

The Institutional Research syntax to create the First Day of Session, One Tenth Day of Session, and End of Session Student Demographic files was revised from previous versions of the Colleague Demographic syntax to incorporate changes made to the Colleague (Datatel) extract process during July 2009. These changes included adjustments previously made to the IR demographic files by adding queried data, changes included in Colleagues patches (or process corrections), and changes based on updated releases of the Colleague product. The primary changes are the updates to race and ethnicity reporting to comply with new IPEDS (Integrated Postsecondary Education Data System) requirements, correction for collection of valid alien status, addition of e-mail, addition of time-stamp fields (term and extract date) to all files, and other formatting changes to make the document more intuitive.

The files will be organized to provide:

- basic demographic information (e.g., address and phone, residency, date of birth and gender)
- academic "history," i.e., high school attended, number of credits transferred in, etc.
- placement and attendance information, i.e., program of study, credits earned, etc.

Particular changes of which to be aware!!

- **RACE/ETHNICITY.** Due to new federal standards, ethnicity and race are being collected and reported differently. The updated demographic file includes the "old" reported race/ethnicity (repace), and the new reported race/ethnicity (RepRaceEthnic). One of the features of the new reporting format is that individuals may report more than one race. If so, they are reported in the "Two or more races" category. For your reference, there will be a file available which reflects all races students report.
- **DECEASED STATUS.** The file will now incorporate "Personal Status." Please be aware that IR has a more updated version of deceased status. Enrollment services staff enter deceased status in student records only when acceptable documentation is submitted. The IR Data Center enters deceased status in an update database upon verbal notification during surveys and registration reminders.
- **TRANSFER INFORMATION.** Rather than including the names of all colleges from which a student might have transferred, there will be a variable with the number of colleges from which a student transferred, and the total credits from all colleges transferred. However, the names of transfer colleges and associated credits are still available.
- **PREVIOUS DEGREE INFORMATION.** Rather than listing all previous OCC degrees and the month and year earned, there is now a variable reflecting the total number of degrees a student might have earned. Please be aware that this variable includes only associate degrees.
- **PLACEMENT INFORMATION.** Three new variables, English Placement, Math Placement, and ESL Placement, will now be available to reflect student's placement data as of the term in question. Note that these variables are based on the College's intake Placement criteria which can be found in the College catalog. The three most recently posted eligible scores are used for evaluation. Note that students can only repeat the placement tests once within a year.
- **E-MAIL.** The OCC, or student e-mail will be provided as well as the "preferred" e-mail.

Terms by Academic Year

Academic Year	File Date	Term	
1999-2000	994	Summer	1999
	995	Fall	1999
	001	Winter	2000
	002	Spring	2000
2000-2001	004	Summer	2000
	005	Fall	2000
	011	Winter	2001
	012	Spring	2001
2001-2002	014	Summer 2	2001
	015	Fall	2001
	021	Winter	2002
	022	Summer 1	2002
2002-2003	024	Summer 2	2002
	025	Fall	2002
	031	Winter	2003
	032	Summer 1	2003
2003-2004	034	Summer 2	2003
	035	Fall	2003
	041	Winter	2004
	042	Summer 1	2004

2004-2005	044	Summer II	2004
	045	Fall	2004
	051	Winter	2005
	052	Summer I	2005

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align
1	id	String	10	0	Student ID	None	None	10	Left
2	ssn	String	11	0	Social Security	None	None	11	Left
3	lname	String	26	0	Last Name	None	None	26	Left
4	fname	String	16	0	First Name	None	None	16	Left
5	mname	String	10	0	Middle Name	None	None	10	Left
6	homcmp	String	2	0	Home Campus	None	None	5	Left
7	prvcmp	String	2	0	Previous Hom	None	None	6	Left
8	fmrlname	String	26	0	Former Last N	None	None	26	Left
9	fmrfname	String	16	0	Former First N	None	None	16	Left
10	fmrname	String	15	0	Former Middle	None	None	15	Left
11	rescnty	String	6	0	County of Resi	None	None	6	Left
12	resstt	String	2	0	State of Resid	None	None	2	Left
13	rescntry	String	5	0	Country of Res	None	None	5	Left
14	alien	String	1	0	Alien Status	None	None	1	Left
15	primlang	String	10	0	Primary Langu	None	None	10	Left
16	immig	String	4	0	Immigration St	{F1, F-1 Stude	None	4	Left
17	area	Numeric	3	0	Area Code	None	None	8	Right
18	phone	String	8	0	Phone Number	None	None	8	Left
19	gender	String	1	0	Gender	None	None	1	Left
20	ethnic	String	2	0	Race/Ethnicity	{AF, African A	None	2	Left
21	brthmo	Numeric	2	0	Birth Month	None	None	8	Right
22	brthda	Numeric	2	0	Birth Day	None	None	8	Right
23	brthyr	Numeric	2	0	Birth Year	None	None	8	Right
24	ferpa	String	1	0	Privacy Code	None	None	1	Left
25	veteran	String	2	0	Veteran Status	None	None	2	Left
26	vetben	String	3	0	Veterans Bene	None	None	3	Left

Demo 041t.sav -

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align
27	rescode	String	4	0	Residency Co	{INID, Internati	None	4	Left
28	citizen	String	1	0	U.S. Citizenshi	None	None	1	Left
29	address	String	26	0	Address	None	None	26	Left
30	address2	String	26	0	Address Line 2	None	None	26	Left
31	city	String	20	0	City	None	None	20	Left
32	state	String	2	0	State	None	None	2	Left
33	zip	String	10	0	Zip Code	None	None	10	Left
34	hscnty	Numeric	7	0	High School C	None	None	8	Right
35	hscode	Numeric	6	0	High School C	None	None	8	Right
36	hsname	String	27	0	High School N	None	None	27	Left
37	hscity	String	16	0	High School Ci	None	None	16	Left
38	hsstate	String	2	0	High School St	None	None	2	Left
39	hsgradmo	Numeric	1	0	High School G	None	None	8	Right
40	hsgradyr	Numeric	1	0	High School G	None	None	8	Right
41	ged	String	1	0	GED	None	None	1	Left
42	appmo	Numeric	2	0	OCC Applicati	None	None	8	Right
43	appday	Numeric	2	0	OCC Applicati	None	None	8	Right
44	appyr	Numeric	2	0	OCC Applicati	None	None	8	Right
45	appsess	String	7	0	Intended Starti	None	None	7	Left
46	admst	String	2	0	Admission Stat	{NC, Non Cred	None	2	Left
47	intent	String	5	0	Educational G	{TA, Transfer	None	5	Left
48	credatt	Numeric	5	2	Credits Attemp	None	None	8	Right
49	credearn	Numeric	5	2	Credits Earned	None	None	8	Right
50	gpacreds	Numeric	5	2	Credits Calcul	None	None	8	Right
51	gpapts	Numeric	6	2	Grade Points	None	None	8	Right
52	prevsess	String	7	0	Last Session A	None	None	7	Left

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align
53	crprog	String	14	0	Credit Progra	None	None	14	Left
54	ncprog	String	7	0	Non-Credit Pro	None	None	7	Left
55	prevdeg	Numeric	1	0	Highest Degre	{1, Doctorate}	None	8	Right
56	run	String	10	0	Extract Date	None	None	10	Left
57	term	String	7	0	Session	None	None	7	Left
58	campah	Numeric	8	2	Courses Take	None	None	8	Right
59	camphl	Numeric	8	2	Courses Take	None	None	8	Right
60	campor	Numeric	8	2	Courses Take	None	None	8	Right
61	campro	Numeric	8	2	Courses Take	None	None	8	Right
62	campsf	Numeric	8	2	Courses Take	None	None	8	Right
63	campdw	Numeric	8	2	Courses Take	None	None	8	Right
64	credah	Numeric	8	2	Credits Regist	None	None	8	Right
65	credhl	Numeric	8	2	Credits Regist	None	None	8	Right
66	credor	Numeric	8	2	Credits Regist	None	None	8	Right
67	credro	Numeric	8	2	Credits Regist	None	None	8	Right
68	credsf	Numeric	8	2	Credits Regist	None	None	8	Right
69	creddw	Numeric	8	2	Credits Regist	None	None	8	Right
70	fpalst	String	1	0		None	None	4	Left
71	reprace	String	8	0	Reported Race	{AF, African A	None	8	Left

USING STUDENT COURSE REGISTRATION FILES

Student course registration files include a record for each course each student has registered during the respective term and snapshot point indicated (e.g. Fall One Tenth Day).

Note: If you need to look at grades, choose the End of Session Student Course Registration File. If using Summer I (Spring) files, look for the All Grades (e.g. Spring 2002 All Grades Course Registration) file to ensure that you evaluate grades for 15 week courses, or other courses that extend beyond the Summer I (Spring) dates of instruction.

The file includes the following variables for each record:

Run – Extract date, or when the data file was extracted from Colleague

Term – The term selected

Pref – Course prefix, or subject abbreviation, e.g. ENG for English

Num – Course number, e.g. 1510, for English 1510

Loc – Campus where course section is held ID – Student ID

Syn – Course synonym, also known as registration number; unique section identifier for the term

Acs --ACS Code -- State Activity Classification Structure code

Strt – Course section start date

End – CourseLSection end date

Type – Credit Type (Credit, Developmental, Non-Credit)

Cred – Credits assigned to the course section

Cont – Contact hours assigned to the course section

Id – Student ID

Hmcp – Student's Home Campus (Where student paperwork is stored)

Last – Student Last Name

First – Student First Name

Midd – Student middle name

Ssn – Student Social Security Number if available

Sec – Course section number

Grde – Populated only in End of Session Course Files

Stat – Student's status in the course

Grdlt – Grade Letter, computed only for End of session Course Files

Grdpt – Associated grade points, computed only for End of Session Course Files

SELECT BY COURSE OR COURSE PREFIX

To choose a set of students taking a particular course or subject, select them out by prefix, or by prefix and number. For instance, to choose only students taking English courses, Select students if subject = 'ENG'. To choose only students taking English 1510, Select students if subject = 'ENG' & Num = '1510'.

TO SELECT BY CREDIT TYPE, USUALLY DEVELOPMENTAL

To choose students who are taking Developmental classes, Select if credtype = 'DEV'. Note that it is not recommended to use ACS codes for this purpose. In limited cases, students might be offered undergraduate level credit rather than developmental credit for some courses. Therefore, attention must be paid to Credit Type rather than ACS.

TO VIEW BY UNDUPLICATED STUDENT COMPLETE COURSE RECORD

If you need to look at the student's complete course taking for a term, restructure or "dedupe" the file.

