

Adult Dental Health Survey

2009

Derived Variable Specification

Contents

CLASSIFICATION **8**

AGE	8
DVAGE Age	8
Ageband1 (D) Grouped age - 6 cats	8
Ageband2 (D) Grouped age - 5 cats	8
Ageband3 (D) Grouped age - 3 cats	8
Ageband4 (D) Grouped age - 7 cats	8
Ageband6 (D) Grouped age - 4 cats, lowest 55+	8
Ageband7 (D) Grouped age - 5 cats, highest 55+	8
AGEBAND5 (D) Grouped age - 8 cats	9
NS SEC SOCIO-ECONOMIC CLASSIFICATION	10
HRPsec8 NS-SEC HRP - Eight-class version	10
HRPsec5 NS-SEC HRP - Five-class version	10
HRPsec3 NS-SEC HRP - Three-class version	10
INCOME	12
TelDV - DV for weekly income amount	12
HlIdDV - DV for weekly income amount	12
ETHNIC GROUP	12
Ethnicg - grouped ethnicity	12
INDEX OF MULTIPLE DEPRIVATION	13
IMDE_D - Deciles - England	13
IMDW_D - Deciles - Wales	13
IMDNI_D - Deciles - Northern Ireland	13

EXAMINATION **15**

EXAMINATION VARIABLES	15
ARCHES (D) Arches with natural teeth	15
EXOUT (D) Examination outcome	15
CORONAL SURFACES (TOOTH VARIABLES)	16
PLUR1 etc (D) Plaque - upper right 1 etc	16
PLLL1 etc (D) Plaque - lower left 1 etc	16
PLUR8p to PLLL8p condition of individual teeth - recoded	17
DUR1 etc (D) Distal surface code - upper right 1 etc	18
OIUR1 etc (D) Occlusal/Incisal surface code - upper right 1 etc	18
MUR1 etc (D) Mesial surface code - upper right 1 etc	18
BUR1 etc (D) Buccal surface code - upper right 1 etc	18
LUR1 etc (D) Lingual/Palatal surface code - upper right 1 etc	18
TSTATUR1 etc (D) Tooth condition - upper right 1 etc	23
CORONAL SURFACES (COUNTS OF TEETH)	28
NUMMISS (D) No. of missing teeth	28
NUMTOOTH (D) No. of teeth	28
NUMANTTH (D) No. of anterior teeth	28
NUMSUT88 (D) No. of sound and untreated teeth (88 def)	28
NUMROS88 (D) No. of restored, otherwise sound teeth (88 def)	28
NUMDU88 (D) No. of decayed or unsound teeth (88 def)	28
NUMSUT98 (D) No. of sound and untreated teeth, not incl. roots (98 def)	28
NUMROS98 (D) No. of restored, otherwise sound teeth (98 def)	28
NUMDU98 (D) No. of decayed or unsound teeth (98 def)	28
NUMREST (D) No. of restored teeth (not incl. roots)	28
NUMFIL (D) No. of filled teeth	28
NUMAMF (D) No. of teeth filled with amalgam	28
NUMNAF (D) No. of teeth with non-amalgam fillings	28
NUMSEAL (D) No. of teeth with sealants	28
NUMSHIM (D) No. of teeth with shims/veneers	28
NUMSF98 (D) No. of teeth with sound fillings	28
NUMSFD98 (D) No. of teeth with sound fillings + other decay	28
NUMFF98 (D) No. of teeth with fractured fillings	28
NUMFFD98 (D) No. of teeth with fractured fillings and decay	28
NUMFFS98 (D) No. of teeth with fractured fillings and no other decay	28
NUMDF98 (D) No. of teeth with decayed fillings	28

NUMDFD98	(D) No. of teeth with decayed fillings + decay	28
NUMDFS98	(D) No. of teeth with decayed fillings + no other decay	28
NUMFND98	(D) No. of filled teeth with no decay	28
NUMDEC98	(D) No. of teeth with primary decay	28
NUMDNR98	(D) No. of teeth with decay and no other fillings/s/v	28
NUMDWR98	(D) No. of teeth with decay and other fillings/s/v	28
NUMUNRST	(D) No. of unrestorable teeth	28
NUMCAV	(D) No. of teeth with cavitated caries	28
NUMVIS	(D) No. of teeth with visual caries	28
NUMTDEC	(D) No. of teeth with decay	28
NUMTPRIM	(D) No. of teeth with primary decay	28
NUMPCAV	(D) No. of teeth with primary cavitated caries	28
NUMPVIS	(D) No. of teeth with primary visual caries	28
NUMTREC	(D) No. of teeth with recurrent decay	28
NUMRCAV	(D) No. of teeth with recurrent cavitated caries	28
NUMRVIS	(D) No. of teeth with recurrent visual caries	28
NUMPRRE	(D) No. of teeth with primary and recurrent caries	28
NUMUSF88	(D) No. of teeth with unsound fillings (88 def)	28
NUMFD88	(D) No. of filled and decayed teeth (88 def)	28
NUMDNT88	(D) No. of decayed teeth with no treatment (88 def)	28
NUMCROWN	(D) No. of crowns	28
UNSCORED	(D) No. of unscorable teeth	28
NUMFRES	(D) No. of teeth with fractured restorations (not incl. roots)	28
NUMIMP	(D) No. of implants	28
NUMBRID	(D) No. of bridge pontics	28
NUMSNAT	(D) No. of natural tooth surfaces (not incl. roots)	28
NUMSNATR	(D) No. of natural tooth surfaces including roots	29
NUMTDECR	(D) Number of teeth with active decay, including roots	32
CORONAL SURFACES (MOUTH VARIABLES – GROUPED ANSWERS CATEGORIES)		32
GRPMISS1	(D) No. of missing teeth - 5 cat	32
GRPSUT88	(D) No. of sound, untreated teeth (88 def)	32
GRPSUT98	(D) No. of sound, untreated teeth (98 def)	32
GRPMISS2	(D) No. of missing teeth - 6 cat	32
GRPSUT6	(D) No. of sound and untreated teeth (6 cat)	32
GRPROS88	(D) No. of restored otherwise sound teeth (88 def)	33
GRPROS98	(D) No. of restored otherwise sound teeth (98 def)	33
GRPDU88	(D) No. of decayed or unsound teeth (88 def)	33
GRPDU98	(D) No. of decayed or unsound teeth (98 def)	33
GRPCROWN	(D) No. of crowns - grouped	33
GRPTOOTH	(D) Number of natural teeth - 6 cat	33
GRPTREC	(D) Number of teeth with recurrent caries - 3 cat	34
NUMTTHG2	(D) No of natural teeth for Wales/Ni chapters	34
CORONAL SURFACES (MOUTH VARIABLES - DICHOTOMOUS)		35
HAS21	(D) Has 21 or more natural teeth	35
HASUT88	(D) Has 18 or more sound, untreated teeth (88 def)	35
HASROS88	(D) Has 12 or more restored otherwise sound teeth (88 def)	35
HASDU88	(D) Has 1 or more decayed or unsound teeth (88 def)	35
HASUT98	(D) Has 18 or more sound, untreated teeth (98 def)	35
HASROS98	(D) Has 12 or more restored otherwise sound teeth (98 def)	35
HASDU98	(D) Has 1 or more decayed or unsound teeth (98 def)	35
HASTDEC	(D) Has 1 or more decayed teeth	35
HASTPRIM	(D) Has 1 or more teeth with primary decay	35
HASPCAV	(D) Has 1 or more teeth with primary cavitated caries	35
HASPVIS	(D) Has 1 or more teeth with primary visual caries	35
HASTREC	(D) Has 1 or more teeth with recurrent caries	35
HASRCAV	(D) Has 1 or more teeth with recurrent cavitated caries	35
HASRVIS	(D) Has 1 or more teeth with recurrent visual caries	35
HASPRRE	(D) Has 1 or more teeth with both primary and recurrent caries	35
HASTDECR	(D) Has 1 or more teeth with active decay, including roots.	35
HASUNRST	(D) Has 1 or more unrestorable teeth	35
HASCROWN	(D) Has 1 or more crowns	35
HASIMP	(D) Has 1 or more implants	35
HASBRID	(D) Has 1 or more bridge pontics	35
CORONAL SURFACES (MOUTH VARIABLES – COUNTS OF SURFACES)		36
NUMSPALL	(D) No. of filled surfaces in post. teeth - all	36
NUMSPAM	(D) No. of filled surfaces in post. teeth - amalgam	36
NUMSPOF	(D) No. of filled surfaces in post. teeth - other	36
NUMSALL	(D) No. of filled surfaces - all	36
NUMSAMA	(D) No. of surfaces filled - amalgam	36
NUMSOFA	(D) No. of surfaces filled - other	36
NUMSSU	(D) No. of sound untreated surfaces (not incl. roots)	36
NUMSRES	(D) No. of restored surfaces (not incl. roots)	36
NUMSREC	(D) No. of surfaces with carious restorations (not incl. roots)	36
NUMSFRC	(D) No. of surfaces with fractured restorations (not incl. roots)	36

PLAQUE	37
PLAQUE (D) Visible plaque present	37
NUMPLQ (D) No. of teeth with visible plaque	37
NUMNOPLQ (D) No. of teeth without visible plaque	37
CONDITION OF ROOT SURFACES	38
ROOTUR1 etc (D) Root condition - upper right 1 etc	38
NUMRS (D) Number of sound roots	40
NUMRE (D) Number of exposed roots	40
NUMRW (D) Number of worn roots	40
NUMRD (D) Total number of decayed and unsound roots	40
NUMRF (D) Number of filled roots (no other decay)	40
NUMRU (D) Number of unscored roots	40
NUMRFILL (D) Total number of filled roots	40
NUMREWFD (D) Total number of exposed, worn, filled and decayed roots	40
NUMRURS (D) Number of unrestorable roots	40
NUMRPD (D) Number of roots with active decay	40
NUMRAD (D) Number of roots with arrested decay	40
NUMRFD (D) Number of filled roots with recurrent decay	40
NUMRFRC (D) Number of roots with fractured restorations	40
NUMRD09 (D) Number of teeth with root decay	41
NUMRE09 (D) Number of vulnerable (exposed) roots	41
NUMRFD09 (D) Number of decayed or filled roots	41
NUMRF09 (D) Number of filled roots	41
NUMSSUR (D) Number of sound surfaces, including roots	41
NUMSRESR (D) Number of restored surfaces, including roots	41
NUMSRECR (D) Number of restored surfaces with recurrent decay, including roots	41
NUMSFRCR (D) Number of restored surfaces with fractured restorations, including roots	41
NUMSUTR (D) Number of sound untreated teeth (including roots)	41
NUMRESTR (D) Number of restored teeth, including crowns and roots	42
NUMFLLT (D) Number of teeth with fillings in crown or root	42
HASFLLT (D) Has teeth with fillings in crown or root	43
HASRESTR (D) Has restored teeth incl roots	43
NUMFSUR (D) Number of filled surfaces, crowns and roots	44
NUMTRECR (D) Number of restored teeth with secondary decay, including crowns and roots	44
HASTRECR (D) Has restored teeth with secondary decay, including crowns and roots	44
NUMTFRSR (D) Number of restored teeth decayed or unsound, incl roots.	44
HASTFRSR (D) Has restored teeth decayed or unsound, incl roots.	45
GRPREWFD (D) Total number of exposed, worn, filled and decayed roots - 4 cat	45
GRPRDEC (D) Number of decayed and unsound roots - 4 cat	45
GRPSUTR (D) Number of sound and untreated teeth including roots.	45
HASSUTR (D) Has 18 or more sound untreated teeth (incl. roots)	45
HASRPD (D) Has any roots with active decay	46
HASRDEC (D) Has any decayed or unsound roots	47
HASRD09 (D) Has any teeth with root decay	47
HASRE09 (D) Has any teeth with vulnerable (exposed) roots	47
HASRFD09 (D) Has any teeth with decayed or filled roots	47
HASRF09 (D) Has any teeth with filled roots	47
TOOTH WEAR	47
WEARBUR1 etc (D) Buccal tooth wear - upper right 1 etc	47
WEARIUR1 etc (D) Incisal tooth wear - upper right 1 etc	47
WEARLUR1 etc (D) Lingual/palatal tooth wear - upper right 1 etc	47
WEARUR1 etc (D) Tooth wear - upper right 1 etc	48
WEARLR1 etc (D) Tooth wear - lower right 1 etc	48
NUMWEAR (D) Number of anterior teeth with wear	49
NUMMOD (D) Number of anterior teeth with moderate wear	49
NUMSEV (D) Number of anterior teeth with severe wear	49
NUMMDSV (D) Number of anterior teeth with moderate or severe wear	49
NUMFRACT (D) Number of fractured anterior teeth	49
HASWEAR (D) Any anterior teeth with wear	50
HASMOD (D) Any anterior teeth with moderate wear	50
HASSEV (D) Any anterior teeth with severe wear	50
HASMDSV (D) Any anterior teeth with moderate or severe wear	50
HASFRACT (D) Any fractured anterior teeth	50
BASIC EROSIVE WEAR EXAMINATION (BEWE)	50
BTEEUR (D) BEWE two or more teeth - UR sextant	50
BTEEUC (D) BEWE two or more teeth - UC sextant	50
BTEEUL (D) BEWE two or more teeth - UL sextant	50
BTEELL (D) BEWE two or more teeth - LL sextant	50
BTEELC (D) BEWE two or more teeth - LC sextant	50
BTEELR (D) BEWE two or more teeth - LR sextant	50
BBOLUR (D) Highest BEWE score - UR sextant	50
BBOLUC (D) Highest BEWE score - UC sextant	50
BBOLUL (D) Highest BEWE score - UL sextant	50
BBOLLL (D) Highest BEWE score - LL sextant	50

BBOLLC (D) Highest BEWE score - LC sextant	50
BBOLLR (D) Highest BEWE score - LR sextant	50
OCCUSAL CONDITION	51
CONTRPM (D) Contact - upper and lower right premolars	51
CONTRML (D) Contact - upper and lower right molars	51
CONTLPM (D) Contact - upper and lower left premolars	51
CONTMLL (D) Contact - upper and lower left molars	51
SPACES	51
SPACEUR1 etc (D) Spaces - upper right 1 etc	51
NUMUSP (D) Number of upper anterior/premolar spaces	52
NUMUSPU (D) Number of unfilled upper spaces	52
NUMUSPD (D) Number of upper spaces filled with denture	52
NUMUSPB (D) Number of upper spaces filled with bridge pontic	52
NUMUSPI (D) Number of upper spaces filled with implant	52
NUMLSP (D) Number of lower anterior/premolar spaces	52
NUMLSPU (D) Number of unfilled lower spaces	52
NUMLSPD (D) Number of lower spaces filled with denture	52
NUMLSPB (D) Number of lower spaces filled with bridge pontic	52
NUMLSPI (D) Number of lower spaces filled with implant	52
HASUSP (D) Any upper anterior/premolar spaces	53
HASUSPU (D) Any unfilled upper spaces	53
HASUSPD (D) Any upper spaces filled with denture	53
HASUSPB (D) Any upper spaces filled with bridge pontic	53
HASUSPI (D) Any upper spaces filled with implant	53
HASLSP (D) Any lower anterior/premolar spaces.	53
HASLSPU (D) Any unfilled lower spaces.	53
HASLSPD (D) Any lower spaces filled with denture.	53
HASLSPB (D) Any lower spaces filled with bridge pontic.	53
HASLSPI (D) Any lower spaces filled with implant.	53
DENTURES	54
WEARDENT (D) Wears denture	54
DENTWEAR (D) Summary of dental wear	54
DENTSUMM (D) Combinations of dentures	54
PUFA INDEX AND OTHER URGENT CONDITIONS	54
PAINDV (D) Experiencing pain related to teeth	54
PHYSPNG (D) OHIP physical pain grouped - 3 cats	55
PUFA (D) PUFA score	55
HASPUFA (D) Has one or more PUFA symptoms	55
SEPS (D) Whether have any fistula/abcess	55
URGENT (D) Urgent conditions score	55
PULPYN (D) Has pulp yes/no	55
ULCYN (D) Has ulc yes/no	55
FISTYN (D) Has fist yes/no	55
ABSCYN (D) Has absc yes/no	55
PERIODONTAL CONDITION, INCLUDING BPE	57
TEEUR etc (D) Two or more teeth - UR sextant etc	57
PKETUR etc (D) Pocket depth - UR sextant etc	57
LOAUR etc (D) Loss of attachment - UR sextant etc	57
CALCUR etc (D) Calculus - UR sextant etc	58
BLEEDUR (D) Bleeding - UR sextant	58
BPEUR (D) BPE - UR sextant	58
ULSEXA etc (D) UL sextant etc - No pocketing or LOA above 3.5 mm	59
NUMSEXA (D) Number of sextants with no pocketing or LOA above 3.5mm	59
ULSEXB etc (D) UL sextant etc - LOA 4 mm or greater etc	60
NUMSEXB (D) Number of sextants with LOA of 4mm or greater	60
ULSEXC etc (D) UL sextant etc - LOA 6 mm or greater etc	60
NUMSEXC (D) Number of sextants with LOA of 6mm or greater	61
ULSEXD etc (D) UL sextant etc - pocketing 4 mm or greater	61
NUMSEXD (D) Number of sextants with pocketing of 4mm or greater	62
ULSEXE etc (D) UL sextant etc - pocketing 6mm or greater	62
NUMSEXE (D) Number of sextants with pocketing of 6mm or greater	63
NUMSEXF (D) Number of missing sextants	63
NUMSEX (D) Number of sextants	63
NUMSCALC (D) Number of sextants with calculus present	63
NUMSBLD (D) Number of sextants with bleeding present	63
CALCULUS (D) Calculus present	63
HASCALC (D) Calculus present	64
HASBLD (D) Any bleeding	64
PCDALL (D) Pocket depth - whole mouth	65
PCD4 (D) Any pocketing 4mm or deeper	65
PCD6 (D) Any pocketing 6mm or deeper	65
PCD6DV (D) pocket depth 6mm or more, no unscored sextants	65
PCD9 (D) Any pocketing 9mm or deeper	66

LOAALL (D) Loss of attachment - whole mouth	66
LOATT4 (D) Any LOA 4mm or deeper	66
LOATT6 (D) Any LOA 6mm or deeper	67
LOATT9 (D) Any LOA 9mm or deeper	67
GUMHLTHY (D) Healthy gums - no pocketing or LoA	67
GUMHLTG4 (D) Healthy gums - four categories	67
DHEALTH (D) 21+ teeth & 18+ SUT (incl. roots) & perio health	68
DHEALTH2 (D) 21+ teeth & 18+ SUT (incl. roots) & perio health and no active decay	68
DHLTHG4 (D) 21+ teeth & 18+ SUT (incl. roots) & perio health in 4 cats	68
DHLTHG3 (D) 21+ teeth & 18+ SUT (incl. roots) & perio health in 3 cats	68
COMPLEXITY SCALE	68
COMPLEX1 (D) Restored surfaces in top quintile.	68
COMPLEX2 (D) Three or more crowns.	69
COMPLEX 3 (D) Any denture, bridge, implant.	69
COMPLEX 4 (D) Any active decay.	69
COMPLEX 5 (D) Active decay: top quintile of those with decay.	69
COMPLEX 6 (D) Pocketing 6mm+ or LoA 9mm+.	69
COMPLEX 7 (D) PUFA score or unrestorable tooth.	70
COMPLEX 8 (D) OHIP: fairly or very often.	70
TOTCOMP (D) Complexity score.	70
TOTCOMPG (D) Complexity score with 5+ grouped.	70
INTERVIEW	71
DENTAL STATUS	71
DENTAL HYGIENE	72
ClnTthG3 Number of times brush teeth per day - grouped	72
ClnTthG4 Number of times brush teeth per day - grouped	72
Fluorid1 (D) grouped fluoride levels	72
SMOKING	73
DVSmoke Smoking status	73
DIET	74
EvrDnAd1 Ever given advice about diet - all people	74
Highsug5 High sugar intake	74
DENTIST ATTENDANCE PATTERNS	75
HowLongG Time since last dental visit	75
NumTrouG Number of times visited dentist with trouble - grouped	75
DVChkUp Reason for last visit - 3 categories	75
FreqDenG Frequency of dentist visits - grouped	75
FreqDnG1 Frequency of dentist visits - grouped 1	76
HowLngG2 Last visit to dentist - grouped 2	76
DENTAL SERVICES	77
SerTypGrp Service type - grouped	77
SerTypNHS Paid for and free NHS combined	77
DVSerTyp Service type for Wales/NI tables	77
Cost	77
Topcoded at £1001 for non-disclosure purposes	77
Costings Cost of treatment- grouped	78
DenPlanD Insurance composite	78
DENTAL TREATMENT RECEIVED	79
Dtotimp If ever had an implant - upper or lower	79
DVisits1 More or less than 1 visit	79
DenFitd Dentures fitted including not applicables	79
DenRepd Dentures repaired including not applicables	79
ValMonD Grouped value for money categories	79
EvrDnAdd Ever had diet advice - Includes no and not applicables	80
SELF-ASSESSED HEALTH	81
GenHlthg Self-reported general health - grouped	81
DenHlthg Self-reported dental health - grouped	81
OHIP - ORAL HEALTH-RELATED QUALITY OF LIFE	82
FunctLim Functional limitation -words taste	82
PhysPain Physical pain - aching foods	82
PsychDisc Psychological discomfort - self tense	82
PhysDisc Physical discomfort -diet meals	82
PsychDisb Psychological disability - relax embarrass	82
SocDisb Social disability - irritabl jobs	82
Handicap Handicap - less functin	82
TotOHIP Total sum across 14 items	84
NumOHIP Number of problems on OHIP items	84

NumOHIPG	Number of problems on OHIP items grouped	84
WordDiff	(D) OHIP Words - any difficulty reported	85
Achingg	(D) Aching (grouped answers)	85
OIDP – ORAL IMPACTS ON DAILY PERFORMANCES		86
IpEatG	Prevalence of difficulty eating	86
IpSpkG	Prevalence of difficulty speaking	86
IpClnG	Prevalence of difficulty cleaning teeth/dentures	86
IpOutG	Prevalence of difficulty going out	86
IpRlxG	Prevalence of difficulty relaxing	86
IpShwG	Prevalence of difficulty showing teeth	86
IpWrkG	Prevalence of difficulty working	86
IpEmtG	Prevalence of problems with emotional stability	86
IpConG	Prevalence of difficulty with contact with others	86
OIDPAII	Overall OIDP prevalence	86
TotOIDP	OIDP severity rating Sum of scores of 9 items	87
HiOIDP	Highest reported score on any OIDP problem	87
NumOIDP	Extent of OIDP score	87
ACCESS TO CARE		88
TrdNHSGp	(D) Tried to make NHS in last 3 years - grouped No and Never	88
ScsNHStr	Successfully made NHS appointment in last 3 years – excluding never tried	88
EFFECT OF COST		88
YesCost	Combined answer to costtyp and costdly	88
DENTAL ANXIETY		89
MdTreatG	How would feel if went to dentist tomorrow - grouped	89
MdWtRmG	How would feel if sitting in waiting room - grouped	89
MdThDrIG	How would feel if about to have tooth drilled - grouped	89
MdScleG	How would feel if about to have scale and polish - grouped	89
MdInjtG	How would feel if about to have local anaesthetic injection - grouped	89
TotMDAS	Total score on MDAS	89
TotMDASG	Total score on MDAS - grouped	89
DTotMDAS	Phobic and nonphobic	89
RELATIONSHIP WITH DENTIST		91
TimDisYN	(D) Was given enough time to discuss problems with the dentist - yes/no only	91
InvDecYN	(D) Was involved as much as would like in decisions - yes/no only	91
AndUndYN	(D) Was given answers that could be understood - yes/no only	91
NumDenR	Number of negative responses to dentist relationship	91
NumDenRG	Number of negative responses to dentist relationship grouped	91
RATING OF DENTAL PRACTICE		92
rteraptkd	rating dental practice in terms of length of wait for a routine appointment including don't knows	92
rteuaptkd	rating dental practice in terms of length of wait for an urgent appointment including don't knows	92
rtewaptkd	rating dental practice in terms of length of wait for a evening or weekend appointment including don't knows	92
rteacsdk	rating dental practice in terms of transport facilities and access including don't knows	92
rteconvdK	rating dental practice in terms of convenience of location including don't knows	92
rtechrGdk	rating dental practice in terms of explanation of NHS charges including don't knows	92
rtdenrep	rating dental practice in terms of reputation of dentists including don't knows	92
rtcredk	rating dental practice in terms of standard and quality of care including don't knows	92

Classification

Age

DVAGE Age

Topcoded at 85 for non-disclosure purposes

```
SPSS Syntax
RECODE
  DVAge (85 thru Highest=85) .
EXECUTE .
```

Ageband1 (D) Grouped age - 6 cats

- 1 16 to 24
- 2 25 to 44
- 3 45 to 54
- 4 55 to 64
- 5 65 to 74
- 6 75 and over

Ageband2 (D) Grouped age - 5 cats

1. 16 to 24
2. 25 to 44
3. 45 to 64
4. 55 to 64
5. 65 to 84
6. 85 and over

Ageband3 (D) Grouped age - 3 cats

1. 16 to 34
2. 35 to 54
3. 55 and over

Ageband4 (D) Grouped age - 7 cats

1. 16 to 24
2. 25 to 34
3. 35 to 44
4. 45 to 54
5. 55 to 64
6. 65 to 74
7. 75 and over

Ageband6 (D) Grouped age - 4 cats, lowest 55+

1. 55 to 64
2. 65 to 74
3. 75 to 84
4. 85 and over

Ageband7 (D) Grouped age - 5 cats, highest 55+

1. 16 to 24
2. 25 to 34
3. 35 to 44
4. 45 to 54

5. 55 and over

AGEBAND5 (D) Grouped age - 8 cats

1. 16 to 24
2. 25 to 34
3. 35 to 44
4. 45 to 54
5. 55 to 64
6. 65 to 74
7. 75 to 84
8. 85 and over

```
SPSS syntax - EXAMPLE
recode dvage (16 thru 24=1) (25 thru 44=2) (45 thru 54=3) (55 thru 64=4) (65 thru 74=5) (75 thru 110=6) into
ageband5.
var lab ageband5 '(D) Grouped age - 8 cats'.
val labs ageband5
1 '16 to 24'
2 '25 to 34'
3 '35 to 44'
4 '45 to 54'
5 '55 to 64'
6 '65 to 74'
7 '75 to 84'
8 '85 and over'.
```

NS SEC Socio-economic classification

HRPsec8 NS-SEC HRP - Eight-class version

- 1.1 Large employers and higher managerial occupations
- 1.2 Higher professional occupations
- 2.0 Lower managerial and professional occupations
- 3.0 Intermediate occupations
- 4.0 Small employers and own account workers
- 5.0 Lower supervisory & technical occupations
- 6.0 Semi-routine Occupations
- 7.0 Routine Occupations
- 8.0 Never worked and long term unemployed
- 97.0 Not classified

HRPsec5 NS-SEC HRP - Five-class version

- 1.0 Managerial and professional occupations
- 2.0 Intermediate occupations
- 3.0 Small employers and own account workers
- 4.0 Lower supervisory & technical occupations
- 5.0 Semi-routine and routine occupations
- 96.0 Never worked and long term unemployed
- 97.0 Not classified

HRPsec3 NS-SEC HRP - Three-class version

- 1.0 Managerial and professional occupations
- 2.0 Intermediate occupations
- 3.0 Routine and manual occupations
- 96.0 Never worked and long term unemployed
- 97.0 Not classified

SPSS syntax

```
*****HRP NSSEC *****.

****to create a household level file with the nssec of the hrp****

select if (DVhrpnum=pnum).

compute hrpsec8=NSSEC8.
compute hrpsec5=NSSEC5.
compute hrpsec3=NSSEC3.

variable labels hrpsec8 'NS-SEC HRP - Eight-class version'/
                 hrpsec5 'NS-SEC HRP - Five-class version'/
                 hrpsec3 'NS-SEC HRP - Three-class version' .
value labels hrpsec8 -9.0 'no answer/refused'
              -8.0 'dont know'
              -7.0 'refused/not obtained'
              -6.0 'schedule not obtained'
              -2.0 'schedule not applicable'
              -1.0 'item not applicable'
              1.1 'Large employers and higher managerial occupations'
              1.2 'Higher professional occupations'
              2.0 'Lower managerial and professional occupations'
              3.0 'Intermediate occupations'
              4.0 'Small employers and own account workers'
              5.0 'Lower supervisory & technical occupations'
              6.0 'Semi-routine Occupations'
              7.0 'Routine Occupations'
              8.0 'Never worked and long term unemployed'
              97.0 'Not classified'/
hrpsec5 -9.0 'no answer/refused'
        -8.0 'dont know'
        -7.0 'refused/not obtained'
        -6.0 'schedule not obtained'
        -2.0 'schedule not applicable'
        -1.0 'item not applicable'
        1.0 'Managerial and professional occupations'
        2.0 'Intermediate occupations'
```

```

3.0 'Small employers and own account workers'
4.0 'Lower supervisory & technical occupations'
5.0 'Semi-routine and routine occupations'
96.0 'Never worked and long term unemployed'
97.0 'Not classified'/
hrpsec3 -9.0 'no answer/refused'
-8.0 'dont know'
-7.0 'refused/not obtained'
-6.0 'schedule not obtained'
-2.0 'schedule not applicable'
-1.0 'item not applicable'
1.0 'Managerial and professional occupations'
2.0 'Intermediate occupations'
3.0 'Routine and manual occupations'
96.0 'Never worked and long term unemployed'
97.0 'Not classified'.

execute.
fre vars = NSSEC8 NSSEC5 NSSEC3.

***** and save resulting household file as adhsechhld.sav*****.

SAVE OUTFILE='D:\deborahs workingfiles\My Documents\adh\adhsechhld.sav'
/keep=Area Address HHold hrpsec8 hrpsec5 hrpsec3
/COMPRESSED.

*****match back to the individual file using the household file as a lookup table*****

GeT FILE='D:\deborahs workingfiles\My Documents\adh\ADHS Final indiv v2 for checking.sav'.

MATCH FILES /TABLE= 'D:\deborahs workingfiles\My Documents\adh\adhsechhld.sav'
/FILE='D:\deborahs workingfiles\My Documents\adh\ADHS Final indiv v2 for
checking.sav'
/ by = Area Address HHold .
EXECUTE.

FREQ VARS nssec8 hrpsec8 nssec5 hrpsec5 nssec3 hrpsec3.

***** 892 cases where HRP not interviewed *****.

recode hrpsec8 hrpsec5 hrpsec3 (sysmis=-1) (else=copy).

list variables Indser HHser pnum dvhrpnum nssec8 hrpsec8 nssec5 hrpsec5 nssec3 hrpsec3
/cases from 1 to 50.

temp.
select if sysmis (hrpsec8).
list variables Indser HHser pnum dvhrpnum nssec8 hrpsec8.

*****to check number of households *****.

aggregate outfile= 'D:\deborahs workingfiles\My Documents\adh\adhhhld.sav'
/break = hhser
/hsex=nu(sex).

***** number = 7227 *****.

get file= 'D:\deborahs workingfiles\My Documents\adh\adhhhld.sav'.
fre hsex.

***** multiply number of each sex by freq in excel = 11372 so correct *****.

