



Maryland Annual Collection Systems

Maryland Approved Program Completer System

(MAPCS)

Maryland Higher Education Commission
Office of Information Technology
6 North Liberty Street • Ground Suite • Baltimore, MD 21201
(410) 767-3301
www.mhec.maryland.gov

Data Web Site: <https://data.mhec.state.md.us>

Document Control

A. Document Information

Title:	Maryland Approved Program Completer System
Security Level:	Unclassified – For Official Use Only
Filename:	MAPCS Manual.doc

B. Document History

Document Version:	Date:	Summary of Change
Version 1	3/30/2018	Piloted MAC2 MAPCS manual
Version 2	7/17/2019	Revised MAC2 MAPCS manual
Version 3	7/03/2020	Revised MAC2 MAPCS manual

C. For Assistance

Name	Phone	Email
		cc: mac2help.mhec@maryland.gov

Table of Contents

Maryland Approved Program Completer System	1
I. Maryland Approved Program Completer System	4
A. Data Use	4
B. Southern Regional Education Board (SREB).....	4
II. Collection Requirements.....	4
A. Changes from Prior Year.....	4
B. Due Date, Record Frequency and Collection Period.....	5
C. Selection Criteria.....	5
D. File Layout and Edit Checks	7
E. Data Cookbook Link.....	12
F. Sample Record Submissions	13
III. Requirements for Data Submission	14
A. MHEC Secure Data Website	14
B. Header Record	15
C. CSV File	15
D. File Transfer Process.....	15
IV. Frequently Asked Questions	18
A. MAPCS FAQs	18
B. General FAQs	20
V. Data Validation Syntax.....	21

I. Maryland Approved Program Completer System

A. Data Use

The purpose of this annual collection, the Maryland Approved Program Completer System (MAPCS), is to identify students who complete a Maryland State Approved Professional Education Program. Any student (degree-seeking or non-degree seeking) who completes a Maryland State Approved Professional Education Program will be reported in this collection. This collection centralizes data collected and aims to help answer questions of interest to institutions and the State regarding Maryland approved teacher program completion.

These include accountability measures included in the agency's Managing for Results as well as research projects as a result of collaboration with the Maryland Longitudinal Data System Center (MLDSC) on such issues as teacher pipeline and teacher workforce outcomes.

B. Southern Regional Education Board (SREB)

This aggregated collection supports MHEC's requirement to report to SREB. SREB is a non-profit organization that works to improve education at every level. The southern region covers 16 states.

II. Collection Requirements

A. Changes from Prior Year

MAPCS was piloted last year and after reviewing submissions and examining MHEC data needs further, it was determined that the following changes be made from the pilot file to the official file requested this year. The 2019-20 year will be the first official collection year for MAPCS.

- Removed the DIS MAPCS match flag field
- Added CIP Code field
- Added additional Program Completion Codes
 - Montessori - 8888
 - Math Instructional Leader (4-9) – 2222
- Expanded Program Completion codes (9999.99)
- Added last name and first name
- Added Capstone Location (optional for 2020 submission)

B. Due Date, Record Frequency and Collection Period

Annual Due Date: December 1st

Record Frequency: One record per student per program completion code. There can be multiple records for the same student if the student completes more than one program within the submission year.

Collection Period: September 1st of the prior year to August 31st of the current year. For example, the MAPCS 2019 should be inclusive of those with a completion date (degree date) between 9/1/18 and 8/31/19.

C. Selection Criteria**a. Who to include**

Degree-seeking students: For students who are degree-seeking at an institution, report any Maryland Approved Program Completion(s) as of the end of the collection period even if the student has not yet fulfilled all requirements to graduate, unless graduation is a requirement for the Approved Program. If graduation is a requirement above and beyond course work, then that student will not be defined as a MAPCS completer and will not be included in the collection until such time as the student graduates. Only report students in MAPCS who have completed all requirements for a Maryland State Approved Professional Education Program. Once a student graduates, if he/she returns to complete additional Maryland State Approved Professional Education Programs that are non-degree, follow the "Non-Degree" reporting instructions. See below in "Who to exclude" for more clarification.

Non-degree students: For students who are non-degree-seeking at an institution, report each Maryland State Approved Professional Education Program completion as it occurs. Specifically, if a student completes a program BUT is still in progress for another program, report only the completed program. See below in "Who to exclude" for more clarification.

b. Who to exclude

Degree-seeking students: Do not report students in MAPCS prior to graduation. If a student is ready to graduate, but will not complete the program requirements by graduation, then report the student as a graduate in DIS, but not as a completer in MAPCS. The student should be included in the MAPCS collection the year she completed the MAPCS program rather than the year she fulfills requirements to graduate.

Non-degree students: Do not report students who are in progress to completing a program. Only report completed programs.

MAC2 Maryland Approved Program Completer System (MAPCS)

One record per student per completion of a Maryland State Approved Professional Education Program submitted 1 time per year.

This collection should contain data on students who complete a Maryland State Approved Professional Education Program (degree and non-degree) in the collection period.

Item #	Minimum Field Length	Maximum Field Length	Field Type	Data Element	Data Dictionary Reference	Edit Checks	Description
1	1	1	Digit	Collection Term	DD2	Must correspond with Term upload selection of Annual code 8.	Use Term Identifier (Annual MAPCS Cycle - 8)
2	4	4	Digits	Collection Year	DD1	Must correspond with upload selection for year.	Collection Year (4 digits).
3	8	8	Digits	OPEID(FICE + 2)	DD3	Must match 6 digit FICE in UserID plus 2 zeros.	Institution Identifier using OPEID codes
4	0	1	Alpha/ Numeric	Sub-Campus Code	DD4	No edit checks.	For campus use (can be blank)
5	9	9	Digits	Identification Number (SSN)	DD5	Must be 9 digits. Must not be 000000000 or 999999999.	Social Security Number, Individual Taxpayer Identification Number or alternative student identifier (9 digits required, blank not valid)
6	1	1	Digit	Identification Number Type	DD6	Must = 1, 2, or 3	1 = SSN, 2=Institution Assigned Identifier, 3=ITIN
7	3	10	Alphanumeric	Local Campus Student ID	DD8	Must not be blank. Expecting at least 3 alphanumeric characters	Campus Assigned Student ID
8	0	10	Digits	SASID	DD9	May be blank if unknown otherwise 10 digits. Must not start with 0.	MSDE assigned Student identifier
9	1	1	Digit	Gender	DD21	Blank not valid. Must be 1, 2, 3, or 4.	Student gender or assigned gender if unknown.
10	1	1	Digit	US Citizenship	DD24	Must not be blank. Must be 1 or 2.	Identifies whether student is in the US Citizenship group or not.
11	0	1	Alphabetic	Hispanic/Latino Ethnicity	DD25	Must be Y or N or blank if unknown.	Identifies whether student is of Hispanic or Latino origin.
12	0	1	Digit	White	DD26.1	Must be 1 or blank.	Student identifies with White Race category.
13	0	1	Digit	Black/African American	DD26.2	Must be 2 or blank.	Student identifies with Black Race category.
14	0	1	Digit	Asian	DD26.3	Must be 3 or blank.	Student identifies with Asian Race category.
15	0	1	Digit	American Indian/ Native Alaskan	DD26.4	Must be 4 or blank.	Student identifies with Native American Race category.
16	0	1	Digit	Native Hawaiian/ Pac. Is.	DD26.5	Must be 5 or blank.	Student identifies with Native Hawaiian/ PI category.
17	1	1	Digit	Entry Term	DD116	Blank not valid. Must be 1, 2, 3, or 4.	Term of initial enrollment at this institution.
18	4	4	Digits	Entry Year	DD117	Blank not valid. Must be 4 digit year.	Year of initial enrollment at this institution.