UNIQUE IDENTIFIER FOR COURSE SECTION IS SYNONYM

Raw Data Evolution from Colleague to SPSS

Colleague Mnemonic (Extract) (can be run by term or ACAD year)	Colleague File (created in HOLD)	Imported to \Research Data\Student Information System***
XIR1-IR Course Section Data	X.IR.COURSE.FILE.yyyymmddtttt	COURSEmmmddyytttt.DAT
XIR2-IR Head Count Extract Select (Automatically ran at 6am daily)	X.IR.HEADCNT.FILE.yyyymmdd X.IR.COURSE.FILE.yyyymmdd	HEADCOUNTmmmddyy.DAT COURSEmmmddyy.DAT
XIR3-Select Student Demographics	X.IR.COLLEGE.FILE.yyyymmdd X.IR.DEGREES.FILE.yyyymmdd X.IR.SPOPS.FILE.yyyymmdd X.IR.STUDENT.FILE.yyyymmdd X.IR.TESTS.FILE.yyyymmdd	COLLEGEmmmddyy.DAT DEGREESmmmddyy.DAT SPOPSmmmddyy.DAT STUDENTmmmddyy.DAT TESTSmmmddyy.DAT

Raw Data File	Demo	Course	DemoCourse
COLLEGEmmmddyy.DAT			
COURSEmmmddyy.DAT			
DEGREESmmmddyy.DAT			
HEADCOUNTmmmddyy.DAT			
SPOPSmmmddyy.DAT			
STUDENTmmmddyy.DAT	X		
TESTSmmmddyy.DAT			

Instructions for creating the Demo file

- Open appropriate STUDENT file found in: ^{Term 2004}
I:\Research Data\Student Information System\Term 200x\Raw Data
- Use Colleague DEMO syntax file located at:
I:\Research Data\Student Information System\Syntax ~~Correction file~~ \Colleague DEMO.SPS
- Adjust the get and output file names and run.

need to add from Colleague
Query Builder

Use:

IR STUDENTS 2004W1

check items and save list to reflect Term

Then use

Save fpa1st for Winter 2004

change

Term on list

& PC

File name

&

Comma Quoted Values

Open

Foreign Person Alien Status (syntax)

Sort on ID

Save as

from
Common
syntax
file

*Use this syntax to create single course registrations from Legacy Course
*Files. Use it when you need to look at particular courses.
*To use this file you will need to replace some file destinations and names.
*Find the "GET FILE" commands (13), and replace the destination and file name
*with the base file, or file you wish to deconstruct. Find the "SAVE OUTFILE"
*commands (15), and replace the destination and file name with the destination
*and file name you will be using for your analysis. Find the "/FILE" commands
with and path and destination other than "", and replace the destination and
*file name with same destination and file name
*you have used for "SAVE OUTFILE."

```
GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\test crse syntax.sav'.
```

```
STRING locat (A2).  
COMPUTE locat = loc1 .  
VARIABLE LABELS locat 'Location' .  
EXECUTE .
```

```
STRING section (A5).  
COMPUTE section = sect1.  
VARIABLE LABELS section 'Section'.
```

```
STRING course (A7).  
COMPUTE course = course1.  
VARIABLE LABELS course 'Course Prefix and Number'.
```

```
STRING prefix (A3).  
COMPUTE prefix = prefix1.  
VARIABLE LABELS prefix 'Prefix'.
```

```
STRING grade (A1).  
COMPUTE grade = grade1.  
VARIABLE LABELS grade 'grade'.  
EXECUTE.
```

```
SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\tempcrs.sav'  
/DROP=loc1 sect1 course1 prefix1 grade1 loc2 sect2 course2 prefix2 grade2  
loc3 sect3 course3 prefix3 grade3 loc4 sect4 course4 prefix4 grade4 loc5  
sect5 course5 prefix5 grade5 loc6 sect6 course6 prefix6 grade6 loc7 sect7  
course7 prefix7 grade7 loc8 sect8 course8 prefix8 grade8 loc9 sect9 course9  
prefix9 grade9 loc10 sect10 course10 prefix10 grade10 loc11 sect11 course11  
prefix11 grade11 loc12 sect12 course12 prefix12 grade12 /COMPRESSED.
```

```
GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\test crse syntax.sav'.
```

```
STRING locat (A2).  
COMPUTE locat = loc2 .  
VARIABLE LABELS locat 'Location' .  
EXECUTE .
```

```
STRING section (A5).  
COMPUTE section = sect2.  
VARIABLE LABELS section 'Section'.
```

```
STRING course (A7).  
COMPUTE course = course2.  
VARIABLE LABELS course 'Course Prefix and Number'.
```

```
STRING prefix (A3).  
COMPUTE prefix = prefix2.  
VARIABLE LABELS prefix 'Prefix'.
```

```
STRING grade (A1).  
COMPUTE grade = grade2.
```

VARIABLE LABELS grade 'grade'.
EXECUTE.

USE ALL.
SELECT IF locat ~= ''.
EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)

/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

'tempcrs.sav'

/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.

EXECUTE.

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

'tempcrs.sav'

GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

'test crse syntax.sav'.

STRING locat (A2).

COMPUTE locat = loc3.

VARIABLE LABELS locat 'Location' .

EXECUTE .

STRING section (A5).

COMPUTE section = sect3.

VARIABLE LABELS section 'Section'.

STRING course (A7).

COMPUTE course = course3.

VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).

COMPUTE prefix = prefix3.

VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).

COMPUTE grade = grade3.

VARIABLE LABELS grade 'grade'.

EXECUTE.

USE ALL.

SELECT IF locat ~= ''.

EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44

```
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)
/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.
EXECUTE.
```

```
SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'
```

```
GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'test crse syntax.sav'.
```

```
STRING locat (A2).
COMPUTE locat = loc4.
VARIABLE LABELS locat 'Location' .
EXECUTE .
```

```
STRING section (A5).
COMPUTE section = sect4.
VARIABLE LABELS section 'Section'.
```

```
STRING course (A7).
COMPUTE course = course4.
VARIABLE LABELS course 'Course Prefix and Number'.
```

```
STRING prefix (A3).
COMPUTE prefix = prefix4.
VARIABLE LABELS prefix 'Prefix'.
```

```
STRING grade (A1).
COMPUTE grade = grade4.
VARIABLE LABELS grade 'grade'.
EXECUTE.
```

```
USE ALL.
SELECT IF locat ~= ''.
EXECUTE .
```

```
ADD FILES /FILE=*
/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)
/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.
EXECUTE.
```

```
SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'
```

```
GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'test crse syntax.sav'.
```

```
STRING locat (A2).
COMPUTE locat = loc5.
```

VARIABLE LABELS locat 'Location' .
EXECUTE .

STRING section (A5).
COMPUTE section = sect5.
VARIABLE LABELS section 'Section'.

STRING course (A7).
COMPUTE course = course5.
VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).
COMPUTE prefix = prefix5.
VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).
COMPUTE grade = grade5.
VARIABLE LABELS grade 'grade'.
EXECUTE.

USE ALL.
SELECT IF locat ~= '' .
EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)

/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

'tempcrs.sav'

/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.

EXECUTE.

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'

GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'test crse syntax.sav'.

STRING locat (A2).
COMPUTE locat = loc6.
VARIABLE LABELS locat 'Location' .
EXECUTE .

STRING section (A5).
COMPUTE section = sect6.
VARIABLE LABELS section 'Section'.

STRING course (A7).
COMPUTE course = course6.
VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).
COMPUTE prefix = prefix6.
VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).
COMPUTE grade = grade6.

VARIABLE LABELS grade 'grade'.
EXECUTE.

USE ALL.
SELECT IF locat ~= ''.
EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)

/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'

/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.

EXECUTE.

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'

GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'test crse syntax.sav'.

STRING locat (A2).
COMPUTE locat = loc7.
VARIABLE LABELS locat 'Location'.
EXECUTE .

STRING section (A5).
COMPUTE section = sect7.
VARIABLE LABELS section 'Section'.

STRING course (A7).
COMPUTE course = course7.
VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).
COMPUTE prefix = prefix7.
VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).
COMPUTE grade = grade7.
VARIABLE LABELS grade 'grade'.
EXECUTE.

USE ALL.
SELECT IF locat ~= ''.
EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)

/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

`tempcrs.sav'

/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.

EXECUTE.

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

`tempcrs.sav'

GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

`test crse syntax.sav'.

STRING locat (A2).

COMPUTE locat = loc8.

VARIABLE LABELS locat 'Location' .

EXECUTE .

STRING section (A5).

COMPUTE section = sect8.

VARIABLE LABELS section 'Section'.

STRING course (A7).

COMPUTE course = course8.

VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).

COMPUTE prefix = prefix8.

VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).

COMPUTE grade8 = grade.

VARIABLE LABELS grade 'grade'.

EXECUTE.

USE ALL.

SELECT IF locat ~= ''.

EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)

/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

`tempcrs.sav'

/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.

EXECUTE.

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

`tempcrs.sav'

GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

`test crse syntax.sav'.

STRING locat (A2).

COMPUTE locat = loc9.

VARIABLE LABELS locat 'Location' .

EXECUTE .

STRING section (A5).
COMPUTE section = sect9.
VARIABLE LABELS section 'Section'.

STRING course (A7).
COMPUTE course = course9.
VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).
COMPUTE prefix = prefix9.
VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).
COMPUTE grade = grade9.
VARIABLE LABELS grade 'grade'.
EXECUTE.

USE ALL.
SELECT IF locat ~= ''.
EXECUTE .

ADD FILES /FILE=*

/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)

/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+

\tempcrs.sav'

/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.

EXECUTE.

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
\tempcrs.sav'

GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
\test crse syntax.sav'.

STRING locat (A2).
COMPUTE locat = loc10.
VARIABLE LABELS locat 'Location'.
EXECUTE .

STRING section (A5).
COMPUTE section = sect10.
VARIABLE LABELS section 'Section'.

STRING course (A7).
COMPUTE course = course10.
VARIABLE LABELS course 'Course Prefix and Number'.

STRING prefix (A3).
COMPUTE prefix = prefix10.
VARIABLE LABELS prefix 'Prefix'.

STRING grade (A1).
COMPUTE grade = grade10.
VARIABLE LABELS grade 'grade'.
EXECUTE.

```
USE ALL.
SELECT IF locat ~= ''.
EXECUTE .
```

```
ADD FILES /FILE=*
```

```
/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)
```

```
/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'
```

```
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57
d58 d59.
```

```
EXECUTE.
```

```
SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'tempcrs.sav'
```

```
GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'test crse syntax.sav'.
```

```
STRING locat (A2).
```

```
COMPUTE locat = loc11.
```

```
VARIABLE LABELS locat 'Location' .
```

```
EXECUTE .
```

```
STRING section (A5).
```

```
COMPUTE section = sect11.
```

```
VARIABLE LABELS section 'Section'.
```

```
STRING course (A7).
```

```
COMPUTE course = course11.
```

```
VARIABLE LABELS course 'Course Prefix and Number'.
```

```
STRING prefix (A3).
```

```
COMPUTE prefix = prefix11.
```

```
VARIABLE LABELS prefix 'Prefix'.
```

```
STRING grade (A1).
```

```
COMPUTE grade = grade11.
```

```
VARIABLE LABELS grade 'grade'.
```

```
EXECUTE.
```

```
USE ALL.
```

```
SELECT IF locat ~= ''.
```

```
EXECUTE .
```

```
ADD FILES /FILE=*
```

```
/RENAME (course1 course10 course11 course12 course2 course3 course4 course5
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)
```

```
/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
```

```
'tempcrs.sav'
```


*To determine if student is FTIAC.

*FIRST RUN A FREQUENCY ON ALL TCOLL.1 -- TCOLL.5 TO OBTAIN LIST OF 'TECH PREP' TCOLLS.

