

Paper 074-2012

Uploading Your IPEDS Data Electronically Using SAS®

James Hume, Western Kentucky University, Bowling Green, Kentucky

ABSTRACT

This paper demonstrates how to use SAS® code to produce a text file that can be used to electronically upload data to the IPEDS Completions Survey. Universities have to provide a large amount of completion data to IPEDS, and this can be a time-consuming task if data is manually entered in the system. Using a few DATA steps, this paper shows you how to upload all of your completion data in a matter of minutes, saving you a tremendous amount of time.

INTRODUCTION

IPEDS is the Integrated Postsecondary Education Data System. Institutions that participate in federal student aid programs are required to enter data on enrollment, program completions, graduation rates, faculty and staff, finances, institutional prices, and student financial aid. This paper focuses specifically on the completions survey. The completions survey provides a count of the total number of undergraduate and graduate degrees and certificates awarded by the institution in the previous 12 months. Each program of study must be reported using a 6-digit Classification of Instructional Program (CIP) code, level awarded (associate's, bachelor's, master's, doctor's, and certificates), race/ethnicity, and gender of recipient.

While it might appear to be easier to manually enter information into the system, the truth remains that this becomes a tedious process where human error is prevalent. SAS will enable you to efficiently streamline this process by generating an accurate text file of all your completion data, where you can then electronically upload the data to IPEDS.

CORRECT FORMAT

In order to upload your completion information to IPEDS, it must be formatted specifically to IPEDS standards. We first need an understanding of the proper variables that are required by IPEDS. Below is a table outlining those variables:

Variable		Variable Description
UNITID	UNITID	Your specific University ID
SURVSECT	Survey Section	"COM" – Represents the Completions Survey
MAJORNUM	First or second major identification	1 = First Major, 2 = Second Major
CIPCODE	Classification of instructional program code	xx.xxxx, valid CIP codes, refer to CIP code table (must have leading zero if applicable)
AWLEVEL*	Award level	1 to 8 and 17 to 19 for MAJORNUM=1 (first major); 3, 5, 7, 17, 18, 19 for MAJORNUM=2 (second major); (refer to award level table) 1 = Postsecondary award, certificate, or diploma (of less than one academic year) 2 = Postsecondary award, certificate, or diploma (of at least one but less than two academic years) 3 = Associate's Degree 4 = Postsecondary award, certificate, or diploma (of at least two but less than four academic years) 5 = Bachelor's Degree or equivalent 6 = Postbaccalaureate Certificate 7 = Master's Degree 8 = Post-Master's Certificate 17 = Doctor's Degree – Research/scholarship 18 = Doctor's Degree – Professional Practice 19 = Doctor's Degree - Other

RACE**	Race	j = 1 to 10; 1 - Nonresident alien 2 - Hispanic/Latino 3 - American Indian or Alaska Native 4 - Asian 5 - Black or African American 6 - Native Hawaiian or Other Pacific Islander 7 - White 8 - Two or more races 9 - Race and ethnicity unknown 10 - Grand total will be generated on the export file. Do not include in import file.
SEX	Sex	k= 1-Men, 2-Women
COUNT	Number of awards	1 to 999999

Table 1. Proper IPEDS Format

TIP: The text file must be sorted in a particular order. CIP Code from smallest to largest. Within each CIP code, it must be sorted by Majornum, Award Level, Race, and then Sex. If you have a count of zero, then that line will be omitted from the text file.

BUILDING THE TEXT FILE

We first need to obtain our data showing which students completed a degree in the previous calendar year. If your CIP Codes are more than 6 characters long, use the SUBSTR function to gather the first 6. At this point we need to keep the Student ID, their gender, race, first major, the first major's award level, second major, and the second major's degree level. From here we'll be able to build on to this data set for the text file.

```

1 DATA DEGREE_FILE;
    SET DEGREE.DEGREES_201011;
    *ONLY NEED SIX DIGIT CIPS;
    MAJOR1_CIP = SUBSTR(MAJOR1_CIP, 1, 6);
    MAJOR2_CIP = SUBSTR(MAJOR2_CIP, 1, 6);
    KEEP STUDENT_ID GENDER RACE DEGREELEVEL_1 MAJOR1_CIP DEGREELEVEL_2
    MAJOR2_CIP;
    PROC SORT; BY MAJOR1_CIP DEGREELEVEL_1 GENDER;
    RUN;
  
```

TIP: A key part to building the text file is using PROC SORT. Play close attention to which variables are being sorted.