MAC2 Maryland Approved Program Completer System (MAPCS)

One record per student per completion of a Maryland State Approved Professional Education Program submitted 1 time per year.


This collection should contain data on students who complete a Maryland State Approved Professional Education Program (degree and non-degree) in the collection period.

Item #	Minimum Field Length	Maximum Field Length	Field Type	Data Element	Data Dictionary Reference	Edit Checks	Description
19	8	8	Digits	Degree Date	DD112	Must be valid date, or if DD unknown use YYYYMM00.	Calendar Date in which the degree was conferred or the program was completed (if no degree conferred). (YYYYMMDD)
20	2	2	Digits	Degree Sought	DD30	Blank not valid. Valid codes are 40, 60, 47, 87 or 00.	Valid bachelor degree (40) or master's degree (60) code or non-degree code undergraduate(47), non-degree graduate (87) or unknown (00).
21	8	8	Digits	Birthdate	DD27	Sudent's date of birth. Format: YYYYMMDD	
22	6	6	Decimal	Program Completion Code	DD144	Blank not valid. Element expanded to 6 digits. 9999.99 Include decimal. Please see list of codes.	Identifies the Maryland State Approved Professional Education program completed by each student. Code list is maintained by MHEC. Include decimal. See List of Codes.
23	3	3	Digits	Geographic Origin	DD23	Permanent legal residence of student at time of application and admission.	
24	1	1	Digit	Residency Code	DD110	1 = MD resident, 2 = Non-resident	Indicates whether or not the student is a current resident of Maryland.
25	1	1	Digit	Military Status	DD145	Blank not valid. Must be 1, 2, 3, 4 , or 5.	Identifies whether student is: 1 = Active duty member of any of the US Armed Forces 2 = Veteran or former active duty member of any of the US Uniformed Forces 3 = Reserve duty member of any of the US Armed Forces, including the National Guard 4 = Spouse or dependent child of an active, reserve, or former member of the US Uniformed Forces 5 = None of the above
26	6	6	Digit	CIP Code	DD73	Use 13.9999 for MAPCS 9999.00	The Classification of Instructional Programs (CIP) code for the instructional program area of the student. Report the CIP code that corresponds to that on the Academic Program Inventory.
27	3	35	Character	Last Name	DD10	Must not be blank or numeric	Last Name (no numbers)
28	1	35	Character	First Name	DD11	Must not be blank or numeric	First Name (no numbers)
29	3	3	Digit	Capstone Location	DD156	Blank not valid. Numeric.	Identifies the county or state where the student completed an edTPA and/or final internship to fulfill requirements for the MPACS program code being reported. See List of Codes.


E. Data Cookbook Link

<https://community.datacookbook.com/institutions/mhec>

Data Cookbook is MHEC’s MAC2 on-line data dictionary. At the site on the “Functional Areas” tab, select the appropriate definitions link in the right column under sharing to view the data elements for one of the MAC2 collections. Each data element name is listed in **alphabetical order** with its functional definition



[Log in](#) [Learn more](#)



Community


Public

Maryland Higher Education Commission

MHEC is the State of Maryland’s higher education coordinating board responsible for establishing statewide policies for Maryland public and private colleges and universities and for-profit career schools. MHEC also administers state financial aid programs that affect students on a statewide basis.

<http://www.mhec.maryland.gov>

Public Contact
[MHEC Administrator](#)



Functional Areas
Shared Definitions
Shared Specifications
Shared Files

Functional Areas for Shared Content

Functional Areas are groupings this organization uses for its shared content. These typically represent different areas within the organization.

Functional Area	Description	Sharing
EIS	Enrollment Information Systems	60 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
EDS	Employee Data Systems	32 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
DIS	Degree Information Systems	33 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
FAIS	Financial Aid Information Systems	31 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
EOTS	End of Term System	19 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
ECS	External Credit System	9 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
CIS	Course Information System	19 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
SRS	System Registration System	17 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
NTS	Non-Degree Teacher Candidate System	10 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶
MAPCS	Maryland Approved Program Completer System	25 DEFINITIONS ▶ 0 SPECIFICATIONS ▶▶

F. Sample Record Submissions

Field#	Data Element Name	Sample Data Set#1	Sample Data Set#2
1.	Collection Term	8	8
2.	Collection Year	2019	2019
3.	OPEID+2	00205700	00205700
4.	Sub-Campus Code		
5.	Identification Number	123456789	123456789
6.	Identification Number Type	1	1
7.	Local Campus Student Identifier	0279853	0254017
8.	SASID	125789529	201459801
9.	Gender	1	2
10.	US Citizen	1	1
11.	Hispanic/Latino Ethnicity	N	Y
12.	White		1
13.	Black/African American	2	
14.	Asian		
15.	American Indian/Native Alaskan		
16.	Native Hawaiian/Pacific Islander		
17.	Entry Term	1	3
18.	Entry Year	2016	2015
19.	Degree Date	20190501	20190605
20.	Degree Sought	40	60
21.	Birthdate	19960820	19950705
22.	Program Completion Code	0321.00	0517.02
23.	Geographic Origin	210	131
24.	Residency Code	1	1
25.	Military Status	5	4
26.	CIP Code	13.0301	13.1318
27.	Last Name	Doe	Due
28.	First Name	John	Jane
29.	Capstone Location	203	215

Data sets are to be submitted without blank spaces between the commas. If you have a field with no data like the sample data below, the commas will be side by side with no embedded space in between.

Sample Data Set 1

8,2019,00205700,,123456789,1,0279853,125789529,1,1,N,,2,,,,,1,2016
 ,20190501,40,19968020,0321.00,210,1,5,13.0301,Doe,John,203

Sample Data Set 2

8,2019,00205700,,123456789,1,0254017,201459801,2,1,Y,1,,,,,3,2015
 ,20190605,60,19950705,0517.02,131,1,4,13.1318,Doe,Jane,215

III. Requirements for Data Submission

A. MHEC Secure Data Website

The Secure Data Website (<http://data.mhec.state.md.us/>) is the starting point for all files and survey submissions. All uploaded unit files MUST be a comma separated value (csv) file and have a .csv extension for a successful submission. Files with any other extension will not upload successfully. The site provides calendars with due dates, file layouts, documentation and the secure area for file transfers. Auxiliary files such as program inventory, trend data and institution codes, are available as well. The site is updated each year to reflect current due dates and updated guidance on documentation and file submissions. For technical issues, guidance and questions/comments, please send an email to mac2help.mhec@maryland.gov.