*LOOK FOR TECH PREP IN THE LIST, AND MAKE SURE ALL OF THE TECH PREPs ARE LISTED IN THE SYNTAX BELOW !!
!

*For Fall 2003 the following ID's need to be eliminated. Asterisk the next section if using this
*syntax on any other term.

*FILTER OFF.

*USE ALL.

*SELECT IF(id ~= '613236' or id ~= '554404' or id NE '616743' or id NE '616744' or id NE '616750'
* or id NE '616751').

*EXECUTE .

STRING ftiac2 (A2).

IF (admst = 'FT' & status = 1 & prevsess = '' & prevdeg > 6 & (tcoll.1 = '' | tcoll.1 =

'TECH PREP BUSINESS INFORMATION SYST' | tcoll.1 = 'TECH PREP DENTAL HEALTH' | tcoll.1 = 'TECH PREP DRAFTING'+

'DESIGN' | tcoll.1 = 'TECH PREP EARLY CHILDHOOD DEVELOPME' | tcoll.1 = 'TECH'+

'PREP HEALTH' | tcoll.1 = 'TECH PREP MEDICAL ASSISTING' | tcoll.1 = 'TECH'+

'PREP STUDENT-APPLICANT') & tcoll.2 = '' & crprog ~= 'GUE' & crprog ~=

'HSG' & awd.1 = ') ftiac2 = 'Y' .

VARIABLE LABELS ftiac 'Calculated FTIAC2' .

EXECUTE .

Course.sps

SET

BLANKS=SYSMIS BLANKS=SYSMIS

UNDEFINED=WARN.

DATA LIST FILE='I:\Research Data\Student Information System\Updated End of Session'+

'\Legacy Data Files\Course-005.dat' FIXED

RECORDS=1 TABLE/1 ssn 1-9 sess1 10

-12(A) loc1 13-14(A) sect1 13-17(A) course1 18-24(A) prefix1 18-20 (A) grade1 25-25(A) loc2 26

-27(A) sect2 26-30(A) course2 31-37(A) prefix2 31-33 (A) grade2 38-38(A) loc3 39-40(A) sect3

39-43(A) course3 44-50(A) prefix3 44-46 (A) grade3 51-51(A) loc4 52-53(A) sect4 52-56(A)

course4 57-63(A) prefix4 57-59 (A) grade4 64-64(A) loc5 65-66(A) sect5 65-69(A) course5 70

-76(A) prefix5 70-72 (A) grade5 77-77(A) loc6 78-79(A) sect6 78-82(A) course6 83-89(A) prefix6 83-85 (A) grade6

90-90(A) loc7 91-92(A) sect7 91-95(A) course7 96-102(A) prefix7 96-98 (A) grade7 103-103(A)

loc8 104-105(A) sect8 104-108(A) course8 109-115(A) prefix8 109-111 (A) grade8 116-116(A) loc9

117-118(A) sect9 117-121(A) course9 122-128(A) prefix9 122-124 (A) grade9 129-129(A) loc10 130

-131(A) sect10 130-134(A) course10 135-141(A) prefix10 135-137 (A) grade10 142-142(A) loc11 143

-144(A) sect11 143-147(A) course11 148-154(A) prefix11 148-150 (A) grade11 155-155(A) loc12 156

-157(A) sect12 156-160(A) course12 161-167(A) prefix12 161-163 (A) grade12 168-168(A) .

EXECUTE.

SAVE OUTFILE='I:\Research Data\Student Information System\Updated End of Session\Course-005E.SAV'.

```
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19  
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38  
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57  
d58 d59.
```

EXECUTE.

```
SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\tempcrs.sav'
```

```
GET FILE 'C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\test crse syntax.sav'.
```

```
STRING locat (A2).  
COMPUTE locat = loc12.  
VARIABLE LABELS locat 'Location' .  
EXECUTE .
```

```
STRING section (A5).  
COMPUTE section = sect12.  
VARIABLE LABELS section 'Section'.
```

```
STRING course (A7).  
COMPUTE course = course12.  
VARIABLE LABELS course 'Course Prefix and Number'.
```

```
STRING prefix (A3).  
COMPUTE prefix = prefix12.  
VARIABLE LABELS prefix 'Prefix'.
```

```
STRING grade (A1).  
COMPUTE grade = grade12.  
VARIABLE LABELS grade 'grade'.  
EXECUTE.
```

```
USE ALL.  
SELECT IF locat ~= '' .  
EXECUTE .
```

```
ADD FILES /FILE=*
```

```
/RENAME (course1 course10 course11 course12 course2 course3 course4 course5  
course6 course7 course8 course9 grade1 grade10 grade11 grade12 grade2 grade3  
grade4 grade5 grade6 grade7 grade8 grade9 loc1 loc10 loc11 loc12 loc2 loc3  
loc4 loc5 loc6 loc7 loc8 loc9 prefix1 prefix10 prefix11 prefix12 prefix2  
prefix3 prefix4 prefix5 prefix6 prefix7 prefix8 prefix9 sect1 sect10 sect11  
sect12 sect2 sect3 sect4 sect5 sect6 sect7 sect8 sect9 = d0 d1 d2 d3 d4 d5  
d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25  
d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44  
d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59)
```

```
/FILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\tempcrs.sav'
```

```
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19  
d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35 d36 d37 d38  
d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57  
d58 d59.
```

EXECUTE.

```
SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+  
'\tempcrs.sav'.
```

```
STRING type (A3).  
IF (course = 'CHE095' or course = 'SPE100' or course = 'TEM101' or course = 'TEM102' or course = 'CHE095'  
or course = 'SPE100' or course = 'ENG050' or course = 'ENG051' or course = 'ENG052' or course = 'ENG053'  
or course = 'ENG054' or course = 'ENG055' or course = 'ENG056' or course = 'ENG102' or course = 'ENG103'  
or course = 'ENG104' or course = 'ENG105' or course = 'ENG106' or course = 'ENG110' or course = 'ENG120'  
or course = 'ENG131' or course = 'MAT101' or course = 'MAT104' or course = 'MAT105' or course = 'MAT107'  
or course = 'MAT110' or course = 'MAT111')
```

TYPE = 'DEV'.
EXECUTE .

SAVE OUTFILE='C:\Multimedia Files\My Documents\Eileen\Sept 2003'+
'\tempcrs.sav'.

```
GET DATA /TYPE = TXT
/FILE = 'I:\Perkins\At Risk December 2003\spops winter 03 121103.dat' ✓
/DELCASE = LINE
/DELIMITERS = ", "
/QUALIFIER = ""
/ARRANGEMENT = DELIMITED
/FIRSTCASE = 2
/IMPORTCASE = ALL
/VARIABLES =
id A10
ssn A11
lname A26
fname A16
mname A10
homcmp A2
V7 A2
spop A3
fund A3
camp A2.
```

```
CACHE.
EXECUTE.
```

VARIABLE LABELS

```
id 'Student ID'
ssn 'Social Security Number'
lname 'Last Name'
fname 'First Name'
mname 'Middle Name'
homcmp 'Home Campus'
spop 'SPOPS Category'
fund 'Funding Category'
camp 'Service Campus'.
EXECUTE.
```

```
value labels spop 'SP' 'Single Parent' 'ID' 'Individuals with Disabilities' 'DH' 'Displaced Homemaker'
'LE' 'Limited English Proficiency' 'AD' 'Academically Disadvantaged'
'ED' 'Economically Disadvantaged' 'NT' 'Non-Traditional'
'HM' 'Homemaker' 'LD' 'Learning Disabled'.
```

```
value labels fund 'ATR' 'At-Risk' 'GEN' 'General' 'PER' 'Perkins'.
```

```
SAVE OUTFILE='I:\Perkins\At Risk December 2003\spops winter 03 121103.sav'. ✓
```

```
SORT CASES BY id .
```

```
CASESTOVARS
```

```
/ID = id
/GROUPBY = INDEX .
```

```
SAVE OUTFILE='I:\Perkins\At Risk December 2003\Unduplicated spops winter 03 121103.sav'. ✓
```

```
*STOP.
```

```
*Look at output for number of funds, spops, and camps.
```

```
*Modify next portion of syntax accordingly.
```

```
FILTER OFF.
```

```
USE ALL.
```

```
SELECT IF(fund.1 = 'ATR' OR fund.2 = 'ATR' or fund.3 = 'ATR' OR fund.4 =
'ATR' or fund.5 = 'ATR' or fund.6 = 'ATR').
```

```
EXECUTE .
```

```
SAVE OUTFILE='I:\Perkins\At Risk December 2003\Unduplicated At Risk Funded winter 03 121103.sav'.
```

```
MATCH FILES /FILE=*
```

```
/TABLE='I:\Perkins\At Risk December 2003\Demo031Er1.sav'
/RENAME (fname homcmp lname mname ssn = d0 d1 d2 d3 d4)
/BY id
```

/DROP= d0 d1 d2 d3 d4.
EXECUTE.

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Unduplicated At Risk Funded winter 03 121103.sav'.

*STOP

*May need to add the following lines back in.
*/MAKE camp FROM camp.1 camp.2 camp.3 camp.4 camp.5
*camp.1 camp.2 camp.3 camp.4 camp.5.

VARSTOCASES /MAKE fund FROM fund.1 fund.2 fund.3 fund.4 fund.5 fund.6
/MAKE spops FROM spop.1 spop.2 spop.3 spop.4 spop.5 spop.6
/KEEP = address address2 admst alien appday appmo appsess appyr area brthda
brthmo brthyr citizen city credatt credearn crprog ethnic ferpa fmrfname
fmrlname fmrname fname ged gender gpacreds gpapts homcmp hscity hscnty
hscode hsgradmo hsgradyr hsname hsstate id immig intent lname mname ncpog
phone prevdeg prevsess primlang prvcmp rescntry rescnty rescode resstt ssn
state v7 vetben veteran zip fund.1 fund.2 fund.3 fund.4 fund.5 fund.6 spop.1 spop.2
spop.3 spop.4 spop.5 spop.6 camp
/NULL = KEEP.

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Duplicated At Risk Cases winter 03 121103.sav'.
FILTER OFF.

USE ALL.

SELECT IF(fund = 'ATR').

EXECUTE .

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Duplicated At Risk Cases winter 03 121103.sav'.

COMPUTE count = 1 .

EXECUTE .

AGGREGATE

/OUTFILE='I:\Perkins\At Risk December 2003\at risk by spops code by program winter 03R1.sav'

/BREAK=crprog spops

/count_1 = SUM(count).

GET

FILE='I:\Perkins\At Risk December 2003\at risk by spops code by program winter 03R1.sav'.

IF (spops = 'AD') adcount = count_1 .
VARIABLE LABELS adcount 'Academically Disadvantaged' .
IF (spops = 'SP') spcount = count_1.
VARIABLE LABELS spcount 'Single Parent'.
IF (spops = 'ID') IDcount =count_1.
VARIABLE LABELS IDcount 'Individuals with Disabilities'.
IF (spops='DH') dhcount = count_1.
VARIABLE LABELS dhcount 'Displaced Homemaker'.
IF (spops = 'LEP') lecount =count_1.
VARIABLE LABELS lecount 'Limited English Proficiency'.
IF (spops = 'ED') edcount =count_1.
VARIABLE LABELS edcount 'Economically Disadvantaged'.
IF (spops = 'NT') ntcoun = count_1.
VARIABLE LABELS ntcoun 'Non-Traditional'.
IF (spops = 'HM') hmcount =count_1.
VARIABLE LABELS hmcount = 'Homemaker'.
IF (spops = 'LD') ldcount = count_1.
VARIABLE LABELS ldcount 'Learning Disabled'.