```

Income

TelDV – DV for weekly income amount

HhldDV – DV for weekly income amount

Topcoded at 10 times average salary for non-disclosure purposes

```
RECODE
  TelDv   (3774 thru Highest=3774)   .
EXECUTE .
RECODE
  HhldDv  (7528 thru Highest=7528)   .
EXECUTE .
```

Ethnic group

Ethnicg – grouped ethnicity

1. White British/Other White
2. Mixed race
3. Asian - Indian
4. Asian - Pakistani & Bangladeshi
5. Asian - Other'
6. Black Caribbean
7. Black African
8. Other Black
9. Other

```
recode ethnic_mer (1,2=1) (3,4,5,6=2) (7=3) (8,9=4) (10=5) (11=6) (12=7) (13=8) (14,15=9) (else=copy) into
ethnicg.
missing values ethnicg (-8,-9).
variable labels ethnicg 'grouped ethnicity'.
value labels ethnicg 1 'White British/Other White' 2 'Mixed race' 3 'Asian - Indian' 4 'Asian -
Pakistani & Bangladeshi'
5 'Asian - Other' 6 'Black Caribbean' 7 'Black African' 8 'Other Black' 9 'Other'.

fre vars = ethnic_mer ethnicg.
crosstabs ethnic_mer by ethnicg.
```

Index of Multiple Deprivation

IMDE_D - Deciles - England

2010 rank scores from the spreadsheet downloaded from the DCLG website, grouped into deciles

IMDW_D - Deciles - Wales

2008 (revised) rank scores from the spreadsheet downloaded from the Stats Wales website, grouped into deciles

IMDNI_D - Deciles - Northern Ireland

2010 rank scores from the spreadsheet downloaded from the NISRA website, grouped into deciles

SPSS Syntax

```
* Stage 1 imports the consolidated Domain sheet of Index of Multiple Deprivation for England added to
* the full spreadsheet downloaded from the DCLG website into SPSS and saves as an SPSS datafile;
* computes ranked decile variables and re-saves the file.
*

GET DATA /TYPE=XLS
  /FILE='V:\DOCUMENTATION\Outputs\IMD\IMD 2010 England.xls'
  /SHEET=name 'IMD 2010'
  /CELLRANGE=full
  /READNAMES=on.

FREQUENCIES
  VARIABLES=RANKOFIMDScorewhereIismostdeprived /FORMAT=NOTABLE
  /NTILES= 10
  /PERCENTILES= 10
  /ORDER= ANALYSIS .

RECODE RANKOFIMDScorewhereIismostdeprived
  (1 thru 3248=1)
  (3249 thru 6496=2)
  (6497 thru 9745=3)
  (9746 thru 12993=4)
  (12994 thru 16241=5)
  (16242 thru 19489=6)
  (19490 thru 22737=7)
  (22738 thru 25986=8)
  (25987 thru 29234=9)
  (29235 thru 32482=10) INTO IMDE_D.
VARIABLE LABELS IMDE_D 'IMD Decile (England)'.
EXECUTE.

SAVE OUTFILE='V:\DOCUMENTATION\Outputs\IMD\IMD_England.sav' /COMPRESSED.

*
* Stage 2 imports the consolidated Domain sheet of Index of Multiple Deprivation for Wales added to
* the full spreadsheet downloaded from the Stats Wales website into SPSS and saves as an SPSS datafile;
* computes ranked decile variables and re-saves the file.
*

GET DATA /TYPE=XLS
  /FILE='V:\DOCUMENTATION\Outputs\IMD\IMD 2008 Wales revised2.xls'
  /SHEET=name 'IMD 2008 Wales revised'
  /CELLRANGE=full
  /READNAMES=on.

FREQUENCIES
  VARIABLES= OverallIndexrank /FORMAT=NOTABLE
  /NTILES= 10
  /PERCENTILES= 10
  /ORDER= ANALYSIS .

RECODE OverallIndexrank
  (1 thru 190=1)
  (191 thru 380=2)
  (381 thru 569=3)
  (570 thru 759=4)
  (760 thru 948=5)
  (949 thru 1138=6)
```

```

(1139 thru 1327=7)
(1328 thru 1517=8)
(1518 thru 1706=9)
(1707 thru 1896=10) INTO IMDW D.
VARIABLE LABELS IMDW D 'IMD Decile (Wales)'.
EXECUTE.

SAVE OUTFILE='V:\DOCUMENTATION\Outputs\IMD\IMD_Wales.sav' /COMPRESSED .

*
* Stage 3 imports the consolidated Domain sheet of Index of Multiple Deprivation for NI added to
* the full spreadsheet downloaded from the NISRA website into SPSS and saves as an SPSS datafile;
* computes ranked decile variables and re-saves the file.
*

GET DATA /TYPE=XLS
  /FILE='V:\DOCUMENTATION\Outputs\IMD\NIMDM_2010_Results_OA.xls'
  /SHEET=name 'MDM 2010'
  /CELLRANGE=full
  /READNAMES=on.

FREQUENCIES
  VARIABLES=RankofMultipleDeprivationMeasureScorewhereIsmostdepr /FORMAT=NOTABLE
  /NTILES= 10
  /ORDER= ANALYSIS .

RECODE RankofMultipleDeprivationMeasureScorewhereIsmostdepr
  (1 thru 502=1)
  (503 thru 1005=2)
  (1006 thru 1507=3)
  (1508 thru 2009=4)
  (2010 thru 2511=5)
  (2512 thru 3014=6)
  (3015 thru 3516=7)
  (3517 thru 4018=8)
  (4019 thru 4520=9)
  (4521 thru 5022=10) INTO IMDNI D.
VARIABLE LABELS IMDNI D 'IMD Decile (Northern Ireland)'.
EXECUTE.

SAVE OUTFILE='V:\DOCUMENTATION\Outputs\IMD\IMD_NI.sav' /COMPRESSED .

*
* Stage 4 opens the ADH file, sorts by SOA2, matches the files of deprivation data for England and Wales,
* re-sorts
* the sample file by OACode, matches the file of deprivation data for NI and re-saves the file.
*

GET
  FILE='V:\Archiving\ADH2009\ADH2009_Dataset for Archive_v2.sav'.
SORT CASES BY SOA1 (A).

MATCH FILES /FILE=*
  /TABLE='V:\DOCUMENTATION\Outputs\IMD\IMD_England.sav'
  /BY SOA1.
EXECUTE.

SAVE OUTFILE='V:\Archiving\ADH2009\ADH2009_Dataset for Archive_v2.sav'.

MATCH FILES /FILE=*
  /TABLE='V:\DOCUMENTATION\Outputs\IMD\IMD_Wales.sav'
  /BY SOA1.
EXECUTE.

SAVE OUTFILE='V:\Archiving\ADH2009\ADH2009_Dataset for Archive v2.sav'.

SORT CASES BY OACode (A).
MATCH FILES /FILE=*
  /TABLE='V:\DOCUMENTATION\Outputs\IMD\IMD_NI.sav'
  /BY OACode.
EXECUTE.

SAVE OUTFILE='V:\Archiving\ADH2009\ADH2009 full dataset Northern Ireland.sav'/keep= Area Address HHold
Indser Quota.

```

Examination

Examination variables

ARCHES (D) Arches with natural teeth

- 1 Teeth in both arches
- 2 Teeth in upper arch only
- 3 Teeth in lower arch only

```
SPSS syntax
if any(bothar,-1,1) arches = bothar.
if bothar = 2 & uppar = 1 arches = 2.
if bothar = 2 & botar = 1 arches = 3.

if bothar=2 & uppar=2 & botar=2 arches=-1.

var lab arches '(D) Arches with natural teeth'.
val lab arches
  1 'Teeth in both arches'
  2 'Teeth in upper arch only'
  3 'Teeth in lower arch only'.
```

EXOUT (D) Examination outcome

- 1 Dentate - exam completed
- 2 Dentate - no examination
- 3 No natural teeth

```
SPSS syntax
recode iout (11 = 1) (12 = 3) (21 22 23 24 25 30 = 2) into exout.

var lab exout '(D) Examination outcome'.

val lab exout
  1 'Dentate - exam completed'
  2 'Dentate - no examination'
  3 'No natural teeth'.
```

Coronal surfaces (tooth variables)

For each tooth variable the last three characters of the variable names refer to position of tooth:

UL - Upper left, UR - Upper right, LL - Lower left, LR - Lower right

1, 2 - incisors, 3 - canine, 4, 5 - premolars, 6, 7, 8 - molars

PLUR1 etc (D) Plaque - upper right 1 etc

- 0 M - missing (not bridge pontic)
- 1 C - clean
- 2 P - plaque
- 6 T - implant
- 7 A - adhesive bridge pontic
- 8 B - conventional bridge pontic or implant pontic

```
SPSS syntax
do repeat
  xxx = pl p12 p13 p14 p15 p16 p17 p18 p19 pl10 pl11 pl12 pl13 pl14 pl15 pl16
  /yyy = PLUR8 PLUR7 PLUR6 PLUR5 PLUR4 PLUR3 PLUR2 PLUR1 PLUL1 PLUL2 PLUL3 PLUL4 PLUL5 PLUL6 PLUL7 PLUL8.
  recode xxx (1=2) (2=1) (3=0) (4=7) (5=8) (else=copy) into yyy.
  if bothar=2 & botar=1 yyy=0.
end repeat.

val lab PLUR8 to PLUL8
  0 'M - missing (not bridge pontic)'
  1 'C - clean'
  2 'P - plaque'
  6 'T - implant'
  7 'A - adhesive bridge pontic'
  8 'B - conventional bridge pontic or implant pontic'.

var lab PLUR8 '(D) Plaque - upper right 8'.
var lab PLUR7 '(D) Plaque - upper right 7'.
var lab PLUR6 '(D) Plaque - upper right 6'.
var lab PLUR5 '(D) Plaque - upper right 5'.
var lab PLUR4 '(D) Plaque - upper right 4'.
var lab PLUR3 '(D) Plaque - upper right 3'.
var lab PLUR2 '(D) Plaque - upper right 2'.
var lab PLUR1 '(D) Plaque - upper right 1'.
var lab PLUL1 '(D) Plaque - upper left 1'.
var lab PLUL2 '(D) Plaque - upper left 2'.
var lab PLUL3 '(D) Plaque - upper left 3'.
var lab PLUL4 '(D) Plaque - upper left 4'.
var lab PLUL5 '(D) Plaque - upper left 5'.
var lab PLUL6 '(D) Plaque - upper left 6'.
var lab PLUL7 '(D) Plaque - upper left 7'.
var lab PLUL8 '(D) Plaque - upper left 8'.
```

PLLL1 etc (D) Plaque - lower left 1 etc

- 0 M - missing (not bridge pontic)
- 1 C - clean
- 2 P - plaque
- 6 T - implant
- 7 A - adhesive bridge pontic
- 8 B - conventional bridge pontic or implant pontic

```
SPSS syntax
do repeat
  xxx = pl17 pl18 pl19 pl20 pl21 pl22 pl23 pl24 pl25 pl26 pl27 pl28 pl29 pl30 pl31 pl32
  /yyy = PLLL8 PLLL7 PLLL6 PLLL5 PLLL4 PLLL3 PLLL2 PLLL1 PLLR1 PLLR2 PLLR3 PLLR4 PLLR5 PLLR6 PLLR7 PLLR8.
  recode xxx (1=2) (2=1) (3=0) (4=7) (5=8) (else=copy) into yyy.
  if bothar=2 & uppar=1 yyy=0.
end repeat.

val lab PLLL8 to PLLR8
  0 'M - missing (not bridge pontic)'
  1 'C - clean'
  2 'P - plaque'
  6 'T - implant'
  7 'A - adhesive bridge pontic'
  8 'B - conventional bridge pontic or implant pontic'.

var lab PLLL8 '(D) Plaque - lower left 8'.
var lab PLLL7 '(D) Plaque - lower left 7'.
var lab PLLL6 '(D) Plaque - lower left 6'.
var lab PLLL5 '(D) Plaque - lower left 5'.
var lab PLLL4 '(D) Plaque - lower left 4'.
```

```

var lab PLLL3 '(D) Plaque - lower left 3'.
var lab PLLL2 '(D) Plaque - lower left 2'.
var lab PLLL1 '(D) Plaque - lower left 1'.
var lab PLLR1 '(D) Plaque - lower right 1'.
var lab PLLR2 '(D) Plaque - lower right 2'.
var lab PLLR3 '(D) Plaque - lower right 3'.
var lab PLLR4 '(D) Plaque - lower right 4'.
var lab PLLR5 '(D) Plaque - lower right 5'.
var lab PLLR6 '(D) Plaque - lower right 6'.
var lab PLLR7 '(D) Plaque - lower right 7'.
var lab PLLR8 '(D) Plaque - lower right 8'.

```

PLUR8p to PLLL8p condition of individual teeth - recoded

- 0 missing
- 1 clean
- 2 plaque
- 3 implant or bridge

SPSS syntax

```

recode PLUR8 PLUR7 PLUR6 PLUR5 PLUR4 PLUR3 PLUR2 PLUR1
      PLUL1 PLUL2 PLUL3 PLUL4 PLUL5 PLUL6 PLUL7 PLUL8
      PLLR8 PLLR7 PLLR6 PLLR5 PLLR4 PLLR3 PLLR2 PLLR1
      PLLL1 PLLL2 PLLL3 PLLL4 PLLL5 PLLL6 PLLL7 PLLL8
  (0=0) (1=1) (2=2) (6,7,8=3) (else=copy) into
      PLUR8p PLUR7p PLUR6p PLUR5p PLUR4p PLUR3p PLUR2p PLUR1p
      PLUL1p PLUL2p PLUL3p PLUL4p PLUL5p PLUL6p PLUL7p PLUL8p
      PLLR8p PLLR7p PLLR6p PLLR5p PLLR4p PLLR3p PLLR2p PLLR1p
      PLLL1p PLLL2p PLLL3p PLLL4p PLLL5p PLLL6p PLLL7p PLLL8p.

missing values PLUR8p PLUR7p PLUR6p PLUR5p PLUR4p PLUR3p PLUR2p PLUR1p
      PLUL1p PLUL2p PLUL3p PLUL4p PLUL5p PLUL6p PLUL7p PLUL8p
      PLLR8p PLLR7p PLLR6p PLLR5p PLLR4p PLLR3p PLLR2p PLLR1p
      PLLL1p PLLL2p PLLL3p PLLL4p PLLL5p PLLL6p PLLL7p PLLL8p (-9 thru -1).

value labels PLUR8p PLUR7p PLUR6p PLUR5p PLUR4p PLUR3p PLUR2p PLUR1p
      PLUL1p PLUL2p PLUL3p PLUL4p PLUL5p PLUL6p PLUL7p PLUL8p
      PLLR8p PLLR7p PLLR6p PLLR5p PLLR4p PLLR3p PLLR2p PLLR1p
      PLLL1p PLLL2p PLLL3p PLLL4p PLLL5p PLLL6p PLLL7p PLLL8p
      0 'missing' 1 'clean' 2 'plaque' 3 'implant or bridge'.

```

DUR1 etc (D) Distal surface code - upper right 1 etc
 OIUR1 etc (D) Occlusal/Incisal surface code - upper right 1 etc
 MUR1 etc (D) Mesial surface code - upper right 1 etc
 BUR1 etc (D) Buccal surface code - upper right 1 etc
 LUR1 etc (D) Lingual/Palatal surface code - upper right 1 etc

0 Sound
 1 Visual caries
 2 Cavitated caries
 3 Unrestorable
 4 Hard arrested decay
 40 Sealant - sound
 41 Sealant with visual caries
 42 Sealant with cavitated caries
 43 Sealant - failed restoration
 44 Sealant with hard arrested decay
 50 Amalgam filling - sound
 51 Amalgam filling with visual caries
 52 Amalgam filling with cavitated caries
 53 Amalgam filling - failed restoration
 54 Amalgam filling with hard arrested decay
 60 Other restoration - sound
 61 Other restoration with visual caries
 62 Other restoration with cavitated caries
 63 Other restoration - failed restoration
 64 Other restoration with hard arrested decay
 70 Shim, veneer - sound
 71 Shim, veneer with visual caries
 72 Shim, veneer with cavitated caries
 73 Shim, veneer - failed restoration
 74 Shim, veneer with hard arrested decay
 80 Artificial crown - sound
 81 Artificial crown with visual caries
 82 Artificial crown with cavitated caries
 83 Artificial crown - failed restoration
 84 Artificial crown with hard arrested decay
 90 Missing
 96 Missing - replaced by implant
 97 Missing - replaced by adhesive bridge
 98 Missing - replaced by conventional bridge
 99 Unscored

```
SPSS syntax

*DISTAL SURFACE.

do repeat xxx = DUR8 DUR7 DUR6 DUR5 DUR4 DUR3 DUR2 DUR1 DUL1 DUL2 DUL3 DUL4 DUL5 DUL6 DUL7 DUL8
  /d = d d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16
  /dm = dm dm2 dm3 dm4 dm5 dm6 dm7 dm8 dm9 dm10 dm11 dm12 dm13 dm14 dm15 dm16
  /pl = pl pl2 pl3 pl4 pl5 pl6 pl7 pl8 pl9 pl10 pl11 pl12 pl13 pl14 pl15 pl16.

COMPUTE      xxx      =-999.
IF d =0      xxx=0.
IF d =19     xxx=0.
IF d =4      xxx=1.
IF d =5      xxx  =2.
IF d =6      xxx  =3.
IF d =8      xxx  =4.
IF d =12 & dm=0 xxx =40.
IF d =12 & dm=4 xxx =41.
IF d =12 & dm=5 xxx =42.
IF d =12 & dm=6 xxx =3.
IF d =12 & dm=7 xxx =43.
IF d =12 & dm=8 xxx =44.
IF d =1 & dm=0 xxx =50.
IF d =1 & dm=4 xxx =51.
IF d =1 & dm=5 xxx =52.
IF d =1 & dm=6 xxx =3.
IF d =1 & dm=7 xxx =53.
IF d =1 & dm=8 xxx =54.
IF d =3 & dm=0 xxx =60.
IF d =3 & dm=4 xxx =61.
IF d =3 & dm=5 xxx =62.
IF d =3 & dm=6 xxx =3.
IF d =3 & dm=7 xxx =63.
IF d =3 & dm=8 xxx =64.
IF d =7 & dm=0 xxx =70.
IF d =7 & dm=4 xxx =71.
IF d =7 & dm=5 xxx =72.
```

```

IF d =7 & dm=6   xxx =3.
IF d =7 & dm=7   xxx =73.
IF d =7 & dm=8   xxx =74.
IF d =2 & dm=0   xxx =80.
IF d =2 & dm=4   xxx =81.
IF d =2 & dm=5   xxx =82.
IF d =2 & dm=6   xxx =3.
IF d =2 & dm=7   xxx =83.
IF d =2 & dm=8   xxx =84.
IF d =9         xxx =99.
IF d <0        xxx =d.
IF pl =3       xxx =90.
IF pl =6       xxx =96.
IF pl =4       xxx =97.
IF pl =5       xxx =98.
IF bothar=2 & botar=1 xxx=90.

end repeat.

*****.
*OCCLUSAL SURFACE.

do repeat xxx = OIUR8 OIUR7 OIUR6 OIUR5 OIUR4 OIUR3 OIUR2 OIUR1 OIUL1 OIUL2 OIUL3 OIUL4 OIUL5 OIUL6 OIUL7
OIUL8
  /oi = oi oi2 oi3 oi4 oi5 oi6 oi7 oi8 oi9 oi10 oi11 oi12 oi13 oi14 oi15 oi16
  /om = om om2 om3 om4 om5 om6 om7 om8 om9 om10 om11 om12 om13 om14 om15 om16
  /pl = pl pl2 pl3 pl4 pl5 pl6 pl7 pl8 pl9 pl10 pl11 pl12 pl13 pl14 pl15 pl16.

COMPUTE      xxx  =-999.
IF oi =0     xxx=0.
IF oi =19    xxx=0.
IF oi =4     xxx=1.
IF oi =5     xxx=2.
IF oi =6     xxx=3.
IF oi =8     xxx=4.
IF oi =12 & om=0   xxx=40.
IF oi =12 & om=4   xxx=41.
IF oi =12 & om=5   xxx=42.
IF oi =12 & om=6   xxx=3.
IF oi =12 & om=7   xxx=43.
IF oi =12 & om=8   xxx=44.
IF oi =1 & om=0   xxx=50.
IF oi =1 & om=4   xxx=51.
IF oi =1 & om=5   xxx=52.
IF oi =1 & om=6   xxx=3.
IF oi =1 & om=7   xxx=53.
IF oi =1 & om=8   xxx=54.
IF oi =3 & om=0   xxx=60.
IF oi =3 & om=4   xxx=61.
IF oi =3 & om=5   xxx=62.
IF oi =3 & om=6   xxx=3.
IF oi =3 & om=7   xxx=63.
IF oi =3 & om=8   xxx=64.
IF oi =7 & om=0   xxx=70.
IF oi =7 & om=4   xxx=71.
IF oi =7 & om=5   xxx=72.
IF oi =7 & om=6   xxx=3.
IF oi =7 & om=7   xxx=73.
IF oi =7 & om=8   xxx=74.
IF oi =2 & om=0   xxx=80.
IF oi =2 & om=4   xxx=81.
IF oi =2 & om=5   xxx=82.
IF oi =2 & om=6   xxx=3.
IF oi =2 & om=7   xxx=83.
IF oi =2 & om=8   xxx=84.
IF oi =9         xxx=99.
IF oi <0        xxx=oi.
IF pl =3       xxx=90.
IF pl =6       xxx=96.
IF pl =4       xxx=97.
IF pl =5       xxx=98.
IF bothar=2 & botar=1 xxx=90.

end repeat.

*****.
*MESIAL SURFACE.

do repeat xxx = MUR8 MUR7 MUR6 MUR5 MUR4 MUR3 MUR2 MUR1 MUL1 MUL2 MUL3 MUL4 MUL5 MUL6 MUL7 MUL8
  /m = m m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 m12 m13 m14 m15 m16
  /mm = mm mm2 mm3 mm4 mm5 mm6 mm7 mm8 mm9 mm10 mm11 mm12 mm13 mm14 mm15 mm16
  /pl = pl pl2 pl3 pl4 pl5 pl6 pl7 pl8 pl9 pl10 pl11 pl12 pl13 pl14 pl15 pl16.

COMPUTE      xxx  =-999.
IF m =0      xxx=0.

```

```

IF m =19    xxx=0.
IF m =4     xxx=1.
IF m =5     xxx=2.
IF m =6     xxx=3.
IF m =8     xxx=4.
IF m =12 & mm=0   xxx=40.
IF m =12 & mm=4   xxx=41.
IF m =12 & mm=5   xxx=42.
IF m =12 & mm=6   xxx=3.
IF m =12 & mm=7   xxx=43.
IF m =12 & mm=8   xxx=44.
IF m =1 & mm=0   xxx=50.
IF m =1 & mm=4   xxx=51.
IF m =1 & mm=5   xxx=52.
IF m =1 & mm=6   xxx=3.
IF m =1 & mm=7   xxx=53.
IF m =1 & mm=8   xxx=54.
IF m =3 & mm=0   xxx=60.
IF m =3 & mm=4   xxx=61.
IF m =3 & mm=5   xxx=62.
IF m =3 & mm=6   xxx=3.
IF m =3 & mm=7   xxx=63.
IF m =3 & mm=8   xxx=64.
IF m =7 & mm=0   xxx=70.
IF m =7 & mm=4   xxx=71.
IF m =7 & mm=5   xxx=72.
IF m =7 & mm=6   xxx=3.
IF m =7 & mm=7   xxx=73.
IF m =7 & mm=8   xxx=74.
IF m =2 & mm=0   xxx=80.
IF m =2 & mm=4   xxx=81.
IF m =2 & mm=5   xxx=82.
IF m =2 & mm=6   xxx=3.
IF m =2 & mm=7   xxx=83.
IF m =2 & mm=8   xxx=84.
IF m =9     xxx=99.
IF m <0     xxx=m.
IF pl =3    xxx=90.
IF pl =6    xxx=96.
IF pl =4    xxx=97.
IF pl =5    xxx=98.
IF bothar=2 & botar=1 xxx=90.

end repeat.

*****.
*BUCCAL SURFACE.

do repeat xxx = BUR8 BUR7 BUR6 BUR5 BUR4 BUR3 BUR2 BUR1 BUL1 BUL2 BUL3 BUL4 BUL5 BUL6 BUL7 BUL8
/b = b b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15 b16
/bm = bm bm2 bm3 bm4 bm5 bm6 bm7 bm8 bm9 bm10 bm11 bm12 bm13 bm14 bm15 bm16
/pl = pl p12 p13 p14 p15 p16 p17 p18 p19 p110 p111 p112 p113 p114 p115 p116.

COMPUTE      xxx  =-999.
IF b =0      xxx=0.
IF b =19     xxx=0.
IF b =4      xxx=1.
IF b =5      xxx=2.
IF b =6      xxx=3.
IF b =8      xxx=4.
IF b =12 & bm=0   xxx=40.
IF b =12 & bm=4   xxx=41.
IF b =12 & bm=5   xxx=42.
IF b =12 & bm=6   xxx=3.
IF b =12 & bm=7   xxx=43.
IF b =12 & bm=8   xxx=44.
IF b =1 & bm=0   xxx=50.
IF b =1 & bm=4   xxx=51.
IF b =1 & bm=5   xxx=52.
IF b =1 & bm=6   xxx=3.
IF b =1 & bm=7   xxx=53.
IF b =1 & bm=8   xxx=54.
IF b =3 & bm=0   xxx=60.
IF b =3 & bm=4   xxx=61.
IF b =3 & bm=5   xxx=62.
IF b =3 & bm=6   xxx=3.
IF b =3 & bm=7   xxx=63.
IF b =3 & bm=8   xxx=64.
IF b =7 & bm=0   xxx=70.
IF b =7 & bm=4   xxx=71.
IF b =7 & bm=5   xxx=72.
IF b =7 & bm=6   xxx=3.
IF b =7 & bm=7   xxx=73.
IF b =7 & bm=8   xxx=74.
IF b =2 & bm=0   xxx=80.
IF b =2 & bm=4   xxx=81.

```

```

IF b =2 & bm=5   xxx=82.
IF b =2 & bm=6   xxx=3.
IF b =2 & bm=7   xxx=83.
IF b =2 & bm=8   xxx=84.
IF b =9         xxx=99.
IF b <0         xxx=b.
IF pl =3       xxx=90.
IF pl =6       xxx=96.
IF pl =4       xxx=97.
IF pl =5       xxx=98.
IF bothar=2 & botar=1 xxx=90.

end repeat.

*****.
*LINGUAL SURFACE.

do repeat xxx = LUR8 LUR7 LUR6 LUR5 LUR4 LUR3 LUR2 LUR1 LUL1 LUL2 LUL3 LUL4 LUL5 LUL6 LUL7 LUL8
  /l = 1 12 13 14 15 16 17 18 19 110 111 112 113 114 115 116
  /lm = lm lm2 lm3 lm4 lm5 lm6 lm7 lm8 lm9 lm10 lm11 lm12 lm13 lm14 lm15 lm16
  /pl = pl pl2 pl3 pl4 pl5 pl6 pl7 pl8 pl9 pl10 pl11 pl12 pl13 pl14 pl15 pl16.

COMPUTE   xxx  =-999.
IF l =0   xxx=0.
IF l =19  xxx=0.
IF l =4   xxx=1.
IF l =5   xxx=2.
IF l =6   xxx=3.
IF l =8   xxx=4.
IF l =12 & lm=0   xxx=40.
IF l =12 & lm=4   xxx=41.
IF l =12 & lm=5   xxx=42.
IF l =12 & lm=6   xxx=3.
IF l =12 & lm=7   xxx=43.
IF l =12 & lm=8   xxx=44.
IF l =1 & lm=0   xxx=50.
IF l =1 & lm=4   xxx=51.
IF l =1 & lm=5   xxx=52.
IF l =1 & lm=6   xxx=3.
IF l =1 & lm=7   xxx=53.
IF l =1 & lm=8   xxx=54.
IF l =3 & lm=0   xxx=60.
IF l =3 & lm=4   xxx=61.
IF l =3 & lm=5   xxx=62.
IF l =3 & lm=6   xxx=3.
IF l =3 & lm=7   xxx=63.
IF l =3 & lm=8   xxx=64.
IF l =7 & lm=0   xxx=70.
IF l =7 & lm=4   xxx=71.
IF l =7 & lm=5   xxx=72.
IF l =7 & lm=6   xxx=3.
IF l =7 & lm=7   xxx=73.
IF l =7 & lm=8   xxx=74.
IF l =2 & lm=0   xxx=80.
IF l =2 & lm=4   xxx=81.
IF l =2 & lm=5   xxx=82.
IF l =2 & lm=6   xxx=3.
IF l =2 & lm=7   xxx=83.
IF l =2 & lm=8   xxx=84.
IF l =9         xxx=99.
IF l <0         xxx=1.
IF pl =3       xxx=90.
IF pl =6       xxx=96.
IF pl =4       xxx=97.
IF pl =5       xxx=98.
IF bothar=2 & botar=1 xxx=90.

end repeat.

var lab DUR8 '(D) Distal surface code - upper right 8'.
var lab DUR7 '(D) Distal surface code - upper right 7'.
var lab DUR6 '(D) Distal surface code - upper right 6'.
var lab DUR5 '(D) Distal surface code - upper right 5'.
var lab DUR4 '(D) Distal surface code - upper right 4'.
var lab DUR3 '(D) Distal surface code - upper right 3'.
var lab DUR2 '(D) Distal surface code - upper right 2'.
var lab DUR1 '(D) Distal surface code - upper right 1'.
var lab DUL1 '(D) Distal surface code - upper left 1'.
var lab DUL2 '(D) Distal surface code - upper left 2'.
var lab DUL3 '(D) Distal surface code - upper left 3'.
var lab DUL4 '(D) Distal surface code - upper left 4'.
var lab DUL5 '(D) Distal surface code - upper left 5'.
var lab DUL6 '(D) Distal surface code - upper left 6'.
var lab DUL7 '(D) Distal surface code - upper left 7'.
var lab DUL8 '(D) Distal surface code - upper left 8'.