The screenshot shows the MHEC Secure Data Web website. At the top, there is a navigation bar with links for HOME, CALENDARS, DOCUMENTATION, FILE TRANSFER, and SURVEYS. Below this, the page is divided into three main sections:

- MAC Calendars:** A list of links for different types of institutions: Community College, Public Four Year, Independent, Private, and Survey Coordinators.
- MHEC Secure Data Web:** A section titled "Morgan State University Lecture" with a photograph of a group of students sitting around a table in a classroom or meeting setting.
- Announcements/News:** A section titled "MHEC uses Data Cookbook for Unit Record Data Collection Systems." It includes a paragraph of text and a link for technical issues: "For technical issues, guidance and questions/comments, please send an email to mac2help.mhec@maryland.gov."

B. Header Record

The header record is required as part of the uploaded csv file and can be retrieved from the data website for each unit record system

http://data.mhec.state.md.us/mac2_documentation.asp.

Collection_Term,Collection_Year,OPEID,Sub_Campus_Code,
Identification_Number_SSN,Identification_Number_Type,
Local_Campus_Student_Identifier,SASID,Gender,US_Citizenship,
Hispanic_Latino_Ethnicity,White,Black_African_American,Asian,
American_Indian_Native_Alaskan,Native_Hawaiian_Pac_Is,Entry_Term,
Entry_Year,Degree_Date,Degree_Sought,Birthdate,Program_Completion_Code,
Geographic_Origin,Residency_Code, Military_Status,
CIP_Code,Last_name,First_name,Capstone_location

C. CSV File

All uploaded unit record files MUST be a comma-separated value (csv) file and have a .csv extension for a successful submission. All records in the file are to be submitted WITHOUT blank spaces between the commas. If there is a field with no data, the commas will be side by side with no embedded space in between.

The csv format is important because it allows flexibility for importing and exporting data with ease. Embedded spaces between commas cause data to be misaligned which cause erroneous results when processing files.

D. File Transfer Process

Once the collection file is ready for submission, upload using the Commission's Data Web Site. This is the most secure and only method for file transfers. The site requires an MHEC supplied USER Name and password.

The USER Name structure for submitting a unit record file is the institution 6-digit code prefaced with a 'P' followed by lower case 'a'. Example: P123456a.

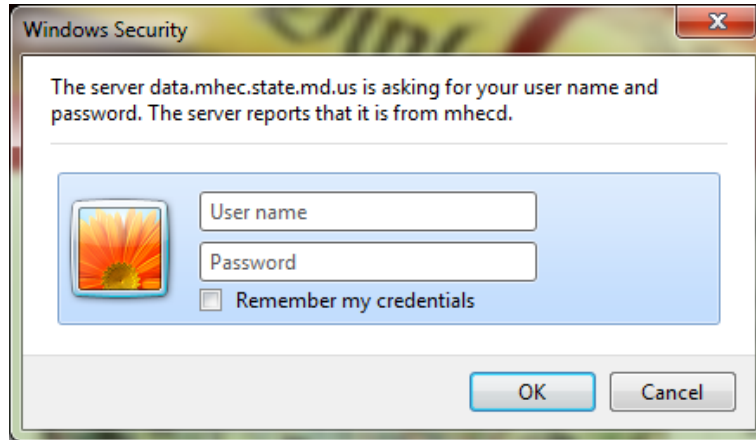
The MAPCS collection requires a FINAL submission only at this time. When submission edits are in place you will be notified. In response to the submission, an email is sent as verification and receipt. The on-line form completed as the final or Official Submission, serves as your official Institutional Certification and Specification form. This form certifies that data submitted is accurate.

1. Final Submission

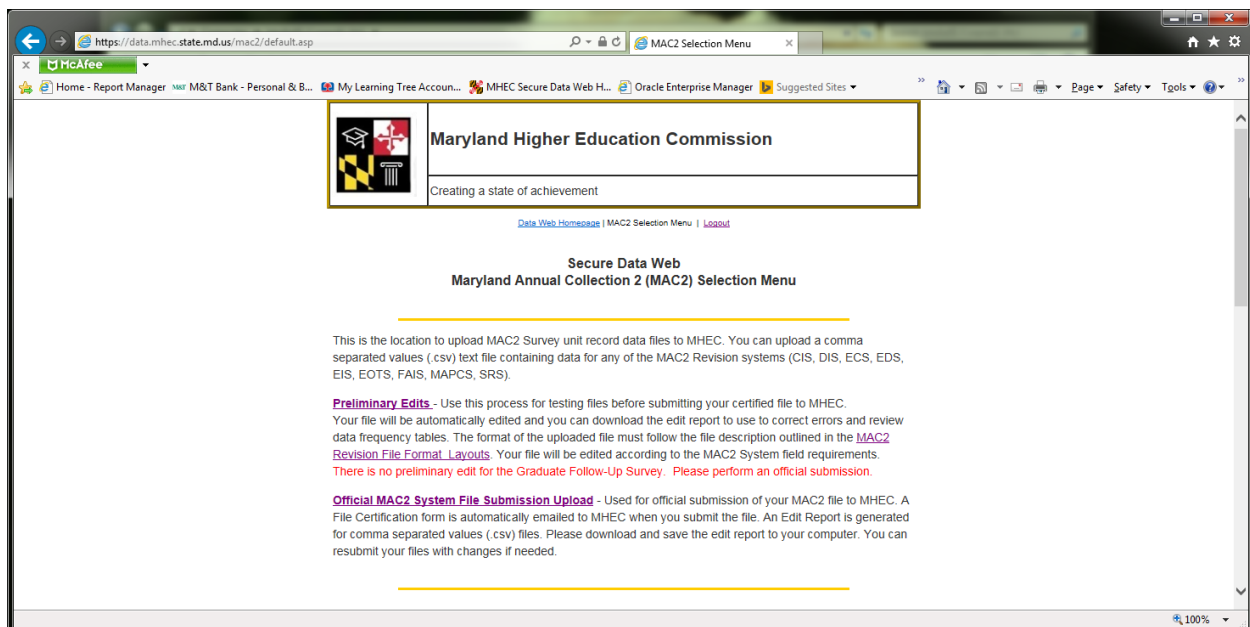
The OFFICIAL MAC2 System File Submission Upload is used for the official submission of the MAC2 file to MHEC. A File Certification form is automatically emailed to MHEC when the file is submitted. An Edit Report is generated for comma separated values (.csv) files. Please download and save the edit report to your computer. The files may be resubmitted with changes if needed.

The steps to submit the Official/Final collection file are:

- a. Under secure file transfer on the left panel of the home screen, select “MAC2 Unit Record.”
- b. The authentication box will open for your User name and Password.



- c. Once logged in, the file upload screen will display.



Once on this screen, select the Official MAC2 System File Submission option to submit the file. A certification form will be automatically emailed when the file is submitted.



Maryland Higher Education Commission

Creating a state of achievement

[Data Web Homepage](#) | [MAC2 Menu](#) | [Logout](#)

Secure Data Web
Maryland Annual Collection 2 (MAC2) - File Certification Form

Upload File for Maryland Higher Education Commission

It is important that all information is completed to ensure accurate transfer of data from your network to the MHEC secure data web. Use the "Browse" button to locate the file on your network that you want to upload. After completing the file certification form use the "Submit" button to initiate the upload .

Submission Date 1/30/2019 for Maryland Higher Education Commission

The file submission upload is for the system indicated below in the standard MHEC record format and prepared in accordance with MHEC instructions and data definitions.