At this point, it's possible to have a student with more than one awarded degree. It is necessary to separate that second major and add that student to a new row. Our completion data is not showing a count per student, it's per CIP Code. So if a student has more than one degree completed during that calendar year, they are counted twice. This next string of code will separate the students who have more than one degree and create a new variable "CIP_CODE" that will be formatted to IPEDS standard.

2

```

DATA DEGREES2;
  SET DEGREES;
  *For First Major;
  CIP_CODE =
  CATX(".", (SUBSR(MAJOR1_CIP,1,2)), (SUBSTR(MAJOR1_CIP,3,6)));
  MAJOR_NUM=1;
OUTPUT;

  *For Second Major;
  IF MAJOR2_CIP NE "" THEN DO;
  CIP_CODE=CATX(".", (SUBSTR(MAJOR2_CIP,1,2)), (SUBSTR(MAJOR2_CIP,3,6
))) ;
  MAJOR_NUM=2;
OUTPUT;
END;
DROP DEGREELEVEL_1 DEGREELEVEL_2 MAJOR1_CIP MAJOR2_CIP;
RUN;

```

The next code places the proper IPEDS format for SEX, and RACE.

3

```

DATA DEGREES3;
  SET DEGREES2;
  IF GENDER="F" THEN GENDER="2";
  ELSE IF GENDER="M" THEN GENDER="1";
  IF RACE="1" THEN RACE="1";
  ELSE IF RACE="5" THEN RACE="2";
  ELSE IF RACE="3" THEN RACE="3";
  ELSE IF RACE="4" THEN RACE="4";
  ELSE IF RACE="2" THEN RACE="5";
  ELSE IF RACE="9" THEN RACE="6";
  ELSE IF RACE="6" THEN RACE="7";
  ELSE IF RACE="8" THEN RACE="8";
  ELSE IF RACE="7" THEN RACE="9";

RUN;
PROC SORT DATA=DEGREES3;
  BY CIP_CODE MAJOR_NUM GENDER RACE AW_LEVEL;
RUN;

```



It is possible to do a PROC FORMAT here for SEX and RACE, however for simplicity, we incorporated if-then statements.

Now we need to build the appropriate variables for the text file. We will create 6 new variables: UNITID, SURVSECT, MAJORNUM, CIPCODE, AWLEVEL, and SEX.

4

```

DATA DEGREES4 (RENAME=(RACE=RACE1));
  LENGTH UNITID $13;
  LENGTH SURVSECT $13;
  LENGTH MAJORNUM $4;
  LENGTH CIPCODE $8;
  LENGTH AWLEVEL $5;
  LENGTH SEX $2;
  SET DEGREES3;
  UNITID="UNITID=123456"; *Instead of 123456, you will enter your
  University's ID;

```

```

SURVSECT=" , SURVSECT=COM";
MAJORNUM=MAJOR_NUM;
CIPCODE=CIP_CODE;
SEX=GENDER;
AWLEVEL=AW_LEVEL;
DROP GENDER CIP_CODE MAJOR_NUM AW_LEVEL STUDENT_ID ;
LABEL RACE=RACE;
RUN;
PROC SORT DATA=DEGREES4;
BY CIPCODE MAJORNUM AWLEVEL RACE1 SEX;
RUN;

```

In order to get the count, we will use by group processing.

```

5 DATA DEGREES5;
  SET DEGREES4;
  BY CIPCODE MAJORNUM AWLEVEL RACE1 SEX;
  IF FIRST.SEX THEN COUNT=0;
  COUNT+1;
  IF LAST.SEX;
RUN;