```

```

var lab OIUR8 '(D) Occlusal/Incisal surface code - upper right 8'.
var lab OIUR7 '(D) Occlusal/Incisal surface code - upper right 7'.
var lab OIUR6 '(D) Occlusal/Incisal surface code - upper right 6'.
var lab OIUR5 '(D) Occlusal/Incisal surface code - upper right 5'.
var lab OIUR4 '(D) Occlusal/Incisal surface code - upper right 4'.
var lab OIUR3 '(D) Occlusal/Incisal surface code - upper right 3'.
var lab OIUR2 '(D) Occlusal/Incisal surface code - upper right 2'.
var lab OIUR1 '(D) Occlusal/Incisal surface code - upper right 1'.
var lab OIUL1 '(D) Occlusal/Incisal surface code - upper left 1'.
var lab OIUL2 '(D) Occlusal/Incisal surface code - upper left 2'.
var lab OIUL3 '(D) Occlusal/Incisal surface code - upper left 3'.
var lab OIUL4 '(D) Occlusal/Incisal surface code - upper left 4'.
var lab OIUL5 '(D) Occlusal/Incisal surface code - upper left 5'.
var lab OIUL6 '(D) Occlusal/Incisal surface code - upper left 6'.
var lab OIUL7 '(D) Occlusal/Incisal surface code - upper left 7'.
var lab OIUL8 '(D) Occlusal/Incisal surface code - upper left 8'.
var lab MUR8 '(D) Mesial surface code - upper right 8'.
var lab MUR7 '(D) Mesial surface code - upper right 7'.
var lab MUR6 '(D) Mesial surface code - upper right 6'.
var lab MUR5 '(D) Mesial surface code - upper right 5'.
var lab MUR4 '(D) Mesial surface code - upper right 4'.
var lab MUR3 '(D) Mesial surface code - upper right 3'.
var lab MUR2 '(D) Mesial surface code - upper right 2'.
var lab MUR1 '(D) Mesial surface code - upper right 1'.
var lab MUL1 '(D) Mesial surface code - upper left 1'.
var lab MUL2 '(D) Mesial surface code - upper left 2'.
var lab MUL3 '(D) Mesial surface code - upper left 3'.
var lab MUL4 '(D) Mesial surface code - upper left 4'.
var lab MUL5 '(D) Mesial surface code - upper left 5'.
var lab MUL6 '(D) Mesial surface code - upper left 6'.
var lab MUL7 '(D) Mesial surface code - upper left 7'.
var lab MUL8 '(D) Mesial surface code - upper left 8'.
var lab BUR8 '(D) Buccal surface code - upper right 8'.
var lab BUR7 '(D) Buccal surface code - upper right 7'.
var lab BUR6 '(D) Buccal surface code - upper right 6'.
var lab BUR5 '(D) Buccal surface code - upper right 5'.
var lab BUR4 '(D) Buccal surface code - upper right 4'.
var lab BUR3 '(D) Buccal surface code - upper right 3'.
var lab BUR2 '(D) Buccal surface code - upper right 2'.
var lab BUR1 '(D) Buccal surface code - upper right 1'.
var lab BUL1 '(D) Buccal surface code - upper left 1'.
var lab BUL2 '(D) Buccal surface code - upper left 2'.
var lab BUL3 '(D) Buccal surface code - upper left 3'.
var lab BUL4 '(D) Buccal surface code - upper left 4'.
var lab BUL5 '(D) Buccal surface code - upper left 5'.
var lab BUL6 '(D) Buccal surface code - upper left 6'.
var lab BUL7 '(D) Buccal surface code - upper left 7'.
var lab BUL8 '(D) Buccal surface code - upper left 8'.
var lab LUR8 '(D) Lingual/Palatal surface code - upper right 8'.
var lab LUR7 '(D) Lingual/Palatal surface code - upper right 7'.
var lab LUR6 '(D) Lingual/Palatal surface code - upper right 6'.
var lab LUR5 '(D) Lingual/Palatal surface code - upper right 5'.
var lab LUR4 '(D) Lingual/Palatal surface code - upper right 4'.
var lab LUR3 '(D) Lingual/Palatal surface code - upper right 3'.
var lab LUR2 '(D) Lingual/Palatal surface code - upper right 2'.
var lab LUR1 '(D) Lingual/Palatal surface code - upper right 1'.
var lab LUL1 '(D) Lingual/Palatal surface code - upper left 1'.
var lab LUL2 '(D) Lingual/Palatal surface code - upper left 2'.
var lab LUL3 '(D) Lingual/Palatal surface code - upper left 3'.
var lab LUL4 '(D) Lingual/Palatal surface code - upper left 4'.
var lab LUL5 '(D) Lingual/Palatal surface code - upper left 5'.
var lab LUL6 '(D) Lingual/Palatal surface code - upper left 6'.
var lab LUL7 '(D) Lingual/Palatal surface code - upper left 7'.
var lab LUL8 '(D) Lingual/Palatal surface code - upper left 8'.

val lab DUR8 to LUL8
  0 'sound' 1 'visual caries' 2 'cavitated caries' 3 'unrestorable' 4 'hard arrested decay'

  40 'sealant - sound' 41 'sealant with visual caries' 42 'sealant with cavitated caries' 43 'sealant -
failed restoration' 44 'sealant with hard arrested decay'

  50 'amalgam filling - sound' 51 'amalgam filling with visual caries' 52 'amalgam filling with cavitated
caries' 53 'amalgam filling - failed restoration' 54 'amalgam filling with hard arrested decay'

  60 'other restoration - sound' 61 'other restoration with visual caries' 62 'other restoration with
cavitated caries' 63 'other restoration - failed restoration' 64 'other restoration with hard arrested
decay'

  70 'shim, veneer - sound' 71 'shim, veneer with visual caries' 72 'shim, veneer with cavitated caries'
73 'shim, veneer - failed restoration' 74 'shim, veneer with hard arrested decay'

  80 'artificial crown - sound' 81 'artificial crown with visual caries' 82 'artificial crown with
cavitated caries' 83 'artificial crown - failed restoration' 84 'artificial crown with hard arrested
decay'

  90 'missing' 96 'missing - replaced by implant' 97 'missing - replaced by adhesive bridge' 98 'missing -
replaced by conventional bridge' 99 'unscored'.

```

TSTATUR1 etc (D) Tooth condition - upper right 1 etc

- 1 All surfaces sound
- 2 Visual caries
- 3 Cavitated caries
- 4 Unrestorable
- 5 Sealant - sound
- 6 Sealant with visual caries
- 7 Sealant with cavitated caries
- 8 Fractured sealant
- 9 Sealant with visual caries + visual caries
- 10 Sealant with visual caries + cavitated caries
- 11 Sealant with cavitated caries + visual caries
- 12 Sealant with cavitated caries + cavitated caries
- 13 Fractured sealant + visual caries
- 14 Fractured sealant + cavitated caries
- 15 Sealant - sound + visual caries
- 16 Sealant - sound + cavitated caries
- 17 Amalgam filling - sound
- 18 Amalgam filling with visual caries
- 19 Amalgam filling with cavitated caries
- 20 Fractured amalgam filling
- 21 Amalgam filling with visual caries + visual caries
- 22 Amalgam filling with visual caries + cavitated caries
- 23 Amalgam filling with cavitated caries + visual caries
- 24 Amalgam filling with cavitated caries + cavitated caries
- 25 Fractured amalgam filling + visual caries
- 26 Fractured amalgam filling + cavitated caries
- 27 Amalgam filling - sound + visual caries
- 28 Amalgam filling - sound + cavitated caries
- 29 Amalgam filling - sound + shim/veneer
- 30 Restoration - sound
- 31 Restoration with visual caries
- 32 Restoration with cavitated caries
- 33 Fractured restoration
- 34 Restoration with visual caries + visual caries
- 35 Restoration with visual caries + cavitated caries
- 36 Restoration with cavitated caries + visual caries
- 37 Restoration with cavitated caries + cavitated caries
- 38 Fractured restoration + visual caries
- 39 Fractured restoration + cavitated caries
- 40 Restoration - sound + visual caries
- 41 Restoration - sound + cavitated caries
- 42 Restoration - sound + shim/veneer
- 43 Shim/veneer - sound
- 44 Shim/veneer with visual caries
- 45 Shim/veneer with cavitated caries
- 46 Fractured shim/veneer
- 47 Shim/veneer with visual caries + visual caries
- 48 Shim/veneer with visual caries + cavitated caries
- 49 Shim/veneer with cavitated caries + visual caries
- 50 Shim/veneer with cavitated caries + cavitated caries
- 51 Fractured shim/veneer + visual caries
- 52 Fractured shim/veneer + cavitated caries
- 53 Shim/veneer - sound + visual caries
- 54 Shim/veneer - sound + cavitated caries
- 55 Crown - sound
- 56 Crown with visual caries
- 57 Crown with cavitated caries
- 58 Fractured crown
- 90 Missing
- 96 Missing replaced by implant
- 97 Missing replaced by adhesive bridge
- 98 Missing replaced by conventional bridge
- 99 Unscorable
- 101 Hard arrested decay, all other surfaces sound
- 102 Hard arrested decay on all surfaces
- 103 Sealant with hard arrested decay
- 104 Sealant - sound + hard arrested decay
- 105 Sealant with hard arrested decay + hard arrested decay
- 106 Sealant with hard arrested decay + visual caries
- 107 Sealant with hard arrested decay + cavitated caries
- 108 Fractured sealant + hard arrested decay
- 109 Sealant with visual caries + hard arrested decay
- 110 Sealant with cavitated caries + hard arrested decay

111 Shim/veneer with hard arrested decay
 112 Shim/veneer - sound + hard arrested decay
 113 Shim/veneer with hard arrested decay + hard arrested decay
 114 Shim/veneer with hard arrested decay + visual caries
 115 Shim/veneer with hard arrested decay + cavitated caries
 116 Fractured shim/veneer + hard arrested decay
 117 Shim/veneer with visual caries + hard arrested decay
 118 Shim/veneer with cavitated caries + hard arrested decay
 119 Restoration with hard arrested decay
 120 Restoration - sound + hard arrested decay
 121 Restoration with hard arrested decay + hard arrested decay
 122 Restoration with hard arrested decay + visual caries
 123 Restoration with hard arrested decay + cavitated caries
 124 Fractured restoration + hard arrested decay
 125 Restoration with visual caries + hard arrested decay
 126 Restoration with cavitated caries + hard arrested decay
 127 Amalgam filling with hard arrested decay
 128 Amalgam filling - sound + hard arrested decay
 129 Amalgam filling with hard arr. decay + hard arr. decay
 130 Amalgam filling with hard arr. decay + visual caries
 131 Amalgam filling with hard arr. decay + cavitated caries
 132 Fractured amalgam filling + hard arrested decay
 133 Amalgam filling with visual caries + hard arr. decay
 134 Amalgam filling with cavitated caries + hard arr. decay
 135 Crown with hard arrested decay

SPSS syntax

```

DO REPEAT xxx = TStatUR8 TStatUR7 TStatUR6 TStatUR5 TStatUR4 TStatUR3 TStatUR2 TStatUR1 TStatUL1 TStatUL2
TStatUL3 TStatUL4 TStatUL5 TStatUL6 TStatUL7 TStatUL8
  /d = dur8 dur7 dur6 dur5 dur4 dur3 dur2 dur1 dul1 dul2 dul3 dul4 dul5 dul6 dul7 dul8
  /oi = oiur8 oiur7 oiur6 oiur5 oiur4 oiur3 oiur2 oiur1 oiul1 oiul2 oiul3 oiul4 oiul5 oiul6 oiul7 oiul8
  /m = mur8 mur7 mur6 mur5 mur4 mur3 mur2 mur1 mul1 mul2 mul3 mul4 mul5 mul6 mul7 mul8
  /b = bur8 bur7 bur6 bur5 bur4 bur3 bur2 bur1 bul1 bul2 bul3 bul4 bul5 bul6 bul7 bul8
  /l = lur8 lur7 lur6 lur5 lur4 lur3 lur2 lur1 lul1 lul2 lul3 lul4 lul5 lul6 lul7 lul8.

compute xxx=-999.

***unscorable, clean or H.
if any(99, d, oi, m, b, l) xxx=99.
if any(0, d, oi, m, b, l) & xxx=99 xxx=1.
if any(4, d, oi, m, b, l) xxx=102.
if any(0, d, oi, m, b, l) & xxx=102 xxx=101.
if (d=0) & (oi=0) & (m=0) & (b=0) & (l=0) xxx=1.

***visual or cavitated caries.
if any(1, d, oi, m, b, l) xxx=2.
if any(2, d, oi, m, b, l) xxx=3.

***sealants.
if any(40, d, oi, m, b, l) xxx=5.
if any(44, d, oi, m, b, l) xxx=103.
if any(43, d, oi, m, b, l) xxx=8.
if any(41, d, oi, m, b, l) xxx=6.
if any(42, d, oi, m, b, l) xxx=7.

if any(4, d, oi, m, b, l) & xxx=5 xxx=104.
if any(1, d, oi, m, b, l) & xxx=5 xxx=15.
if any(2, d, oi, m, b, l) & xxx=5 xxx=16.

if any(4, d, oi, m, b, l) & xxx=103 xxx=105.
if any(1, d, oi, m, b, l) & xxx=103 xxx=106.
if any(2, d, oi, m, b, l) & xxx=103 xxx=107.

if any(4, d, oi, m, b, l) & xxx=8 xxx=108.
if any(1, d, oi, m, b, l) & xxx=8 xxx=13.
if any(2, d, oi, m, b, l) & xxx=8 xxx=14.

if any(4, d, oi, m, b, l) & xxx=6 xxx=109.
if any(1, d, oi, m, b, l) & xxx=6 xxx=9.
if any(2, d, oi, m, b, l) & xxx=6 xxx=10.

if any(4, d, oi, m, b, l) & xxx=7 xxx=110.
if any(1, d, oi, m, b, l) & xxx=7 xxx=11.
if any(2, d, oi, m, b, l) & xxx=7 xxx=12.

***shims or veneers.
if any(70, d, oi, m, b, l) xxx=43.
if any(74, d, oi, m, b, l) xxx=111.
if any(73, d, oi, m, b, l) xxx=46.
if any(71, d, oi, m, b, l) xxx=44.
if any(72, d, oi, m, b, l) xxx=45.
  
```

```

if any(4, d, oi, m, b, l) & xxx=43 xxx=112.
if any(1, d, oi, m, b, l) & xxx=43 xxx=53.
if any(2, d, oi, m, b, l) & xxx=43 xxx=54.

if any(4, d, oi, m, b, l) & xxx=111 xxx=113.
if any(1, d, oi, m, b, l) & xxx=111 xxx=114.
if any(2, d, oi, m, b, l) & xxx=111 xxx=115.

if any(4, d, oi, m, b, l) & xxx=46 xxx=116.
if any(1, d, oi, m, b, l) & xxx=46 xxx=51.
if any(2, d, oi, m, b, l) & xxx=46 xxx=52.

if any(4, d, oi, m, b, l) & xxx=44 xxx=117.
if any(1, d, oi, m, b, l) & xxx=44 xxx=47.
if any(2, d, oi, m, b, l) & xxx=44 xxx=48.

if any(4, d, oi, m, b, l) & xxx=45 xxx=118.
if any(1, d, oi, m, b, l) & xxx=45 xxx=49.
if any(2, d, oi, m, b, l) & xxx=45 xxx=50.

***other restorations.
if any(60, d, oi, m, b, l) xxx=30.
if any(64, d, oi, m, b, l) xxx=119.
if any(63, d, oi, m, b, l) xxx=33.
if any(61, d, oi, m, b, l) xxx=31.
if any(62, d, oi, m, b, l) xxx=32.

if any(4, d, oi, m, b, l) & xxx=30 xxx=120.
if any(70, d, oi, m, b, l) & xxx=30 xxx=42.
if any(73, d, oi, m, b, l) & xxx=30 xxx=42.
if any(1, d, oi, m, b, l) & xxx=30 xxx=40.
if any(2, d, oi, m, b, l) & xxx=30 xxx=41.

if any(4, d, oi, m, b, l) & xxx=119 xxx=121.
if any(1, d, oi, m, b, l) & xxx=119 xxx=122.
if any(2, d, oi, m, b, l) & xxx=119 xxx=123.

if any(4, d, oi, m, b, l) & xxx=33 xxx=124.
if any(1, d, oi, m, b, l) & xxx=33 xxx=38.
if any(2, d, oi, m, b, l) & xxx=33 xxx=39.

if any(4, d, oi, m, b, l) & xxx=31 xxx=125.
if any(1, d, oi, m, b, l) & xxx=31 xxx=34.
if any(2, d, oi, m, b, l) & xxx=31 xxx=35.

if any(4, d, oi, m, b, l) & xxx=31 xxx=126.
if any(1, d, oi, m, b, l) & xxx=31 xxx=36.
if any(2, d, oi, m, b, l) & xxx=31 xxx=37.

***amalgam fillings.
if any(50, d, oi, m, b, l) xxx=17.
if any(54, d, oi, m, b, l) xxx=127.
if any(53, d, oi, m, b, l) xxx=20.
if any(51, d, oi, m, b, l) xxx=18.
if any(52, d, oi, m, b, l) xxx=19.

if any(4, d, oi, m, b, l) & xxx=17 xxx=128.
if any(70, d, oi, m, b, l) & xxx=17 xxx=29.
if any(73, d, oi, m, b, l) & xxx=17 xxx=29.
if any(1, d, oi, m, b, l) & xxx=17 xxx=27.
if any(2, d, oi, m, b, l) & xxx=17 xxx=28.

if any(4, d, oi, m, b, l) & xxx=127 xxx=129.
if any(1, d, oi, m, b, l) & xxx=127 xxx=130.
if any(2, d, oi, m, b, l) & xxx=127 xxx=131.

if any(4, d, oi, m, b, l) & xxx=20 xxx=132.
if any(1, d, oi, m, b, l) & xxx=20 xxx=25.
if any(2, d, oi, m, b, l) & xxx=20 xxx=26.

if any(4, d, oi, m, b, l) & xxx=18 xxx=133.
if any(1, d, oi, m, b, l) & xxx=18 xxx=21.
if any(2, d, oi, m, b, l) & xxx=18 xxx=22.

if any(4, d, oi, m, b, l) & xxx=19 xxx=134.
if any(1, d, oi, m, b, l) & xxx=19 xxx=23.
if any(2, d, oi, m, b, l) & xxx=19 xxx=24.

***crowns.
if any(80, d, oi, m, b, l) xxx=55.
if any(84, d, oi, m, b, l) xxx=135.
if any(83, d, oi, m, b, l) xxx=58.
if any(81, d, oi, m, b, l) xxx=56.
if any(82, d, oi, m, b, l) xxx=57.

```

```

***unrestorable, missing, uncoded.
if any(3, d, oi, m, b, l) xxx=4.
if d=90 xxx=90.
if d=96 xxx=96.
if d=97 xxx=97.
if d=98 xxx=98.

if d=-1 xxx=-1.
IF bothar=2 & botar=1 xxx=90.

END REPEAT.

var lab tstatur8 '(D) Tooth condition - upper right 8'.
var lab tstatur7 '(D) Tooth condition - upper right 7'.
var lab tstatur6 '(D) Tooth condition - upper right 6'.
var lab tstatur5 '(D) Tooth condition - upper right 5'.
var lab tstatur4 '(D) Tooth condition - upper right 4'.
var lab tstatur3 '(D) Tooth condition - upper right 3'.
var lab tstatur2 '(D) Tooth condition - upper right 2'.
var lab tstatur1 '(D) Tooth condition - upper right 1'.
var lab tstatul1 '(D) Tooth condition - upper left 1'.
var lab tstatul2 '(D) Tooth condition - upper left 2'.
var lab tstatul3 '(D) Tooth condition - upper left 3'.
var lab tstatul4 '(D) Tooth condition - upper left 4'.
var lab tstatul5 '(D) Tooth condition - upper left 5'.
var lab tstatul6 '(D) Tooth condition - upper left 6'.
var lab tstatul7 '(D) Tooth condition - upper left 7'.
var lab tstatul8 '(D) Tooth condition - upper left 8'.

val lab tstatur8 to tstatul8
1 'All surfaces sound'
2 'Visual caries'
3 'Cavitated caries'
4 'Unrestorable'
5 'Sealant - sound'
6 'Sealant with visual caries'
7 'Sealant with cavitated caries'
8 'Fractured sealant'
9 'Sealant with visual caries + visual caries'
10 'Sealant with visual caries + cavitated caries'
11 'Sealant with cavitated caries + visual caries'
12 'Sealant with cavitated caries + cavitated caries'
13 'Fractured sealant + visual caries'
14 'Fractured sealant + cavitated caries'
15 'Sealant - sound + visual caries'
16 'Sealant - sound + cavitated caries'
17 'Amalgam filling - sound'
18 'Amalgam filling with visual caries'
19 'Amalgam filling with cavitated caries'
20 'Fractured amalgam filling '
21 'Amalgam filling with visual caries + visual caries'
22 'Amalgam filling with visual caries + cavitated caries'
23 'Amalgam filling with cavitated caries + visual caries'
24 'Amalgam filling with cavitated caries + cavitated caries'
25 'Fractured amalgam filling + visual caries'
26 'Fractured amalgam filling + cavitated caries'
27 'Amalgam filling - sound + visual caries'
28 'Amalgam filling - sound + cavitated caries'
29 'Amalgam filling - sound + shim/veneer'
30 'Restoration - sound'
31 'Restoration with visual caries'
32 'Restoration with cavitated caries'
33 'Fractured restoration '
34 'Restoration with visual caries + visual caries'
35 'Restoration with visual caries + cavitated caries'
36 'Restoration with cavitated caries + visual caries'
37 'Restoration with cavitated caries + cavitated caries'
38 'Fractured restoration + visual caries'
39 'Fractured restoration + cavitated caries'
40 'Restoration - sound + visual caries'
41 'Restoration - sound + cavitated caries'
42 'Restoration - sound + shim/veneer'
43 'Shim/veneer - sound'
44 'Shim/veneer with visual caries'
45 'Shim/veneer with cavitated caries'
46 'Fractured shim/veneer'
47 'Shim/veneer with visual caries + visual caries'
48 'Shim/veneer with visual caries + cavitated caries'
49 'Shim/veneer with cavitated caries + visual caries'
50 'Shim/veneer with cavitated caries + cavitated caries'
51 'Fractured shim/veneer + visual caries'
52 'Fractured shim/veneer + cavitated caries'
53 'Shim/veneer - sound + visual caries'
54 'Shim/veneer - sound + cavitated caries'
55 'Crown - sound'
56 'Crown with visual caries'

```

57 'Crown with cavitated caries'
 58 'Fractured crown '
 90 'Missing'
 96 'Missing replaced by implant'
 97 'Missing replaced by adhesive bridge'
 98 'Missing replaced by conventional bridge'
 99 'Unscorable'
 101 'Hard arrested decay, all other surfaces sound'
 102 'Hard arrested decay on all surfaces'
 103 'Sealant with hard arrested decay'
 104 'Sealant - sound + hard arrested decay'
 105 'Sealant with hard arrested decay + hard arrested decay'
 106 'Sealant with hard arrested decay + visual caries'
 107 'Sealant with hard arrested decay + cavitated caries'
 108 'Fractured sealant + hard arrested decay'
 109 'Sealant with visual caries + hard arrested decay'
 110 'Sealant with cavitated caries + hard arrested decay'
 111 'Shim/veneer with hard arrested decay'
 112 'Shim/veneer - sound + hard arrested decay'
 113 'Shim/veneer with hard arrested decay + hard arrested decay'
 114 'Shim/veneer with hard arrested decay + visual caries'
 115 'Shim/veneer with hard arrested decay + cavitated caries'
 116 'Fractured shim/veneer + hard arrested decay'
 117 'Shim/veneer with visual caries + hard arrested decay'
 118 'Shim/veneer with cavitated caries + hard arrested decay'
 119 'Restoration with hard arrested decay'
 120 'Restoration - sound + hard arrested decay'
 121 'Restoration with hard arrested decay + hard arrested decay'
 122 'Restoration with hard arrested decay + visual caries'
 123 'Restoration with hard arrested decay + cavitated caries'
 124 'Fractured restoration + hard arrested decay'
 125 'Restoration with visual caries + hard arrested decay'
 126 'Restoration with cavitated caries + hard arrested decay'
 127 'Amalgam filling with hard arrested decay'
 128 'Amalgam filling - sound + hard arrested decay'
 129 'Amalgam filling with hard arr. decay + hard arr. decay'
 130 'Amalgam filling with hard arr. decay + visual caries'
 131 'Amalgam filling with hard arr. decay + cavitated caries'
 132 'Fractured amalgam filling + hard arrested decay'
 133 'Amalgam filling with visual caries + hard arr. decay'
 134 'Amalgam filling with cavitated caries + hard arr. decay'
 135 'Crown with hard arrested decay'.

Coronal surfaces (counts of teeth)

NUMMISS	(D) No. of missing teeth
NUMTOOTH	(D) No. of teeth
NUMANTTH	(D) No. of anterior teeth
NUMSUT88	(D) No. of sound and untreated teeth (88 def)
NUMROS88	(D) No. of restored, otherwise sound teeth (88 def)
NUMDU88	(D) No. of decayed or unsound teeth (88 def)
NUMSUT98	(D) No. of sound and untreated teeth, not incl. roots (98 def)
NUMROS98	(D) No. of restored, otherwise sound teeth (98 def)
NUMDU98	(D) No. of decayed or unsound teeth (98 def)
NUMREST	(D) No. of restored teeth (not incl. roots)
NUMFIL	(D) No. of filled teeth
NUMAMF	(D) No. of teeth filled with amalgam
NUMNAF	(D) No. of teeth with non-amalgam fillings
NUMSEAL	(D) No. of teeth with sealants
NUMSHIM	(D) No. of teeth with shims/veneers
NUMSF98	(D) No. of teeth with sound fillings
NUMSFD98	(D) No. of teeth with sound fillings + other decay
NUMFF98	(D) No. of teeth with fractured fillings
NUMFFD98	(D) No. of teeth with fractured fillings and decay
NUMFFS98	(D) No. of teeth with fractured fillings and no other decay
NUMDF98	(D) No. of teeth with decayed fillings
NUMDFD98	(D) No. of teeth with decayed fillings + decay
NUMDFS98	(D) No. of teeth with decayed fillings + no other decay
NUMFND98	(D) No. of filled teeth with no decay
NUMDEC98	(D) No. of teeth with primary decay
NUMDNR98	(D) No. of teeth with decay and no other fillings/s/v
NUMDWR98	(D) No. of teeth with decay and other fillings/s/v
NUMUNRST	(D) No. of unrestorable teeth
NUMCAV	(D) No. of teeth with cavitated caries
NUMVIS	(D) No. of teeth with visual caries
NUMTDEC	(D) No. of teeth with decay
NUMTPRIM	(D) No. of teeth with primary decay
NUMPCAV	(D) No. of teeth with primary cavitated caries
NUMPVIS	(D) No. of teeth with primary visual caries
NUMTREC	(D) No. of teeth with recurrent decay
NUMRCAV	(D) No. of teeth with recurrent cavitated caries
NUMRVIS	(D) No. of teeth with recurrent visual caries
NUMPRRE	(D) No. of teeth with primary and recurrent caries
NUMUSF88	(D) No. of teeth with unsound fillings (88 def)
NUMFD88	(D) No. of filled and decayed teeth (88 def)
NUMDNT88	(D) No. of decayed teeth with no treatment (88 def)
NUMCROWN	(D) No. of crowns
UNSCORED	(D) No. of unscorable teeth
NUMFRES	(D) No. of teeth with fractured restorations (not incl. roots)
NUMIMP	(D) No. of implants
NUMBRID	(D) No. of bridge pontics
NUMSNAT	(D) No. of natural tooth surfaces (not incl. roots)

NUMSNATR (D) No. of natural tooth surfaces including roots

```
SPSS syntax
**counts of different types of teeth**.
```

```
compute nummiss=-999.
compute numtooth=-999.
compute numantth=-999.
compute numsut88=-999.
compute numros88=-999.
compute numdu88=-999.
compute numsut98=-999.
compute numros98=-999.
compute numdu98=-999.
compute numrest=-999.
compute numfil=-999.
compute numamf=-999.
compute numnaf=-999.
compute numseal=-999.
compute numshim=-999.
compute numsf98=-999.
compute numsf98=-999.
compute numff98=-999.
compute numffd98=-999.
compute numffs98=-999.
compute numdf98=-999.
compute numdfd98=-999.
compute numdfs98=-999.
compute numfnd98=-999.
compute numdec98=-999.
compute numdnr98=-999.
compute numdwr98=-999.
compute numunrst=-999.
compute numcav=-999.
compute numvis=-999.
compute numtdec=-999.
compute numtprim=-999.
compute numpcav=-999.
compute numpvis=-999.
compute numtrec=-999.
compute numrcav=-999.
compute numrvis=-999.
compute numpre=-999.
compute numusf88=-999.
compute numfd88=-999.
compute numdnt88=-999.
compute numcrown=-999.
compute unscored=-999.
compute numfres=-999.
compute numimp=-999.
compute numbrid=-999.
compute numsnat=-999.
compute numsnatr=-999.
```

```
do if any(exout,2,3).
compute nummiss=-1.
compute numtooth=-1.
compute numantth=-1.
compute numsut88=-1.
compute numros88=-1.
compute numdu88=-1.
compute numsut98=-1.
compute numros98=-1.
compute numdu98=-1.
compute numrest=-1.
compute numfil=-1.
compute numamf=-1.
compute numnaf=-1.
compute numseal=-1.
compute numshim=-1.
compute numsf98=-1.
compute numsf98=-1.
compute numff98=-1.
compute numffd98=-1.
compute numffs98=-1.
compute numdf98=-1.
compute numdfd98=-1.
compute numdfs98=-1.
compute numfnd98=-1.
compute numdec98=-1.
compute numdnr98=-1.
compute numdwr98=-1.
compute numunrst=-1.
```

```

compute numcav=-1.
compute numvis=-1.
compute numtdec=-1.
compute numtprim=-1.
compute numpcav=-1.
compute numpvis=-1.
compute numtrec=-1.
compute numrcav=-1.
compute numrvis=-1.
compute numprre=-1.
compute numusf88=-1.
compute numfd88=-1.
compute numdnt88=-1.
compute numcrown=-1.
compute unscored=-1.
compute numfres=-1.
compute numimp=-1.
compute numbrid=-1.
compute numsnat=-1.
compute numsnatr=-1.

else.
  count nummiss      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (90,96,97,98).

  count numtooth      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (1 thru 58, 99, 101 thru 135).

  count numantth      = tstatll3 tstatll2 tstatll1 tstatlr1 tstatlr2 tstatlr3 tstatul1 tstatul2 tstatul3
tstatul3 tstatul2 tstatul1 (1 thru 58, 99, 101 thru 135).

  count numsut88      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (1,2,5,6,8,9,13,15).

  count numros88      = tstatur8 to tstatul8 tstatll8 to tstatlr8
(17,18,21,27,29,30,31,34,40,42,43,44,47,53,55,56).

  count numdu88       = tstatur8 to tstatul8 tstatll8 to tstatlr8
(3,4,7,10,11,12,14,16,19,20,22 thru 26,28,32,33,35 thru 39,41,45,46,48 thru 52,54,57,58).

  count numsut98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (1,5,8,101 thru 105, 108).

  count numros98      = tstatur8 to tstatul8 tstatll8 to tstatlr8
(17,29,30,42,43,55,111,112,119,120,127,128,135).

count numdu98        = tstatur8 to tstatul8 tstatll8 to tstatlr8 (2,3,4,6,7,9 thru 16, 18 thru 28, 31 thru
41, 44 thru 54, 56,57,58,106,107,109,110,114,115,117,118,122,123,125,126,130,131,133,134).

  count numrest       = tstatur8 to tstatul8 tstatll8 to tstatlr8 (17 thru 58,111 thru 135).

  count numfil        = tstatur8 to tstatul8 tstatll8 to tstatlr8 (17 thru 42,119 thru 134).

  count numamf        = tstatur8 to tstatul8 tstatll8 to tstatlr8 (17 thru 29, 127 thru 134).

  count numnaf        = tstatur8 to tstatul8 tstatll8 to tstatlr8 (30 thru 42, 119 thru 126).

  count numseal       = tstatur8 to tstatul8 tstatll8 to tstatlr8 (5 thru 16, 103 thru 110).

  count numshim       = tstatur8 to tstatul8 tstatll8 to tstatlr8 (29,42 thru 54,111 thru 118).

  count numsf98       = tstatur8 to tstatul8 tstatll8 to tstatlr8
(17,27 thru 30,40,41,42,119 thru 121,127,128,129).

  count numsf98       = tstatur8 to tstatul8 tstatll8 to tstatlr8 (27,28,40,41,122,123,130,131).

  count numff98       = tstatur8 to tstatul8 tstatll8 to tstatlr8 (20,25,26,33,38,39,124,132).

  count numffd98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (25,26,38,39).

  count numffs98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (20,33,124,132).

  count numdf98       = tstatur8 to tstatul8 tstatll8 to tstatlr8
(18,19,21 thru 24,31,32,34 thru 37,125,126,133,134).

  count numdfd98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (21 thru 24, 34 thru 37).

  count numdfs98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (18,19,31,32,125,126,133,134).

  count numfnd98      = tstatur8 to tstatul8 tstatll8 to tstatlr8
(17,20,29,30,33,42,119 thru 121,124,127 thru 129,132).

  count numdec98      = tstatur8 to tstatul8 tstatll8 to tstatlr8
(2 thru 4,6,7,9 thru 16,21 thru 28,34 thru 41,47 thru 54).

  count numdnr98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (2 thru 4,6,7,9 thru 16).

  count numdwr98      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (21 thru 28,34 thru 41,47 thru 54).

  count numunrst      = tstatur8 to tstatul8 tstatll8 to tstatlr8 (4).