Name:

Title:

Email:

Phone:

Please supply information about system file submitted.

Select System Name	Year	Term	Number of Records
Collection Type	Year	Choose Term	

Select Choose File button below to find the file on your network:

No file selected.

Please enter any notes or comments here:

By submitting this form you certify that the information submitted on this file is correct and true to the best of your knowledge. Click the submit button to continue with the upload.

There is a comment box on the screen that can be used to convey messages to the MHEC staff. Typical uses of the comment are to document if there is a valid reason why a fatal error(s) cannot be resolved.

Complete the form and select submit.

IV. Frequently Asked Questions

A. MAPCS FAQs

1. Which institutions are required to submit MAPCS files to MHEC?

Only four-year public and independent institutions that have Maryland Approved Programs are required to submit a MAPCS file.

2. How many times is MAPCS collected during the year?

MAPCS is collected once per year. The file is due on December 1st. The file should contain one record per student per program completion code. There can be multiple records for the same student if the student completes more than one program within the submission year.

3. Who should be included?

Degree-seeking students: For students that are degree-seeking at an institution, report any Maryland Approved Program Completion(s) as of the end of the collection period. Only report students in MAPCS that have completed all requirements for a Maryland State Approved Professional Education Program. Once a student graduates, if he/she returns to complete additional Maryland State Approved Professional Education Programs, follow the "Non-Degree" reporting instructions. If a degree-seeking student has completed the Approve Program requirement in a collection year, but, is not yet ready to graduate, INCLUDE that student in the MAPCS collection, but not in the DIS collection, unless graduation is a requirement for the Approved Program. If graduation is a requirement above and beyond course work, then that student will not be defined as a MAPCS completer and will not be included in the collection until such time as the student graduates.

Non-degree students: For students who are non-degree-seeking at an institution, report each Maryland State Approved Professional Education Program completion as it occurs. Specifically, if a student completes an Approve Program BUT is still in progress for another program, report only the completed program.

4. Who should be excluded?

Degree-seeking students: If a student is ready to graduate, but will not complete the Approve Program requirements by graduation, then report the student as a graduate in DIS, but not as a completer in MAPCS. If a degree-seeking student has completed the Approve Program requirement in a collection year, but, is not yet ready to graduate, INCLUDE that student in the MAPCS collection, but not in the DIS collection.

Non-degree students: Do not report non-degree seeking students who are in-progress to completing an Approved Program. Only report non-degree seeking students with completed Approved Programs.

5. What if there is no program completion code or CIP code that aligns with the student's program?

If there is no program completion code that aligns with the student's program, the records should be submitted with a code of 9999.00. If there is no CIP code that aligns with the student's program, the record should be submitted with a CIP code of 13. 9999.

6. What entry term and entry year am I to use?

The entry term and entry year recorded with the each record should correspond to the term and year the student began that specific Approved Program, not the degree program. This is not the term and year a student first enrolled at an institution. It is the term and year each Approved Program was begun.

7. What geographic code and residency code am I to use?

The geographic code should correspond to the student's legal resident location status at the time the student was admitted to the Approved Program, not when the student was admitted to the institutions or degree program. The residency code should reflect the student's residency as of the MAPCS reporting period.

8. How are data used from this collection?

These include accountability measures included in the agency's Managing for Results as well as research projects as a result of collaboration with the Maryland Longitudinal Data System Center (MLDSC) on such issues as teacher pipeline and teacher workforce outcomes.

B. General FAQs**1. What should I do if my submission will be late?**

You may send an email to the data collection manager and to mac2help.mhec@maryland.gov to request an extension up to two weeks beyond the established calendar due date. If the extension is granted, you must keep the data collection manager abreast of the progress of your file submission. It often results in missed deadlines, and extra time and effort to accommodate a late submission. Institutions are encouraged to submit data by the deadline to avoid the need for extensions.

2. What happens with a re-submitted file?

Institutions are encouraged to validate and submit the best quality data by the deadline to minimize the need for file changes and resubmissions.

3. If I resubmit a file after the collection window closes what happens with my submitted changes?

MHEC will accept the file with the changes. MHEC will review the discrepancy between the old and new submissions to determine whether the errors are substantial enough to warrant updating the data tables and reports.

4. What should I do if I have a social security number correction?

Notify the data collection manager for the system immediately as a revised file may be requested.

5. What if I cannot resolve a fatal error?

If a fatal error cannot be resolved, then an explanation must be documented in the comment box on the official file submission page. MHEC may have follow-up questions regarding the explanation and a revised file could be requested.

6. Why is it important to use the same student identifier with each file submission?

MHEC produces trend analyses and reports that link identifiers from year to year and it is important to ensure that the identifier is consistent with the prior year for an accurate results. In addition, MHEC links identifiers from one data collection to another; again, it is important that identifiers are consistent across collections for accurate results. When a student moves from one institution to another, MHEC must be able to detect the movement and the only way to do so is through the use of a universally available identifier which is why social security number is the preferred identifier for all submissions. An institutionally assigned identifier should only be used for students without social security numbers.

V. Data Validation Syntax

The following Stata code is used to validate all file submissions. Additional validation testing may be completed contingent upon the results of this baseline code. New validation steps may be added over time as the scope of the collection changes.

```

*file population*
*compare totals for each tab to prior years*
*compare sic to fice for correct sic construction*
table fice /*also known as OPEID*?
table sic /*MHEC constructed institution level identifier*/
table degree
table mdpgcode

*count unique populations*
*first instance of each unit record within an fice is marked as unique, additional instances as
duplciate*
*add to excel tab and cross check against prior year totals*
sort fice idn
egen num_unique=tag(fice idn)

label define counts 1 "unique" 0 "duplicate"
label values num_unique counts

tab num_unique
tab fice if num_unique==1

*****
*validate FICE-SIC alignment*

gen sicfice=.
replace sicfice=1 if fice==143600& sic==260700
replace sicfice=1 if fice==205700& sic==110100
replace sicfice=1 if fice==205800& sic==110200
replace sicfice=1 if fice==206100& sic==111300
replace sicfice=1 if fice==206200& sic==120600
replace sicfice=1 if fice==206300& sic==111250
replace sicfice=1 if fice==206400& sic==111000
replace sicfice=1 if fice==206500& sic==263300
replace sicfice=1 if fice==206700& sic==261200
replace sicfice=1 if fice==206800& sic==121400
replace sicfice=1 if fice==207100& sic==111700
replace sicfice=1 if fice==207200& sic==121800
replace sicfice=1 if fice==207300& sic==262000
replace sicfice=1 if fice==207400& sic==112100
replace sicfice=1 if fice==207500& sic==112200
replace sicfice=1 if fice==207600& sic==262300