```

One more data step will concatenate certain variables to incorporate the proper IPEDS format.

```

6 DATA DEGREES6;
  LENGTH ID $13;
  LENGTH SECTION $13;
  LENGTH MAJOR $11;
  LENGTH CODE $16;
  LENGTH LEVEL $11;
  LENGTH RACE $7;
  LENGTH DW_SEX $6;
  LENGTH COUNT1 $10;
  SET DEGREES5;
  ID=UNITID;
  SECTION=SURVSECT;
  MAJOR=CATS(' , MAJORNUM=' , MAJORNUM) ;
  CODE=CATS(' , CIPCODE=' , CIPCODE) ;
  LEVEL=CATS(' , AWLEVEL=' , AWLEVEL) ;
  RACE=CATS(' , RACE=' , RACE1) ;
  DW_SEX=CATS(' , SEX=' , SEX) ;
  COUNT1=CATS(' , COUNT=' , COUNT) ;
  DROP UNITID SURVSECT MAJORNUM CIPCODE AWLEVEL SEX RACE1 COUNT;
RUN;

```

Now use a simple PROC EXPORT to generate your text file:

```

7 DATA EXPORT;
  SET DEGREES6;
  FILE 'C:\COMPLETIONS_IMPORT.TXT' DLM= ',';
  PUT ID $13. SECTION $13. MAJOR $11. CODE $16. LEVEL $11.
    RACE $7. DW_SEX $6. COUNT1 $10.;
RUN;

```

Here is an excerpt of what your text file will look like:

```
UNITID=123456,SURVSECT=COM,MAJORNUM=1,CIPCODE=01.0000,AWLEVEL=7
,RACE=7,SEX=2,COUNT=4
UNITID=123456,SURVSECT=COM,MAJORNUM=2,CIPCODE=01.0000,AWLEVEL=5
,RACE=8,SEX=1,COUNT=1
UNITID=123456,SURVSECT=COM,MAJORNUM=1,CIPCODE=01.0301,AWLEVEL=3
,RACE=7,SEX=1,COUNT=1
UNITID=123456,SURVSECT=COM,MAJORNUM=1,CIPCODE=04.0901,AWLEVEL=5
,RACE=5,SEX=1,COUNT=1
UNITID=123456,SURVSECT=COM,MAJORNUM=1,CIPCODE=04.0901,AWLEVEL=5
,RACE=7,SEX=1,COUNT=8
UNITID=123456,SURVSECT=COM,MAJORNUM=1,CIPCODE=04.0901,AWLEVEL=5
,RACE=7,SEX=2,COUNT=4
UNITID=123456,SURVSECT=COM,MAJORNUM=1,CIPCODE=05.0207,AWLEVEL=6
,RACE=2,SEX=2,COUNT=1
```



Remember that the proper order is by CIPCODE, AWLEVEL, RACE, and then SEX. If it is not in this order, it will not load properly to IPEDS.

UPLOAD TO IPEDS

Now return to your IPEDS screen and go to “File Import/Upload” under the Tools section. Make sure to select the Completions Survey in the drop down box and click “Key Value Pair” above the file name. After this point, you are free to upload.

CONCLUSION

As you can see through the use of SAS, we are able to generate the proper text file to upload our completion data to IPEDS. Manually entering data to IPEDS can be such a monotonous and confusing process, human error is likely to happen. This text file is an excellent way to streamline efficiency in your office by uploading necessary information in a matter of minutes.

BIOGRAPHY

James Hume is currently the Database Analyst for the Office of Institutional Research at Western Kentucky University in Bowling Green, KY. James creates, maintains, and ensures data quality of all department databases. James prepares, validates, and submits data to the Kentucky Council on Postsecondary Education and the Integrated Postsecondary Education data System (IPEDS) in accordance with state and federal guidelines. James graduated from Western Kentucky University in 2007 and has worked at WKU since 2009.

Your comments and questions are valued and encouraged. Contact the author at:

Name:	James Hume
Enterprise:	Western Kentucky University
Address:	1906 College Heights Blvd.
City, State ZIP:	Bowling Green, KY 42101
Work Phone:	(270) 745-3250
E-mail:	james.hume@wku.edu

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.