```

```

count numcav      = tstatu8 to tstatu8 tstatl18 to tstatl8
(3,7,10 thru 12,14,16,22,24,26,28,35,37,39,41,45,48,50,52,54,107,110,115,118,123,126,131,134) .

count numvis      = tstatu8 to tstatu8 tstatl18 to tstatl8
(2,6,9,13,15,21,23,25,27,34,36,38,40,44,47,49,51,53,106,109,114,117,122,125,130,133) .

count numtdec     = tstatu8 to tstatu8 tstatl18 to tstatl8 (2 thru 4,6,7,9 thru 16,18,19,21 thru
28,31,32,34 thru 41,44,45,47 thru 54,56,57,106,107,109,110,114,115,117,118,122,123,125,126,130,131,134) .

count numtprim    = tstatu8 to tstatu8 tstatl18 to tstatl8
(2,3,6,7,9 thru 16,21 thru 28,34 thru 41,47 thru 54,106,107,114,115,122,123,130,131) .

count numpcav     = tstatu8 to tstatu8 tstatl18 to tstatl8
(3,7,10 thru 12,14,16,22,24,26,28,35,37,39,41,48,50,52,54,107,115,123,131) .

count numpvis     = tstatu8 to tstatu8 tstatl18 to tstatl8
(2,6,9,13,15,21,23,25,27,34,36,38,40,47,49,51,53,106,114,122,130) .

count numtrec     = tstatu8 to tstatu8 tstatl18 to tstatl8
(18,19,21 thru 24,31,32,34 thru 37,44,45,47 thru 50,56,57,109,110,117,118,125,126,133,134) .

count numrcav     = tstatu8 to tstatu8 tstatl18 to tstatl8
(19,23,24,32,36,37,45,49,50,57,110,118,126,134) .

count numrvis     = tstatu8 to tstatu8 tstatl18 to tstatl8
(18,21,22,31,34,35,44,47,48,56,109,117,125,133) .

count numprre     = tstatu8 to tstatu8 tstatl18 to tstatl8 (21 thru 24,34 thru 37,47 thru 50) .

count numusf88    = tstatu8 to tstatu8 tstatl18 to tstatl8 (20,25,33,38,46,51) .

count numfd88     = tstatu8 to tstatu8 tstatl18 to tstatl8
(19,22,23,24,26,28,32,35,36,37,39,41,45,48,49,50,52,54) .

count numdnt88    = tstatu8 to tstatu8 tstatl18 to tstatl8 (3,7,10,11,12,14,16) .

count numcrown    = tstatu8 to tstatu8 tstatl18 to tstatl8 (55 thru 58,135) .

count unscored    = tstatu8 to tstatu8 tstatl18 to tstatl8 (99) .

count numfres     = tstatu8 to tstatu8 tstatl18 to tstatl8
(18,19,21 thru 24,31,32,34 thru 37,44,45,47 thru 50,56,57,117,118,125,126) .

count numimp      = tstatu8 to tstatu8 tstatl18 to tstatl8 (96) .

count numbrid     = tstatu8 to tstatu8 tstatl18 to tstatl8 (97,98) .

count xxx         = tstatu8 to tstatu8 tstatl18 to tstatl8 (1 thru 58, 99, 101 thru 135) .

compute numsnat=xxx*5.
compute numsnatr=xxx*6.
end if.

var lab nummiss '(D) No. of missing teeth'.
var lab numtooth '(D) No. of teeth'.
var lab numantth '(D) No. of anterior teeth'.
var lab numsut88 '(D) No. of sound and untreated teeth (88 def)'.
var lab numros88 '(D) No. of restored, otherwise sound teeth (88 def)'.
var lab numdu88 '(D) No. of decayed or unsound teeth (88 def)'.
var lab numsut98 '(D) No. of sound and untreated teeth, not incl. roots (98 def)'.
var lab numros98 '(D) No. of restored, otherwise sound teeth (98 def)'.
var lab numdu98 '(D) No. of decayed or unsound teeth (98 def)'.
var lab numrest '(D) No. of restored teeth (not incl. roots)'.
var lab numfil '(D) No. of filled teeth'.
var lab numamf '(D) No. of teeth filled with amalgam'.
var lab numnaf '(D) No. of teeth with non-amalgam fillings '.
var lab numseal '(D) No. of teeth with sealants'.
var lab numshim '(D) No. of teeth with shims/veneers'.
var lab numsf98 '(D) No. of teeth with sound fillings'.
var lab numsf98 '(D) No. of teeth with sound fillings + other decay'.
var lab numff98 '(D) No. of teeth with fractured fillings'.
var lab numffd98 '(D) No. of teeth with fractured fillings and decay'.
var lab numffs98 '(D) No. of teeth with fractured fillings and no other decay'.
var lab numdf98 '(D) No. of teeth with decayed fillings '.
var lab numdfd98 '(D) No. of teeth with decayed fillings + decay'.
var lab numdfs98 '(D) No. of teeth with decayed fillings + no other decay'.
var lab numfnd98 '(D) No. of filled teeth with no decay'.
var lab numdec98 '(D) No. of teeth with primary decay'.
var lab numdnr98 '(D) No. of teeth with decay and no other fillings/s/v'.
var lab numdwr98 '(D) No. of teeth with decay and other fillings/s/v'.
var lab numunrst '(D) No. of unrestorable teeth'.
var lab numcav '(D) No. of teeth with cavitated caries'.
var lab numvis '(D) No. of teeth with visual caries'.
var lab numtdec '(D) No. of teeth with decay '.
var lab numtprim '(D) No. of teeth with primary decay'.
var lab numpcav '(D) No. of teeth with primary cavitated caries '.
var lab numpvis '(D) No. of teeth with primary visual caries'.

```

```

var lab numtrec '(D) No. of teeth with recurrent decay'.
var lab numrcav '(D) No. of teeth with recurrent cavitated caries '.
var lab numrvis '(D) No. of teeth with recurrent visual caries'.
var lab numprre '(D) No. of teeth with primary and recurrent caries'.
var lab numusf88 '(D) No. of teeth with unsound fillings (88 def)'.
var lab numfd88 '(D) No. of filled and decayed teeth (88 def)'.
var lab numdnt88 '(D) No. of decayed teeth with no treatment (88 def)'.
var lab numcrown '(D) No. of crowns'.
var lab unscored '(D) No. of unscorable teeth'.
var lab numfres '(D) No. of teeth with fractured restorations (not incl. roots)'.
var lab numimp '(D) No. of implants'.
var lab numbrid '(D) No. of bridge pontics'.
var lab numsnat '(D) No. of natural tooth surfaces (not incl. roots)'.
var lab numsnatr '(D) No. of natural tooth surfaces including roots'.

```

NUMTDECR (D) Number of teeth with active decay, including roots

```

SPSS syntax
do repeat xxx = tstatur8 tstatur7 tstatur6 tstatur5 tstatur4 tstatur3 tstatur2 tstatur1
                tstatul1 tstatul2 tstatul3 tstatul4 tstatul5 tstatul6 tstatul7 tstatul8
                tstatl18 tstatl17 tstatl16 tstatl15 tstatl14 tstatl13 tstatl12 tstatl11
                tstatlr1 tstatlr2 tstatlr3 tstatlr4 tstatlr5 tstatlr6 tstatlr7 tstatlr8
                /yyy = rootur8 rootur7 rootur6 rootur5 rootur4 rootur3 rootur2 rootur1
                rootul1 rootul2 rootul3 rootul4 rootul5 rootul6 rootul7 rootul8
                rootl18 rootl17 rootl16 rootl15 rootl14 rootl13 rootl12 rootl11
                rootlr1 rootlr2 rootlr3 rootlr4 rootlr5 rootlr6 rootlr7 rootlr8
/zzz = drur8 drur7 drur6 drur5 drur4 drur3 drur2 drur1
        drul1 drul2 drul3 drul4 drul5 drul6 drul7 drul8
        drl18 drl17 drl16 drl15 drl14 drl13 drl12 drl11
        drlr1 drlr2 drlr3 drlr4 drlr5 drlr6 drlr7 drlr8.

compute zzz=-999.
if exout>=2 zzz=-1.

if
any(xxx,2,3,4,6,7,9,10,11,12,13,14,15,16,18,19,21,22,23,24,25,26,27,28,31,32,34,35,36,37,38,39,40,41,44,45
,47,48,49,50,51,52,53,54,
        56,57,106,107,109,110,114,115,117,118,122,123,125,126,130,131,134) | any(yyy,13,14,51,61)
zzz=1.

end repeat.

compute numtdecr = -999.
do if any(exout,2,3).
compute numtdecr =-1.
else.
count numtdecr = drur8 to drlr8 (1).
end if.

var lab numtdecr "(D) Number of teeth with active decay, including roots".

```

Coronal surfaces (mouth variables – grouped answers categories)

GRPMISS1 (D) No. of missing teeth - 5 cat

GRPSUT88 (D) No. of sound, untreated teeth (88 def)

GRPSUT98 (D) No. of sound, untreated teeth (98 def)

0 None
1 1 to 5
2 6 to 11
3 12 to 17
4 18 or more'

GRPMISS2 (D) No. of missing teeth - 6 cat

GRPSUT6 (D) No. of sound and untreated teeth (6 cat)

0 None
1 1 to 5
2 6 to 11
3 12 to 17
4 18 to 23
5 24 or more

GRPROS88 (D) No. of restored otherwise sound teeth (88 def)

GRPROS98 (D) No. of restored otherwise sound teeth (98 def)

0 None
1 1 to 5
2 6 to 11
3 12 or more

GRPDU88 (D) No. of decayed or unsound teeth (88 def)

GRPDU98 (D) No. of decayed or unsound teeth (98 def)

0 None
1 1 to 5
2 6 or more

GRPCROWN (D) No. of crowns - grouped

0 None
1 One
2 Two
3 Three or more

```
SPSS syntax
recode nummiss (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru 17=3) (18 thru hi=4) (else = copy) into grpmiss1.
recode nummiss (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru 17=3) (18 thru 23=4) (24 thru hi=5) (else = copy)
into grpmiss2.
recode numsut88 (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru 17=3) (18 thru hi=4) (else = copy) into grpsut88.
recode numros88 (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru hi=3) (else=copy) into grpros88.
recode numdu88 (0=0) (1 thru 5=1) (6 thru hi=2) (else=copy) into grpdu88.
recode numsut98 (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru 17=3) (18 thru hi=4) (else = copy) into grpsut98.
recode numros98 (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru hi=3) (else=copy) into grpros98.
recode numdu98 (0=0) (1 thru 5=1) (6 thru hi=2) (else=copy) into grpdu98.
recode numcrown (0=0) (1=1) (2=2) (3 thru hi=3) (else=copy) into grpcrown.
recode numsut98 (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru 17=3) (18 thru 23=4) (24 thru hi=5) (else = copy)
into grpsut6.

variable labels  grpmiss1 '(D) No. of missing teeth - 5 cat' /
  grpmiss2 '(D) No. of missing teeth - 6 cat' /
  grpsut88 '(D) No. of sound, untreated teeth (88 def)' /
  grpros88 '(D) No. of restored otherwise sound teeth (88 def)' /
  grpdu88 '(D) No. of decayed or unsound teeth (88 def)' /
  grpsut98 '(D) No. of sound, untreated teeth (98 def)' /
  grpros98 '(D) No. of restored otherwise sound teeth (98 def)' /
  grpdu98 '(D) No. of decayed or unsound teeth (98 def)' /
  grpcrown '(D) No. of crowns - grouped' /
  grpsut6 '(D) Number of sound and untreated teeth (6 cats)' .

val lab grpmiss2 grpsut6
0 'None'
1 '1 to 5'
2 '6 to 11'
3 '12 to 17'
4 '18 to 23'
5 '24 or more'.

val lab grpmiss1 grpsut88 grpsut98
0 'None'
1 '1 to 5'
2 '6 to 11'
3 '12 to 17'
4 '18 or more'.

val lab grpros88 grpros98
0 'None'
1 '1 to 5'
2 '6 to 11'
3 '12 or more'.

val lab grpdu88 grpdu98
0 'None'
1 '1 to 5'
2 '6 or more'.

val lab grpcrown
0 'None'
1 'One'
2 'Two'
3 'Three or more'.
```

GRPTOOTH (D) Number of natural teeth - 6 cat

1 1 to 8
2 9 to 14

3 15 to 20
4 21 to 26
5 27 to 31
6 32

```
SPSS syntax
compute Grptooth = 0.
RECODE Numtooth (1 thru 8 = 1) (9 thru 14 = 2) (15 thru 20 = 3) (21 thru 26 = 4) (27 thru 31 = 5) (32 = 6)
  (else = copy) INTO GrpTooth.
Var Lab GrpTooth "(D) Number of natural teeth - 6 cat".
Value Lab GrpTooth
1  "1-8"
2  "9-14"
3  "15-20"
4  "21-26"
5  "27-31"
6  "32".
```

GRPTREC (D) Number of teeth with recurrent caries - 3 cat

1 None
2 One
3 Two or more

```
SPSS syntax
compute Grptrec = 0.
RECODE Numtrec (2 thru hi = 2) (else = copy) INTO GrpTrec.
Var Lab GrpTrec "(D) Number of teeth with recurrent caries - 3 cat".
Value Lab GrpTrec
0  "None"
1  "One"
2  "Two or more".
```

NUMTTHG2 (D) No of natural teeth for Wales/NI chapters

1 1 to 14
2 15 to 20
3 21 to 26
4 27 to 31
5 32

```
SPSS syntax
compute numtthg2 = 0.
RECODE Numtooth (1 thru 14 = 1) (15 thru 20 = 2) (21 thru 26 = 3) (27 thru 31 = 4) (32 = 5) (else = copy)
  INTO numtthg2.
Var Lab numtthg2 "(D) Number of natural teeth - 6 cat".
Value Lab numtthg2
1  "1-14"
2  "15-20"
3  "21-26"
4  "27-31"
5  "32".
```

Coronal surfaces (mouth variables - dichotomous)

HAS21	(D) Has 21 or more natural teeth
HASSUT88	(D) Has 18 or more sound, untreated teeth (88 def)
HASROS88	(D) Has 12 or more restored otherwise sound teeth (88 def)
HASDU88	(D) Has 1 or more decayed or unsound teeth (88 def)
HASSUT98	(D) Has 18 or more sound, untreated teeth (98 def)
HASROS98	(D) Has 12 or more restored otherwise sound teeth (98 def)
HASDU98	(D) Has 1 or more decayed or unsound teeth (98 def)
HASTDEC	(D) Has 1 or more decayed teeth
HASTPRIM	(D) Has 1 or more teeth with primary decay
HASPCAV	(D) Has 1 or more teeth with primary cavitated caries
HASPVIS	(D) Has 1 or more teeth with primary visual caries
HASTREC	(D) Has 1 or more teeth with recurrent caries
HASRCAV	(D) Has 1 or more teeth with recurrent cavitated caries
HASRVIS	(D) Has 1 or more teeth with recurrent visual caries
HASPRRE	(D) Has 1 or more teeth with both primary and recurrent caries
HASTDECR	(D) Has 1 or more teeth with active decay, including roots.
HASUNRST	(D) Has 1 or more unrestorable teeth
HASCROWN	(D) Has 1 or more crowns
HASIMP	(D) Has 1 or more implants
HASBRID	(D) Has 1 or more bridge pontics

SPSS syntax

```

recode numtooth (0 thru 20=2) (21 thru hi=1) (else=copy) into has21.
recode numsut88 (0 thru 17=2) (18 thru hi=1) (else=copy) into hassut88.
recode numros88 (0 thru 11=2) (12 thru hi=1) (else=copy) into hasros88.
recode numdu88 (0=2) (1 thru hi=1) (else=copy) into hasdu88.
recode numsut98 (0 thru 17=2) (18 thru hi=1) (else=copy) into hassut98.
recode numros98 (0 thru 11=2) (12 thru hi=1) (else=copy) into hasros98.
recode numdu98 (0=2) (1 thru hi=1) (else=copy) into hasdu98.
recode numtdec (0=2) (1 thru hi=1) (else=copy) into hastdec.
recode numtprim (0=2) (1 thru hi=1) (else=copy) into hastprim.
recode numpcav (0=2) (1 thru hi=1) (else=copy) into haspcav.
recode numpvis (0=2) (1 thru hi=1) (else=copy) into haspvis.
recode numtrec (0=2) (1 thru hi=1) (else=copy) into hastrec.
recode numrcav (0=2) (1 thru hi=1) (else=copy) into hasrcav.
recode numrvis (0=2) (1 thru hi=1) (else=copy) into hasrvis.
recode numprre (0=2) (1 thru hi=1) (else=copy) into hasprre.
recode numtdecr (0=2) (1 thru hi = 1) (else = copy) into hastdecr .
recode numunrst (0=2) (1 thru hi=1) (else=copy) into hasunrst.
recode numcrown (0=2) (1 thru hi=1) (else=copy) into hascrown.
recode numimp (0=2) (1 thru hi=1) (else=copy) into hasimp.
recode numbrid (0=2) (1 thru hi=1) (else=copy) into hasbrid.

variable labels has21 '(D) Has 21 or more natural teeth'/
hassut88 '(D) Has 18 or more sound, untreated teeth (88 def)'/
hasros88 '(D) Has 12 or more restored otherwise sound teeth (88 def)'/
hasdu88 '(D) Has 1 or more decayed or unsound teeth (88 def)'/
hassut98 '(D) Has 18 or more sound, untreated teeth (98 def)'/
hasros98 '(D) Has 12 or more restored otherwise sound teeth (98 def)'/
hasdu98 '(D) Has 1 or more decayed or unsound teeth (98 def)'/
hastdec '(D) Has 1 or more decayed teeth'/
hastprim '(D) Has 1 or more teeth with primary decay'/
haspcav '(D) Has 1 or more teeth with primary cavitated caries'/
haspvis '(D) Has 1 or more teeth with primary visual caries'/
hastrec '(D) Has 1 or more teeth with recurrent caries'/
hasrcav '(D) Has 1 or more teeth with recurrent cavitated caries'/
hasrvis '(D) Has 1 or more teeth with recurrent visual caries'/
hasprre '(D) Has 1 or more teeth with both primary and recurrent caries'/
hastdecr '(D) Has 1 or more teeth with active decay, including roots'/
hasunrst '(D) Has 1 or more unrestorable teeth'/
hascrown '(D) Has 1 or more crowns'/
hasimp '(D) Has 1 or more implants'/
hasbrid '(D) Has 1 or more bridge pontics'/.

val lab has21 to hasbrid
1 'Yes'
2 'No'.

```

Coronal surfaces (mouth variables – counts of surfaces)

NUMSPALL	(D) No. of filled surfaces in post. teeth - all
NUMSPAM	(D) No. of filled surfaces in post. teeth - amalgam
NUMSPOF	(D) No. of filled surfaces in post. teeth - other
NUMSALL	(D) No. of filled surfaces - all
NUMSAMA	(D) No. of surfaces filled - amalgam
NUMSOFA	(D) No. of surfaces filled - other
NUMSSU	(D) No. of sound untreated surfaces (not incl. roots)
NUMSRES	(D) No. of restored surfaces (not incl. roots)
NUMSREC	(D) No. of surfaces with carious restorations (not incl. roots)
NUMSFRC	(D) No. of surfaces with fractured restorations (not incl. roots)

SPSS syntax

```
compute numspall=-999.
compute numspam=-999.
compute numspof=-999.
compute numshall=-999.
compute numsama=-999.
compute numsofa=-999.
compute numssu=-999.
compute numbres=-999.
compute numbre=-999.
compute numsfrc=-999.

do if any(exout,2,3).
  compute numspall=-1.
  compute numspam=-1.
  compute numspof=-1.
  compute numshall=-1.
  compute numsama=-1.
  compute numsofa=-1.
  compute numssu=-1.
  compute numbres=-1.
  compute numbre=-1.
  compute numsfrc=-1.
else.
  count numspall      =  DUR8 to DUR4 OIUR8 to OIUR4 MUR8 to MUR4 BUR8 to BUR4 LUR8 to LUR4
                        DUL4 to DUL8 OIUL4 to OIUL8 MUL4 to MUL8 BUL4 to BUL8 LUL4 to LUL8
                        DLL8 to DLL4 OILL8 to OILL4 MLL8 to MLL4 BLL8 to BLL4 LLL8 to LLL4
                        DLR4 to DLR8 OILR4 to OILR8 MLR4 to MLR8 BLR4 to BLR8 LLR4 to LLR8
                        (50 thru 54, 60 thru 64).
  count numspam       =  DUR8 to DUR4 OIUR8 to OIUR4 MUR8 to MUR4 BUR8 to BUR4 LUR8 to LUR4
                        DUL4 to DUL8 OIUL4 to OIUL8 MUL4 to MUL8 BUL4 to BUL8 LUL4 to LUL8
                        DLL8 to DLL4 OILL8 to OILL4 MLL8 to MLL4 BLL8 to BLL4 LLL8 to LLL4
                        DLR4 to DLR8 OILR4 to OILR8 MLR4 to MLR8 BLR4 to BLR8 LLR4 to LLR8
                        (50 thru 54).
  count numspof       =  DUR8 to DUR4 OIUR8 to OIUR4 MUR8 to MUR4 BUR8 to BUR4 LUR8 to LUR4
                        DUL4 to DUL8 OIUL4 to OIUL8 MUL4 to MUL8 BUL4 to BUL8 LUL4 to LUL8
                        DLL8 to DLL4 OILL8 to OILL4 MLL8 to MLL4 BLL8 to BLL4 LLL8 to LLL4
                        DLR4 to DLR8 OILR4 to OILR8 MLR4 to MLR8 BLR4 to BLR8 LLR4 to LLR8
                        (60 thru 64).
  count numshall      =  DUR8 to LUL8 DLL8 to LLR8 (50 thru 54, 60 thru 64).
  count numsama       =  DUR8 to LUL8 DLL8 to LLR8 (50 thru 54).
  count numsofa       =  DUR8 to LUL8 DLL8 to LLR8 (60 thru 64).
  count numssu        =  DUR8 to LUL8 DLL8 to LLR8 (0, 4, 40, 44).
  count numbres        =  DUR8 to LUL8 DLL8 to LLR8 (50 thru 84).
  count numbre        =  DUR8 to LUL8 DLL8 to LLR8 (51, 52, 61, 62, 71, 72, 81, 82).
  count numsfrc       =  DUR8 to LUL8 DLL8 to LLR8 (53, 63, 73, 83).
end if.

var lab numspall '(D) No. of filled surfaces in post. teeth - all'.
var lab numspam '(D) No. of filled surfaces in post. teeth - amalgam'.
var lab numspof '(D) No. of filled surfaces in post. teeth - other'.
var lab numshall '(D) No. of filled surfaces - all'.
var lab numsama '(D) No. of surfaces filled - amalgam'.
var lab numsofa '(D) No. of surfaces filled - other'.
var lab numssu '(D) No. of sound untreated surfaces (not incl. roots)'.
var lab numbres '(D) No. of restored surfaces (not incl. roots)'.
var lab numbre '(D) No. of surfaces with carious restorations (not incl. roots)'.
var lab numsfrc '(D) No. of surfaces with fractured restorations (not incl. roots)'.
```

Plaque

PLAQUE (D) Visible plaque present

- 1 Visible plaque
- 2 No visible plaque

```
SPSS syntax
*plaque.
  compute plaque=-999.
do if any(exout,2,3).
  compute plaque=-1.
else.
  compute plaque=2.
  if any(2,plur8 to plul8 ,plll8 to pll8) plaque=1.
end if.
var lab plaque '(D) Visible plaque present'.
val lab plaque
  1 'Visible plaque'
  2 'No visible plaque'.
```

NUMPLQ (D) No. of teeth with visible plaque

NUMNOPLQ (D) No. of teeth without visible plaque

```
SPSS syntax
*numplq.
count numplq = plur8 to plul8 plll8 to pll8 (2).
if any(exout,2,3) numplq=-1.
var lab numplq '(D) No. of teeth with visible plaque'.

*numnoplq.
count numnoplq = plur8 to plul8 plll8 to pll8 (1).
if any(exout,2,3) numnoplq=-1.
var lab numnoplq '(D) No. of teeth without visible plaque'.
```

Condition of root surfaces

For each root variable the last three characters of the variable names refer to position of tooth:

UL – Upper left, UR – Upper right, LL – Lower left, LR – Lower right

1, 2 – incisors, 3 – canine, 4, 5 – premolars, 6, 7, 8 – molars

ROOTUR1 etc (D) Root condition - upper right 1 etc

- 10 Sound - no root exposed
- 11 Exposed root - no caries/rest.
- 12 Worn root (2mm or more) - no caries/rest
- 13 Active caries
- 14 Unrestorable/pulp involved
- 15 Hard arrested decay
- 50 Amalgam filling - sound
- 51 Amalgam filling - caries
- 52 Amalgam filling - failed rest.
- 53 Amalgam filling - hard arrested decay
- 60 Restoration - sound
- 61 Restoration - caries
- 62 Restoration - failed rest.
- 63 Restoration - hard arrested decay
- 90 Missing
- 96 Missing - replaced by implant
- 97 Missing - replaced by adhesive bridge
- 98 Missing - replaced by conventional bridge
- 99 Unscorable

```
SPSS syntax
*Upper.
do repeat xxx = RootUR8 RootUR7 RootUR6 RootUR5 RootUR4 RootUR3 RootUR2 RootUR1 RootUL1 RootUL2 RootUL3
RootUL4 RootUL5 RootUL6 RootUL7 RootUL8
  /root = root root2 root3 root4 root5 root6 root7 root8 root9 root10 root11 root12 root13 root14 root15
root16
  /rootm = rootm rootm2 rootm3 rootm4 rootm5 rootm6 rootm7 rootm8 rootm9 rootm10 rootm11 rootm12 rootm13
rootm14 rootm15 rootm16
  /pl = pl pl2 pl3 pl4 pl5 pl6 pl7 pl8 pl9 pl10 pl11 pl12 pl13 pl14 pl15 pl16.

COMPUTE xxx = -999.
IF root = 0 xxx = 11.
IF root = 1 xxx = 10.
IF root = 4 xxx = 13.
IF root = 6 xxx = 14.
IF root = 7 xxx = 12.
IF root = 8 xxx = 15.
IF root = 9 xxx = 99.
IF root = 2 & rootm = 0 xxx = 50.
IF root = 2 & rootm = 4 xxx = 51.
IF root = 2 & rootm = 6 xxx = 14.
IF root = 2 & rootm = 7 xxx = 52.
IF root = 2 & rootm = 8 xxx = 53.
IF root = 3 & rootm = 0 xxx = 60.
IF root = 3 & rootm = 4 xxx = 61.
IF root = 3 & rootm = 6 xxx = 14.
IF root = 3 & rootm = 7 xxx = 62.
IF root = 3 & rootm = 8 xxx = 63.
IF root <0    xxx = root.
IF pl =3     xxx =90.
IF pl =6     xxx =96.
IF pl =4     xxx =97.
IF pl =5     xxx =98.
IF bothar=2 & botar=1 xxx=90.

end repeat.

*Lower.
do repeat xxx = RootLL8 RootLL7 RootLL6 RootLL5 RootLL4 RootLL3 RootLL2 RootLL1 RootLR1 RootLR2 RootLR3
RootLR4 RootLR5 RootLR6 RootLR7 RootLR8
  /root = root17 root18 root19 root20 root21 root22 root23 root24 root25 root26 root27 root28 root29 root30
root31 root32
  /rootm = rootm17 rootm18 rootm19 rootm20 rootm21 rootm22 rootm23 rootm24 rootm25 rootm26 rootm27 rootm28
rootm29 rootm30 rootm31 rootm32
  /pl = pl17 pl18 pl19 pl20 pl21 pl22 pl23 pl24 pl25 pl26 pl27 pl28 pl29 pl30 pl31 pl32.

COMPUTE xxx = -999.
IF root = 0 xxx = 11.
IF root = 1 xxx = 10.
IF root = 4 xxx = 13.
IF root = 6 xxx = 14.
IF root = 7 xxx = 12.
```

```

IF root = 8 xxx = 15.
IF root = 9 xxx = 99.
IF root = 2 & rootm = 0 xxx = 50.
IF root = 2 & rootm = 4 xxx = 51.
IF root = 2 & rootm = 6 xxx = 14.
IF root = 2 & rootm = 7 xxx = 52.
IF root = 2 & rootm = 8 xxx = 53.
IF root = 3 & rootm = 0 xxx = 60.
IF root = 3 & rootm = 4 xxx = 61.
IF root = 3 & rootm = 6 xxx = 14.
IF root = 3 & rootm = 7 xxx = 62.
IF root = 3 & rootm = 8 xxx = 63.
IF root <0    xxx = root.
IF pl =3     xxx =90.
IF pl =6     xxx =96.
IF pl =4     xxx =97.
IF pl =5     xxx =98.
IF bothar=2 & uppar=1 xxx=90.

end repeat.

val lab RootUR8 to RootLR8
10 'Sound - no root exposed'
11 'Exposed root - no caries/rest.'
12 'Worn root (2mm or more) - no caries/rest'
13 'Active caries'
14 'Unrestorable/pulp involved'
15 'Hard arrested decay'
50 'Amalgam filling - sound'
51 'Amalgam filling - caries'
52 'Amalgam filling - failed rest.'
53 'Amalgam filling - hard arrested decay'
60 'Restoration - sound'
61 'Restoration - caries'
62 'Restoration - failed rest.'
63 'Restoration - hard arrested decay'
90 'missing'
96 'missing - replaced by implant'
97 'missing - replaced by adhesive bridge'
98 'missing - replaced by conventional bridge'
99 'unscorable'.

var lab RootUR8 '(D) Root condition - upper right 8'.
var lab RootUR7 '(D) Root condition - upper right 7'.
var lab RootUR6 '(D) Root condition - upper right 6'.
var lab RootUR5 '(D) Root condition - upper right 5'.
var lab RootUR4 '(D) Root condition - upper right 4'.
var lab RootUR3 '(D) Root condition - upper right 3'.
var lab RootUR2 '(D) Root condition - upper right 2'.
var lab RootUR1 '(D) Root condition - upper right 1'.
var lab RootUL1 '(D) Root condition - upper left 1'.
var lab RootUL2 '(D) Root condition - upper left 2'.
var lab RootUL3 '(D) Root condition - upper left 3'.
var lab RootUL4 '(D) Root condition - upper left 4'.
var lab RootUL5 '(D) Root condition - upper left 5'.
var lab RootUL6 '(D) Root condition - upper left 6'.
var lab RootUL7 '(D) Root condition - upper left 7'.
var lab RootUL8 '(D) Root condition - upper left 8'.
var lab RootLL8 '(D) Root condition - lower left 8'.
var lab RootLL7 '(D) Root condition - lower left 7'.
var lab RootLL6 '(D) Root condition - lower left 6'.
var lab RootLL5 '(D) Root condition - lower left 5'.
var lab RootLL4 '(D) Root condition - lower left 4'.
var lab RootLL3 '(D) Root condition - lower left 3'.
var lab RootLL2 '(D) Root condition - lower left 2'.
var lab RootLL1 '(D) Root condition - lower left 1'.
var lab RootLR1 '(D) Root condition - lower right 1'.
var lab RootLR2 '(D) Root condition - lower right 2'.
var lab RootLR3 '(D) Root condition - lower right 3'.
var lab RootLR4 '(D) Root condition - lower right 4'.
var lab RootLR5 '(D) Root condition - lower right 5'.
var lab RootLR6 '(D) Root condition - lower right 6'.
var lab RootLR7 '(D) Root condition - lower right 7'.
var lab RootLR8 '(D) Root condition - lower right 8'.