```

```

replace sicfice=1 if fice==207700& sic==262500
replace sicfice=1 if fice==207800& sic==262600
replace sicfice=1 if fice==208000& sic==262700
replace sicfice=1 if fice==208300& sic==143000
replace sicfice=1 if fice==208600& sic==263100
replace sicfice=1 if fice==208900& sic==113600
replace sicfice=1 if fice==209100& sic==123900
replace sicfice=1 if fice==209200& sic==263700
replace sicfice=1 if fice==209500& sic==154000
replace sicfice=1 if fice==209900& sic==124200
replace sicfice=1 if fice==210200& sic==124400
replace sicfice=1 if fice==210300& sic==124700
replace sicfice=1 if fice==210400& sic==124500
replace sicfice=1 if fice==210500& sic==124600
replace sicfice=1 if fice==210600& sic==124800
replace sicfice=1 if fice==210700& sic==265000
replace sicfice=1 if fice==210800& sic==265200
replace sicfice=1 if fice==210900& sic==265400
replace sicfice=1 if fice==465000& sic==111100
replace sicfice=1 if fice==691100& sic==112970
replace sicfice=1 if fice==817500& sic==112400
replace sicfice=1 if fice==830800& sic==110900
replace sicfice=1 if fice==1001400& sic==111900
replace sicfice=1 if fice==1164400& sic==124900
replace sicfice=1 if fice==2073900& sic==115470
replace sicfice=1 if fice==3100700& sic==110770

```

```
tab sicfice
```

```
*****
```

```
*Step 2: Duplicate Check*
```

```
*this must be done first as variables will be constructed throughout validation that will
```

```
*alter results when looking for duplicates across all variables*
```

```
sort fice idn
```

```
*dups on all variables*
```

```
*no records should be identical on all elements*
```

```
duplicates tag, generate(dups)
```

```
tab dups
```

```
*validate counts are same on idn, tssn and tssn2 to make sure idn is correctly converted*
```

```
*dups on ssn only*
```

```
duplicates tag idn, generate (dupssn)
```

```
duplicates tag tssn, generate (duptssn)
```

```
duplicates tag tssn2, generate (duptssn2)
```

```
tab dupssn
tab duptssn
tab duptssn2
```

duplicate checking matching on ssn, fice, mapcs code

no records should be included twice with same code

sort fice idn mdpgcode

duplicates tag idn fice mdpgcode, generate(ssn_pg)

duplicates tag tssn fice mdpgcode, generate(tssn_pg)

duplicates tag tssn2 fice mdpgcode, generate(tssn2_pg)

tab ssn_pg

tab tssn_pg

tab tssn2_pg

tab fice ssn_pg if ssn_pg!=0

validate that when multiple records are present, key data elements are identical in all records

static elements shifting, must encode string hispanic

other elements may change between mapcs award (entry year, entry term, georg, res, etc. are specific to the MAPCS code, demographics should not change

encode hispanic, gen (hislat)

tab hispanic

tab hislat

bysort fice idn: generate chcit = citizen - citizen[_n-1]

bysort fice idn: generate chhis = hislat - hislat[_n-1]

bysort fice idn: generate chwh = rwhite - rwhite[_n-1]

bysort fice idn: generate chbk = rblack - rblack[_n-1]

bysort fice idn: generate chas = rasian - rasian[_n-1]

bysort fice idn: generate chai = raian - raian[_n-1]

bysort fice idn: generate chnh = rnhpi - rnhpi[_n-1]

bysort fice idn: generate chsex = sex - sex[_n-1]

bysort fice idn: generate chidt = idt - idt[_n-1]

bysort fice idn: generate chflg = mapcsflg - mapcsflg[_n-1]

tab chcit /*0 is no change, all other numbers are measure of change*/

tab fice if (chcit!=0& chcit!=.) & num_unique==0 /*changer*/

tab chhis /*0 is no change, all other numbers are measure of change*/

tab fice if (chhis!=0&chhis!=.) & num_unique==0 /*changer*/

tab chwh /*0 is no change, all other numbers are measure of change*/

```

tab fice if (chwh!=0 &chwh!=.)& num_unique==0 /*changer*/

tab chbk /*0 is no change, all other numbers are measure of change*/
tab fice if (chbk!=0&chbk!=.)& num_unique==0 /*changer*/

tab chas /*0 is no change, all other numbers are measure of change*/
tab fice if (chas!=0 & chas!=.) & num_unique==0 /*changer*/

tab chai /*0 is no change, all other numbers are measure of change*/
tab fice if (chai!=0 & chai!=.) & num_unique==0 /*changer*/

tab chnh /*0 is no change, all other numbers are measure of change*/
tab fice if (chnh!=0 &chnh!=.) & num_unique==0 /*changer*/

tab chsex /*0 is no change, all other numbers are measure of change*/
tab fice if (chsex!=0&chsex!=.) & num_unique==0 /*changer*/

tab chidt /*0 is no change, all other numbers are measure of change*/
tab fice if (chidt!=0&chidt!=.) & num_unique==0 /*changer*/

tab chflg /*0 is no change, all other numbers are measure of change*/
tab fice if (chflg!=0&chflg!=.) & num_unique==0 /*changer*/

*****
***
*validate individual data elements*
*valid fice, 1 valid, -9 not valid*

gen vfice=.
replace vfice=1 if fice==143600 | fice==205700 | fice==205800 | fice==206100 | fice==206200 |
fice==206300 | fice==206400 | ///
fice==206500 | fice==206700 | fice==206800 | fice==207100 | ///
fice==207200 | fice==207300 | fice==207400 | fice==207500 | fice==207600 | fice==207700 | ///
fice==207800 | fice==208000 | fice==208600 | fice==208700 | fice==208900 | ///
fice==209100 | fice==209200 | fice==209500 | fice==209600 | fice==209900 | fice==210200 |
fice==210300 | ///
fice==210400 | fice==210500 | fice==210600 | fice==210700 | fice==210800 | fice==210900 |
fice==465000 | ///
fice== 691100 | fice==795900 | fice==817500 | fice==830800 | fice==1001400 | fice==1164400 |
fice== 2073900 | ///
fice==208300 | fice==2127900 | fice==2314800 | fice==2578400 | fice==3100700 | fice==3455500 |
fice==90795900 | fice==99765900

mvdecode vfice, mv(-9)

tab vfice /*valid fice, 1 valid, -9 not valid*/