EXECUTE .

RECODE

adcount count_1 dhcount edcount hmcount idcount ldcount lecount ntcoun
spcount (MISSING=0) .

EXECUTE.

AGGREGATE

/OUTFILE='!:\Perkins\At Risk December 2003\at risk by program by spops code winter 03R1.sav'

/BREAK=crprog

/count__1 = SUM(count_1) /adcoun_1 = SUM(adcount) /dhcoun_1 = SUM(dhcount)

/edcoun_1 = SUM(edcount) /hmcoun_1 = SUM(hmcount) /idcoun_1 = SUM(idcount)

/ldcoun_1 = SUM(ldcount) /lecoun_1 = SUM(lecoun) /ntcoun_1 = SUM(ntcount)

/spcoun_1 = SUM(spcoun).

'DC' 'Obtain OCC degree or certificate'
'ER' 'Employment Related Knowledge'
'EN' 'Gain knowledge unrelated to employment'
'UN' 'Undecided'.

value labels immig 'PR' 'Permanent Resident'
'TR' 'Temporary Resident'
'RE' 'Refugee/Asylee'
'RA' 'Resident Alien'
'NI' 'Non-Immigrant'
'NR' 'Non-Resident Alien'
'UN' 'Unknown'
'AU' 'Au Pair'
'F1' 'F-1 Student'
'OI' 'Other International'.

value labels rescode 'REVN' 'Residency Verification Needed' 'SCOD' 'Invalid Code'
'OUSV' 'Virtual College Out of State' 'OUDV' 'Virtual College Out of District'
'SCIN' 'Invalid Code' 'INOS' 'International Out of State' 'INOD' 'International Out of District'
'INID' 'International In District' 'INDI' 'In District' 'OUST' 'Out of State'
'INDV' 'Virtual College In District' 'OUDI' 'Out of District'.

value labels prevdeg 1 'Doctorate' 2 'Master' 3 'Bachelor' 5 'Associate' 6 'Certificate'
7 'High School' 8 'GED' 9 'No determination'.

value labels ethnic 'AF' 'African American' 'AS' 'Asian' 'HI' 'Hispanic' 'NI' 'Native American'
'UN' 'Race Unknown' 'WH' 'White'.

SAVE OUTFILE='C:\SPOPS\Demo025.sav'.

Notes

STUDENT DEC 03 03.DAT

SET BLANKS=SYSMIS BLANKS=SYSMIS UNDEFINED=WARN.

GET DATA /TYPE = TXT

/FILE = 'G:\SPOPS\STUDENT\DEC18.DAT'

I:\Perkins\ At Risk December 2003\STUDENT DEZ0303.DAT

/DELCASE = LINE

/DELIMITERS = ", "

/QUALIFIER = ""

/ARRANGEMENT = DELIMITED

/FIRSTCASE = 2

/IMPORTCASE = ALL

/VARIABLES =

id A10
ssn A11
lname A26
fname A16
mname A10
homcmp A2
prvcmp A2
fmrlname A26
fmrfname A16
fmrname A15
rescnty A6
resstt A2
rescntry A5
alien A1
primlang A10
immig A4
area F3.0
phone A8
gender A1
ethnic A2
brthmo F2.0
brthda F2.0
brthyr F2.0
ferpa A1
veteran A2
vetben A3
rescode A4
citizen A1
address A26
address2 A26
city A20
state A2
zip A10
hscnty F7.0
hscode F6.0
hsname A27
hscity A16
hsstate A2
hsgradmo F1.0
hsgradyr F1.0
ged A1
appmo F2.0
appday F2.0
appyr F2.0
appsess A7
admst A2
intent A5
credreg 5 X
credatt F5.2
credearn F5.2
gpacreds F5.2
gpapts F6.2
prevsess A7
crprog A14
sdprog 14 X
ncprog A7
prevdeg F1.0

CACHE.
EXECUTE.

VARIABLE LABELS

id 'Student ID'
ssn 'Social Security Number'
lname 'Last Name'
fname 'First Name'
mname 'Middle Name'
homcmp 'Home Campus'
prvcmp 'Previous Home Campus'
fmrlname 'Former Last Name'
fmrfname 'Former First Name'
fmrmmname 'Former Middle Name'
rescnty 'County of Residence'
resstt 'State of Residence'
rescntry 'Country of Residence'
alien 'Alien Status'
primlang 'Primary Language'
immig 'Immigration Status'
area 'Area Code'
phone 'Phone Number'
gender 'Gender'
ethnic 'Race/Ethnicity'
brthmo 'Birth Month'
brthda 'Birth Day'
brthyr 'Birth Year'
ferpa 'Privacy Code'
veteran 'Veteran Status'
vetben 'Veterans Benefits'
rescode 'Residency Code'
citizen 'U.S. Citizenship'
address 'Address'
address2 'Address Line 2'
city 'City'
state 'State'
zip 'Zip Code'
hscnty 'High School County'
hscode 'High School Code'
hsname 'High School Name'
hscity 'High School City'
hsstate 'High School State'
hsgradmo 'High School Graduation Month'
hsgradyr 'High School Graduation Year'
ged 'GED'
appmo 'OCC Application Month'
appday 'OCC Application Day'
appyr 'OCC Application Year'
appsess 'Intended Starting Session'
admst 'Admission Status'
intent 'Educational Goal'
credatt 'Credits Attempted OCC Career'
credearn 'Credits Earned OCC Career'
gpacreds 'Credits Calculated in GPA'
gpapts 'Grade Points Earned OCC Career'
prevsess 'Last Session Attended'
cprog 'Credit Program'
ncprog 'Non-Credit Program'
prevdeg 'Highest Degree Obtained'

EXECUTE.

value labels admst 'CG' 'College Guest' 'FT' 'First Time in Any College' 'HD' 'Dual Enrollment'
'HG' 'High School Guest' 'NC' 'Non Credit' 'PR' 'Previously Attended OCC' 'TR' 'Transfer'.

value labels intent 'TW' 'Transfer without obtaining degree or certificate'
'TA' 'Transfer after obtaining degree or certificate'.

Begin At Risk Fall 03 .SPS

GET DATA /TYPE = TXT
/FILE = 'I:\Perkins\At Risk December 2003\SPOPSMAR2803.DAT'
/DELCASE = LINE
/DELIMITERS = ", "
/QUALIFIER = ""
/ARRANGEMENT = DELIMITED
/FIRSTCASE = 2
/IMPORTCASE = ALL
/VARIABLES =
id A10
ssn A11
lname A26
fname A16
mname A10
homcmp A2
V7 A2
spop A3
fund A3
camp A2.

SEPOSD3

CACHE.
EXECUTE.

VARIABLE LABELS
id 'Student ID'
ssn 'Social Security Number'
lname 'Last Name'
fname 'First Name'
mname 'Middle Name'
homcmp 'Home Campus'
spop 'SPOPS Category'
fund 'Funding Category'
camp 'Service Campus'.
EXECUTE.

value labels spop 'SP' 'Single Parent' 'ID' 'Individuals with Disabilities' 'DH' 'Displaced Homemaker'
'LE' 'Limited English Proficiency' 'AD' 'Academically Disadvantaged'
'ED' 'Economically Disadvantaged' 'NT' 'Non-Traditional'
'HM' 'Homemaker' 'LD' 'Learning Disabled'.

value labels fund 'ATR' 'At-Risk' 'GEN' 'General' 'PER' 'Perkins'.

SAVE OUTFILE='I:\Perkins\At Risk December 2003\SPOPSMAR2803.sav'.

SORT CASES BY id .
CASESTOVARS
/ID = id
/GROUPBY = INDEX .

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Unduplicated SPOPSMAR2803.sav'.

*STOP.
*Look at output for number of funds,spops, and camps.
*Modify next portion of syntax accordingly.

FILTER OFF.
USE ALL.
SELECT IF(fund.1 = 'ATR' OR fund.2 = 'ATR' or fund.3 = 'ATR' OR fund.4 =
'ATR' or fund.5 = 'ATR' or fund.6 = 'ATR').
EXECUTE .

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Unduplicated At Risk Funded 032803.sav'.

MATCH FILES /FILE=*
/TABLE='I:\Perkins\At Risk December 2003\Demo 031E.sav'
/RENAME (fname homcmp lname mname ssn = d0 d1 d2 d3 d4)
/BY id

/DROP= d0 d1 d2 d3 d4.
EXECUTE.

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Unduplicated At Risk Funded 032803.sav'.

*STOP

*May need to add the following lines back in.

*/MAKE camp FROM camp.1 camp.2 camp.3 camp.4 camp.5

*camp.1 camp.2 camp.3 camp.4 camp.5.

VARSTOCASES /MAKE fund FROM fund.1 fund.2 fund.3 fund.4 fund.5 fund.6
/MAKE spops FROM spop.1 spop.2 spop.3 spop.4 spop.5 spop.6
/KEEP = address address2 admst alien appday appmo appsess appyr area brthda
brthmo brthyr citizen city credatt credearn crprog ethnic ferpa fmrfname
fmrlname fmrmname fname ged gender gpacreds gpapts homcmp hscity hscnty
hscode hsgradmo hsgradyr hsname hsstate id immig intent lname mname ncpog
phone prevdeg prevsess primlang prvcmp rescntry rescnty rescode resstt ssn
state v7 vetben veteran zip fund.1 fund.2 fund.3 fund.4 fund.5 fund.6 spop.1 spop.2
spop.3 spop.4 spop.5 spop.6 camp
/NULL = KEEP.

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Duplicated At Risk Cases 032803.sav'.

FILTER OFF.

USE ALL.

SELECT IF(fund = 'ATR').

EXECUTE .

SAVE OUTFILE='I:\Perkins\At Risk December 2003\Duplicated At Risk Cases 032803.sav'.

COMPUTE count = 1 .

EXECUTE .

AGGREGATE

/OUTFILE='I:\Perkins\At Risk December 2003\at risk by spops code by program winter 03.sav'

/BREAK=crprog spops

/count_1 = SUM(count).

GET

FILE='I:\Perkins\At Risk December 2003\at risk by spops code by program winter 03.sav'.

IF (spops = 'AD') adcount = count_1 .

VARIABLE LABELS adcount 'Academically Disadvantaged' .

IF (spops = 'SP') spcount = count_1.

VARIABLE LABELS spcount 'Single Parent'.

IF (spops = 'ID') IDcount =count_1.

VARIABLE LABELS IDcount 'Individuals with Disabilities'.

IF (spops='DH') dhcount = count_1.

VARIABLE LABELS dhcount 'Displaced Homemaker'.

IF (spops = 'LEP') lecount =count_1.

VARIABLE LABELS lecount 'Limited English Proficiency'.

IF (spops = 'ED') edcount =count_1.