```

NUMRS	(D) Number of sound roots
NUMRE	(D) Number of exposed roots
NUMRW	(D) Number of worn roots
NUMRD	(D) Total number of decayed and unsound roots
NUMRF	(D) Number of filled roots (no other decay)
NUMRU	(D) Number of unscored roots
NUMRFILL	(D) Total number of filled roots
NUMREWFD	(D) Total number of exposed, worn, filled and decayed roots
NUMRURS	(D) Number of unrestorable roots
NUMRPD	(D) Number of roots with active decay
NUMRAD	(D) Number of roots with arrested decay
NUMRFD	(D) Number of filled roots with recurrent decay
NUMRFRC	(D) Number of roots with fractured restorations

SPSS syntax

```
*Counts of roots.

compute numrs=-999.
compute numre=-999.
compute numrw=-999.
compute numrd=-999.
compute numrf=-999.
compute numru=-999.
compute numrfill=-999.
compute numrewfd=-999.
compute numrurs=-999.
compute numrpd=-999.
compute numrad=-999.
compute numrfd=-999.
compute numrfrc=-999.

do if any(exout,2,3).
  compute numrs=-1.
  compute numre=-1.
  compute numrw=-1.
  compute numrd=-1.
  compute numrf=-1.
  compute numru=-1.
  compute numrfill=-1.
  compute numrewfd=-1.
  compute numrurs=-1.
  compute numrpd=-1.
  compute numrad=-1.
  compute numrfd=-1.
  compute numrfrc=-1.
else.
  count numrs      = RootUR8 to RootLR8 (10).
  count numre      = RootUR8 to RootLR8 (11).
  count numrw      = RootUR8 to RootLR8 (12).
  count numrd      = RootUR8 to RootLR8 (13,14,15,51,53,61,63).
  count numrf      = RootUR8 to RootLR8 (50,52,60,62).
  count numru      = RootUR8 to RootLR8 (99).
  count numrfill   = RootUR8 to RootLR8 (50,51,52,53,60,61,62,63).
  count numrewfd   = RootUR8 to RootLR8 (11,12,13,14,15,50,51,52,53,61,62,63).
  count numrurs    = RootUR8 to RootLR8 (14).
  count numrpd     = RootUR8 to RootLR8 (13).
  count numrad     = RootUR8 to RootLR8 (15).
  count numrfd     = RootUR8 to RootLR8 (51,53,61,63).
  count numrfrc    = RootUR8 to RootLR8 (52,62).
end if.

var lab numrs      '(D) Number of sound roots'.
var lab numre      '(D) Number of exposed roots'.
var lab numrw      '(D) Number of worn roots'.
var lab numrd      '(D) Total number of decayed and unsound roots'.
var lab numrf      '(D) Number of filled roots (no other decay)'.
var lab numru      '(D) Number of unscored roots'.
var lab numrfill   '(D) Total number of filled roots'.
var lab numrewfd   '(D) Total number of exposed, worn, filled and decayed roots'.
var lab numrurs    '(D) Number of unrestorable roots'.
var lab numrpd     '(D) Number of roots with active decay'.
var lab numrad     '(D) Number of roots with arrested decay'.
var lab numrfd     '(D) Number of filled roots with recurrent decay'.
var lab numrfrc    '(D) Number of roots with fractured restorations'.
```

NUMRD09 (D) Number of teeth with root decay
 NUMRE09 (D) Number of vulnerable (exposed) roots
 NUMRFD09 (D) Number of decayed or filled roots
 NUMRF09 (D) Number of filled roots

```
SPSS syntax
compute numrd09 =-999.
compute numre09 =-999.
compute numrfd09 =-999.
compute numrf09 =-999.
do if any(exout,2,3).
  compute numrd09 =-1.
  compute numre09 =-1.
  compute numrfd09 =-1.
  compute numrf09 =-1.
else.
  count numrd09 =RootUR8 to RootLR8 (13, 14, 51, 61).
  count numre09 =RootUR8 to RootLR8 (11 thru 63).
  count numrfd09 =RootUR8 to RootLR8 (13, 14, 50, 51, 52, 53, 60, 61, 62, 63).
  count numrf09 =RootUR8 to RootLR8 (50, 51, 52, 53, 60, 61, 62, 63).
end if.
var lab numrd09 '(D) No. teeth with root decay'.
var lab numre09 '(D) Number of vulnerable (exposed) roots'.
var lab numrfd09 '(D) Number of decayed or filled roots'.
var lab numrf09 '(D) No. teeth with filled roots'.
```

NUMSSUR (D) Number of sound surfaces, including roots
 NUMSRESR (D) Number of restored surfaces, including roots
 NUMSRECR (D) Number of restored surfaces with recurrent decay, including roots
 NUMSFRCCR (D) Number of restored surfaces with fractured restorations, including roots

```
SPSS syntax

*counts of surfaces, including roots.
compute numssur=-999.
compute numsresr=-999.
compute numsreocr=-999.
compute numsfrccr=-999.

do if any(exout,2,3).
  compute numssur=-1.
  compute numsresr=-1.
  compute numsreocr=-1.
  compute numsfrccr=-1.
else.
  compute numssur = numssu + numrs.
  compute numsresr = numsres + numrfill.
  compute numsreocr = numsrec + numrfd.
  compute numsfrccr = numsfrc + numrfrc.
end if.

var lab numssur '(D) Number of sound surfaces, including roots'.
var lab numsresr '(D) Number of restored surfaces, including roots'.
var lab numsreocr '(D) Number of restored surfaces with recurrent decay, including roots'.
var lab numsfrccr '(D) Number of restored surfaces with fractured restorations, including roots'.
```

NUMSUTR (D) Number of sound untreated teeth (including roots)

```
SPSS syntax
do repeat xxx = tstat8 tstat7 tstat6 tstat5 tstat4 tstat3 tstat2 tstat1
               tstat11 tstat12 tstat13 tstat14 tstat15 tstat16 tstat17 tstat18
               tstat118 tstat117 tstat116 tstat115 tstat114 tstat113 tstat112 tstat111
               tstat1r1 tstat1r2 tstat1r3 tstat1r4 tstat1r5 tstat1r6 tstat1r7 tstat1r8
               /yyy = rootur8 rootur7 rootur6 rootur5 rootur4 rootur3 rootur2 rootur1
               rootul1 rootul2 rootul3 rootul4 rootul5 rootul6 rootul7 rootul8
               rootl18 rootl17 rootl16 rootl15 rootl14 rootl13 rootl12 rootl11
               rootlr1 rootlr2 rootlr3 rootlr4 rootlr5 rootlr6 rootlr7 rootlr8
               /zzz = sutur8 sutur7 sutur6 sutur5 sutur4 sutur3 sutur2 sutur1
               sutul1 sutul2 sutul3 sutul4 sutul5 sutul6 sutul7 sutul8
               sutl18 sutl17 sutl16 sutl15 sutl14 sutl13 sutl12 sutl11
               sutlr1 sutlr2 sutlr3 sutlr4 sutlr5 sutlr6 sutlr7 sutlr8.

compute zzz=-8.
if exout>=2 zzz=-1.
if any(xxx,1,5,8,101,102,103,104,105,108) zzz=1.
```

```

if (xxx>0 & xxx<>1 & xxx<>5 & xxx<>8 & xxx<>101 & xxx<>102 & xxx<>103 & xxx<>104 & xxx<>105 & xxx<>108)
zzz=2.
if yyy=13 | yyy=14 | yyy>15 zzz=2.
end repeat.
compute numsutr = -999.
do if any(exout,2,3).
compute numsutr =-1.
else.
count numsutr = sutur8 to sutlr8 (1).
end if.
var lab numsutr "(D) Number of sound untreated teeth, including crowns and roots".

```

NUMRESTR (D) Number of restored teeth, including crowns and roots

```

SPSS syntax
do repeat xxx = tstatur8 tstatur7 tstatur6 tstatur5 tstatur4 tstatur3 tstatur2 tstatur1
tstatul1 tstatul2 tstatul3 tstatul4 tstatul5 tstatul6 tstatul7 tstatul8
tstatl18 tstatl17 tstatl16 tstatl15 tstatl14 tstatl13 tstatl12 tstatl11
tstatlr1 tstatlr2 tstatlr3 tstatlr4 tstatlr5 tstatlr6 tstatlr7 tstatlr8
/yyy = rootur8 to rootlr8
/zzz = trur8 trur7 trur6 trur5 trur4 trur3 trur2 trur1
trul1 trul2 trul3 trul4 trul5 trul6 trul7 trul8
trl18 trl17 trl16 trl15 trl14 trl13 trl12 trl11
trlr1 trlr2 trlr3 trlr4 trlr5 trlr6 trlr7 trlr8.

compute zzz=-999.
if exout>=2 zzz=-1.
if (xxx >= 17 & xxx <= 58) | (xxx >= 111 & xxx <= 135) | (yyy >= 50 & yyy <= 63) zzz=1.
end repeat.

compute numrestr = -999.
do if any(exout,2,3).
compute numrestr =-1.
else.
count numrestr= trur8 to trlr8 (1,2).
end if.

var lab numrestr '(D) Number of restored teeth, including crowns and roots'.

```

NUMFLLT (D) Number of teeth with fillings in crown or root

```

do repeat xxx = tstatur8 tstatur7 tstatur6 tstatur5 tstatur4 tstatur3 tstatur2 tstatur1
tstatul1 tstatul2 tstatul3 tstatul4 tstatul5 tstatul6 tstatul7 tstatul8
tstatl18 tstatl17 tstatl16 tstatl15 tstatl14 tstatl13 tstatl12 tstatl11
tstatlr1 tstatlr2 tstatlr3 tstatlr4 tstatlr5 tstatlr6 tstatlr7 tstatlr8
/yyy = rootur8 rootur7 rootur6 rootur5 rootur4 rootur3 rootur2 rootur1
rootul1 rootul2 rootul3 rootul4 rootul5 rootul6 rootul7 rootul8
rootl18 rootl17 rootl16 rootl15 rootl14 rootl13 rootl12 rootl11
rootlr1 rootlr2 rootlr3 rootlr4 rootlr5 rootlr6 rootlr7 rootlr8
/zzz = fillur8 fillur7 fillur6 fillur5 fillur4 fillur3 fillur2 fillur1
fillul1 fillul2 fillul3 fillul4 fillul5 fillul6 fillul7 fillul8
filll18 filll17 filll16 filll15 filll14 filll13 filll12 filll11
filllr1 filllr2 filllr3 filllr4 filllr5 filllr6 filllr7 filllr8.

compute zzz=-999.
if exout>=2 zzz=-1.
if any(xxx, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39,
40, 41, 42,
119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134) | any(yyy, 51, 52, 61, 62)
zzz=1.
end repeat.

compute numfllt = -999.
do if any(exout,2,3).
compute numfllt =-1.
else.
count numfllt = fillur8 to filllr8 (1).
end if.

var lab numfllt '(D) Number of teeth with fillings in crown or root'.

```

HASFLLT (D) Has teeth with fillings in crown or root

- 1 Yes
- 2 No

```
recode numfltl (1 thru hi = 1) (0=2) (else=copy) into hastrecr .  
var lab hasfltl '(D) Has teeth with fillings in crown or root'.  
val lab hasfltl  
1 'Yes'  
2 'No'.
```

HASRESTR (D) Has restored teeth incl roots

- 1 Yes
- 2 No

```
recode numrestr (1 thru hi = 1) (0=2) (else=copy) into hasrestr .  
var lab hasrestr '(D) Has restored teeth, incl roots'.  
val lab hasrestr hasfrsr  
1 'Yes'  
2 'No'.
```

NUMFSUR (D) Number of filled surfaces, crowns and roots

```
compute numfsur = numsal1 + numrfill.  
  
if numsal1<0 numfsur =numsal1 .  
  
var lab numfsur '(D) Number of filled surfaces, crowns and roots' .
```

NUMTRECR (D) Number of restored teeth with secondary decay, including crowns and roots

```
do repeat xxx = tstat8 tstat7 tstat6 tstat5 tstat4 tstat3 tstat2 tstat1  
              tstat11 tstat12 tstat13 tstat14 tstat15 tstat16 tstat17 tstat18  
              tstat118 tstat117 tstat116 tstat115 tstat114 tstat113 tstat112 tstat111  
              tstat1r1 tstat1r2 tstat1r3 tstat1r4 tstat1r5 tstat1r6 tstat1r7 tstat1r8  
/yyy = root8 root7 root6 root5 root4 root3 root2 root1  
       root11 root12 root13 root14 root15 root16 root17 root18  
       root118 root117 root116 root115 root114 root113 root112 root111  
       root1r1 root1r2 root1r3 root1r4 root1r5 root1r6 root1r7 root1r8  
/zzz = rdur8 rdur7 rdur6 rdur5 rdur4 rdur3 rdur2 rdur1  
       rdul1 rdul2 rdul3 rdul4 rdul5 rdul6 rdul7 rdul8  
       rdll18 rdll17 rdll16 rdll15 rdll14 rdll13 rdll12 rdll11  
       rdrl1 rdrl2 rdrl3 rdrl4 rdrl5 rdrl6 rdrl7 rdrl8.  
  
compute zzz=-999.  
if exout>=2 zzz=-1.  
if any(xxx, 18, 19, 21, 22, 23, 24, 31, 32, 34, 35, 36, 37, 44, 45, 47, 48, 49, 50, 56, 57, 109, 110,  
117, 118, 125, 126, 133, 134) | any(yyy, 51, 61) zzz=1.  
end repeat.  
  
compute numtreocr = -999.  
  
do if any(exout,2,3).  
compute numtreocr =-1.  
else.  
count numtreocr = rdur8 to rdrl8 (1,2).  
end if.  
  
var lab numtreocr '(D) Number of restored teeth with secondary decay, including crowns and roots' .
```

HASTRECR (D) Has restored teeth with secondary decay, including crowns and roots

- 1 Yes
- 2 No

```
recode numtreocr (1 thru hi = 1) (0=2) (else=copy) into hastreocr .  
var lab hastreocr '(D) Has restored teeth with secondary decay, including crowns and roots'.  
val lab hastreocr  
1 'Yes'  
2 'No'.
```

NUMTFRSR (D) Number of restored teeth decayed or unsound, incl roots.

```
do repeat xxx = tstat8 tstat7 tstat6 tstat5 tstat4 tstat3 tstat2 tstat1  
              tstat11 tstat12 tstat13 tstat14 tstat15 tstat16 tstat17 tstat18  
              tstat118 tstat117 tstat116 tstat115 tstat114 tstat113 tstat112 tstat111  
              tstat1r1 tstat1r2 tstat1r3 tstat1r4 tstat1r5 tstat1r6 tstat1r7 tstat1r8  
/yyy = root8 root7 root6 root5 root4 root3 root2 root1  
       root11 root12 root13 root14 root15 root16 root17 root18  
       root118 root117 root116 root115 root114 root113 root112 root111  
       root1r1 root1r2 root1r3 root1r4 root1r5 root1r6 root1r7 root1r8  
/zzz = rdur8 rdur7 rdur6 rdur5 rdur4 rdur3 rdur2 rdur1  
       rdul1 rdul2 rdul3 rdul4 rdul5 rdul6 rdul7 rdul8  
       rdll18 rdll17 rdll16 rdll15 rdll14 rdll13 rdll12 rdll11  
       rdrl1 rdrl2 rdrl3 rdrl4 rdrl5 rdrl6 rdrl7 rdrl8.  
  
compute zzz=-999.  
if exout>=2 zzz=-1.  
if any(xxx, 18, 19, 20, 21, 22, 23, 24, 25, 26, 31, 32, 33, 34, 35, 36, 37, 38, 39, 44, 45, 46, 47, 48,  
49, 50, 51, 52, 56, 57, 58, 116, 117, 118, 124,  
125, 126, 132, 133, 134) | any(yyy, 51, 52, 61, 62) zzz=1.
```

```

end repeat.

compute numtfrsr = -999.
do if any(exout,2,3).
compute numtfrsr =-1.
else.
count numtfrsr = rdur8 to rdlr8 (1,2).
end if.

var lab numtfrsr '(D) Number of restored teeth decayed or unsound, incl roots'.

```

HASTFRSR (D) Has restored teeth decayed or unsound, incl roots.

```

recode numtreocr (1 thru hi = 1) (0=2) (else=copy) into hastreocr .
var lab hastreocr '(D) Has restored teeth decayed or unsound, incl roots'.
val lab hastreocr
1 'Yes'
2 'No'.

```

GRPREWFD (D) Total number of exposed, worn, filled and decayed roots - 4 cat

0 None
1 1 to 5
2 6 to 11
3 12 or more

```

SPSS syntax
recode numrewfd (0=0) (1 thru 5=1) (6 thru 11=2) (12 thru hi=3) (else = copy) into grprewfd.
var labels grprewfd '(D) Total number of exposed, worn, filled and decayed roots - 4 cat'.
val lab grprewfd
0 'None'
1 '1 to 5'
2 '6 to 11'
3 '12 or more'.

```

GRPRDEC (D) Number of decayed and unsound roots - 4 cat

0 None
1 1
2 2
3 3 or more

```

SPSS syntax
recode numrd (0=0) (1=1) (2=2) (3 thru hi=3) (else = copy) into grprdec.
var labels grprdec '(D) Number of decayed and unsound roots - 4 cat'.
val lab grprdec
0 'None'
1 '1'
2 '2'
3 '3 or more'.

```

GRPSUTR (D) Number of sound and untreated teeth including roots.

0 None
1 1 to 5
2 6 to 11
3 12 to 17
4 18 to 23
5 24 or more

```

SPSS syntax
RECODE numsutr (0 = 0) (1 thru 5 = 1) (6 thru 11 = 2) (12 thru 17 = 3) (18 thru 23 = 4) (24 thru high = 5)
(else = copy) INTO GrpSutr.
Var Lab GrpSutr "(D) Number of sound and untreated teeth including roots (6 cats)".
Value Lab GrpSutr
0 "None"
1 "1-5"
2 "6-11"
3 "12-17"
4 "18-23"
5 "24 or more".

```

HASSUTR (D) Has 18 or more sound untreated teeth (incl. roots)

1 Yes

2No

```
SPSS syntax
recode numsutr (0 thru 17=2) (18 thru hi=1) (else=copy) into hassutr.
val lab hassutr
1 "Yes"
2 "No".
var lab hassutr '(D) Has 18 or more sound untreated teeth (incl. roots)'.
```

HASRPD (D) Has any roots with active decay

1 Yes

2 No

```
SPSS syntax
recode numrpd (0=2) (1 thru hi=1) (else=copy) into hasrpd.
val lab hasrpd
1 'Yes'
2 'No'.
var lab hasrpd '(D) Has any roots with active decay'.
```

HASRDEC (D) Has any decayed or unsound roots

- 1 Yes
- 2 No

```
SPSS syntax
recode numrd (0 thru 0 = 0) (1 thru hi = 1)(else = copy) INTO hasrdec.
val lab hasrdec
  1 'Yes'
  2 'No'.
var lab hasrdec '(D) Has any decayed or unsound roots'.
```

HASRD09 (D) Has any teeth with root decay

HASRE09 (D) Has any teeth with vulnerable (exposed) roots

HASRFD09 (D) Has any teeth with decayed or filled roots

HASRF09 (D) Has any teeth with filled roots

- 1 Yes
- 2 No

```
SPSS syntax
recode numrd09 (0 thru 0 = 0) (1 thru hi = 1)(else = copy) INTO hasrd09.
recode numre09 (0 thru 0 = 0) (1 thru hi = 1)(else = copy) INTO hasre09.
recode numrfd09 (0 thru 0 = 0) (1 thru hi = 1)(else = copy) INTO hasrfd09.
recode numrf09 (0 thru 0 = 0) (1 thru hi = 1)(else = copy) INTO hasrf09.
val lab hasrd09 hasre09 hasrfd09 hasrf09
  1 'Yes'
  2 'No'.
var lab hasrd09 '(D) Has any teeth with root decay'.
var lab hasre09 '(D) Has any teeth with vulnerable (exposed) roots'.
var lab hasrfd09 '(D) Has any teeth with decayed or filled roots'.
var lab hasrf09 '(D) Has any teeth with filled roots'.
```

Tooth wear

UR3-UL3, LR3-LL3, only

Wear on individual surfaces only recorded for upper teeth

WEARBUR1 etc (D) Buccal tooth wear - upper right 1 etc

WEARIUR1 etc (D) Incisal tooth wear - upper right 1 etc

WEARLUR1 etc (D) Lingual/palatal tooth wear - upper right 1 etc

- 0 Missing
- 1 Sound - no wear exposing dentine
- 2 Loss of enamel - just exposing dentine
- 3 Loss of enamel exposing dentine on >1/3 of surface
- 4 Complete loss enamel/pulp exposure/exposure sec. dentine
- 8 Fractured tooth
- 9 Unscorable, >75% surface obscured

```
SPSS syntax
*Upper.
do repeat
  xxx = twearb twearb2 twearb3 twearb4 twearb5 twearb6
        tweari tweari2 tweari3 tweari4 tweari5 tweari6
        twearl twearl2 twearl3 twearl4 twearl5 twearl6
  /yyy = WearBUR3 WearBUR2 WearBUR1 WearBUL1 WearBUL2 WearBUL3
        WearIUR3 WearIUR2 WearIUR1 WearIUL1 WearIUL2 WearIUL3
        WearLUR3 WearLUR2 WearLUR1 WearLUL1 WearLUL2 WearLUL3.
  recode xxx (0=1) (1=2) (2=3) (3=4) (8=8) (9=9) (90=0) (else=copy) into yyy.
  IF bothar=2 & botar=1 yyy=0.
end repeat.

val lab wearBUR3 to wearLUL3
  0 'Missing'
  1 'Sound - no wear exposing dentine'
  2 'Loss of enamel - just exposing dentine'
  3 'Loss of enamel exposing dentine on >1/3 of surface'
  4 'Complete loss enamel/pulp exposure/exposure sec. dentine'
  8 'Fractured tooth'
  9 'Unscorable, >75% surface obscured'.

var lab WearBUR3 '(D) Buccal tooth wear - upper right 3'.
var lab WearBUR2 '(D) Buccal tooth wear - upper right 2'.
var lab WearBUR1 '(D) Buccal tooth wear - upper right 1'.
var lab WearBUL1 '(D) Buccal tooth wear - upper left 1'.
```

```

var lab WearBUL2 '(D) Buccal tooth wear - upper left 2'.
var lab WearBUL3 '(D) Buccal tooth wear - upper left 3'.
var lab WearIUR3 '(D) Incisal tooth wear - upper right 3'.
var lab WearIUR2 '(D) Incisal tooth wear - upper right 2'.
var lab WearIUR1 '(D) Incisal tooth wear - upper right 1'.
var lab WearIUL1 '(D) Incisal tooth wear - upper left 1'.
var lab WearIUL2 '(D) Incisal tooth wear - upper left 2'.
var lab WearIUL3 '(D) Incisal tooth wear - upper left 3'.
var lab WearLUR3 '(D) Lingual/palatal tooth wear - upper right 3'.
var lab WearLUR2 '(D) Lingual/palatal tooth wear - upper right 2'.
var lab WearLUR1 '(D) Lingual/palatal tooth wear - upper right 1'.
var lab WearLUL1 '(D) Lingual/palatal tooth wear - upper left 1'.
var lab WearLUL2 '(D) Lingual/palatal tooth wear - upper left 2'.
var lab WearLUL3 '(D) Lingual/palatal tooth wear - upper left 3'.

```

WEARUR1 etc (D) Tooth wear - upper right 1 etc

- 0 Missing
- 1 Sound - no wear exposing dentine
- 2 Loss of enamel - just exposing dentine
- 3 Loss of enamel exposing dentine on >1/3 of surface
- 4 Complete loss enamel/pulp exposure/exposure sec. dentine
- 8 Fractured tooth
- 9 Unscorable, >75% surface obscured

```

SPSS syntax
do repeat bbb = WearBUR3 WearBUR2 WearBUR1 WearBUL1 WearBUL2 WearBUL3
    /iii = WearIUR3 WearIUR2 WearIUR1 WearIUL1 WearIUL2 WearIUL3
    /lll = WearLUR3 WearLUR2 WearLUR1 WearLUL1 WearLUL2 WearLUL3
    /yyy = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3.

compute yyy=-999.
do if any(exout,2,3).
    compute yyy=-1.
else.
    recode bbb (9=-99) (else=copy) into wb.
    recode iii (9=-99) (else=copy) into wi.
    recode lll (9=-99) (else=copy) into wl.
    compute yyy=max(wb,wi,wl).
    if yyy=-99 yyy=9.
end if.
end repeat.

var lab WearUR3 '(D) Tooth wear - upper right 3'.
var lab WearUR2 '(D) Tooth wear - upper right 2'.
var lab WearUR1 '(D) Tooth wear - upper right 1'.
var lab WearUL1 '(D) Tooth wear - upper left 1'.
var lab WearUL2 '(D) Tooth wear - upper left 2'.
var lab WearUL3 '(D) Tooth wear - upper left 3'.

val lab wearUR3 to wearUL3
0 'Missing'
1 'Sound - no wear exposing dentine'
2 'Loss of enamel - just exposing dentine'
3 'Loss of enamel exposing dentine on >1/3 of surface'
4 'Complete loss enamel/pulp exposure/exposure sec. dentine'
8 'Fractured tooth'
9 'Unscorable, >75% surface obscured'.

```

WEARLR1 etc (D) Tooth wear - lower right 1 etc

- 0 Missing
- 1 Sound - no wear exposing dentine
- 2 Loss of enamel - just exposing dentine
- 3 Loss of enamel exposing dentine on >1/3 of surface
- 4 Complete loss enamel/pulp exposure/exposure sec. dentine
- 8 Fractured tooth
- 9 Unscorable, >75% surface obscured

```

SPSS syntax
*Lower.
do repeat
    xxx = twearb7 twearb8 twearb9 twearb10 twearb11 twearb12
    /yyy = WearLL3 WearLL2 WearLL1 WearLR1 WearLR2 WearLR3.
    recode xxx (0=1) (1=2) (2=3) (3=4) (8=8) (9=9) (90=0) (else=copy) into yyy.
    IF bothar=2 & uppar=1 yyy=0.
end repeat.

val lab wearll3 to wearlr3
0 'Missing'
1 'Sound - no wear exposing dentine'
2 'Loss of enamel - just exposing dentine'
3 'Loss of enamel exposing dentine on >1/3 of surface'
4 'Complete loss enamel/pulp exposure/exposure sec. dentine'

```

```

8 'Fractured tooth'
9 'Unscorable, >75% surface obscured'.

var lab WearLL3 '(D) Tooth wear - lower left 3'.
var lab WearLL2 '(D) Tooth wear - lower left 2'.
var lab WearLL1 '(D) Tooth wear - lower left 1'.
var lab WearLR1 '(D) Tooth wear - lower right 1'.
var lab WearLR2 '(D) Tooth wear - lower right 2'.
var lab WearLR3 '(D) Tooth wear - lower right 3'.

```

NUMWEAR (D) Number of anterior teeth with wear
NUMMOD (D) Number of anterior teeth with moderate wear
NUMSEV (D) Number of anterior teeth with severe wear
NUMMDSV (D) Number of anterior teeth with moderate or severe wear
NUMFRACT(D) Number of fractured anterior teeth

```

SPSS syntax
compute numwear=-999.
compute nummod=-999.
compute numsev=-999.
compute nummdsv=-999.
compute numfract=-999.

do if any(exout,2,3).
compute numwear=-1.
compute nummod=-1.
compute numsev=-1.
compute nummdsv=-1.
compute numfract=-1.

else.
  count numwear      = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1 WearLR1
WearLR2 WearLR3 (2,3,4).
  count nummod      = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1 WearLR1
WearLR2 WearLR3 (3).
  count numsev      = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1 WearLR1
WearLR2 WearLR3 (4).
  count nummdsv     = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1 WearLR1
WearLR2 WearLR3 (3,4).
  count numfract    = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1
WearLR1 WearLR2 WearLR3 (8).
end if.

count xxx = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1 WearLR1 WearLR2
WearLR3 (0, 9).
count yyy = WearUR3 WearUR2 WearUR1 WearUL1 WearUL2 WearUL3 WearLL3 WearLL2 WearLL1 WearLR1 WearLR2
WearLR3 (0).

*Teeth either missing or unscorable.
if xxx = 12 numwear = -8.
if xxx = 12 nummod = -8.
if xxx = 12 numsev = -8.
if xxx = 12 nummdsv = -8.
if xxx = 12 numfract = -8.

*All anterior teeth missing.
if yyy = 12 numwear = -7.
if yyy = 12 nummod = -7.
if yyy = 12 numsev = -7.
if yyy = 12 nummdsv = -7.
if yyy = 12 numfract = -7.

var lab numwear '(D) Number of anterior teeth with wear'.
var lab nummod '(D) Number of anterior teeth with moderate wear'.
var lab numsev '(D) Number of anterior teeth with severe wear'.
var lab nummdsv '(D) Number of anterior teeth with moderate or severe wear'.
var lab numfract '(D) Number of fractured anterior teeth'.

val lab numwear nummod numsev nummdsv numfract
-9 "No answer/refused"
-8 "Unscorable"
-7 "All anterior teeth missing"
-1 "Item not applicable".