```

```
tab fice if vface== -9 /*list of invalid fice numbers*/
```

```
*****
```

```
*valid sic, 1 valid, -9 not valid*
```

```
gen vsic=.
```

```
replace vsic=1 if sic==110100 | sic==110200 | sic==110770 | sic==110900 | sic==111000 | ///
sic==111100 | sic==111250 | sic==111300 | sic==111700 | sic==111900 | sic==112100 | ///
sic==112200 | sic==112400 | sic==112970 | sic==113600 | sic==115470 | sic==120600 | ///
sic==121400 | sic==121800 | sic==123900 | sic==124200 | sic==124400 | sic==124500 | ///
sic==124600 | sic==124700 | sic==124800 | sic==124900 | sic==124960 | sic==124965 | ///
sic==143000 | sic==154000 | sic==260400 | sic==260410 | sic==260469 | sic==260537 | ///
sic==260700 | sic==262000 | sic==262050 | sic==262230 | sic==262300 | sic==262487 | ///
sic==262500 | sic==262570 | sic==262600 | sic==262650 | sic==262700 | sic==262767 | ///
sic==263100 | sic==263124 | sic==263200 | sic==263300 | sic==263500 | sic==263700 | ///
sic==263915 | sic==264100 | sic==264150 | sic==264182 | sic==264187 | sic==264300 | ///
sic==265000 | sic==261200 | sic==265100 | sic==265200 | sic==265300 | sic==265400 | ///
sic==265420 | sic==265600 | sic==261575 | sic==263575 | sic==260500 | sic==261150 | ///
sic==261450 | sic==110800 | sic==111500 | sic==111600 | sic==112800 | sic==112900 | ///
sic==112950 | sic==260250
```

```
mvdecode vsic, mv(-9)
```

```
tab vsic, missing /*1 valid, -9 not valid*/
```

```
tab sic if vsic== -9 /*list of invalid sic numbers*/
```

```
*****
```

```
*valid collection term, 1 is valid, 2 is not valid*
```

```
gen vct = .
```

```
*select one of the following*
```

```
replace vct = 1 if coltm == 8
```

```
replace vct = 2 if coltm !=8
```

```
tab vct /*1 is valid, 2 is not valid*/
```

```
tab fice if vct==2 /*1 is valid, 2 is not valid*/
```

```
*****
```

```
*valid collection year, 1 is valid, 2 is not valid*
```

```
*change year as needed*
```

```
gen vcy = .
```

```
*select one of the following*
```

```
replace vcy = 1 if colyr == 2018
```

```
replace vcy = 2 if colyr != 2018
```

```
replace vcy = 1 if colyr == 2019
replace vcy = 2 if colyr != 2019
```

```
replace vcy = 1 if colyr == 2020
replace vcy = 2 if colyr != 2020
```

```
tab vcy /*1 is valid, 2 is not valid*/
tab fice if vcy==2 /*1 is valid, 2 is not valid*/
```

```
*****
```

```
*sex, 1 valid, 2 not valid*
gen vsex=.
replace vsex=1 if sex==1 | sex==2 | sex==3 | sex==4
replace vsex=2 if sex>=5 | sex==. | sex<1
```

```
tab sex
tab vsex /*1 is valid, 2 is not valid*/
tab fice if vsex==2 /*1 is valid, 2 is not valid*/
```

```
*****
```

```
*citizenship, 1 valid, 2 not valid*
gen vcit=.
replace vcit=1 if citizen==1 | citizen==2
replace vcit=2 if citizen<1 | citizen>2 | citizen==.
```

```
tab citizen
tab vcit /*1 is valid, 2 is not valid*/
tab fice if vcit==2 /*1 is valid, 2 is not valid*/
```

```
*****
```

```
*race/ethnicity, 1 valid, 2 not valid*

gen vhispanic=.
replace vhispanic=1 if hislat==1 | hislat==2 | hislat==.
replace vhispanic=2 if hislat!=1 & hislat!=2 & hislat!=.
mvencode hislat, mv(9)
tab hispanic
tab hislat
tab vhispanic /*1 is valid, 2 is not valid*/
tab fice if vhispanic==2 /*1 is valid, 2 is not valid*/
```

```
gen vrwhite=.
replace vrwhite=1 if rwhite==1 | rwhite==.
replace vrwhite=2 if rwhite!=1 & rwhite!=.
tab vrwhite /*1 is valid, 2 is not valid*/
tab fice if vrwhite==2 /*1 is valid, 2 is not valid*/
```

```

gen vrblack=.
replace vrblack=1 if rblack==2 | rblack==.
replace vrblack=2 if rblack!=2 & rblack!=.
tab vrblack /*1 is valid, 2 is not valid*/
tab fice if vrblack==2 /*1 is valid, 2 is not valid*/

```

```

gen vrasian=.
replace vrasian=1 if rasian==3 | rasian==.
replace vrasian=2 if rasian!=3 & rasian!=.
tab vrasian /*1 is valid, 2 is not valid*/
tab fice if vrasian==2 /*1 is valid, 2 is not valid*/

```

```

gen vamer=.
replace vamer=1 if raian==4 | raian==.
replace vamer=2 if raian!=4 & raian!=.
tab vamer /*1 is valid, 2 is not valid*/
tab fice if vamer==2 /*1 is valid, 2 is not valid*/

```

```

gen vhawi=.
replace vhawi=1 if rnhpi==5 | rnhpi==.
replace vhawi=2 if rnhpi!=5 & rnhpi!=.
tab vhawi /*1 is valid, 2 is not valid*/
tab fice if vhawi==2 /*1 is valid, 2 is not valid*/

```

```

*****
*birthdt - must be 8 digits with at least a valid 4 digit year + 0000 or can be 00000000 if unknown*
*considering birthdt between 1900 and 2017 valid, tagging unknowns as 3 and 4*

```

```

gen vbirth=.

replace vbirth=1 if birthdt >=19170000 & birthdt <=20100000
replace vbirth=2 if birthdt <19170000 & birthdt!=0
replace vbirth=2 if birthdt >20100000 & birthdt!=0
replace vbirth=3 if birthdt==.
replace vbirth=4 if birthdt==0

tab vbirth /*1 is valid, 2 is out of range, 3 is blank, 4 is 0*/
tab fice vbirth if vbirth>1

```

```

*****
gen vmilstat=.
replace vmilstat=1 if milstat>=1 & milstat <=5
replace vmilstat=2 if milstat==.
replace vmilstat=3 if milstat>5
replace vmilstat=4 if milstat<1

```



```

tab milstat
tab vmilstat /*1 is valid, 2 is blank, 3 is out of range*/
tab fice if vmilstat>1

```

```

*****

```

```

*geographic orgin*
*001 - foreign, 100 - unknown US state, 101-199, 200- maryland unknown count, 201-224, 000-
unknown*

```

```

gen vgeorg=.
replace vgeorg=1 if georg==1
replace vgeorg=3 if georg==100
replace vgeorg=4 if georg==0
replace vgeorg=5 if georg==200
replace vgeorg=6 if georg>200 & georg<=224
replace vgeorg=7 if georg>100 & georg<=199
replace vgeorg=2 if georg==. |georg>224|georg<0

```

```

tab vgeorg
tab fice if vgeorg==3 /*100 - unknown US state*/
tab fice if vgeorg==4 /*unknown*/
tab fice if vgeorg==5 /*Unknown MD county*/
tab fice if vgeorg==2 /*value out of range*/

```

```

*****

```

```

*mdresidency code*
gen vmdres=.
replace vmdres=1 if mdres==1|mdres==2
replace vmdres=3 if mdres>2
replace vmdres=2 if mdres==.
replace vmdres=3 if mdres<1

```

```

tab vmdres /*1 is valid, 2 is blank, 3 is out of range*/
tab fice if vmdres>1

```

```

*****

```

```

*valid ID Type, valid 1, invalid 2*
gen vtype=.
replace vtype = 1 if idt ==1 | idt==2 | idt==3
replace vtype = 2 if idt !=1 & idt !=2 & idt!=3 & idt !=.