VARIABLE LABELS edcount 'Economically Disadvantaged'.

IF (spops = 'NT') ntcoun = count_1.

VARIABLE LABELS ntcoun 'Non-Traditional'.

IF (spops = 'HM') hmcount =count_1.

VARIABLE LABELS hmcount = 'Homemaker'.

IF (spops = 'LD') ldcount = count_1.

VARIABLE LABELS ldcount 'Learning Disabled'.

EXECUTE .

RECODE

adcount count_1 dhcount edcount hmcount idcount ldcount lecount ntcoun

spcount (MISSING=0) .

EXECUTE.

AGGREGATE

/OUTFILE='I:\Perkins\At Risk December 2003\at risk by program by spops code winter 03.sav'

/BREAK=crprog

/count__1 = SUM(count_1) /adcoun_1 = SUM(adcount) /dhcoun_1 = SUM(dhcount)

/edcoun_1 = SUM(edcount) /hmcoun_1 = SUM(hmcount) /idcoun_1 = SUM(idcount)

/ldcoun_1 = SUM(ldcount) /lecoun_1 = SUM(lecoun) /ntcoun_1 = SUM(ntcount)

/spcoun_1 = SUM(spcount).

GET DATA /TYPE = TXT
/FILE = C:\SPOPS\SPOPSMAR2803.DAT'
/DELCASE = LINE
/DELIMITERS = ", "
/QUALIFIER = ""
/ARRANGEMENT = DELIMITED
/FIRSTCASE = 2
/IMPORTCASE = ALL
/VARIABLES =
id A10
ssn A11
lname A26
fname A16
mname A10
homcmp A2
V7 A2
spop A3
fund A3
camp A2.

I:\Perkins\At Risk December 2003\

CACHE.
EXECUTE.

VARIABLE LABELS

id 'Student ID'
ssn 'Social Security Number'
lname 'Last Name'
fname 'First Name'
mname 'Middle Name'
homcmp 'Home Campus'
spop 'SPOPS Category'
fund 'Funding Category'
camp 'Service Campus'.
EXECUTE.

value labels spop 'SP' 'Single Parent' 'ID' 'Individuals with Disabilities' 'DH' 'Displaced Homemaker'
'LE' 'Limited English Proficiency' 'AD' 'Academically Disadvantaged'
'ED' 'Economically Disadvantaged' 'NT' 'Non-Traditional'
'HM' 'Homemaker' 'LD' 'Learning Disabled'.

value labels fund 'ATR' 'At-Risk' 'GEN' 'General' 'PER' 'Perkins'.

SAVE OUTFILE=C:\SPOPS\SPOPSMAR2803.sav'.
I Perkins ...

SORT CASES BY id .
CASESTOVARS
/ID = id
/GROUPBY = INDEX .

SAVE OUTFILE=C:\SPOPS\Unduplicated SPOPSMAR2803.sav'.

FILTER OFF.
USE ALL.
SELECT IF(fund.1 = 'ATR' OR fund.2 = 'ATR' or fund.3 = 'ATR' OR fund.4 =
'ATR' or fund.5 = 'ATR').
EXECUTE .

SAVE OUTFILE=C:\SPOPS\Unduplicated At Risk Funded 121802.sav'.
32803

MATCH FILES /FILE=*
/TABLE=C:\SPOPS\Demo025.sav'
/RENAME (fname homcmp lname mname ssn = d0 d1 d2 d3 d4)
/BY id
/DROP= d0 d1 d2 d3 d4.
EXECUTE.

I:\Perkins\At Risk December 2003\ Demo 031. sav

SAVE OUTFILE=C:\SPOPS\Unduplicated At Risk Funded 121802.sav'.
I:\Perkins\At Risk December 2003

* LOOK FOR # of
fund, SPOPS

VARSTOCASES /MAKE fund FROM fund.1 fund.2 fund.3 fund.4 fund.5 *fund 6*
/MAKE spops FROM spop.1 spop.2 spop.3 spop.4 spop.5 *spop 6*
/MAKE camp FROM camp.1 camp.2 camp.3 camp.4 camp.5
/KEEP = address address2 admst alien appday appmo appsess appyr area brthda
brthmo brthyr citizen city credatt credearn crprog ethnic ferpa fmrfname
fmrlname fmrname fname ged gender gpacreds gpapts homcmp hscity hscnty
hscode hsgradmo hsgradyr hsname hsstate id immigr intent lname mname ncprog
phone prevdeg prevsess primlang prvcmp rescntry rescnty rescode resstt ssn
state v7 vetben veteran zip fund.1 fund.2 fund.3 fund.4 fund.5 spop.1 spop.2
spop.3 spop.4 spop.5 camp.1 camp.2 camp.3 camp.4 camp.5
/NULL = KEEP. *fund. 6*

I:\Perkins... *032803*
SAVE OUTFILE='C:\SPOPS\Duplicated At Risk Cases 121802.sav'.
FILTER OFF.
USE ALL.
SELECT IF(fund = 'ATR').
EXECUTE. *I Perkins...* *032803*
SAVE OUTFILE='C:\SPOPS\Duplicated At Risk Cases 121802.sav'.

COMPUTE count = 1 .
EXECUTE .

AGGREGATE
/OUTFILE='C:\SPOPS\at risk by spops code by program.sav'
/BREAK=crprog spops
/count_1 = SUM(count).

GET
FILE='C:\SPOPS\at risk by spops code by program.sav'.

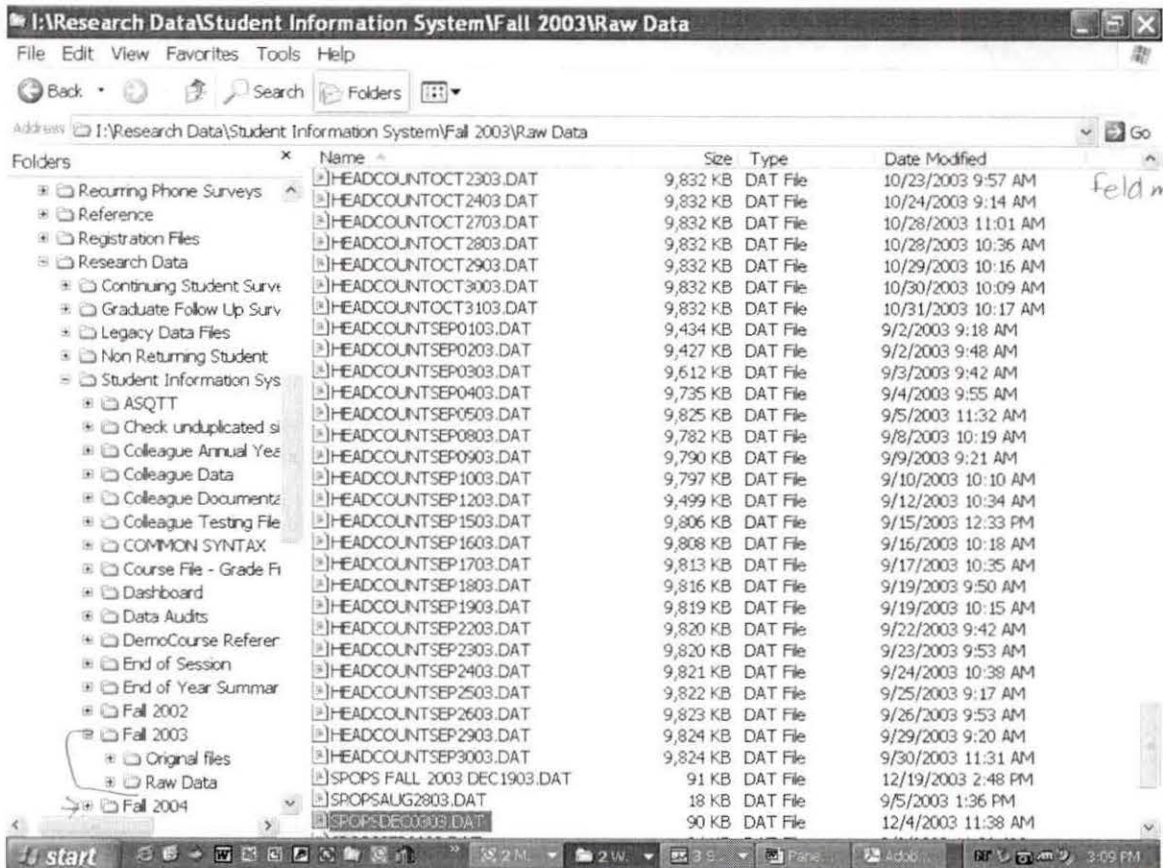
IF (spops = 'AD') adcount = count_1 .
VARIABLE LABELS adcount 'Academically Disadvantaged' .
IF (spops = 'SP') spcount = count_1 .
VARIABLE LABELS spcount 'Single Parent'.
IF (spops = 'ID') IDcount =count_1 .
VARIABLE LABELS IDcount 'Individuals with Disabilities'.
IF (spops='DH') dhcount = count_1 .
VARIABLE LABELS dhcount 'Displaced Homemaker'.
IF (spops = 'LEP') lecount =count_1 .
VARIABLE LABELS lecount 'Limited English Proficiency'.
IF (spops = 'ED') edcount =count_1 .
VARIABLE LABELS edcount 'Economically Disadvantaged'.
IF (spops = 'NT') ntcoun = count_1 .
VARIABLE LABELS ntcoun 'Non-Traditional'.
IF (spops = 'HM') hmcoun =count_1 .
VARIABLE LABELS hmcoun = 'Homemaker'.
IF (spops = 'LD') ldcount = count_1 .
VARIABLE LABELS ldcount 'Learning Disabled'.

EXECUTE .

RECODE
adcount count_1 dhcount edcount hmcoun idcount ldcount lecount ntcoun
spcount (MISSING=0) .
EXECUTE.

AGGREGATE
/OUTFILE='C:\SPOPS\at risk by program by spops code.sav'
/BREAK=crprog
/count_1 = SUM(count_1) /adcoun_1 = SUM(adcount) /dhcoun_1 = SUM(dhcount)
/edcoun_1 = SUM(edcount) /hmcoun_1 = SUM(hmcoun) /idcoun_1 = SUM(idcount)
/ldcoun_1 = SUM(ldcount) /lecoun_1 = SUM(lecount) /ntcoun_1 = SUM(ntcoun)
/spcoun_1 = SUM(spcount).

6/22 - 27



Saved as via Windows Explorer!

Oct 14, 2003 →
need to be re copied
into Raw Data.

13 - wrong data

10/14/03	10/27/03	11/11/03	12/01/03	22
10/15/03	28	12	12/02/03	23
10/16/03	29	13	12/03/03	24
10/17/03	30	14	04	25
10/20/03	31	15	05	26
21	11/03/03	18	06	27
22	04	19	07	29
23	05	20	12/08/03	30
24	06	21	9	31
	07	11/24/03	10	15
		25	11	16
		26	12	17
		27	13	18
		28		19
		29		20

11/04
11/3/04
5
6
7
8
9
10

34
37
71

Foster, Gail

From: Showers, Nancy
Sent: Tuesday, August 17, 2004 11:23 AM
To: Brennan, Eileen
Cc: Foster, Gail
Subject: Status of SU II 04 OTD file

Follow Up Flag: Follow up
Due By: Tuesday, August 24, 2004 12:00 AM
Flag Status: Flagged

Can you please advise when the demo file will be available? I need it for a project I'm working on for Carla. Thanks.