```

HASWEAR (D) Any anterior teeth with wear
 HASMOD (D) Any anterior teeth with moderate wear
 HASSEV (D) Any anterior teeth with severe wear
 HASMDSV (D) Any anterior teeth with moderate or severe wear
 HASFRACT (D) Any fractured anterior teeth
 1 Yes
 2 No

```
SPSS syntax
recode numwear (0=2) (1 thru hi=1) (else=copy) into haswear.
recode nummod (0=2) (1 thru hi=1) (else=copy) into hasmod.
recode numsev (0=2) (1 thru hi=1) (else=copy) into hassev.
recode nummdsv (0=2) (1 thru hi=1) (else=copy) into hasmdsv.
recode numfract (0=2) (1 thru hi=1) (else=copy) into hasfract.

variable labels  haswear  '(D) Any anterior teeth with wear'.
variable labels  hasmod   '(D) Any anterior teeth with moderate wear'.
variable labels  hassev   '(D) Any anterior teeth with severe wear'.
variable labels  hasmdsv  '(D) Any anterior teeth with moderate or severe wear'.
variable labels  hasfract '(D) Any fractured anterior teeth'.

val lab haswear to hasfract
-9 "No answer/refused"
-8 "Unscorable"
-7 "All anterior teeth missing"
-1 "Item not applicable"
1 'Yes'
2 'No'.
```

Basic Erosive Wear Examination (BEWE)

BTEEUR (D) BEWE two or more teeth - UR sextant
 BTEEUC (D) BEWE two or more teeth - UC sextant
 BTEEUL (D) BEWE two or more teeth - UL sextant
 BTEELL (D) BEWE two or more teeth - LL sextant
 BTEELC (D) BEWE two or more teeth - LC sextant
 BTEELR (D) BEWE two or more teeth - LR sextant
 1 Yes
 2 No

```
SPSS syntax
rename var (TwoTee = BTeeUR).
rename var (TwoTee2 = BTeeUC).
rename var (TwoTee3 = BTeeUL).
rename var (TwoTee4 = BTeeLL).
rename var (TwoTee5 = BTeeLC).
rename var (TwoTee6 = BTeeLR).

var lab BTeeUR '(D) BEWE two or more teeth - UR sextant'.
var lab BTeeUC '(D) BEWE two or more teeth - UC sextant'.
var lab BTeeUL '(D) BEWE two or more teeth - UL sextant'.
var lab BTeeLL '(D) BEWE two or more teeth - LL sextant'.
var lab BTeeLC '(D) BEWE two or more teeth - LC sextant'.
var lab BTeeLR '(D) BEWE two or more teeth - LR sextant'.
```

BBOLUR (D) Highest BEWE score - UR sextant
 BBOLUC (D) Highest BEWE score - UC sextant
 BBOLUL (D) Highest BEWE score - UL sextant
 BBOLLL (D) Highest BEWE score - LL sextant
 BBOLLC (D) Highest BEWE score - LC sextant
 BBOLLR (D) Highest BEWE score - LR sextant
 0 Sound, no surface loss
 1 loss of enamel
 2 distinct defect, hard tissue loss <50% of surface area
 3 hard tissue loss >50% of surface area
 9 unscorable (>25% of tooth surface obscured)

```

SPSS syntax
rename var (Bol = BBolUR).
rename var (Bol2 = BBolUC).
rename var (Bol3 = BBolUL).
rename var (Bol4 = BBolLL).
rename var (Bol5 = BBolLC).
rename var (Bol6 = BBolLR).

var lab BBolUR '(D) Highest BEWE score - UR sextant'.
var lab BBolUC '(D) Highest BEWE score - UC sextant'.
var lab BBolUL '(D) Highest BEWE score - UL sextant'.
var lab BBolLL '(D) Highest BEWE score - LL sextant'.
var lab BBolLC '(D) Highest BEWE score - LC sextant'.
var lab BBolLR '(D) Highest BEWE score - LR sextant'.

```

Occlusal condition

CONTRPM (D) Contact - upper and lower right premolars
 CONTRML (D) Contact - upper and lower right molars
 CONTLPM (D) Contact - upper and lower left premolars
 CONTLML (D) Contact - upper and lower left molars

```

SPSS syntax
do repeat
  xxx = contact contact2 contact3 contact4
  /yyy = contrpm contrml contlpm contlml.
  recode xxx (lo thru hi=copy) into yyy.
  if uppar=1 yyy=0.
end repeat.

val lab contrpm to contlml
0 'No contact'
1 'Contact'.

var lab contrpm '(D) Contact - upper and lower right premolars'.
var lab contrml '(D) Contact - upper and lower right molars'.
var lab contlpm '(D) Contact - upper and lower left premolars'.
var lab contlml '(D) Contact - upper and lower left molars'.

```

Spaces

UR5-UL5, LR5-LL5 only

SPACEUR1 etc (D) Spaces - upper right 1 etc

- 0 N - no space
- 1 S - space >= half the size of expected tooth
- 2 D - space restored by a removable prosthesis
- 3 B - space restored by a fixed bridge
- 4 T - implant so no space

```

SPSS syntax
do repeat
  xxx = space space2 space3 space4 space5 space6 space7 space8 space9 space10 space11 space12 space13
  space14 space15 space16 space17 space18 space19 space20
  /yyy = spaceur1 spaceur2 spaceur3 spaceur4 spaceur5 spaceu11 spaceu12 spaceu13 spaceu14 spaceu15
  space111 space112 space113 space114 space115 space1r1 space1r2 space1r3 space1r4 space1r5.
  recode xxx (0=0) (1=4) (2=1) (3=2) (4=3) (else=copy) into yyy.
end repeat.

val lab spaceur1 to space1r5
0 'N - no space'
1 'S - space >= half the size of expected tooth'
2 'D - space restored by a removable prosthesis'
3 'B - space restored by a fixed bridge'
4 'T - implant so no space'.

var lab SpaceUR1 '(D) Spaces - upper right 1'.
var lab SpaceUR2 '(D) Spaces - upper right 2'.
var lab SpaceUR3 '(D) Spaces - upper right 3'.
var lab SpaceUR4 '(D) Spaces - upper right 4'.
var lab SpaceUR5 '(D) Spaces - upper right 5'.
var lab SpaceUL1 '(D) Spaces - upper left 1'.

```

```

var lab SpaceUL2 '(D) Spaces - upper left 2'.
var lab SpaceUL3 '(D) Spaces - upper left 3'.
var lab SpaceUL4 '(D) Spaces - upper left 4'.
var lab SpaceUL5 '(D) Spaces - upper left 5'.
var lab SpaceLL1 '(D) Spaces - lower left 1'.
var lab SpaceLL2 '(D) Spaces - lower left 2'.
var lab SpaceLL3 '(D) Spaces - lower left 3'.
var lab SpaceLL4 '(D) Spaces - lower left 4'.
var lab SpaceLL5 '(D) Spaces - lower left 5'.
var lab SpaceLR1 '(D) Spaces - lower right 1'.
var lab SpaceLR2 '(D) Spaces - lower right 2'.
var lab SpaceLR3 '(D) Spaces - lower right 3'.
var lab SpaceLR4 '(D) Spaces - lower right 4'.
var lab SpaceLR5 '(D) Spaces - lower right 5'.

```

NUMUSP (D) Number of upper anterior/premolar spaces
 NUMUSPU (D) Number of unfilled upper spaces
 NUMUSPD (D) Number of upper spaces filled with denture
 NUMUSPB (D) Number of upper spaces filled with bridge pontic
 NUMUSPI (D) Number of upper spaces filled with implant

```

SPSS syntax
*Upper.
compute numusp=-999.
compute numuspu=-999.
compute numuspd=-999.
compute numuspb=-999.
compute numuspi=-999.
do if any(exout,2,3) | arches=3.
  compute numusp=-1.
  compute numuspu=-1.
  compute numuspd=-1.
  compute numuspb=-1.
  compute numuspi=-1.
else.
  count numusp = spaceurl to spaceul5 (1 thru 4).
  count numuspu = spaceurl to spaceul5 (1).
  count numuspd = spaceurl to spaceul5 (2).
  count numuspb = spaceurl to spaceul5 (3).
  count numuspi = spaceurl to spaceul5 (4).
end if.

var lab numusp "(D) Number of upper anterior/premolar spaces".
var lab numuspu "(D) Number of unfilled upper spaces".
var lab numuspd "(D) Number of upper spaces filled with denture".
var lab numuspb "(D) Number of upper spaces filled with bridge pontic".
var lab numuspi "(D) Number of upper spaces filled with implant".

```

NUMLSP (D) Number of lower anterior/premolar spaces
 NUMLSPU (D) Number of unfilled lower spaces
 NUMLSPD (D) Number of lower spaces filled with denture
 NUMLSPB (D) Number of lower spaces filled with bridge pontic
 NUMLSPI (D) Number of lower spaces filled with implant

```

SPSS syntax
*Lower.
compute numlsp=-999.
compute numlspu=-999.
compute numlspd=-999.
compute numlspb=-999.
compute numlspi=-999.
do if any(exout,2,3) | arches=2.
  compute numlsp=-1.
  compute numlspu=-1.
  compute numlspd=-1.
  compute numlspb=-1.
  compute numlspi=-1.
else.
  count numlsp = spacell1 to spacelr5 (1 thru 4).
  count numlspu = spacell1 to spacelr5 (1).
  count numlspd = spacell1 to spacelr5 (2).
  count numlspb = spacell1 to spacelr5 (3).
  count numlspi = spacell1 to spacelr5 (4).
end if.

var lab numlsp "(D) Number of lower anterior/premolar spaces".
var lab numlspu "(D) Number of unfilled lower spaces".

```

```
var lab numlspd "(D) Number of lower spaces filled with denture".
var lab numlspb "(D) Number of lower spaces filled with bridge pontic".
var lab numlspl "(D) Number of lower spaces filled with implant".
```

HASUSP (D) Any upper anterior/premolar spaces
HASUSPU (D) Any unfilled upper spaces
HASUSPD (D) Any upper spaces filled with denture
HASUSPB (D) Any upper spaces filled with bridge pontic
HASUSPI (D) Any upper spaces filled with implant
1 Yes
2 No

```
SPSS syntax
recode numusp (0=2) (1 thru hi=1) (else=copy) into hasusp.
recode numuspu (0=2) (1 thru hi=1) (else=copy) into hasuspu.
recode numuspd (0=2) (1 thru hi=1) (else=copy) into hasuspd.
recode numuspb (0=2) (1 thru hi=1) (else=copy) into hasuspb.
recode numuspi (0=2) (1 thru hi=1) (else=copy) into hasuspi.

var lab hasusp "(D) Any upper anterior/premolar spaces".
var lab hasuspu "(D) Any unfilled upper spaces".
var lab hasuspd "(D) Any upper spaces filled with denture".
var lab hasuspb "(D) Any upper spaces filled with bridge pontic".
var lab hasuspi "(D) Any upper spaces filled with implant".

val lab hasusp to haslspi
  1 'Yes'
  2 'No'.

recode numusp (0=2) (1 thru hi=1) (else=copy) into hasusp.
recode numuspu (0=2) (1 thru hi=1) (else=copy) into hasuspu.
recode numuspd (0=2) (1 thru hi=1) (else=copy) into hasuspd.
recode numuspb (0=2) (1 thru hi=1) (else=copy) into hasuspb.
recode numuspi (0=2) (1 thru hi=1) (else=copy) into hasuspi.

var lab hasusp "(D) Any upper anterior/premolar spaces".
var lab hasuspu "(D) Any unfilled upper spaces".
var lab hasuspd "(D) Any upper spaces filled with denture".
var lab hasuspb "(D) Any upper spaces filled with bridge pontic".
var lab hasuspi "(D) Any upper spaces filled with implant".
```

HASLSP (D) Any lower anterior/premolar spaces.
HASLSPU (D) Any unfilled lower spaces.
HASLSPD (D) Any lower spaces filled with denture.
HASLSPB (D) Any lower spaces filled with bridge pontic.
HASLSPI (D) Any lower spaces filled with implant.

```
SPSS syntax
recode numlsp (0=2) (1 thru hi=1) (else=copy) into haslsp.
recode numlspu (0=2) (1 thru hi=1) (else=copy) into haslspu.
recode numlspd (0=2) (1 thru hi=1) (else=copy) into haslspd.
recode numlspb (0=2) (1 thru hi=1) (else=copy) into haslspb.
recode numlspl (0=2) (1 thru hi=1) (else=copy) into haslspl.

var lab haslsp "(D) Any upper anterior/premolar spaces".
var lab haslspu "(D) Any unfilled upper spaces".
var lab haslspd "(D) Any upper spaces filled with denture".
var lab haslspb "(D) Any upper spaces filled with bridge pontic".
var lab haslspl "(D) Any upper spaces filled with implant".

recode numlsp (0=2) (1 thru hi=1) (else=copy) into haslsp.
recode numlspu (0=2) (1 thru hi=1) (else=copy) into haslspu.
recode numlspd (0=2) (1 thru hi=1) (else=copy) into haslspd.
recode numlspb (0=2) (1 thru hi=1) (else=copy) into haslspb.
recode numlspl (0=2) (1 thru hi=1) (else=copy) into haslspl.

var lab haslsp "(D) Any upper anterior/premolar spaces".
var lab haslspu "(D) Any unfilled upper spaces".
var lab haslspd "(D) Any upper spaces filled with denture".
var lab haslspb "(D) Any upper spaces filled with bridge pontic".
var lab haslspl "(D) Any upper spaces filled with implant".
```

Dentures

WEARDENT (D) Wears denture

- 1 Yes
- 2 No

```
SPSS syntax
recode denture2 (lo thru hi=copy) into weardent.
var lab weardent '(D) Wears denture'.
val lab weardent
  1 'Yes' 2 'No'.
```

DENTWEAR (D) Summary of dental wear

- 0 No denture worn
- 1 Upper
- 2 Lower
- 3 Both

```
SPSS syntax
recode where (lo thru hi=copy) into dentwear.
if weardent=2 dentwear=0.
var lab dentwear '(D) Summary of dental wear'.
val lab dentwear
  0 'No denture worn'
  1 'Upper'
  2 'Lower'
  3 'Both'.
```

DENTSUMM (D) Combinations of dentures

- 0 'no dentures worn'
- 1 'partial in one arch'
- 2 'partial in both arches'
- 3 'complete in one arch plus partial'
- 4 'complete in one arch, no partial'

```
compute dentsumm=999.
if eweight>=0 dentsumm=-1.
if dentwear=0 dentsumm=0 .
if dentwear=1 & utype=1 dentsumm=1 .
if dentwear=1 & utype=4 dentsumm=1 .
if dentwear=2 & ltype=1 dentsumm=1 .
if dentwear=2 & ltype=4 dentsumm=1 .
if dentwear=3 & (utype=1 or utype=4) & (ltype=1 or ltype=4) dentsumm=2.
if dentwear=3 & (utype=1 or utype=4) & (ltype=2 or ltype=3) dentsumm=3 .
if dentwear=3 & (utype=2 or utype=3) & (ltype=1 or ltype=4) dentsumm=3 .
if dentwear=1 & (utype=2 or utype=3) dentsumm=4 .
if dentwear=2 & (ltype=2 or ltype=3) dentsumm=4 .

var lab dentsumm '(D) combinations of dentures'.
val lab dentsumm 0 'no dentures worn' 1 'partial in one arch' 2 'partial in both arches' 3 'complete in
one arch plus partial' 4 'complete in one arch, no partial' .
```

PUFA index and other urgent conditions

PAINDV (D) Experiencing pain related to teeth

- 1 Yes
- 2 No

```
SPSS syntax
recode pain (1=1) (0=2) (else=copy) into paindv.
if paintth=0 paindv=2.
val lab paindv
  1 'Yes'
  2 'No'.
var lab paindv '(D) Experiencing pain related to teeth'.
```

PHYSPNG (D) OHIP physical pain grouped - 3 cats

- 1 Never/hardly ever
- 2 Occasionally
- 3 Fairly/very often

```
SPSS syntax
recode physpain (1 thru 2=1) (4 thru 5=5) (else=copy) into physpng
val lab physpng
1 'Never/hardly ever'
2 'Occasionally'
3 'Fairly/very often.'
var lab physpng '(D) Physical pain grouped - 3 cats'.
```

PUFA (D) PUFA score

```
SPSS syntax
compute pufa=0.
if pulp>0 pufa = pufa+pulp.
if ulc>0 pufa = pufa+ulc.
if fist>0 pufa = pufa+fist.
if absc>0 pufa = pufa+absc.
if any (-9, pulp, ulc, fist, absc) pufa=-9.
if any (-8, pulp, ulc, fist, absc) pufa=-8.
if any (-1, pulp, ulc, fist, absc) pufa=-1.
var lab pufa '(D) PUFA score'.
```

HASPUFA (D) Has one or more PUFA symptoms

- 1 Yes
- 2 No

```
SPSS syntax
recode pufa (0=2) (1 thru hi=1) (else=copy) into haspufa.
val lab haspufa
1 'Yes'
2 'No'.
var lab haspufa '(D) Has one or more PUFA symptoms'.
```

SEPS (D) Whether have any fistula/abcess

- 1 Yes
- 2 No

```
SPSS syntax
compute seps=-1.
if pulp>0 seps=0.
if fist>0 seps=1.
if absc>0 seps=1.
val lab seps
1 'Yes'
2 'No'.
var lab seps '(D) Has fistula/abcess'.
```

URGENT (D) Urgent conditions score

```
SPSS syntax
compute urgent1=0.
if paindv=1 urgent1= urgent1+1 .
if physpain> 3 urgent1= urgent1+1 .
if haspufa=1 urgent1 = urgent1+1 .
if hasunrst = 1 urgent1 =urgent1+1 .

if eweight<=0 urgent1=-1 .

var lab urgent1 '(D) Urgent conditions score (out of 4)'.
```

PULPYN (D) Has pulp yes/no

ULCYN (D) Has ulc yes/no

FISTYN (D) Has fist yes/no

ABSCYN (D) Has absc yes/no

- 1 Yes
- 2 No

SPSS syntax

```
recode pulp ulc fist absc (0=2) (1 thru hi=1) (else=copy) into pulpyn ulcyn fistyn abscyn.  
var lab pulpyn '(D) has pulp yes/no'.  
var lab ulcyn '(D) has ulc yes/no' .  
var lab fistyn '(D) has fist yes/no' .  
var lab abscyn '(D) has absc yes/no'.  
val lab pulpyn ulcyn fistyn abscyn 1 'Yes' 2 'No'.
```

Periodontal condition, including BPE

TEEUR etc (D) Two or more teeth - UR sextant etc

- 1 Yes
- 2 No

```
SPSS syntax
rename var (TwoTee7 = TeeUR).
rename var (TwoTee8 = TeeUC).
rename var (TwoTee9 = TeeUL).
rename var (TwoTee10 = TeeLL).
rename var (TwoTee11 = TeeLC).
rename var (TwoTee12 = TeeLR).

var lab TeeUR '(D) Two or more teeth - UR sextant'.
var lab TeeUC '(D) Two or more teeth - UC sextant'.
var lab TeeUL '(D) Two or more teeth - UL sextant'.
var lab TeeLL '(D) Two or more teeth - LL sextant'.
var lab TeeLC '(D) Two or more teeth - LC sextant'.
var lab TeeLR '(D) Two or more teeth - LR sextant'.

val lab TeeUR to TeeLR
1 'Yes'
2 'No'.
```

PKETUR etc (D) Pocket depth - UR sextant etc

- 1 Up to 3.5mm
- 2 4-5.5mm
- 3 6-8.5mm
- 4 9mm and over
- 9 Unscorable

```
SPSS syntax
do repeat
  xxx = pocket pocket2 pocket3 pocket4 pocket5 pocket6
  /yyy = PcketUR PcketUC PcketUL PcketLL PcketLC PcketLR
  /zzz = TeeUR TeeUC TeeUL TeeLL TeeLC TeeLR.
  recode xxx (0=1) (1=2) (2=3) (3=4) (9=9) (else=copy) into yyy.
  if any(zzz,-9,-8) | any(xxx,-9,-8) yyy=-8.
end repeat.

var lab PcketUR '(D) Pocket depth - UR sextant'.
var lab PcketUC '(D) Pocket depth - UC sextant'.
var lab PcketUL '(D) Pocket depth - UL sextant'.
var lab PcketLL '(D) Pocket depth - LL sextant'.
var lab PcketLC '(D) Pocket depth - LC sextant'.
var lab PcketLR '(D) Pocket depth - LR sextant'.

val lab pcketur to pcketlr
1 'Up to 3.5mm'
2 '4-5.5mm'
3 '6-8.5mm'
4 '9mm and over'
9 'Unscorable'.
```

LOAUR etc (D) Loss of attachment - UR sextant etc

- 1 Up to 3.5mm
- 2 4-5.5mm
- 3 6-8.5mm
- 4 9mm and over
- 9 Unscorable

```
SPSS syntax
do repeat
  xxx = loa loa2 loa3 loa4 loa5 loa6
  /yyy = LOAUR LOAUC LOAUL LOALL LOALC LOALR
  /zzz = TeeUR TeeUC TeeUL TeeLL TeeLC TeeLR.
  recode xxx (0=1) (1=2) (2=3) (3=4) (9=9) (else=copy) into yyy.
  if any(zzz,-9,-8) | any(xxx,-9,-8) yyy=-8.
  if dvage<55 yyy = -1.
end repeat.

val lab loaur to loalr
1 'Up to 3.5mm'
2 '4-5.5mm'
```

```

3 '6-8.5mm'
4 '9mm and over'
9 'Unscorable'.

var lab LOAUR '(D) Loss of attachment - UR sextant'.
var lab LOAUC '(D) Loss of attachment - UC sextant'.
var lab LOAUL '(D) Loss of attachment - UL sextant'.
var lab LOALL '(D) Loss of attachment - LL sextant'.
var lab LOALC '(D) Loss of attachment - LC sextant'.
var lab LOALR '(D) Loss of attachment - LR sextant'.

```

CALCUR etc (D) Calculus - UR sextant etc

- 1 No visible calculus
- 2 Visible calculus
- 9 Unscorable

```

SPSS syntax
do repeat xxx = calc calc2 calc3 calc4 calc5 calc6
  /yyy = CalcUR CalcUC CalcUL CalcLL CalcLC CalcLR
  /zzz = TeeUR TeeUC TeeUL TeeLL TeeLC TeeLR.
  recode xxx (0=1) (1=2) (9=9) (else=copy) into yyy.
  if any(zzz,-9,-8) | any(xxx,-9,-8) yyy=-8.
end repeat.

val lab calcur to calclr
  1 'No visible calculus'
  2 'Visible calculus'
  9 'Unscorable'.

var lab CalcUR '(D) Calculus - UR sextant'.
var lab CalcUC '(D) Calculus - UC sextant'.
var lab CalcUL '(D) Calculus - UL sextant'.
var lab CalcLL '(D) Calculus - LL sextant'.
var lab CalcLC '(D) Calculus - LC sextant'.
var lab CalcLR '(D) Calculus - LR sextant'.

```

BLEEDUR (D) Bleeding - UR sextant

- 1 Absent
- 2 Present

```

SPSS syntax
do repeat xxx = bleed bleed2 bleed3 bleed4 bleed5 bleed6
  /yyy = BleedUR BleedUC BleedUL BleedLL BleedLC BleedLR
  /zzz = TeeUR TeeUC TeeUL TeeLL TeeLC TeeLR.
  recode xxx (0=1) (1=2) (else=copy) into yyy.
  if any(zzz,-9,-8) | any(xxx,-9,-8) yyy=-8.
end repeat.

var lab BleedUR '(D) Bleeding - UR sextant'.
var lab BleedUC '(D) Bleeding - UC sextant'.
var lab BleedUL '(D) Bleeding - UL sextant'.
var lab BleedLL '(D) Bleeding - LL sextant'.
var lab BleedLC '(D) Bleeding - LC sextant'.
var lab BleedLR '(D) Bleeding - LR sextant'.

val lab bleedur to bleedlr
  1 'Absent'
  2 'Present'.

```

BPEUR (D) BPE - UR sextant

- 0 no bleeding or pocketing
- 1 bleeding on probing - no pocketing >3.5mm
- 2 plaque retentive factors present - no pocketing > 3.5mm
- 3 pockets >3.5mm <5.5mm deep
- 4 pockets >5.5mm deep

```

SPSS syntax
rename var (BPE = BPEUR).
rename var (BPE2 = BPEUC).
rename var (BPE3 = BPEUL).
rename var (BPE4 = BPELL).
rename var (BPE5 = BPELC).
rename var (BPE6 = BPELR).

var lab BPEUR '(D) BPE - UR sextant'.
var lab BPEUC '(D) BPE - UC sextant'.
var lab BPEUL '(D) BPE - UL sextant'.
var lab BPELL '(D) BPE - LL sextant'.
var lab BPELC '(D) BPE - LC sextant'.

```

```
var lab BPELR '(D) BPE - LR sextant'.
```

ULSEXa etc (D) UL sextant etc - No pocketing or LOA above 3.5 mm

1 Yes
2 No
9 Sextant not scored

```
SPSS syntax
*ULSEXa - no pocketing or LOA above 3.5mm.
recode PcketUL (1=1) (2,3,4=2) (9=9) (else=copy) into ULSEXa.
if any(LOAUL,2,3,4) ULSEXa=2.
if LOAUL=9 & PcketUL=9 ULSEXa=9.
var lab ulsexa '(D) UL sextant - No pocketing or LOA above 3.5 mm'.
val lab ulsexa
  1 'Yes'
  2 'No'
  9 'Sextant not scored'.
freq ulsexa.
temp.
select if pcketul=1 & loa=1.

*UCSEXa - no pocketing or LOA above 3.5mm.
recode PcketUC (1=1) (2,3,4=2) (9=9) (else=copy) into UCSEXa.
if any(LOAUC,2,3,4) UCSEXa=2.
if LOAUC=9 & PcketUC=9 UCSEXa=9.
var lab UCsexa '(D) UC sextant - No pocketing or LOA above 3.5 mm'.
val lab UCsexa
  1 'Yes'
  2 'No'
  9 'Sextant not scored'.

*URSEXa - no pocketing or LOA above 3.5mm.
recode PcketUR (1=1) (2,3,4=2) (9=9) (else=copy) into URSEXa.
if any(LOAUR,2,3,4) URSEXa=2.
if LOAUR=9 & PcketUR=9 URSEXa=9.
var lab URsexa '(D) UR sextant - No pocketing or LOA above 3.5 mm'.
val lab URsexa
  1 'Yes'
  2 'No'
  9 'Sextant not scored'.

*LLSEXa - no pocketing or LOA above 3.5mm.
recode PcketLL (1=1) (2,3,4=2) (9=9) (else=copy) into LLSEXa.
if any(LOALL,2,3,4) LLSEXa=2.
if LOALL=9 & PcketLL=9 LLSEXa=9.
var lab LLsexa '(D) LL sextant - No pocketing or LOA above 3.5 mm'.
val lab LLsexa
  1 'Yes'
  2 'No'
  9 'Sextant not scored'.

*LCSEXa - no pocketing or LOA above 3.5mm.
recode PcketLC (1=1) (2,3,4=2) (9=9) (else=copy) into LCSEXa.
if any(LOALC,2,3,4) LCSEXa=2.
if LOALC=9 & PcketLC=9 LCSEXa=9.
var lab LCsexa '(D) LC sextant - No pocketing or LOA above 3.5 mm'.
val lab LCsexa
  1 'Yes'
  2 'No'
  9 'Sextant not scored'.

*LRSEXa - no pocketing or LOA above 3.5mm.
recode PcketLR (1=1) (2,3,4=2) (9=9) (else=copy) into LRSEXa.
if any(LOALR,2,3,4) LRSEXa=2.
if LOALR=9 & PcketLR=9 LRSEXa=9.
var lab LRsexa '(D) LR sextant - No pocketing or LOA above 3.5 mm'.
val lab LRsexa
  1 'Yes'
  2 'No'
  9 'Sextant not scored'.
```

NUMSEXa (D) Number of sextants with no pocketing or LOA above 3.5mm

```
SPSS syntax
compute numsexa=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=2 & TeeLC=2 & TeeLR=2).
  compute numsexa=-1.
else.
  count numsexa = ulsexa ucsexa ursexa llsexa lcsexa lrsexa (1).
end if.
var lab numsexa '(D) Number of sextants with no pocketing or LOA above 3.5mm'.
```

ULSEXB etc (D) UL sextant etc - LOA 4 mm or greater etc

- 1 Yes
- 2 No
- 9 Unscorable

```
SPSS syntax
*ULSEXB - LOA4mm or greater.
recode LOAUL (1=2) (2,3,4=1) (9=9) (else=copy) into ULSEXB.
var lab ulsexb '(D) UL sextant - LOA 4 mm or greater'.
val lab ulsexb
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*UCSEXB - LOA4mm or greater.
recode LOAUC (1=2) (2,3,4=1) (9=9) (else=copy) into UCSEXB.
var lab ucsexb '(D) UC sextant - LOA 4 mm or greater'.
val lab ucsexb
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*URSEXB - LOA4mm or greater.
recode LOAUR (1=2) (2,3,4=1) (9=9) (else=copy) into URSEXB.
var lab ursexb '(D) UR sextant - LOA 4 mm or greater'.
val lab ursexb
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LLSEXB - LOA4mm or greater.
recode LOALL (1=2) (2,3,4=1) (9=9) (else=copy) into LLSEXB.
var lab llsexb '(D) LL sextant - LOA 4 mm or greater'.
val lab llsexb
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LCSEXB - LOA4mm or greater.
recode LOALC (1=2) (2,3,4=1) (9=9) (else=copy) into LCSEXB.
var lab lcsexb '(D) LC sextant - LOA 4 mm or greater'.
val lab lcsexb
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LRSEXB - LOA4mm or greater.
recode LOALR (1=2) (2,3,4=1) (9=9) (else=copy) into LRSEXB.
var lab lrsexb '(D) LR sextant - LOA 4 mm or greater'.
val lab lrsexb
  1 'Yes'
  2 'No'
  9 'Unscorable'.
```

NUMSEXB (D) Number of sextants with LOA of 4mm or greater

```
SPSS syntax
compute numsexb=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=1 & TeeLC=2 & TeeLR=2) | dvage<55.
  compute numsexb=-1.
else.
  count numsexb = ulsexb ucsexb ursexb llsexb lcsexb lrsexb (1).
end if.
var lab numsexb '(D) Number of sextants with LOA of 4mm or greater'.
```

ULSEXC etc (D) UL sextant etc - LOA 6 mm or greater etc

- 1 Yes
- 2 No
- 9 Unscorable

```
SPSS syntax
*ULSEXC - LOA6mm or greater.
recode LOAUL (1,2=2) (3,4=1) (9=9) (else=copy) into ULSEXC.
var lab ulsexc '(D) UL sextant - LOA 6 mm or greater'.
val lab ulsexc
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*UCSEXC - LOA6mm or greater.
```

```

recode LOAUC (1,2=2) (3,4=1) (9=9) (else=copy) into UCSEXC.
var lab ucsexc '(D) UC sextant - LOA 6 mm or greater'.
val lab ucsexc
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*URSEXC - LOA6mm or greater.
recode LOAUR (1,2=2) (3,4=1) (9=9) (else=copy) into URSEXC.
var lab ursexc '(D) UR sextant - LOA 6 mm or greater'.
val lab ursexc
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LLSEXC - LOA6mm or greater.
recode LOALL (1,2=2) (3,4=1) (9=9) (else=copy) into LLSEXC.
var lab llsexc '(D) LL sextant - LOA 6 mm or greater'.
val lab llsexc
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LCSEXC - LOA6mm or greater.
recode LOALC (1,2=2) (3,4=1) (9=9) (else=copy) into LCSEXC.
var lab lcsexc '(D) LC sextant - LOA 6 mm or greater'.
val lab lcsexc
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LRSEXC - LOA6mm or greater.
recode LOALR (1,2=2) (3,4=1) (9=9) (else=copy) into LRSEXC.
var lab lrsexc '(D) LR sextant - LOA 6 mm or greater'.
val lab lrsexc
  1 'Yes'
  2 'No'
  9 'Unscorable'.