```

```

tab vtype /*1 is valid, 2 not valid*/

```

```

*****

```

```

*valid SSN, 1 is valid, 2 is 8 digits, 3 is 10 or more digits, 4 is 7 or less digits*
*8 digits may indicate missing leading 0*

```

```
gen vssn =idn
tostring vssn, replace
```

```
gen vssn2=.
replace vssn2=1 if strlen(vssn)==9
replace vssn2=2 if strlen(vssn)==8
replace vssn2=3 if strlen(vssn)>=10
replace vssn2=4 if strlen(vssn)<=7
tab vssn2
```

```
*validate against id type*
tab vssn2 if idt==1 &(vssn2==3|vssn2==4) /*number of invalid IDNs with ID Type =SSN*/
tab fice if idt==1 &(vssn2==3|vssn2==4) /*fice with number of invalid IDNs with ID Type =SSN*/
```

```
tab vssn2 idt
```

```
*****
```

```
*validate TSSN*
*tssn should be 10 characters*
gen vtssn=.
replace vtssn=3 if strlen(tssn)>10
replace vtssn=2 if strlen(tssn)<10
replace vtssn=1 if strlen(tssn)==10
tab vtssn /*1 is valid, 2 & 3 not valid*/
```

```
*****
```

```
*validate TSSN2*
*tssn2 should be 7 characters*
gen vtssn2=.
replace vtssn2=3 if strlen(tssn2)>7
replace vtssn2=2 if strlen(tssn2)<7
replace vtssn2=1 if strlen(tssn2)==7
tab vtssn2 /*1 is valid, 2 & 3 not valid*/
```

```
*****
```

```
*valid campusID 1 valid, 2 not valid*
*campus id is 10 or less characters*
gen vcamp=.
replace vcamp=2 if strlen(tssn)>=11
replace vcamp=1 if strlen(tssn)<=10
tab vcamp /*1 is valid, 2 not valid*/
```

```
*****
```

```
*valid SASID, code missing values as 9*
```

```
gen vsasid =sasid
mvencode vsasid, mv(9)
tostring vsasid, replace
```

```
*valid SASID*, 1 and 2 are valid, 3 and 4 are not valid
gen vsasid2=.
replace vsasid2=4 if strlen(vsasid)>10 /* not valid, too long*/
replace vsasid2=3 if strlen(vsasid)<10 &strlen(vsasid)>1 /* not valid, too short*/
replace vsasid2=2 if strlen(vsasid)==10 /*valid*/
replace vsasid2=1 if strlen(vsasid)==1 /*valid coded as 9 missing value*/
```

```
tab vsasid2
tab fice if vsasid2>=3 /*fice with SASID not valid*/
```

```
*****
```

```
*Valid degree sought, 1 valid, 2 not valid*
gen vdegree=.
```

```
replace vdegree=1 if degree==10 | degree==20 | degree==30 | degree==40 | degree==50 |
degree==60| ///
degree== 65|degree== 70| degree==81| degree==85| degree==86| degree==99 |degree==47 |
degree==87 | degree==0
```

```
replace vdegree=2 if degree ==.
```

```
replace vdegree=3 if degree !=10 & degree!=20 & degree!=30 & degree!=40 & degree!=50 & ///
degree!=60 & degree!= 65 & degree!= 70 & degree!=81 & degree!=85 & degree!=86 & degree!=99 &
degree!=47 & degree!=87 & degree!=0
```

```
tab degree vdegree
tab fice if vdegree>1
```

```
*****
```

```
*valid Degree Date*
*update sequence to aling to expected graduating year for collection*
*collection is 9/1 to 8/31*
*9/1/17 to 8/31/18 for MAPCS18*
```

```
gen vddate=.
replace vddate=1 if degdate !=. & (degdate>=20170901 & degdate<=20180831) /*valid*/
replace vddate=4 if degdate ==. /*blank invalid*/
replace vddate=2 if degdate <=20170831 /*date too early for collection range*/
replace vddate=3 if degdate >=20180901 /*date too late for collection range*/
tab vddate
tab fice vddate if vddate>1
```

entry term

gen veterm=.

replace veterm=1 if entrm==1 | entrm==2 | entrm==3 | entrm==4 /*valid*/

replace veterm=2 if entrm!=1 & entrm!=2 & entrm!=3 & entrm!=4 & entrm!=. /*invalid*/

replace veterm=3 if entrm==. /*blank invalid*/

tab veterm

tab fice if veterm>1

entry year

gen veyear=.

replace veyear=1 if (entyr >=1990 & entyr <2018) /*acceptable range*/

replace veyear=2 if entyr<=1989 /*invalid*/

replace veyear=3 if entyr>=2018 /*invalid*/

replace veyear=3 if entyr==. /*blank invalid*/

tab veyear

tab fice veyear if veyear>=2

validate MAPCS code is from current list

old codes may be in file if student is completing discontinued program

gen vmdpgcode=.