Nancy, ext. 2141

*SU II 04 OTD = 7/9/04 data
use 7/12 XIR3 data*

Demo 044T

Follow Instructions rather than syntax

TERM DEMOGRAPHIC AND COURSE FILE PROCESS

Seven files are extracted out of Colleague to form the basic IR demographic and course files. The "Headcount" and "Course" files are generated automatically. These two files provide information about student course registrations and the courses themselves.

The following files are produced by running the Colleague program "XIR3": "Student," "Colleges," "Degrees," "Tests," and "SPOPS." The first three files produce the SPSS "Demo" file. Selected elements from Test are added in as well. SPOPS is produced solely for the purpose of State Reporting.

RUN THE EXTRACTS

1. In Colleague, enter the mnemonic "XIR3." A window requesting the term appears. Enter the term, hit the save icon, and save out of the ensuing windows.
2. After the program runs the report, the files must be imported to the IR network drive (I). To do this, select "Import file" under the Options menu. The Host file is HOLD.
 - a. An easy way to select the files is to click on the Item button, which produces a search window. Select the "begins" radio button, and enter the following starting sequence for each report. Note that only one search item can be entered.
 - b. This list is provided for each report to import:
"X.IR.STUDENT.FILE.#####" , "X.IR.COLLEGE.FILE.#####" ,
"X.IR.DEGREES.FILE.#####" , "X.IR.TESTS.FILE.#####" , AND
"X.IR.SPOPS.FILE.#####" . The ##### indicates entering the year and month to narrow the search. (E.g. X.IR.STUDENT.FILE.200409).
 - c. Each report must be imported by itself. The Local file to which the data should be imported is the term file, the raw data folder. The naming convention is to use the Colleague file name (e.g. STUDENT) followed by the date (no spaces) and given the "DAT" extension. For example, STUDENT09082004.DAT.

CREATE THE SPSS DATA BASE – BASIC DEMOGRAPHIC INFORMATION

1. The first syntax is "Colleague Demo." This converts the "Student" file into SPSS.
2. Note that the ID selection criteria eliminates test and dummy ids known to IR.
3. Until the extract program is corrected, (as of July 2004 it had not been), the correct "Alien Status" must be imported. (Until July 2003 this information was contained in the PERSON file; after that time the information is stored in the FOREIGN.PERSON file for students). Run the query, "IR Students by term" to generate a saved list of students, and then the query "Save Foreign Person Alien Status."
4. Run the SPSS syntax "Foreign Person Alien Status" to convert the Colleague extract file into SPSS. Return to the "Demo" File, and merge the newly created fpalst variable into the Demo file.

5. Run the syntax "Reported RaceEthnicity" to get a Reported Race Ethnicity variable incorporating Non Resident Alien status.

CREATE THE TRANSFER COLLEGE FILE

1. Run the "Transfers" syntax, which imports the "Colleges.dat" file into SPSS.
2. Please note that the variable "tcred" is computed to reduce the credits to the correct size. Decimals were not included in the extract resulting in an import of credits multiplied by 100.
3. Note that the syntax restructures the file to unduplicated by student id. The number of "tcoll" and "tcred" (Transfer college and transfer credits) variables may change to term depending on the maximum number of transfers by any student.

CREATE THE PREVIOUS OCC DEGREES FILE

1. Run the SPSS syntax "Previous OCC Degrees after Summer I 2003." (Prior to Summer I 2003 the information was extracted differently). This converts the Degrees.dat file into SPSS.
2. Note that the syntax restructures the file to unduplicated by student id. The number of "occdeg" (OCC degree program) variables may change to term depending on the maximum number of degrees earned by any student.

ENROLLMENT DATA

1. Information regarding student credit enrollment is available as summary data (credits per campus, total credits) through the Official Enrollment Report base data.
2. Information regarding individual student course taking is available through a duplicated student database in the "Student Course Registration" folder. Instructions regarding the use of this database will be included in that folder.

MERGE THE INFORMATION FOR THE DEMO FILE

1. Open the "Demo" file if it is not open already. Merge in the transfer file, adding only the "tcoll" and "tcred" variables. (Recall, the "cred" variables are the transfer credits multiplied by 100).
2. Merge in the previous OCC degrees file. Add only the "awdmo" (Award Month), "awdyr" (Award Year) and "occdeg" (OCC Degree) variables.
3. Run the 'FTIAC' syntax to incorporate the calculated FTIAC variable.

Syntax: "Colleague DEMO.SPS"

GET DATA /TYPE = TXT

/FILE = 'I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\STUDENTJUL1204.DAT'

/DELCASE = LINE

/DELIMITERS = ", "

/QUALIFIER = ""

/ARRANGEMENT = DELIMITED

/FIRSTCASE = 2

/IMPORTCASE = ALL

/VARIABLES =

id A10

ssn A11

lname A26

fname A16

mname A10

homcmp A2

prvcmp A2

fmrlname A26

fmrfname A16

fmrname A15

rescnty A6

resstt A2

rescntry A5

alien A1

primlang A10

immig A4

area F3.0

phone A8

gender A1

ethnic A2

brthmo F2.0

brthda F2.0

brthyr F2.0

ferpa A1

veteran A2

vetben A3

rescode A4

citizen A1

address A26

address2 A26

city A20

state A2

zip A10

hscnty F7.0

hscode F6.0

hsname A27

hscity A16

hsstate A2

hsgradmo F1.0

hsgradyr F1.0

ged A1

appmo F2.0

appday F2.0

appyr F2.0

appsess A7

admst A2

intent A5

credreg 5 X

credatt F5.2

credearn F5.2

gpacreds F5.2

gpapts F6.2

prevsess A7

crprog A14

sdprog 14 X

ncprog A7

prevdeg F1.0

CACHE.

EXECUTE.

VARIABLE LABELS

id 'Student ID'
ssn 'Social Security Number'
lname 'Last Name'
fname 'First Name'
mname 'Middle Name'
homcmp 'Home Campus'
prvcmp 'Previous Home Campus'
fmrlname 'Former Last Name'
fmrfname 'Former First Name'
fmrname 'Former Middle Name'
rescnty 'County of Residence'
resstt 'State of Residence'
rescntry 'Country of Residence'
alien 'Alien Status'
primlang 'Primary Language'
immig 'Immigration Status'
area 'Area Code'
phone 'Phone Number'
gender 'Gender'
ethnic 'Race/Ethnicity'
brthmo 'Birth Month'
brthda 'Birth Day'
brthyr 'Birth Year'
ferpa 'Privacy Code'
veteran 'Veteran Status'
vetben 'Veterans Benefits'
rescode 'Residency Code'
citizen 'U.S. Citizenship'
address 'Address'
address2 'Address Line 2'
city 'City'
state 'State'
zip 'Zip Code'
hscnty 'High School County'
hscode 'High School Code'
hsname 'High School Name'
hscity 'High School City'
hsstate 'High School State'
hsgradmo 'High School Graduation Month'
hsgradyr 'High School Graduation Year'
ged 'GED'
appmo 'OCC Application Month'
appday 'OCC Application Day'
appyr 'OCC Application Year'
appsess 'Intended Starting Session'
admst 'Admission Status'
intent 'Educational Goal'
credatt 'Credits Attempted OCC Career'
credeam 'Credits Earned OCC Career'
gpacreds 'Credits Calculated in GPA'
gpapts 'Grade Points Earned OCC Career'
prevsess 'Last Session Attended'
crprog 'Credit Program'
ncprog 'Non-Credit Program'
prevdeg 'Highest Degree Obtained'

EXECUTE.

value labels admst 'CG' 'College Guest' 'FT' 'First Time in Any College' 'HD' 'Dual Enrollment'
'HG' 'High School Guest' 'NC' 'Non Credit' 'PR' 'Previously Attended OCC' 'TR' 'Transfer'.

value labels intent 'TW' 'Transfer without obtaining degree or certificate'
'TA' 'Transfer after obtaining degree or certificate'
'DC' 'Obtain OCC degree or certificate'

'ER' 'Employment Related Knowledge'
'EN' 'Gain knowledge unrelated to employment'
'UN' 'Undecided'.

value labels immig 'PR' 'Permanent Resident'
'TR' 'Temporary Resident'
'RE' 'Refugee/Asylee'
'RA' 'Resident Alien'
'NI' 'Non-Immigrant'
'NR' 'Non-Resident Alien'
'UN' 'Unknown'
'AU' 'Au Pair'
'F1' 'F-1 Student'
'OI' 'Other International'.

value labels rescode 'REVN' 'Residency Verification Needed' 'SCOD' 'Invalid Code'
'OUSV' 'Virtual College Out of State' 'OUDV' 'Virtual College Out of District'
'SCIN' 'Invalid Code' 'INOS' 'International Out of State' 'INOD' 'International Out of District'
'INID' 'International In District' 'INDI' 'In District' 'OUST' 'Out of State'
'INDV' 'Virtual College In District' 'OUDI' 'Out of District'.

value labels prevdeg 1 'Doctorate' 2 'Master' 3 'Bachelor' 5 'Associate' 6 'Certificate'
7 'High School' 8 'GED' 9 'No determination'.

value labels ethnic 'AF' 'African American' 'AS' 'Asian' 'HI' 'Hispanic' 'NI' 'Native American'
'UN' 'Race Unknown' 'WH' 'White'.

*To prepare file for merge.
SORT CASES BY
id (A) .

*To remove blanks and test accounts.
FILTER OFF.
USE ALL.
SELECT IF (id NE '' & id NE '0616743' & id NE '0616744' & id NE '0616750'
& id NE '0616751' & id NE '0613236' & id NE '0554404').
EXECUTE .

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+
'\Demo 044T.sav'.

* Add fpalst (Corrected Alien Status).
MATCH FILES /FILE=*
/TABLE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\Foreign Person 044T.sav'
/BY id.
EXECUTE.

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer I 2004 Raw Data'+
'\Demo 042E.sav'.

*STOP Go run "Foreign Person Alien Status", "Reported RaceEthnicity", "Transfers", "Previous OCC Degrees after Summer I
2003".

*To merge in the transfer file variables.
MATCH FILES /FILE=*
/TABLE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\Transfer044T.sav'
/RENAME (cred.1 cred.2 cred.3 cred.4 cred.5 cred.6 fname homcmp lname mname prvcmp ssn = d0 d1 d2 d3 d4 d5 d6 d7 d8 d9
d10
d11)
/BY id
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11.
EXECUTE.

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+
'\Demo 044T.sav'.

*To merge in the Previous OCC degrees variables.
MATCH FILES /FILE=*

```
/TABLE='I:\Research Data\Student Information System\Summer 2004\Summer I 2004 Raw Data\Previous OCC Degrees  
042E.sav'  
/RENAME (awd.1 awd.2 awd.3 fname homcmp lname major.1 major.2 major.3 mname prvcmp spec.1 spec.2 spec.3 ssn = d0 d1  
d2 d3 d4  
d5 d6 d7 d8 d9 d10 d11 d12 d13 d14)  
/BY id  
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14.  
EXECUTE.
```

```
SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+  
'\Demo 044T.sav'.
```

*To prepare to run "FTIAC" syntax, you must add "stat" from the Official Enrollment unduplicated headcount file.
*stat is the status in each course

*Next you have to compute status for the term.
IF (stat = 'A' & (credah > 0 | credhl > 0 | credor > 0 | credro > 0 | credsf > 0 | creddw > 0)) status = 1 .
VARIABLE LABELS status 'Student Status in Term' .
EXECUTE .