```

NUMSEXC (D) Number of sextants with LOA of 6mm or greater

```

SPSS syntax
compute numsexc=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=2 & TeeLC=2 & TeeLR=2) | dvage<55.
compute numsexc=-1.
else.
  count numsexc = ulsexc ucsexc ursexc llsexc lcsexc lrsexc (1).
end if.
var lab numsexc '(D) Number of sextants with LOA of 6mm or greater'.

```

ULSEXD etc (D) UL sextant etc - pocketing 4 mm or greater

```

1 Yes
2 No
9 Unscorable

```

```

SPSS syntax
*ULSEXD - pocketing 4mm or greater.
recode PcketUL (1=2) (2,3,4=1) (9=9) (else=copy) into ULSEXD.
var lab ulsexd '(D) UL sextant - pocketing 4 mm or greater'.
val lab ulsexd
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*UCSEXD - pocketing 4mm or greater.
recode PcketUC (1=2) (2,3,4=1) (9=9) (else=copy) into UCSEXD.
var lab ucsexd '(D) UC sextant - pocketing 4 mm or greater'.
val lab ucsexd
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*URSEXD - pocketing 4mm or greater.
recode PcketUR (1=2) (2,3,4=1) (9=9) (else=copy) into URSEXD.
var lab ursexd '(D) UR sextant - pocketing 4 mm or greater'.
val lab ursexd
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LLSEXD - pocketing 4mm or greater.
recode PcketLL (1=2) (2,3,4=1) (9=9) (else=copy) into LLSEXD.
var lab llsexd '(D) LL sextant - pocketing 4 mm or greater'.

```

```

val lab llsexd
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LCSEXD - pocketing 4mm or greater.
recode PcketLC (1=2) (2,3,4=1) (9=9) (else=copy) into LCSEXD.
var lab lcsexd '(D) LC sextant - pocketing 4 mm or greater'.
val lab lcsexd
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LRSEXD - pocketing 4mm or greater.
recode PcketLR (1=2) (2,3,4=1) (9=9) (else=copy) into LRSEXD.
var lab lrsexd '(D) LR sextant - pocketing 4 mm or greater'.
val lab lrsexd
  1 'Yes'
  2 'No'
  9 'Unscorable'.

```

NUMSEXD (D) Number of sextants with pocketing of 4mm or greater

```

SPSS syntax
  compute numsexd=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=2 & TeeLC=2 & TeeLR=2).
  compute numsexd=-1.
else.
  count numsexd = ulsexd ucsexd ursexd llsexd lcsexd lrsexd (1).
end if.
var lab numsexd '(D) Number of sextants with pocketing of 4mm or greater'.

```

ULSEXE etc (D) UL sextant etc - pocketing 6mm or greater

```

1 Yes
2 No
9 Unscorable

```

```

SPSS syntax
*ULSEXE - pocketing 6mm or greater.
recode PcketUL (1,2=2) (3,4=1) (9=9) (else=copy) into ULSEXE.
var lab ulsexex '(D) UL sextant - pocketing 6 mm or greater'.
val lab ulsexex
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*UCSEXE - pocketing 6mm or greater.
recode PcketUC (1,2=2) (3,4=1) (9=9) (else=copy) into UCSEXE.
var lab ucsexex '(D) UC sextant - pocketing 6 mm or greater'.
val lab ucsexex
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*URSEXE - pocketing 6mm or greater.
recode PcketUR (1,2=2) (3,4=1) (9=9) (else=copy) into URSEXE.
var lab ursexex '(D) UR sextant - pocketing 6 mm or greater'.
val lab ursexex
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LLSEXE - pocketing 6mm or greater.
recode PcketLL (1,2=2) (3,4=1) (9=9) (else=copy) into LLSEXE.
var lab llsexex '(D) LL sextant - pocketing 6 mm or greater'.
val lab llsexex
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LCSEXE - pocketing 6mm or greater.
recode PcketLC (1,2=2) (3,4=1) (9=9) (else=copy) into LCSEXE.
var lab lcsexex '(D) LC sextant - pocketing 6 mm or greater'.
val lab lcsexex
  1 'Yes'
  2 'No'
  9 'Unscorable'.

*LRSEXE - pocketing 6mm or greater.
recode PcketLR (1,2=2) (3,4=1) (9=9) (else=copy) into LRSEXE.
var lab lrsexex '(D) LR sextant - pocketing 6 mm or greater'.
val lab lrsexex
  1 'Yes'

```

```
2 'No'
9 'Unscorable'.
```

NUMSEXE (D) Number of sextants with pocketing of 6mm or greater

```
SPSS syntax
  compute numsexe=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=2 & TeeLC=2 & TeeLR=2).
  compute numsexe=-1.
else.
  count numsexe = ulsexe ucsexe ursexe llsexe lcsexe lrsexe (1).
end if.
var lab numsexe '(D) Number of sextants with pocketing of 6mm or greater'.
```

NUMSEXF (D) Number of missing sextants

```
SPSS syntax
  compute numsexf=-999.
do if any(exout,2,3).
  compute numsexf=-1.
else.
  count numsexf = TeeUR TeeUC TeeUL TeeLL TeeLC TeeLR (2).
end if.
var lab numsexf '(D) Number of missing sextants'.
```

NUMSEX (D) Number of sextants

```
SPSS syntax
  compute numsex=-999.
do if any(exout,2,3).
  compute numsex=-1.
else.
  compute numsex = 6 - numsexf.
end if.
var lab numsex '(D) Number of sextants'.
```

NUMSCALC (D) Number of sextants with calculus present

```
SPSS syntax
  compute numscalc=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=2 & TeeLC=2 & TeeLR=2).
  compute numscalc=-1.
else.
  count numscalc = calcur to calclr (2).
end if.
var lab numscalc '(D) Number of sextants with calculus present'.
```

NUMSBLD (D) Number of sextants with bleeding present

```
SPSS syntax
  compute numsbld=-999.
do if any(exout,2,3) | (TeeUR=2 & TeeUC=2 & TeeUL=2 & TeeLL=2 & TeeLC=2 & TeeLR=2).
  compute numsbld=-1.
else.
  count numsbld = bleedur to bleedlr (2).
end if.
var lab numsbld '(D) Number of sextants with bleeding present'.
```

CALCULUS (D) Calculus present

- 1 Calculus
- 2 No calculus
- 9 All sextants unscorable

```
SPSS syntax
recode numscalc (0=2) (1 thru hi=1) (else=copy) into calculus.
count yyy = calcur to calclr (1).
if yyy=0 & calculus=2 calculus=9.
if numsex=0 calculus=-1.
if calcur=-8 & calcuc=-8 & calcul=-8 & calcll=-8 & calclc=-8 & calclr=-8 calculus=-8.
var lab calculus '(D) Calculus present'.
val lab calculus
  1 'Calculus'
  2 'No calculus'
  9 'All sextants unscorable'.
```

HASCALC (D) Calculus present

- 1 Calculus
- 2 No calculus

```
SPSS syntax
recode numscal (0=2) (1 thru hi=1) (else=copy) into hascalc.
count yyy = calcur calcuc calcul calccl calcclc calcclr (1).
if yyy=0 & hascalc=2 hascalc=-8.
if numsex=0 hascalc=-1.
if calcur=-8 & calcuc=-8 & calcul=-8 & calccl=-8 & calcclc=-8 & calcclr=-8 hascalc=-8.
var lab hascalc "(D) Calculus present".
val lab hascalc
  1 "Calculus"
  2 "No calculus".
```

HASBLD (D) Any bleeding

- 1 Yes
- 2 No

```
SPSS syntax
recode numsbld (0=2) (1 thru hi=1) (else=copy) into hasbld.
if numsex=0 hasbld=-1.
if bleedur=-8 & bleeduc=-8 & bleedul=-8 & bleedll=-8 & bleedlc=-8 & bleedlr=-8 hasbld=-8.
val lab hasbld
  1 "Yes"
  2 "No".
var lab hasbld "(D) Any bleeding".
```

PCDALL (D) Pocket depth - whole mouth

- 1 Up to 3.5mm
- 2 4-5.5mm
- 3 6-8.5mm
- 4 9mm and over
- 9 All sextants unscorable

```
SPSS syntax
do repeat xxx = PcketUR PcketUC PcketUL PcketLL PcketLC PcketLR
    /yyy = PkUR PkUC PkUL PkLL PkLC PkLR.
compute yyy = xxx.
if xxx = 9 yyy = -2.
end repeat.
compute pcdall=max(PkUR, PkUC, PkUL, PkLL, PkLC, PkLR).
if pcdall = -2 | pcdall=-1 pcdall = 9.
if any(exout,2,3) | numsex=0 pcdall=-1.
var lab pcdall '(D) Pocket depth - whole mouth'.
val lab pcdall
  1 'Up to 3.5mm'
  2 '4-5.5mm'
  3 '6-8.5mm'
  4 '9mm and over'
  9 'All sextants unscorable'.
```

PCD4 (D) Any pocketing 4mm or deeper

- 1 Yes
- 2 No
- 9 All sextants unscorable

```
SPSS syntax
recode numsexd (0=2) (1 thru hi=1) (else=copy) into pcd4.
count yyy = PcketUR to PcketLR (1).
if yyy=0 & pcd4=2 pcd4=9.
if numsex=0 pcd4=-1.
if pcketur=-8 & pcketuc=-8 & pcketul=-8 & pcketll=-8 & pcketlc=-8 & pcketlr=-8 pcd4=-8.
var lab pcd4 '(D) Any pocketing 4mm or deeper'.
val lab pcd4
  1 'Yes'
  2 'No'
  9 'All sextants unscorable'.
```

PCD6 (D) Any pocketing 6mm or deeper

- 1 Yes
- 2 No
- 9 All sextants unscorable

```
SPSS syntax
recode numsex (0=2) (1 thru hi=1) (else=copy) into pcd6.
count yyy = PcketUR to PcketLR (1, 2).
if yyy=0 & pcd6=2 pcd6=9.
if numsex=0 pcd6=-1.
if pcketur=-8 & pcketuc=-8 & pcketul=-8 & pcketll=-8 & pcketlc=-8 & pcketlr=-8 pcd6=-8.
var lab pcd6 '(D) Any pocketing 6mm or deeper'.
val lab pcd6
  1 'Yes'
  2 'No'
  9 'All sextants unscorable'.
```

PCD6DV (D) pocket depth 6mm or more, no unscored sextants

- 1 Yes
- 2 No

```
SPSS syntax
recode pcd6 (9=sysmis) (else=copy) into pcd6DV.
var lab pcd6 '(D) pocket depth 6mm or more, no unscored sextants'.
val lab pcd6
  1 'Yes'
  2 'No'.
```

PCD9 (D) Any pocketing 9mm or deeper

- 1 Yes
- 2 No
- 9 All sextants unscorable

SPSS syntax

```
compute pcd9=-999.
do if any(exout,2,3).
  compute pcd9=-1.
else.
  count pcd9 = PcketUR PcketUC PcketUL PcketLL PcketLC PcketLR (4).
  recode pcd9 (0=2) (1 thru hi=1).
  count yyy = PcketUR PcketUC PcketUL PcketLL PcketLC PcketLR (1,2,3).
  if yyy=0 & pcd9=2 pcd9=9.
  if pcketur=-8 & pcketuc=-8 & pcketul=-8 & pcketll=-8 & pcketlc=-8 & pcketlr=-8 pcd9=-8.
  if numsex=0 pcd9=-1.
end if.
var lab pcd9 "(D) Any pocketing 9mm or deeper".
val lab pcd9
  1 "Yes"
  2 "No"
  9 "All sextants unscorable".
```

LOAALL (D) Loss of attachment - whole mouth

- 1 Up to 3.5mm
- 2 4-5.5mm
- 3 6-8.5mm
- 4 9mm and over
- 9 All sextants unscorable

SPSS syntax

```
do repeat xxx = LOAUR LOAUC LOAUL LOALL LOALC LOALR
  /yyy = LUR LUC LUL LLL LLC LLR.
  compute yyy = xxx.
  if xxx = 9 yyy = -2.
end repeat.
compute loaall=max(LUR, LUC, LUL, LLL, LLC, LLR).
if loaall = -2 | loaall=-1 loaall = 9.
if any(exout,2,3) | numsex=0 | dvage<55 loaall=-1.
var lab loaall "(D) Loss of attachment - whole mouth".
val lab loaall
  1 'Up to 3.5mm'
  2 '4-5.5mm'
  3 '6-8.5mm'
  4 '9mm and over'
  9 'All sextants unscorable'.
```

LOATT4 (D) Any LOA 4mm or deeper

- 1 Yes
- 2 No
- 9 All sextants unscorable

SPSS syntax

```
recode numsexb (0=2) (1 thru hi=1) (else=copy) into LOAtt4.
count yyy = LOAUR to LOALR (1).
if yyy=0 & loatt4=2 loatt4=9.
if loaur=-8 & loauc=-8 & loaul=-8 & loall=-8 & loalc=-8 & loalr=-8 loatt4=-8.
if numsex=0 loatt4=-1.
var lab loatt4 "(D) Any LOA 4mm or deeper".
val lab loatt4
  1 'Yes'
  2 'No'
  9 'All sextants unscorable'.
```

LOATT6 (D) Any LOA 6mm or deeper

- 1 Yes
- 2 No
- 9 All sextants unscorable

```
SPSS syntax
recode numsexc (0=2) (1 thru hi=1) (else=copy) into LOAtt6.
count yyy = LOAUR to LOALR (1, 2).
if yyy=0 & loatt6=2 loatt6=9.
if loaur=-8 & loauc=-8 & loaul=-8 & loall=-8 & loalc=-8 & loalr=-8 loatt6=-8.
if numsex=0 loatt6=-1.
var lab loatt6 '(D) Any LOA 6mm or deeper'.
val lab loatt6
  1 'Yes'
  2 'No'
  9 'All sextants unscorable'.
```

LOATT9 (D) Any LOA 9mm or deeper

- 1 Yes
- 2 No
- 9 All sextants unscorable

```
SPSS syntax
compute LOAtt9=-999.
do if any(exout,2,3) | dvage<55.
  compute LOAtt9=-1.
else.
  count LOAtt9= LOAUR to LOALR (4).
  recode LOAtt9 (0=2) (1 thru hi=1).
  count yyy = LOAUR to LOALR (1,2,3).
  if yyy=0 & loatt9=2 loatt9=9.
  if loaur=-8 & loauc=-8 & loaul=-8 & loall=-8 & loalc=-8 & loalr=-8 loatt9=-8.
  if numsex=0 loatt9=-1.
end if.
var lab loatt9 '(D) Any LOA 9mm or deeper'.
val lab loatt9
  1 'Yes'
  2 'No'
  9 'All sextants unscorable'.
```

GUMHLTHY (D) Healthy gums - no pocketing or LoA

- 1 No pocketing or LoA
- 2 Pocketing or LoA
- 9 All sextants unscorable

```
SPSS syntax
recode pcdall (1=1) (2 thru 4=2) (else=copy) into gumhlthy.
if range(loaall,2,4) gumhlthy=2.
val lab gumhlthy
  1 'No pocketing or LoA'
  2 'Pocketing or LoA'
  9 'All sextants unscorable'.
var lab gumhlthy '(D) Healthy gums - no pocketing or LoA'.
```

GUMHLTG4 (D) Healthy gums - four categories

- 1 No pocketing, LoA, calculus or bleeding
- 2 Bleeding, but no pocketing, LoA or calculus
- 3 Calculus (with or without bleeding), but no pocketing or LoA
- 4 Some pocketing or LoA
- 9 All sextants unscorable

```
SPSS syntax
recode gumhlthy (2=4) (else=copy) into gumhltg4.
if hasbld=1 & gumhltg4<>4 gumhltg4=2.
if calculus=1 & gumhltg4<>4 gumhltg4=3.
val lab gumhltg4
  1 "No pocketing, LoA, calculus or bleeding"
  2 "Bleeding, but no pocketing, LoA or calculus"
  3 "Calculus (with or without bleeding), but no pocketing or LoA"
  4 "Some pocketing or LoA"
  9 "All sextants unscorable".
var lab gumhltg4 "(D) Healthy gums - four categories".
```

DHEALTH (D) 21+ teeth & 18+ SUT (incl. roots) & perio health

- 1 Yes
- 2 No

```
SPSS syntax
recode has21 (lo thru hi=copy) into dhealth.
if hassutr=2 dhealth=2.
if gumhlthy=2 dhealth=2.
if gumhlthy=-9 dhealth=-8.
if gumhlthy<0 & dhealth<>-1 dhealth=-9.
val lab dhealth
1 "Yes"
2 "No".
var lab dhealth '(D) 21+ teeth & 18+ SUT (incl. roots) & perio health'.
```

DHEALTH2 (D) 21+ teeth & 18+ SUT (incl. roots) & perio health and no active decay

- 1 Yes
- 2 No

```
SPSS syntax
recode dhealth (lo thru hi=copy) into dhealth2.
if hastdecr=1 dhealth2=2.
if gumhlthg4=2 dhealth2=2.
if gumhlthg4=3 dhealth2=2.
val lab dhealth2
1 "Yes"
2 "No"
-1 "Item not applicable"
-8 "Don't know"
-9 "No answer/refused".
var lab dhealth2 '(D) 21+ teeth & 18+ SUT (incl. roots) & perio health and no active decay'.
```

DHLTHG4 (D) 21+ teeth & 18+ SUT (incl. roots) & perio health in 4 cats

```
SPSS syntax
recode dhealth (2=4) (else=copy) into dhlthg4.
if gumhlthg4=2 & dhlthg4<>4 dhlthg4=2.
if gumhlthg4=3 & dhlthg4<>4 dhlthg4=3.
val lab dhlthg4
1 "Yes, including all perio healthy"
2 "Yes, bleeding, otherwise perio healthy"
3 "Yes, calculus, otherwise perio healthy"
4 "No".
var lab dhlthg4 "(D) 21+ teeth & 18+ SUT (incl. roots) & perio health in 4 cats".
```

DHLTHG3 (D) 21+ teeth & 18+ SUT (incl. roots) & perio health in 3 cats

```
SPSS syntax
recode dhlthg3 (3=2) (else=copy) into dhlthg3.
val lab dhlthg4
1 "Yes, including all perio healthy"
2 "Yes, bleeding or calculus, otherwise perio healthy"
3 "No".
var lab dhlthg3 "(D) 21+ teeth & 18+ SUT (incl. roots) & perio health in 3 cats".
```

Complexity scale

COMPLEX1 (D) Restored surfaces in top quintile.

- 0 No
- 1 Yes

```
SPSS syntax
compute complex1=0.
if numsresr>=32 complex1=1.
if numsresr<0 complex1=-1.
val lab complex1 0 'No' 1 'Yes'.
var lab complex1 '(D) Restored surfaces in top quintile'.
```

COMPLEX2 (D) Three or more crowns.

0 No
1 Yes

```
SPSS syntax
compute complex2=0.
if numcrown>=3 complex2=1.
if numcrown<0 complex2=-1 .
val lab complex1 0 'No' 1 'Yes'.
var lab complex2 '(D) Three or more crowns'.
```

COMPLEX 3 (D) Any denture, bridge, implant.

0 No
1 Yes

```
SPSS syntax
compute complex3=0.
if denture=1 complex3=1.
if hasbrid=1 complex3=1 .
if hasimp=1 complex3=1 .
if eweight<0 complex3=-1 .
val lab complex1 0 'No' 1 'Yes'.
var lab complex3 '(D) Any denture, bridge, implant' .
```

COMPLEX 4 (D) Any active decay.

0 No
1 Yes

```
SPSS syntax
count xxx=RootUR8 to RootLR8 (13,14,51,61).
compute complex4=0.
if hastdec=1 complex4=1.
if hasunrst=1 complex4=1 .
if xxx>0 complex4=1.
if hastdec<0 complex4=-1 .
val lab complex1 0 'No' 1 'Yes'.
var lab complex4 '(D) Any active decay' .
```

COMPLEX 5 (D) Active decay: top quintile of those with decay.

0 No
1 Yes

```
SPSS syntax
count xxx=DUR8 to LUR8 DUR7 to LUR7 DUR6 to LUR6 DUR5 to LUR5 DUR4 to LUR4 DUR3 to LUR3 DUR2 to LUR2 DUR1
to LUR1
DUL1 to LUL1 DUL2 to LUL2 DUL3 to LUL3 DUL4 to LUL4 DUL5 to LUL5 DUL6 to LUL6 DUL7 to LUL7 DUL8 to LUL8
DLL8 to LLL8 DLL7 to LLL7 DLL6 to LLL6 DLL5 to LLL5 DLL4 to LLL4 DLL3 to LLL3 DLL2 to LLL2 DLL1 to LLL1
DLR1 to LLR1 DLR2 to LLR2 DLR3 to LLR3 DLR4 to LLR4 DLR5 to LLR5 DLR6 to LLR6 DLR7 to LLR7 DLR8 to LLR8
(1,2,3,41,42,51,52,61,62,71,72,81,82) .

count yyy=RootUR8 to RootLR8 (13,14,51,61).

compute aaaa=xxx+yyy.
if PLUR8<0 aaaa=-1.

*top quintile of active decay = 8 .
compute complex5=0.
if aaaa>=8 complex5=1.
if aaaa<0 complex5=-1 .
val lab complex1 0 'No' 1 'Yes'.
var lab complex5 '(D) Active decay: top quintile of those with decay'.
```

COMPLEX 6 (D) Pocketing 6mm+ or LoA 9mm+.

0 No
1 Yes

```
SPSS syntax
compute complex6=0.
if pcd6=1 complex6=1.
if loatt9=1 complex6=1 .
if pcdall<0 complex6=-1 .
val lab complex1 0 'No' 1 'Yes'.
```

```
var lab complex6 '(D) Pocketing 6mm+ or LoA 9mm+'.
```

COMPLEX 7 (D) PUFA score or unrestorable tooth.

0 No
1 Yes

```
SPSS syntax
compute complex7=0.
if haspufa=1 complex7=1.
if numunrst>0 complex7=1 .
if numrurs>0 complex7=1 .
if haspufa<0 complex7=-1 .
val lab complex1 0 'No' 1 'Yes'.
var lab complex7 '(D) PUFA score or unrestorable tooth'.
```

COMPLEX 8 (D) OHIP: fairly or very often.

0 No
1 Yes

```
SPSS syntax
compute complex8=0.
if hiohip>3 complex8=1.
if hiohip<0 complex8=-1 .
val lab complex1 0 'No' 1 'Yes'.
var lab complex8 '(D) OHIP: fairly or very often'.
```

TOTCOMP (D) Complexity score.

```
SPSS syntax
compute totcomp=0.
if complex1=1 totcomp= totcomp+1.
if complex2=1 totcomp= totcomp+1.
if complex3=1 totcomp= totcomp+1.
if complex4=1 totcomp= totcomp+1.
if complex5=1 totcomp= totcomp+1.
if complex6=1 totcomp= totcomp+1.
if complex7=1 totcomp= totcomp+1.
if complex8=1 totcomp= totcomp+1.
if (eweight<=0) totcomp=-1.
var lab totcomp '(D) Complexity score' .
```

TOTCOMPG (D) Complexity score with 5+ grouped.

```
SPSS syntax
recode totcomp (5 thru hi=5) (else=copy) into totcompg .
var lab totcompg '(D) Complexity score with 5+ grouped' .
val lab totcompg
0 '0'
1 '1'
2 '2'
3 '3'
4 '4'
5 '5+' .
```

Interview

Dental status

Dentate Natural teeth or edentate

1. Edentate
2. Dentate

SPSS syntax

```
recode natural (1=1) (2 thru hi=2) (else=copy) into dentate.  
variable labels dentate 'dental status'  
value labels dentate 1 'edentate' 2 'dentate' -9 'No answer/refused' -8 'Don't know' -1 'not applicable'.  
missing values dentate (-9 thru -1).  
fre vars = natural dentate.
```

DVDent Natural teeth and use of dentures

1. Edentate, no dentures
2. Edentate, has dentures
3. Dentate, has dentures
4. Dentate, no dentures

SPSS syntax

```
*****Computation of DVDent - dental status - natural teeth and use of dentures.  
  
Do if (Natural=1 AND Denture=2).  
Compute DVDENT=1.  
Else if (Natural=1 AND Denture=1).  
Compute DVDENT=2.  
Else if (Natural>1 AND Denture=1).  
Compute DVDENT=3.  
Else if (Natural>1 AND Denture=2).  
Compute DVDENT=4.  
End if.  
variable labels DVDENT 'dental status'.  
value labels DVDENT  
1 'Edentate, no dentures'  
2 'Edentate, has dentures'  
3 'Dentate, has dentures'  
4 'Dentate, no dentures'.
```

DVDent2 Natural teeth and use of dentures 2

1. Edentate
2. Dentate, has dentures
3. Dentate, no dentures

SPSS syntax

```
*****Computation of DVDent2 - dental status - natural teeth and use of dentures.  
  
Do if (Natural=1).  
Compute DVDENT2=1.  
Else if (Natural>1 AND Denture=1).  
Compute DVDENT2=2.  
Else if (Natural>1 AND Denture=2).  
Compute DVDENT2=3.  
End if.  
variable labels DVDENT2 'dental status'.  
value labels DVDENT2  
1 'Edentate'  
2 'Dentate, has dentures'  
3 'Dentate, no dentures'.
```

Dental hygiene

ClntthG3 Number of times brush teeth per day – grouped

1. Twice a day or more
2. Once a day
3. Never, less than once a day

ClntthG4 Number of times brush teeth per day – grouped

1. Twice a day or more
2. Once a day or less
3. Less than once a day
4. Never

SPSS syntax

```
recode Clntth (1,2=1) (3=2) (4,5=3) (6,7,8,9=3) (94,95=-7) (else=copy) into ClntthG3.
variable labels ClntthG3 'number of times brush teeth per day - 3 groups'.
value labels ClntthG3
1 'twice a day or more'
2 'once a day'
3 'never, less than once a day'
-7 'other'
-8 'dont know'
-1 ' item not applicable' .

recode Clntth (1,2=1) (3=2) (4=3) (5=4) (6,7,8,9=3) (94,95=-7) (else=copy) into ClntthG4.
variable labels ClntthG4 'number of times brush teeth per day - 4 groups'.
value labels ClntthG4
1 'twice a day or more'
2 'once a day'
3 'less than once a day'
4 'never'
-7 'other'
-8 'dont know'
-1 ' item not applicable'.

missing values ClntthG2 ClntthG3 ClntthG4 (-8, -1, -7).
```

Fluorid1 (D) grouped fluoride levels

1. 1350-1500ppm
2. 1000-1350 ppm'
3. 550 ppm or less or no fluoride
5. cant code/none of these brands

SPSS syntax

```
recode fluoride (1=1) (2=2) (3,4=3) (5,999=5) (else=copy) into fluorid1.
variable labels fluorid1 'grouped fluoride levels'.
value labels fluorid1
1 '1350-1500ppm'
2 '1000-1350 ppm'
3 '550 ppm or less or no fluoride'
5 'cant code/none of these brands'
-8 'don't know'
-1 'item not applicable'.

missing values fluorid1 (-8, -1, -7).
```

Smoking

DVSmoke Smoking status

1. Currently smoke
2. Use to smoke
3. Never smoked

SPSS syntax

```
missing values cignow smokever ().
execute.

Do if (SmokEver=-8 and cignow=-1) .
Compute DVSMOKE=-8.
Else if (SmokEver=-9 and cignow=-1) .
Compute DVSMOKE=-9.
Else if (SmokEver=2).
Compute DVSMOKE=3.
Else if (CigNow=1).
Compute DVSMOKE=1.
Else if (CigNow=2 AND SmokEver=1) .
Compute DVSMOKE=2.
Else .
Compute DVSMOKE=4.
end if.
Variable labels DVSMOKE 'smoking status'.
Value labels DVSMOKE 1 'Currently Smoke' 2 'Used to Smoke' 3 'Never smoked' 4 'error-investigate'
-8 'dont know' -1 ' item not applicable' -9 'no answer/refused'.
execute.

missing values cignow smokever DVSMOKE (-8,-1,-9).
```

Diet

EvrDnAd1 Ever given advice about diet - all people

1. Yes
2. No
3. Not asked about diet

SPSS syntax

```
missing values evrdiet evrdnad ().
if EvrDnAd=1 EvrDnAd1=1.
if EvrDnAd=2 EvrDnAd1=2.
if EvrDnAd=-8 EvrDnAd1=-8.
if EvrDiet=-1 EvrDnAd1=-1.
if EvrDiet=-8 EvrDnAd1=-8.
if EvrDiet=2 EvrDnAd1=3.

variable labels EvrDnAd1 'ever given advice about diet - all people'.
value labels EvrDnAd1 1 'yes' 2 'no' 3 'not asked about diet' -8 'dont know' -1 'not applicable'.
execute.
missing values evrdiet evrdnad EvrDnAd1(-1,-8).
```

Highsug5 High sugar intake

1. high sugar intake
2. not high

SPSS syntax

```
missing values ncakes sweets sofdrnk ().
execute.

do if (ncakes=-9) or (sweets = -9) or (sofdrnk =-9).
compute highsug5=-9.
else if (ncakes=-8) or (sweets = -8) or (sofdrnk =-8).
compute highsug5=-8.
else if (ncakes=1) or (sweets=1) or (sofdrnk =1).
compute highsug5=1.
else .
compute highsug5=2.
end if.

variable labels highsug5 'high sugar intake - version 5'.
value lables highsug5 1 'high sugar intake' 2 'not high '.
missing values ncakes sweets sofdrnk highsug5 (-9,-8).
```

Dentist attendance patterns

HowLongG Time since last dental visit

1. Less than 1 year
2. Between 1 and 5 years
3. Over 5 up to 10 years
4. Over 10 years

SPSS syntax

```
*****Computation of HowLongG - how long since visited dentist

recode HowLong (1,2=1) (3,4,5=2) (6=3) (7=4) (else=copy) into HowLongG.

Value labels HowLongG
1 'less than 1 year'
2 'between 1 and 5 years'
3 'over 5 up to 10 years'
4 'over 10 years'
-1 'item not applicable'
-8 'dont know'
Variable labels HowLongG 'how long since visited dentist - grouped'.

missing values HowlongG (-1, -8).
```

NumTrouG Number of times visited dentist with trouble – grouped

1. 1
2. 2
3. 3
4. 4 or more

SPSS syntax

```
*****Computation of NumTrouG - how how many times visited dentist with trouble - grouped

recode NumTroub (1=1) (2=2) (3=3) (4 thru hi=4) (else=copy) into NumTrouG.

Value labels NumTrouG
1 '1'
2 '2'
3 '3'
4 '4 or more'
-1 'item not applicable'
-8 'dont know'
Variable labels NumTrouG 'how many visits to dentist with trouble- grouped'.

missing values NumTrouG (-1, -8).
```

