT exists WATSEWRT=WATSEWRT+WATAMT else don't change WATSEWRT

If SEWAMT exists WATSEWRT=WATSEWRT+SEWAMT else don't change WATSEWRT

If 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 WSINCAMT exists WATSEWRT=WATSEWRT+WSINCAMT
Else if WATERINC exists and =1 or SEWERINC exists and =1 and WSINCAMT=-1 (values of waterinc or sewerinc have been imputed but WSINCAMT has been skipped) call WATER WSINCAMT=WSINCAMT+WATER
Else don't change WATSEWRT

-1Not applicable to this case : Scotland

-2Not 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**
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

1 Definition

This variable is coded as

YOUNGCHThe age of the youngest child in the benefit unit.

-1Not applicable to this case

-2Unable 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,

CodeCondition

YOUNGCHFrom CHILD table,

If DEPCHILDB > 0, use MINR function to obtain smallest value of age.

-1Not applicable to this case - DEPCHILDB = 0

-2Unable to derive due to age variable missing

3 Results

Tabulation to show number of youngest children

4 Test Cases