*Run "FTIAC" syntax.
*Save file.

"ColleagueDEMO.SPS"

JUL1204.DAT

GET DATA /TYPE = TXT

/FILE = 'I:\Research Data\Student Information System\Summer 2004\STUDENT070604 (2004SU).DAT'

/DELCASE = LINE

/DELIMITERS = ", "

/QUALIFIER = ""

/ARRANGEMENT = DELIMITED

/FIRSTCASE = 2

/IMPORTCASE = ALL

/VARIABLES =

id A10
ssn A11
lname A26
fname A16
mname A10
homcmp A2
prvcmp A2
fmrlname A26
fmrfname A16
fmrmname A15
rescnty A6
resstt A2
rescntry A5
alien A1
primlang A10
immig A4
area F3.0
phone A8
gender A1
ethnic A2
brthmo F2.0
brthda F2.0
brthyr F2.0
ferpa A1
veteran A2
vetben A3
rescode A4
citizen A1
address A26
address2 A26
city A20
state A2
zip A10
hscnty F7.0
hscode F6.0
hsname A27
hscity A16
hsstate A2
hsgradmo F1.0
hsgradyr F1.0
ged A1
appmo F2.0
appday F2.0
appyr F2.0
appsess A7
admst A2
intent A5
credreg 5 X
credatt F5.2
credearn F5.2
gpacreds F5.2
gpapts F6.2
prevsess A7
crprog A14
sdprog 14 X
ncprog A7
prevdeg F1.0

✓
Summer II 2004 Raw Data

CACHE.

EXECUTE.

VARIABLE LABELS

id 'Student ID'
ssn 'Social Security Number'
lname 'Last Name'
fname 'First Name'
mname 'Middle Name'
homcmp 'Home Campus'
prvcmp 'Previous Home Campus'
fmrlname 'Former Last Name'
fmrfname 'Former First Name'
fmrname 'Former Middle Name'
rescnty 'County of Residence'
resstt 'State of Residence'
rescntry 'Country of Residence'
alien 'Alien Status'
primlang 'Primary Language'
immig 'Immigration Status'
area 'Area Code'
phone 'Phone Number'
gender 'Gender'
ethnic 'Race/Ethnicity'
brthmo 'Birth Month'
brthda 'Birth Day'
brthyr 'Birth Year'
ferpa 'Privacy Code'
veteran 'Veteran Status'
vetben 'Veterans Benefits'
rescode 'Residency Code'
citizen 'U.S. Citizenship'
address 'Address'
address2 'Address Line 2'
city 'City'
state 'State'
zip 'Zip Code'
hscnty 'High School County'
hscode 'High School Code'
hsname 'High School Name'
hscity 'High School City'
hsstate 'High School State'
hsgradmo 'High School Graduation Month'
hsgradyr 'High School Graduation Year'
ged 'GED'
appmo 'OCC Application Month'
appday 'OCC Application Day'
appyr 'OCC Application Year'
appsess 'Intended Starting Session'
admst 'Admission Status'
intent 'Educational Goal'
credatt 'Credits Attempted OCC Career'
credearn 'Credits Earned OCC Career'
gpacreds 'Credits Calculated in GPA'
gpapts 'Grade Points Earned OCC Career'
prevsess 'Last Session Attended'
crprog 'Credit Program'
ncprog 'Non-Credit Program'
prevdeg 'Highest Degree Obtained'

EXECUTE.

value labels admst 'CG' 'College Guest' 'FT' 'First Time in Any College' 'HD' 'Dual Enrollment'
'HG' 'High School Guest' 'NC' 'Non Credit' 'PR' 'Previously Attended OCC' 'TR' 'Transfer'.

value labels intent 'TW' 'Transfer without obtaining degree or certificate'
'TA' 'Transfer after obtaining degree or certificate'
'DC' 'Obtain OCC degree or certificate'

'ER' 'Employment Related Knowledge'
'EN' 'Gain knowledge unrelated to employment'
'UN' 'Undecided'.

value labels immig 'PR' 'Permanent Resident'
'TR' 'Temporary Resident'
'RE' 'Refugee/Asylee'
'RA' 'Resident Alien'
'NI' 'Non-Immigrant'
'NR' 'Non-Resident Alien'
'UN' 'Unknown'
'AU' 'Au Pair'
'F1' 'F-1 Student'
'OI' 'Other International'.

value labels rescode 'REVN' 'Residency Verification Needed' 'SCOD' 'Invalid Code'
'OUSV' 'Virtual College Out of State' 'OUDV' 'Virtual College Out of District'
'SCIN' 'Invalid Code' 'INOS' 'International Out of State' 'INOD' 'International Out of District'
'INID' 'International In District' 'INDI' 'In District' 'OUST' 'Out of State'
'INDV' 'Virtual College In District' 'OUDI' 'Out of District'.

value labels prevdeg 1 'Doctorate' 2 'Master' 3 'Bachelor' 5 'Associate' 6 'Certificate'
7 'High School' 8 'GED' 9 'No determination'.

value labels ethnic 'AF' 'African American' 'AS' 'Asian' 'HI' 'Hispanic' 'NI' 'Native American'
'UN' 'Race Unknown' 'WH' 'White'.

*To prepare file for merge.
SORT CASES BY
id (A).

*To remove blanks and test accounts.
FILTER OFF.
USE ALL.
SELECT IF (id NE '' & id NE '0616743' & id NE '0616744' & id NE '0616750'
& id NE '0616751' & id NE '0613236' & id NE '0554404').
EXECUTE .

don't remove "II"

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+
'\Demo 042E.sav'.
4T

* Add fpalst (Corrected Alien Status).
MATCH FILES /FILE=*
/TABLE='I:\Research Data\Student Information System\Summer 2004\fpalst summer I EOS & II FDS-2004.sav'
/BY id.
EXECUTE.

*1 Summer II 2004 Raw Data /
Foreign Person 044T...sav*

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer I 2004 Raw Data'+
'\Demo 042E.sav'.

*STOP Go run "Foreign Person Alien Status", "Reported RaceEthnicity", "Transfers", "Previous OCC Degrees after Summer I 2003".

** Re open Demo*
*To merge in the transfer file variables.
MATCH FILES /FILE=*
/TABLE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\Transfer042E.sav'
/RENAME (cred.1 cred.2 cred.3 cred.4 cred.5 cred.6 fname homcmp lname mname prvcmp ssn = d0 d1 d2 d3 d4 d5 d6 d7 d8 d9
d10
d11)
/BY id
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11.
EXECUTE.

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+
'\Demo 042E.sav'.
4T

*To merge in the Previous OCC degrees variables.
MATCH FILES /FILE=*

```
./TABLE='I:\Research Data\Student Information System\Summer 2004\Summer I 2004 Raw Data\Previous OCC Degrees  
042E.sav' 044T  
/RENAME (awd.1 awd.2 awd.3 fname homcmp lname major.1 major.2 major.3 mname prvcmp spec.1 spec.2 spec.3 ssn = d0 d1  
d2 d3 d4  
d5 d6 d7 d8 d9 d10 d11 d12 d13 d14)  
/BY id  
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14.  
EXECUTE.
```

```
SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer I 2004 Raw Data'+  
'\Demo 042E.sav'. 47
```

*To prepare to run "FTIAC" syntax, you must add "stat" from the Official Enrollment unduplicated headcount file.
*stat is the status in each course

```
MATCH FILES /FILE=*  
/TABLE='I:\Internal Reporting\Enrollment\Official Enrollment\Raw Data\EOS Spring 04\Summer I 2004HEADCOUNTjun3004+  
' Unduplicated.SAV' OTD Summer Jul 1004  
/RENAME (acs.1 acs.2 acs.3 acs.4 acs.5 acs.6 AH cont.1 cont.2 cont.3 cont.4 cont.5 cont.6 cred.1 cred.2 cred.3 cred.4 cred.5  
cred.6 CreditStudent DW end.1 end.2 end.3 end.4 end.5 end.6 first grde.1 grde.2 grde.3 grde.4 grde.5 grde.6 HL last loc.1  
loc.2 loc.3 loc.4 loc.5 loc.6 midd num.1 num.2 num.3 num.4 num.5 num.6 Orch pref.1 pref.2 pref.3 pref.4 pref.5 pref.6 RO run  
SF ssn strt.1 strt.2 strt.3 strt.4 strt.5 strt.6 syn.1 syn.2 syn.3 syn.4 syn.5 syn.6 term type.1 type.2 type.3 type.4 type.5  
type.6 = d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25 d26 d27 d28 d29  
d30  
d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59  
d60 d61  
d62 d63 d64 d65 d66 d67 d68 d69 d70 d71 d72 d73 d74 d75 d76 d77 d78)  
/BY id  
/DROP= d0 d1 d2 d3 d4  
d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31 d32 d33 d34 d35  
d36  
d37 d38 d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51 d52 d53 d54 d55 d56 d57 d58 d59 d60 d61 d62 d63 d64 d65  
d66 d67  
d68 d69 d70 d71 d72 d73 d74 d75 d76 d77 d78.  
EXECUTE.
```

```
*Next you have to compute status for the term.  
IF (stat = 'A' & (credah > 0 | credhl > 0 | credor > 0 | credro > 0 | credsf > 0 | creddw > 0)) status = 1 .  
VARIABLE LABELS status 'Student Status in Term' .  
EXECUTE .
```

```
*Run "FTIAC" syntax.  
*Save file.
```

SYNTAX: Transfers.SPS

```
GET DATA /TYPE = TXT  
/FILE = 'I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+  
'\COLLEGEJul1204.DAT'  
/DELCASE = LINE  
/DELIMITERS = ", "  
/QUALIFIER = ""  
/ARRANGEMENT = DELIMITED  
/FIRSTCASE = 2  
/IMPORTCASE = ALL  
/VARIABLES =  
id A10  
ssn A11  
lname A11  
fname A11  
mname A8  
homcmp A2  
prvcmp A2  
tcoll A35  
cred F8.2
```

```
.  
CACHE.  
EXECUTE.
```

```
FILTER OFF.  
USE ALL.  
SELECT IF(id ~= ' ' & id NE ' ').  
EXECUTE .
```

```
COMPUTE tcred = cred * .01 .  
EXECUTE .
```

```
SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+  
'\Transfer044T.sav'.
```

```
VARIABLE LABELS  
tcoll 'Transfer College'  
tcred 'Credits from Transfer College'.  
EXECUTE.
```

```
SORT CASES BY id .  
CASESTOVARS  
/ID = id  
/GROUPBY = INDEX .
```

```
SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+  
'\Transfer044T.sav'.
```

GET DATA /TYPE = TXT

/FILE = 'I:\Research Data\Student Information System\Summer 2004\Summer 11 2004 Raw Data'

~~'COLLEGE070604 (2004SU).DAT'~~

/DELCASE = LINE

/DELIMITERS = ", "

/QUALIFIER = ""

/ARRANGEMENT = DELIMITED

/FIRSTCASE = 2

/IMPORTCASE = ALL

/VARIABLES =

id A10

ssn A11

lname A11

fname A11

mname A8

homcmp A2

prvcmp A2

tcoll A35

cred F8.2

CACHE.