DVChkUp Reason for last visit – 3 categories

1. Routine checkup
2. Emergency treatment
3. other treatment

SPSS syntax

```
RECODE CheckUp (1 =1 ) (2=2) (3,5, 94,95=3) (Else=copy) INTO DVChkUp .
VARIABLE LABELS DVChkUp 'Reason for last visit - 3 categories'.
VALUE LABELS DVChkUp 1 'routine checkup' 2 'emergency treatment' 3 'other treatment'
-7 'other' -8 'dont know' -1 'item not applicable'.
EXECUTE .
```

FreqDenG Frequency of dentist visits – grouped

1. At least every 6 or 12 months
2. Less freq than 12 months

SPSS syntax

```
RECODE FreqDen (1 thru 2=1) (3,4,5=2) (Else=copy) INTO FreqDenG .
```

```
VARIABLE LABELS FreqDenG 'frequency of dental visits grouped'.  
VALUE LABELS FreqDenG 1 'at least every 6 or 12 months' 2 'less freq than every 12 months'  
-7 'other' -8 'dont know' -1 'item not applicable'.  
EXECUTE .
```

FreqDnG1 Frequency of dentist visits – grouped 1

1. At least every two years
2. Less freq than every 2 years

```
SPSS syntax  
RECODE FreqDen (1 thru 3=1) (4,5=2) (Else=copy) INTO FreqDnG1 .  
VARIABLE LABELS FreqDnG1 'frequency of dental visits grouped 1'.  
VALUE LABELS FreqDnG1 1 'at least every 2 years' 2 'less freq than every 2 years'  
-7 'other' -8 'dont know' -1 'item not applicable'.  
EXECUTE .
```

HowLngG2 Last visit to dentist – grouped 2

1. Less than two years
- 2 More than 2 years

```
SPSS syntax  
  
RECODE  
  HowLong (1 thru 3=1) (4 thru 7=2) (else=copy) INTO HowLngG2 .  
VARIABLE LABELS HowLngG2 'last visit to dentist was less than 2 years ago'.  
VALUE LABELS HowLngG2 1 'less than 2 years' 2 'more than 2 years' -7 'other' -8 'dont know' -1 'item not  
applicable'.  
EXECUTE .
```

Dental services

SerTypGrp Service type – grouped

1. Private
2. NHS paid for
3. NHS free
4. NHS followed by private dental care
5. Other
6. Not sure

SPSS syntax

***** other category now also includes dental hospital, treated abroad, vague/irrelevant and other specific.

```
RECODE
  SerTyp
    (1=Copy) (2=Copy) (3=Copy) (4=Copy) (5=Copy) (6=Copy) (7=5) (8=5) (94=5) (95=5) INTO
  Sertypgrp .
VARIABLE LABELS Sertypg 'service type grouped'.
EXECUTE .
```

SerTypNHS Paid for and free NHS combined

1. not NHS paid/free
2. NHS paid/free composite

SPSS syntax

```
RECODE
  SerTyp (1=1) (2=2) (3=2) (4=1) (5=1) (6=1) (7=1) INTO SerTypNHS .
VARIABLE LABELS SerTypNHS 'Paid for and free NHS'.
VALUE LABELS SerTypNHS 1 'not NHS paid/free' 2 'NHS paid/free composite'.
EXECUTE .
```

DVSerTyp Service type for Wales/NI tables

1. Private
2. NHS paid for
3. NHS free
4. NHS and private dental care
5. Not sure
6. Other

SPSS syntax

***** other category now also includes dental hospital, treated abroad, vague/irrelevant and other specific.

```
RECODE
  SerTyp (1=Copy) (2=Copy) (3=Copy) (4=Copy) (5=6) (6=5) (7=6) (8=6) (94=6) (95=6) INTO
  DVSerTyp .
VARIABLE LABELS DVSerTyp 'Service type for Wales/NI tables '.
EXECUTE .
```

Cost

Topcoded at £1001 for non-disclosure purposes

SPSS syntax

```
RECODE
  Cost (1001 thru Highest=1001) .
EXECUTE .
```

Costings Cost of treatment- grouped

1. Zero
2. Less than £20
3. £21 to £50
4. £51 to £100
5. £101 to £200
6. £201 to £500
7. £501 +

SPSS syntax

```
RECODE
  Cost
  (0=1) (1 thru 20=2) (21 thru 50=3) (51 thru 100=4) (101 thru 200=5) (201 thru 500=6) (501 thru
Highest=7) INTO
  costings .
EXECUTE .
VARIABLE LABELS Costings 'grouped cost of last dental treatment in pounds'.
VALUE LABELS Costings 1 '0' 2 '<20' 3 '21-50' 4 '51-100' 5 '101-200' 6 '201-500' 7 '501+' -7 'other' -8
'dont know' -1 'item not applicable'.
EXECUTE .
```

DenPlanD Insurance composite

1. yes
2. no
3. don't know

SPSS syntax

```
***DenPlan yes/no DenPlanD*****.
RECODE DenPlan (1=1) (2=1) (3=1) (4=2) (5=3) INTO DenPlanD .
VARIABLE LABELS DenPlanD 'Insurance composite'.
VALUE LABELS DenPlanD 1 'Yes' 2 'No' 3 'don't know'.
EXECUTE .
```

Dental treatment received

Dtotimp If ever had an implant – upper or lower

1. yes
2. no

SPSS syntax

```
DO IF (EvrImpU=1and EvrImpL=1) or (EvrImpU=1 and EvrImpL=2) or (EvrImpU=2
and EvrImpL=1) .
COMPUTE dtotimp =1.
ELSE IF (EvrImpU=2 and EvrImpL=2) .
COMPUTE DTOTIMP =2.
END IF.
EXECUTE.
```

DVisits1 More or less than 1 visit

1. 1 visit
2. more than 1 visit

RECODE

```
DVisits
(1=1) (2 thru 5=2) INTO DVisits1 .
VARIABLE LABELS DVisits1 'More or less than 1 visit'.
VALUE LABELS DVisits1 1 'one visit' 2 'more than one visit'.
EXECUTE .
```

DenFitd Dentures fitted including not applicables

1. Yes
2. No
3. NAs and DKs

DenRepd Dentures repaired including not applicables

1. Yes
2. No
3. NAs and DKs

SPSS Syntax

```
*****Dentures fitted at last treatment (include dk's) - Denfitd*****.
RECODE
  DenFitt
  (1=1) (2=2) (-8=3) (-1=3) INTO Denfitd .
EXECUTE .

****Dentures repaired at last treatment (include dk's) - Denrepd****.
RECODE
  DenRepr
  (1=1) (2=2) (-8=3) (-1=3) INTO Denrepd .
EXECUTE .
VALUE LABELS Denfitd Denrepd 1 'yes' 2 'no' 3 'NAs and DKs'.
```

ValMonD Grouped value for money categories

1. Very good/good
2. Fair
3. Poor/very poor
4. Don't know

SPSS Syntax

```
*****ValMon grouped - ValMonD*****.
RECODE
  ValMon
  (3=2) (6=4) (1 thru 2=1) (4 thru 5=3) INTO ValMonD .
VARIABLE LABELS ValMonD 'Grouped values for money categories'.
```

```
VALUE LABELS ValMonD 1 'Very good/good' 2 'Fair' 3 'Poor/very poor' 4 'Don't know'.  
EXECUTE .
```

EvrDnAdd Ever had diet advice - Includes no and not applicables

1. yes
2. no NAs DKs

```
*****Diet advice including all 'no's from evrdiet*****.  
RECODE  
  EvrDnAd  
  (1=1) (2=2) (-9=2) (-8=2) (-7=2) (-6=2) (-2=2) (-1=2) INTO evrdsnadd .  
VARIABLE LABELS evrdsnadd 'Includes no and not applicables'.  
VALUE LABELS 1 'yes' 2 'no NAs DKs '  
EXECUTE .
```

Self-assessed health

GenHlthg Self-reported general health - grouped

- 1 Very good/good
- 2 Fair
- 3 Very bad/ bad

DenHlthg Self-reported dental health - grouped

- 1 Very good/good
- 2 Fair
- 3 Very bad/ bad

SPSS Syntax

```
Recode QHealth1 (1,2=1) (3=2) (4,5=3) (else=copy) into GenHlthg.  
Recode DenHeal (1,2=1) (3=2) (4,5=3) (else=copy) into DenHlthg.
```

```
variable labels  
  GenHlthg 'self-reported general health - grouped'  
  DenHlthg 'self-reported dental health - grouped'.
```

```
Value labels GenHlthg DenHlthg  
  1 'Very good/good'  
  2 'Fair'  
  3 'Very bad/ bad'  
 -8 'don't know'.
```

```
Missing values GenHlthg DenHlthg (-1, -8, -9).
```

OHIP – oral health-related quality of life

FuncLim Functional limitation –words taste

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

PhysPain Physical pain – aching foods

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

PyscDisc Psychological discomfort – self tense

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

PhysDisc Physical discomfort –diet meals

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

PsycDisb Psychological disability – relax embarass

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

SocDisb Social disability – irritabl jobs

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

Handicap Handicap – less functin

1. Never
2. Hardly ever
3. Occasionally
4. Fairly often
5. Very often

SPSS syntax

*****OHIP variables *****.

```
Do if (Words=5 or Taste=5) .
Compute FunctLim=5.
Else if (Words=4 or Taste=4) .
Compute FunctLim=4.
Else if (Words=3 or Taste=3) .
Compute FunctLim=3.
Else if (Words=2 or Taste=2) .
Compute FunctLim=2.
Else if (Words=1 or Taste=1).
Compute FunctLim=1.
Else if (words=-8 or taste=-8).
Compute functlm=-8.
end if.
```

```
Do if (Aching=5 or Foods=5) .
Compute PhysPain=5.
Else if (Aching=4 or Foods=4) .
Compute PhysPain=4.
Else if (Aching=3 or Foods=3) .
Compute PhysPain=3.
Else if (Aching=2 or Foods=2) .
Compute PhysPain=2.
Else if (Aching=1 or Foods=1).
Compute PhysPain=1.
Else if (Aching=-8 or Foods=-8).
Compute PhysPain=-8.
end if.
```

```
Do if (Self=5 or Tense=5) .
Compute PsycDisc=5.
Else if (Self=4 or Tense=4) .
Compute PsycDisc=4.
Else if (Self=3 or Tense=3) .
Compute PsycDisc=3.
Else if (Self=2 or Tense=2) .
Compute PsycDisc=2.
Else if (Self=1 or Tense=1).
Compute PsycDisc=1.
Else if (Self=-8 or Tense=-8).
Compute PsycDisc=-8.
end if.
```

```
Do if (Diet=5 or Meals=5) .
Compute PhysDisb=5.
Else if (Diet=4 or Meals=4) .
Compute PhysDisb=4.
Else if (Diet=3 or Meals=3) .
Compute PhysDisb=3.
Else if (Diet=2 or Meals=2) .
Compute PhysDisb=2.
Else if (Diet=1 or Meals=1).
Compute PhysDisb=1.
Else if (Diet=-8 or Meals=-8).
Compute PhysDisb=-8.
end if.
```

```
Do if (Relax=5 or Embarass=5) .
Compute PsycDisb=5.
Else if (Relax=4 or Embarass=4) .
Compute PsycDisb=4.
Else if (Relax=3 or Embarass=3) .
Compute PsycDisb=3.
Else if (Relax=2 or Embarass=2) .
Compute PsycDisb=2.
Else if (Relax=1 or Embarass=1).
Compute PsycDisb=1.
Else if (Relax=-8 or Embarass=-8).
Compute PsycDisb=-8.
end if.
```

```
Do if (Irritabl=5 or Jobs=5) .
Compute SocDisb=5.
Else if (Irritabl=4 or Jobs=4) .
Compute SocDisb=4.
Else if (Irritabl=3 or Jobs=3) .
Compute SocDisb=3.
Else if (Irritabl=2 or Jobs=2) .
Compute SocDisb=2.
Else if (Irritabl=1 or Jobs=1).
Compute SocDisb=1.
Else if (Irritabl=-8 or Jobs=-8).
Compute SocDisb=-8.
end if.
```

```

Do if (Less=5 or Functin=5) .
Compute Handicap=5.
Else if (Less=4 or Functin=4) .
Compute Handicap=4.
Else if (Less=3 or Functin=3) .
Compute Handicap=3.
Else if (Less=2 or Functin=2 ) .
Compute Handicap=2.
Else if (Less=1 or Functin=1).
Compute Handicap=1.
Else if (Less=-8 or Functin=-8).
Compute Handicap=-8.
end if.

Variable labels FunctLim 'Functional limitation - words taste'.
Variable labels PhysPain 'Physical pain - aching foods'.
Variable labels PsycDisc 'Psychological discomfort - self tense'.
Variable labels PhysDisb 'Physical disability - diet meals'.
Variable labels PsycDisb 'Psychological disability - relax embarass'.
Variable labels SocDisb 'Social disability - Irritabl Jobs'.
Variable labels Handicap 'Handicap - Less Functin'.

Value labels FunctLim PhysPain PsycDisc PhysDisb PsycDisb SocDisb Handicap
1 'Never'
2 'Hardly ever'
3 'Occasionally'
4 'Fairly often'
5 'very often'
-8 'dont know'.
missing values functlim PhysPain PsycDisc PhysDisb PsycDisb SocDisb Handicap (-8).

```

TotOHIP Total sum across 14 items

SPSS syntax

```

**** using x + y etc as opposed to Sum means if any = missing TotOHIP = missing *****.

do if (words=-8 or Taste=-8 or Aching=-8 or Foods=-8 or Self =-8 or Tense=-8 or Diet =-8 or
      Meals=-8 or Relax =-8 or Embarass=-8 or Irritabl=-8 or Jobs=-8 or Less=-8 or
      Functin=-8 ).
compute TotOHIP=-8.
else.
compute TotOHIP = Words + Taste + Aching + Foods + Self + Tense + Diet + Meals + Relax +
      Embarass + Irritabl + Jobs + Less + Functin.
end if.

fre vars = TotOHIP.
missing values Words Taste Aching Foods Self Tense Diet Meals Relax Embarass Irritabl Jobs Less Functin
TotOHIP(-8).

```

NumOHIP Number of problems on OHIP items

NumOHIPG Number of problems on OHIP items grouped

```

0 0
1 1
2 2-4
3 5 or more

```

SPSS syntax

```

do if (Words ge 1).
Count NumOHIP=Words Taste Aching Foods Self Tense Diet Meals Relax Embarass Irritabl Jobs Less Functin
(3,4,5).
else if (words=-8 or Taste=-8 or Aching=-8 or Foods=-8 or Self =-8 or Tense=-8 or Diet =-8 or Meals=-8
      or Relax =-8 or Embarass=-8 or Irritabl=-8 or Jobs=-8 or Less=-8 or Functin=-8 ).
compute NumOHIP=-8.
end if.
variable labels NumOHIP 'number of problems on 14 OHIP items'.
recode NumOHIP (0=0) (1=1) (2 thru 4=2) (5 thru 14=3) (-8=-8) into NumOHIPG.
variable labels NumOHIPG 'number of problems on 14 OHIP items grouped'.

```

```
value labels NumOHIPG
0 '0'
1 '1'
2 '2-4'
3 '5 or more'
-8 'don't know'.
```

WordDiff (D) OHIP Words – any difficulty reported

- 1 yes
- 2 no

SPSS syntax

```
recode Words (1=1) (2 thru hi=2) (else=copy) into WordDiff.
variable labels WordDiff '(D) OHIP Words- any difficulty reported'.
value labels WordDiff 1 'yes' 2 'no'.
```

As above for TastDiff AcheDiff FoodDiff SelfDiff TensDiff DietDiff MealDiff RelxDiff EmbrDiff
IrrtDiff JobsDiff LessDiff FuncDiff AnyDiff

Achingg (D) Aching (grouped answers)

- 1 Never/hardly ever
- 2 Occasionally
- 3 Fairly often/very often

SPSS syntax

```
recode Aching (1,2=1) (3=2) (4,5=3) (else=copy) into Achingg.
variable labels Achingg '(D) Aching (grouped answers)'
value labels achingg 1 'Never/hardly ever' 2 'Occasionally' 3 'Fairly often/very often'.
```

OIDP – oral impacts on daily performances

IpEatG Prevalence of difficulty eating

- 0 No difficulty eating
- 1 Difficulty eating

IpSpkG Prevalence of difficulty speaking

- 0 No difficulty speaking
- 1 Difficulty speaking

IpClnG Prevalence of difficulty cleaning teeth/dentures

- 0 No difficulty cleaning teeth/dentures
- 1 Difficulty cleaning teeth/dentures

IpOutG Prevalence of difficulty going out

- 0 No difficulty going out
- 1 Difficulty going out

IpRlxG Prevalence of difficulty relaxing

- 0 No difficulty relaxing
- 1 Difficulty relaxing

IpShwG Prevalence of difficulty showing teeth

- 0 No difficulty showing teeth
- 1 Difficulty showing teeth

IpWrkG Prevalence of difficulty working

- 0 No difficulty showing teeth
- 1 Difficulty showing teeth

IpEmtG Prevalence of problems with emotional stability

- 0 No problems with emotional stability
- 1 Problems with emotional stability

IpConG Prevalence of difficulty with contact with others

- 0 No difficulty with contact with others
- 1 Difficulty with contact with others

OIDPAll Overall OIDP prevalence

- 0 No items with OIDP score
- 1 At least 1 item with OIDP score

SPSS Syntax

```
*****OIDP variables *****.

recode IpEat IpSpk IpCln IpOut IpRlx IpShw IpWrk IpEmt IpCon (0=0) (1 thru 5=1) (else=copy) into IpEatG
      IpSpkG IpClnG IpOutG IpRlxG IpShwG IpWrkG IpEmtG IpConG.
missing values IpEatG IpSpkG IpClnG IpOutG IpRlxG IpShwG IpWrkG IpEmtG IpConG (-8).

value labels IpEatG IpSpkG IpClnG IpOutG IpRlxG IpShwG IpWrkG IpEmtG IpConG 0 'not affected' 1 'affected'.

missing values IpEatG IpSpkG IpClnG IpOutG IpRlxG IpShwG IpWrkG IpEmtG IpConG ().
execute.
do if (IpEatG=-8 or IpSpkG=-8 or IpClnG=-8 or IpOutG=-8 or IpRlxG=-8 or IpShwG=-8 or IpWrkG=-8 or
IpEmtG=-8 or IpConG=-8) .
compute OIDPAll=-8.
else if (IpEatG=1 or IpSpkG=1 or IpClnG=1 or IpOutG=1 or IpRlxG=1 or IpShwG=1 or IpWrkG=1 or
IpEmtG=1 or IpConG=1) .
compute OIDPAll=1.
else.
compute OIDPAll=0.
end if.

variable labels OIDPAll 'overall OIDP prevalence'.
value labels OIDPAll
0 'no items with OIDP score'
1 'at least 1 item with OIDP score'.
```

```
missing values IpEatG IpSpkG IpClnG IpOutG IpRlxG IpShwG IpWrkG IpEmtG IpConG OIDPall (-8,-9,-1).
```

TotOIDP ODP severity rating Sum of scores of 9 items

SPSS Syntax

```
*** using x + y etc as opposed to Sum means if any = missing TotOIDP = missing *****.

missing values IpEat IpSpk IpCln IpOut IpRlx IpShw IpWrk IpEmt IpCon ().
execute.
Do if (IpEat=-8 or IpSpk=-8 or IpCln=-8 or IpOut=-8 or IpRlx=-8 or IpShw=-8 or IpWrk=-8 or IpEmt=-8
or IpCon=-8) .
Compute TotOIDP=-8.
else.
compute TotOIDP =(IpEat+ IpSpk+ IpCln +IpOut+ IpRlx+ IpShw +IpWrk +IpEmt+ IpCon)/45 *100.
end if.
variable labels TotOIDP 'severity of ODP score- sum'.

Missing values IpEat IpSpk IpCln IpOut IpRlx IpShw IpWrk IpEmt IpCon TotOIDP (-8,-9,-1).
```

HiOIDP Highest reported score on any ODP problem

SPSS syntax

```
missing values IpEat IpSpk IpCln IpOut IpRlx IpShw IpWrk IpEmt IpCon ().
execute.
Do if (IpEat=5 or IpSpk=5 or IpCln=5 or IpOut=5 or IpRlx=5 or IpShw=5 or IpWrk=5 or IpEmt=5 or
IpCon =5 ) .
Compute HiOIDP=5.
Else if (IpEat=4 or IpSpk=4 or IpCln=4 or IpOut=4 or IpRlx=4 or IpShw=4 or IpWrk=4 or IpEmt=4 or
IpCon =4 ) .
Compute HiOIDP=4.
Else if (IpEat=3 or IpSpk=3 or IpCln=3 or IpOut=3 or IpRlx=3 or IpShw=3 or IpWrk=3 or IpEmt=3 or
IpCon =3 ) .
Compute HiOIDP=3.
Else if (IpEat=2 or IpSpk=2 or IpCln=2 or IpOut=2 or IpRlx=2 or IpShw=2 or IpWrk=2 or IpEmt=2 or
IpCon =2 ) .
Compute HiOIDP=2.
Else if (IpEat=1 or IpSpk=1 or IpCln=1 or IpOut=1 or IpRlx=1 or IpShw=1 or IpWrk=1 or IpEmt=1 or
IpCon =1) .
Compute HiOIDP=1.
Else if (IpEat=0 or IpSpk=0 or IpCln=0 or IpOut=0 or IpRlx=0 or IpShw=0 or IpWrk=0 or IpEmt=0 or
IpCon =0) .
Compute HiOIDP=0
Else .
Compute HiOIDP=-8.
end if.

Variable labels HiOIDP 'highest reported score on any ODP problem'.
missing values HiOIDP (-8,-9,-1).
```

NumOIDP Extent of ODP score

SPSS Syntax

```
missing values IpEat IpSpk IpCln IpOut IpRlx IpShw IpWrk IpEmt IpCon ().
execute.
Do if (IpEat=-8 or IpSpk=-8 or IpCln=-8 or IpOut=-8 or IpRlx=-8 or IpShw=-8 or IpWrk=-8 or IpEmt=-8
or IpCon=-8) .
Compute NumOIDP=-8.
else.
Count NumOIDP= IpEatG IpSpkG IpClnG IpOutG IpRlxG IpShwG IpWrkG IpEmtG IpConG (1).
end if.
variable labels NumOIDP 'number items with ODP score'.
missing values IpEat IpSpk IpCln IpOut IpRlx IpShw IpWrk IpEmt IpCon NumOIDP (-8,-9,-1).
execute.
```

Access to care

TrdNHSGp (D) Tried to make NHS in last 3 years - grouped No and Never

- 1 Yes
- 2 No

SPSS syntax

```
recode Trdnhs (1=1) (2,3=2) (else=copy) into TrdNHSGp.  
missing values TrdnhsGp (-9 to -1).  
variable labels TrdnhsGp '(D) Tried to make NHS in last 3 years - grouped No and Never'.  
value labels TrdnhsGp 1 'Yes' 2 'No'.
```

ScsNHStr Successfully made NHS appointment in last 3 years – excluding never tried

- 1 Yes, went to appt
- 2 Yes, didn't go to appt
- 3 No

SPSS syntax

```
recode scsnhs (4=-4) (else=copy) into ScsNHStr.  
missing values scsnhstr (-9 to -1).  
variable labels scsnhstr 'Successfully made NHS appointment in last 3 years- excluding never tried'.  
value labels scsnhstr 1 'Yes, went to appointment' 2 'Yes, didn t go to appointment' 3 'No'  
-4 'Never tried to make NHS appt (spont only)' -8 'Dont know' -1 'Item not applicable' .
```

Effect of cost

YesCost Combined answer to costtyp and costdly

- 1 yes to both
- 2 no to either

SPSS syntax

```
missing values CostTyp CostDly ().  
execute.  
Do if (CostTyp=-8 or CostDly=-8).  
Compute YesCost=-8.  
Else if (CostTyp=1 AND CostDly=1) .  
Compute YesCost=1.  
Else .  
Compute YesCost=4.  
end if.  
execute.  
Variable labels YesCost 'combined answer to costtyp and costdly'.  
value labels YesCost 1 'yes to both' 2 'no to either' .  
crosstabs tables = costtyp by costdly.  
missing values CostTyp CostDly YesCost (-8,-1).
```

Dental Anxiety

MdTreatG How would feel if went to dentist tomorrow - grouped

- 1 Not anxious
- 2 Slightly/fairly anxious
- 3 Very/extremely anxious

MdWtRmG How would feel if sitting in waiting room - grouped

- 1 Not anxious
- 2 Slightly/fairly anxious
- 3 Very/extremely anxious

MdThDrlG How would feel if about to have tooth drilled- grouped

- 1 Not anxious
- 2 Slightly/fairly anxious
- 3 Very/extremely anxious

MdScleG How would feel if about to have scale and polish - grouped

- 1 Not anxious
- 2 Slightly/fairly anxious
- 3 Very/extremely anxious

MdInjtG How would feel if about to have local anaesthetic injection - grouped

- 1 Not anxious
- 2 Slightly/fairly anxious
- 3 Very/extremely anxious

SPSS Syntax

```
recode MdTreat MdWtRm MdThDrl MdScle MdInjt (1=1) (2,3=2) (4,5=3) (-8=-8) (-1=-1) into MdTreatG MdWtRmG
MdThDrlG MdScleG MdInjtG.
value labels MdTreatG MdWtRmG MdThDrlG MdScleG MdInjtG
  1 'Not anxious' 2 'slightly/fairly anxious' 3 'very/extremely anxious' -8 'dont know' -1 'not
applicable'.
missing values MdTreatG MdWtRmG MdThDrlG MdScleG MdInjtG (-8 -1).
```

TotMDAS Total score on MDAS

TotMDASG Total score on MDAS - grouped

- 1 0-9
- 2 10-18
- 3 19+

DTotMDAS Phobic and nonphobic

- 1 phobic
- 2 nonphobic

SPSS Syntax

```
**** using x + y etc as opposed to Sum means if any = missing TotOHIP = missing *****.
missing values MdTreat MdWtRm MdThDrl MdScle MdInjt ().
execute.

do if ( MdTreat=-1 and MdWtRm=-1 and MdThDrl=-1 and MdScle=-1 and MdInjt=-1).
compute TotMDAS=-1.
else if (MdTreat=-8 or MdWtRm=-8 or MdThDrl=-8 or MdScle=-8 or MdInjt=-8).
```

```

compute TotMDAS=-8.
else .
compute TotMDAS =MdTreat + MdWtRm + MdThDrl + MdScle + MdInjt .
end if.
execute.
variable labels TotMDAS 'total score on MDAS'.
recode TotMDAS (0 thru 9=1) (10 thru 18=2) (19 thru 25=3) (else=copy) into TotMDASG.

variable labels TotMDASG 'total score on MDAS grouped'.
value labels TotMDASG 1 '0-9' 2 '10-18' 3 '19+' -1 'item not applicable' -8 'dont know'.

Recode TotMDAS (0 thru 18=2) (19 thru 25=1) (else=copy) into DTotMDAS.
variable labels DTotMDAS 'phobic and nonphobic'.
value labels TotMDASG 1 'phobic' 2 'nonphobic' -1 'item not applicable' -8 'dont know'.

missing values MdTreat MdWtRm MdThDrl MdScle MdInjt TotMDAS TotMDASG DTotMDAS (-1,-8).

```

Relationship with dentist

TimDisYN (D) Was given enough time to discuss problems with the dentist - yes/no only

InvDecYN (D) Was involved as much as would like in decisions - yes/no only

AndUndYN (D) Was given answers that could be understood - yes/no only

- 1 Yes
- 2 No

SPSS syntax

```
recode TimDis InvDec AndUnd (1=1) (2=2)(else=copy) into TimDisYN InvDecYN AndUndYN.
missing values TimDisYN InvDecYN AndUndYN (-9 to -1,3).
variable labels TimDisYN `(D) Was given enough time to discuss problems with the dentist - yes/no only'/
InvDecYN `(D) Was involved as much as would like in decisions - yes/no only'/
AndUndYN `(D) Was given answers that could be understood - yes/no only'.

value labels TimDisYN InvDecYN AndUndYN 1 'Yes' 2 'No'.
```

NumDenR Number of negative responses to dentist relationship

NumDenRG Number of negative responses to dentist relationship grouped

- 0
- 1
- 2 2-4
- 3 5 or more

SPSS syntax

```
missing values Listen TimDis InvDec AnsUnd ResTre Respec ConTrus().
execute.

do if (Listen =-8 or TimDis =-8 or InvDec =-8 or AnsUnd =-8 or ResTre =-8 or Respec =-8 or ConTrus =-8).
compute NumDenR = -8.
else if (Listen =-1 or TimDis =-1 or InvDec =-1 or AnsUnd =-1 or ResTre =-1 or Respec =-1 or ConTrus =-1).
compute NumDenR = -1.
else if (Listen =-9 or TimDis =-9 or InvDec =-9 or AnsUnd =-9 or ResTre =-9 or Respec =-9 or ConTrus =-9).
compute NumDenR = -9.
else if (Listen ge 1 and TimDis ge 1 and InvDec ge 1 and AnsUnd ge 1 and ResTre ge 1 and Respec ge 1 and
ConTrus ge 1).
Count NumDenR=Listen TimDis InvDec AnsUnd ResTre Respec ConTrus (2).
end if.
execute.
variable labels NumDenR 'number of negative responses to dentist relationship'.
value labels NumDenR -1 'not applicable' -8 'dont know' -9 'no answer/refused'.
recode NumDenR (0=0) (1=1) (2 thru 4=2) (5 thru 14=3) (else=copy) into NumDenRG.
variable labels NumDenRG 'number of negative responses to dentist relationship grouped'.
value labels NumDenRG 0 '0' 1 '1' 2 '2-4' 3 '5 or more' -1 'not applicable' -8 'dont know' -9 'no
answer/refused'.

missing values Listen TimDis InvDec AnsUnd ResTre Respec ConTrus NumDenR NumDenRG (-1,-8,-9).
fre vars = Listen TimDis InvDec AnsUnd ResTre Respec ConTrus NumDenR NumDenRG.
```

Rating of dental practice

rteraptdk rating dental practice in terms of length of wait for a routine appointment including don't knows

rteuaptdk rating dental practice in terms of length of wait for an urgent appointment including don't knows

rtewaptdk rating dental practice in terms of length of wait for a evening or weekend appointment including don't knows

rteaccsdk rating dental practice in terms of transport facilities and access including don't knows

rteconvdk rating dental practice in terms of convenience of location including don't knows

rtchrgdk rating dental practice in terms of explanation of NHS charges including don't knows

rtdenrep rating dental practice in terms of reputation of dentists including don't knows

rtcredk rating dental practice in terms of standard and quality of care including don't knows

1. Very good
2. Good
3. Fair
4. Poor
5. Very poor
- 8 Don't know

SPSS Syntax

```
*****Use filter for rterapt rteuapt and rtewapt to filter out not applicable/never had appointment
for all ratings tables*****
RECODE
  RteRApt RteUApt RteWApt RteAccs RteConv RteCre RteChrg RteDen Next (-8=7) into rteraptdk
rteuaptdk rtewaptdk rteaccsdk rteconvdk Rtechrgdk rtdenrep rtcredk.
EXECUTE .
```