EXECUTE.

FILTER OFF.

USE ALL.

SELECT IF(id ~='□' & id NE ' ').

EXECUTE .

COMPUTE tcred = cred * .01 .

EXECUTE .

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer 11 2004 Raw Data'+
47
'\Transfer042E.sav'.

VARIABLE LABELS

tcoll 'Transfer College'

tcred 'Credits from Transfer College'.

EXECUTE.

SORT CASES BY id .

CASESTOVARS

/ID = id

/GROUPBY = INDEX .

SAVE OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer 11 2004 Raw Data'+
'\Transfer042E.sav'.

Syntax : " Previous OCC Degrees after Summer I 2003 "

```
DATA /TYPE = TXT
FILE = 'I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data'+
'DEGREESJul1204.DAT'
/DELIMITERS = ", "
/QUALIFIER = ""
/ARRANGEMENT = DELIMITED
/FIRSTCASE = 2
/IMPORTCASE = ALL
/VARIABLES =
id A10
ssn A11
lname A26
fname A15
mname A8
homcmp A2
prvcmp A2
major A3
spec A3
awd A4
awdmo A2
awdyr A2.
```

VARIABLE LABELS

```
major 'Primary Subject Area'
spec 'Specialization Area'
awd 'Previous OCC Award Type'
awdmo 'Previous OCC Degree Month'
awdyr 'Previous OCC Degree Year'.
```

```
CACHE.
EXECUTE.
```

```
STRING occdeg (A12).
VARIABLE LABELS occdeg 'Previous OCC Degree program' .
IF (spec ~= ' ') occdeg = CONCAT(major, '.', spec, '.', awd) .
IF (spec = ' ') occdeg = CONCAT (major, '.', awd).
EXECUTE .
```

```
FILTER OFF.
USE ALL.
SELECT IF(id ~= ' ' & id NE ' ').
EXECUTE .
```

```
SAVE OUTFILE='I:\Research Data\Student Information System'+
'\Summer 2004\Summer II 2004 Raw Data\Previous OCC Degrees 044T.sav'.
```

```
SORT CASES BY id .
CASESTOVARS
/ID = id
/GROUPBY = INDEX .
```

```
SAVE OUTFILE='I:\Research Data\Student Information System'+
'\Summer 2004\Summer II 2004 Raw Data\Previous OCC Degrees 044T.sav'.
```


Summer 11 2004 Raw Data

```
DATA /TYPE = TXT
FILE = 'I:\Research Data\Student Information System\Summer 2004\'+
'DEGREES070604 (2004SU).DAT' Jul 12 04, DAT
/DELIMITERS = ", "
/QUALIFIER = ""
/ARRANGEMENT = DELIMITED
/FIRSTCASE = 2
/IMPORTCASE = ALL
/VARIABLES =
id A10
ssn A11
lname A26
fname A15
mname A8
homcmp A2
prvcmp A2
major A3
spec A3
awd A4
awdmo A2
awdyr A2.
```

```
VARIABLE LABELS
major 'Primary Subject Area'
spec 'Specialization Area'
awd 'Previous OCC Award Type'
awdmo 'Previous OCC Degree Month'
awdyr 'Previous OCC Degree Year'.
```

```
CACHE.
EXECUTE.
```

```
STRING occdeg (A12).
VARIABLE LABELS occdeg 'Previous OCC Degree program' .
IF (spec ~= ' ') occdeg = CONCAT(major, '.', spec, '.', awd) .
IF (spec = ' ') occdeg = CONCAT (major, '.', awd).
EXECUTE .
```

```
FILTER OFF.
USE ALL.
SELECT IF(id ~= ' ' & id NE ' ').
EXECUTE .
```

```
SAVE OUTFILE='I:\Research Data\Student Information System'+
'\Summer 2004\Summer 11 2004 Raw Data\Previous OCC Degrees 042E.sav'. HT
```

```
SORT CASES BY id .
CASESTOVARS
/ID = id
/GROUPBY = INDEX .
```

```
SAVE OUTFILE='I:\Research Data\Student Information System'+
'\Summer 2004\Summer 11 2004 Raw Data\Previous OCC Degrees 042E.sav'. HT
```

Syntax: 'Reported Race Ethnicity.SPS'

STRING reprice (A8).

IF (fpalst = 'Y') reprice = 'NR' .

IF (fpalst NE 'Y') reprice = ethnic .

VARIABLE LABELS reprice 'Reported Race/Ethnicity' .

EXECUTE .

value labels reprice 'AF' 'African American' 'AS' 'Asian' 'HI' 'Hispanic' 'NI' 'Native American'
'UN' 'Race Unknown' 'WH' 'White' 'NR' 'Non Resident Alien'.

Syntax: "Campus credits for the Demo file, SPS"

**** Using duplicated course registration file, set the file up to calculate credits and attendance by campus.
**** This information will be merged into the Demographic file.

*Select all active students.
FILTER OFF.
USE ALL.
SELECT IF(stat = 'A').
EXECUTE .

IF (loc = 'AH') credah = cred .
IF (loc = 'HL') credhl = cred.
IF (loc = 'OR') credor = cred.
IF (loc = 'RO') credro = cred.
IF (loc = 'SF') credsf = cred.
IF (loc NE 'AH' & loc NE 'HL' & loc NE 'OR' & loc NE 'RO' & loc NE 'SF') creddw = cred.
VARIABLE LABELS credah 'Credits taken at Auburn Hills'
credhl 'Credits taken at Highland Lakes'
credor 'Credits taken at Orchard Ridge'
credro 'Credits taken at Royal Oak'
credsf 'Credits taken at Southfield'
creddw 'Credits taken District wide'.
EXECUTE .

AGGREGATE

/OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\Student campus credit counts'+
'044T.sav'
/BREAK=run term id stat
/credah 'Credits Taken at Auburn Hills' = SUM(credah)
/credhl 'Credits Taken at Highland Lakes' = SUM(credhl)
/credor 'Credits Taken at Orchard Ridge' = SUM(credor)
/credro 'Credits Taken at Royal Oak' = SUM(credro)
/credsf 'Credits Taken at Southfield' = SUM(credsf)
/creddw 'Credits Taken District-Wide' = SUM(creddw).

GET

FILE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\Student campus credit counts
044T.sav'.
SORT CASES BY
id (A) .

COMPUTE credreg = credah + credhl + credor + credro + credsf + creddw .
VARIABLE LABELS credreg 'Credits Registered This Term' .
EXECUTE .

*Return to the developing demographic file. Add all of the variables from the Aggregate file.

MATCH FILES /FILE=*

/TABLE='I:\Research Data\Student Information System\Summer 2004\Summer II 2004 Raw Data\Student campus credit counts
044T.sav'
/BY id.
EXECUTE.

**** For Summer I and Summer II select out those students not enrolled in the relevant portion of the term by selecting
**** those with run date (extract date) not equal to blank.

FILTER OFF.
USE ALL.
SELECT IF(run ~= ' ').
EXECUTE .

Syntax: "Campus credits for the Demo file .SPS"

**** Using duplicated course registration file, set the file up to calculate credits and attendance by campus. -

**** This information will be merged into the Demographic file.

"headcount summer II 2004.sav"

```
IF (loc = 'AH') credah = cred .
IF (loc = 'HL') credhl = cred.
IF (loc = 'OR') credor = cred.
IF (loc = 'RO') credro = cred.
IF (loc = 'SF') credsf = cred.
IF (loc NE 'AH' & loc NE 'HL' & loc NE 'OR' & loc NE 'RO' & loc NE 'SF') creddw = cred.
VARIABLE LABELS credah 'Credits taken at Auburn Hills'
credhl 'Credits taken at Highland Lakes'
credor 'Credits taken at Orchard Ridge'
credro 'Credits taken at Royal Oak'
credsf 'Credits taken at Southfield'
creddw 'Credits taken District wide'.
EXECUTE .
```

AGGREGATE

```
/OUTFILE='I:\Research Data\Student Information System\Summer 2004\Summer I\2004 Raw Data\Student campus credit counts'+
'042.sav'
/BREAK=run term id
/credah = SUM(credah) /credhl = SUM(credhl) /credor = SUM(credor) /credro = SUM(credro) /credsf = SUM(credsf) /creddw =
SUM(creddw).
```

GET

```
FILE='I:\Research Data\Student Information System\Summer 2004\Summer I\2004 Raw Data\Student campus credit counts
042.sav'.
SORT CASES BY
id (A) .
```

Return to the developing demographic file. Add all of the variables from the Aggregate file.

MATCH FILES /FILE=*

```
/TABLE='I:\Research Data\Student Information System\Summer 2004\Summer I\2004 Raw Data\Student campus credit counts
042.sav'
/BY id.
EXECUTE.
```

**** For Summer I and Summer II select out those students not enrolled in the relevant portion of the term by selecting
**** those with run date (extract date) not equal to blank.

FILTER OFF.

```
USE ALL.
SELECT IF(run ~= '').
EXECUTE .
```

Syntax: "FTIAC.SPS"

****Calculate FTIAC (First Time in Any College). Note the following: status refers to active
****status in at least one course. 'admst' (Admit Status) refers to the student's self-reported
****admission status, with FT being First Time in Any College. Previous Session (prevsess)
****null indicates no prior registration at OCC. Highest degree obtained previously (prevdeg)
****greater than 6 indicates high school graduate, GED, unknown, or hasn't graduated high
****school. Transfer college (tcoll) variables indicate activity at other colleges, except those
****indicated within the syntax which are Tech Prep programs at OCC. A credit program (crprog)
****of Guest or High School Dual Enrollment eliminates a student from being considered as
****first time in any college. Finally, presence of occdeg.1 (Previous OCC degree) would indicate
****attendance at OCC.

STRING ftiac (A8).

```
IF (status = 1 & admst = 'FT' & prevsess = '' & prevdeg > 6 & (tcoll.1 = '' | tcoll.1 =  
'TECH PREP BUSINESS INFORMATION SYST' | tcoll.1 = 'TECH PREP DRAFTING'+  
'/DESIGN' | tcoll.1 = 'TECH PREP EARLY CHILDHOOD DEVELOPME' | tcoll.1 = 'TECH'+  
'PREP HEALTH' | tcoll.1 = 'TECH PREP MEDICAL ASSISTING' | tcoll.1 = 'TECH'+  
'PREP STUDENT-APPLICANT' | tcoll.1 = 'ADVANCED PLACEMENT EXAMINATION-CEEB'  
| tcoll.1 = 'COLLEGE ENTRANCE EXAMINATION BOARD') & tcoll.2 = '' & crprog ~= 'GUE' & crprog ~=  
'HSG' & occdeg.1 = '') ftiac = 'Y' .
```

VARIABLE LABELS ftiac 'Calculated FTIAC' .
EXECUTE .