replace vmdpgcode= 1 if mdpgcode==517

replace vmdpgcode= 1 if mdpgcode==300

replace vmdpgcode= 1 if mdpgcode==1561

replace vmdpgcode= 1 if mdpgcode==1580

replace vmdpgcode= 1 if mdpgcode==1303

replace vmdpgcode= 1 if mdpgcode==412

replace vmdpgcode= 1 if mdpgcode==1213

replace vmdpgcode= 1 if mdpgcode==1305

replace vmdpgcode= 1 if mdpgcode==1214

replace vmdpgcode= 1 if mdpgcode==1581

replace vmdpgcode= 1 if mdpgcode==1250

replace vmdpgcode= 1 if mdpgcode==335

replace vmdpgcode= 1 if mdpgcode==104

replace vmdpgcode= 1 if mdpgcode==1215

replace vmdpgcode= 1 if mdpgcode==1230

replace vmdpgcode= 1 if mdpgcode==105

replace vmdpgcode= 1 if mdpgcode==1242

replace vmdpgcode= 1 if mdpgcode==1560

replace vmdpgcode= 1 if mdpgcode==249

replace vmdpgcode= 1 if mdpgcode==1312

```
replace vmdpgcode= 1 if mdpcode==1200
replace vmdpgcode= 1 if mdpcode==1231
replace vmdpgcode= 1 if mdpcode==1203
replace vmdpgcode= 1 if mdpcode==1557
replace vmdpgcode= 1 if mdpcode==1568
replace vmdpgcode= 1 if mdpcode==1311
replace vmdpgcode= 1 if mdpcode==1232
replace vmdpgcode= 1 if mdpcode==1582
replace vmdpgcode= 1 if mdpcode==1583
replace vmdpgcode= 1 if mdpcode==1584
replace vmdpgcode= 1 if mdpcode==421
replace vmdpgcode= 1 if mdpcode==1309
replace vmdpgcode= 1 if mdpcode==1534
replace vmdpgcode= 1 if mdpcode==1244
replace vmdpgcode= 1 if mdpcode==1331
replace vmdpgcode= 1 if mdpcode==1332
replace vmdpgcode= 1 if mdpcode==1338
replace vmdpgcode= 1 if mdpcode==1339
replace vmdpgcode= 1 if mdpcode==1316
replace vmdpgcode= 1 if mdpcode==1319
replace vmdpgcode= 1 if mdpcode==1222
replace vmdpgcode= 1 if mdpcode==1223
replace vmdpgcode= 1 if mdpcode==1237
replace vmdpgcode= 1 if mdpcode==1210
replace vmdpgcode= 1 if mdpcode==324
replace vmdpgcode= 1 if mdpcode==1233
replace vmdpgcode= 1 if mdpcode==321
replace vmdpgcode= 1 if mdpcode==1207
replace vmdpgcode= 1 if mdpcode==428
replace vmdpgcode= 1 if mdpcode==527
replace vmdpgcode= 1 if mdpcode==1234
replace vmdpgcode= 1 if mdpcode==1236
replace vmdpgcode= 1 if mdpcode==1205
replace vmdpgcode= 1 if mdpcode==340
replace vmdpgcode= 1 if mdpcode==343
replace vmdpgcode= 1 if mdpcode==342
replace vmdpgcode= 1 if mdpcode==341
replace vmdpgcode= 1 if mdpcode==385
replace vmdpgcode= 1 if mdpcode==387
replace vmdpgcode= 1 if mdpcode==1323
replace vmdpgcode= 1 if mdpcode==420
replace vmdpgcode= 1 if mdpcode==1734
replace vmdpgcode= 1 if mdpcode==1313
replace vmdpgcode= 1 if mdpcode==1241
replace vmdpgcode= 1 if mdpcode==1586
replace vmdpgcode= 1 if mdpcode==9999
```

```

replace vmdpgcode=2 if mdpgcode==. /*blank, not valid*/

tab vmdpgcode

tab fice vmdpgcode if vmdpgcode==2 /*blank, not valid*/

*****population distributions*****
tab idt /*high levels of 2 requires review*/
tab citizen
tab sex

*race/ethnicity*
tab hispanic, missing

*construct multi race*
gen rmulti=.
replace rmulti=1 if (rwhite==1 & rblack==2 & rasian==3 & raian==4 & rnhpi==5 & hislat!=2)
replace rmulti=1 if (rwhite==1 & rblack==2 & hislat!=2)
replace rmulti=1 if (rwhite==1 & rasian==3 & hislat!=2)
replace rmulti=1 if (rwhite==1 & raian==4 & hislat!=2)
replace rmulti=1 if (rwhite==1 & rnhpi==5 & hislat!=2)
replace rmulti=1 if (rblack==2 & rasian==3 & hislat!=2)
replace rmulti=1 if (rblack==2 & raian==4 & hislat!=2)
replace rmulti=1 if (rblack==2 & rnhpi==5 & hislat!=2)
replace rmulti=1 if (rasian==3 & raian==4 & hislat!=2)
replace rmulti=1 if (rasian==3 & rnhpi==5 & hislat!=2)
replace rmulti=1 if (raian==4 & rnhpi==5 & hislat!=2)

tab rmulti

*unknown race*
gen runknown=.
replace runknown=2 if rwhite==. & rblack==. & rasian==. & raian==. & rnhpi==. & hislat!=2
tab runknown /*high levels of unknown requires review*/

**distribution of population by race**
tab rwhite if hislat!=2 & rmulti!=1
tab rblack if hislat!=2 & rmulti!=1
tab hispanic
tab hislat
tab rasian if hislat!=2 & rmulti!=1
tab raian if hislat!=2 & rmulti!=1
tab rnhpi if hislat!=2 & rmulti!=1
tab rmulti

```

tab runknown

tab degree

```
table georg if georg==001
table georg if georg==100
tabstat georg if georg>100 & georg <200, statistics( count )
table georg if georg==200
tabstat georg if georg>200 & georg <225, statistics( count )
table georg if georg==000
```

tab mdres

tab milstat

tab mapcsflg /*high rate of 9999 requires review*/

tabulate mdpgcode, nolabel

population issues

tab fice if idt!=1 /*id provided is not SSN*/

tab fice if idt!=1 & citizen==1 /*id provided is not SSN & student is US citizen*/

tab fice if mdpgcode==9999 /*no mapcs code*/

tab fice if runknown==2 /*race/ethnicity unknown*/

tab fice if vgeorg==3|vgeorg==4 /*100 - unknown US state*//*unknown*/

tab fice if vgeorg==5 /*Unknown MD county*/

prep DIS

drop coltm colyr subcamp sasid ptax fmjrd cumgpa cncrhre ccrhra pdflg entrm

drop entyr revtrn rqcrhr teachcan sex citizen hispanic-tssn_pg dup2-num_unique2

drop sic idt

rename program_tax_orig program

parse program

format %06.0f program

tostring program, gen (program2)

construct majord and disp

first two digits are MAJORDS

first four digits are DISCPLN

```
*MAJORDS, first two digits*
gen majords=substr(program2, 1, length(program2)-4)
destring majords, replace
```

```
*DISCIPLN - first four digits*
gen discipln=substr(program2, 1, length(program2)-2)
destring discipln, replace
gen prog=discipln
```

```
*drop program2*
drop program2
```

```
*step 1: convert to string*
tostring majorlk, gen(fmjrd)
```

```
*start with the first number and keep 2 digits*
*this pulls out the degree sought code from the first_major string*
gen fmdeg=substr(fmjrd, 1, 2)
```

```
*start with the third digit and keep that digit and 3 additional)
*this pulls out the program taxonomy code from the first_major string*
gen fmpt=substr(fmjrd,3, 4)
```

```
*finally, destring to reinstate numbers for validation both degree code and program taxonomy*
destring fmdeg fmpt, replace
```

```
*****
```

```
*tag potential teachers*
*these are likely to be in the MAPCS file*
gen teach=.
replace teach=1 if fmpt==801 | fmpt==802 | fmpt==803 | fmpt==804
replace teach=1 if fmpt==805 | fmpt==806 | fmpt==807 | fmpt==808
replace teach=1 if fmpt==809 | fmpt==810 | fmpt==811
replace teach=1 if fmpt==812 | fmpt==813 | fmpt==814 | fmpt==815
replace teach=1 if fmpt==816 | fmpt==817 | fmpt==818 | fmpt==819
replace teach=1 if fmpt==820 | fmpt==823
replace teach=1 if fmpt==830 | fmpt==831 | fmpt==832
replace teach=1 if fmpt==833 | fmpt==834 | fmpt==835 | fmpt==836
replace teach=1 if fmpt==837 | fmpt==838 | fmpt==839
```

```
replace teach=1 if prog==801 | prog==802 | prog==803 | prog==804
replace teach=1 if prog==805 | prog==806 | prog==807 | prog==808
replace teach=1 if prog==809 | prog==810 | prog==811
```



```

replace teach=1 if prog==812|prog==813|prog==814|prog==815
replace teach=1 if prog==816|prog==817|prog==818|prog==819
replace teach=1 if prog==820|prog==823
replace teach=1 if prog==830|prog==831|prog==832
replace teach=1 if prog==833|prog==834|prog==835|prog==836
replace teach=1 if prog==837|prog==838|prog==839

```

```

tab fice teach if program!=.

```

```

*restrict population to in scope degree date*

```

```

*update dates each year*

```

```

gen inscope=.

```

```

replace inscope=1 if teach!=. & (degdate>=20170901 & degdate<=20180831)

```

```

tab inscope

```

```

tab fice if inscope==1

```

```

tab teach inscope

```

```

tab fice inscope if teach==1

```

```

*save and drop all observations that are not teach=1 and in-scope=1*

```

```

drop if teach!=1

```

```

drop if inscope!=1

```

```

*identify unique possible teachers in DIS*

```

```

gsort fice idn inscope

```

```

egen teach_unique=tag(fice idn)

```

```

tab teach_unique

```

```

*keep only one record per idn/fice*

```

```

drop if teach_unique!=1

```

```

drop tssn campusid degdate degree program majorlk fmjrd majords discipln fmdeg fmpt prog teach
inscope teach_unique

```

```

*this is final population of potential unit records for MAPCS*

```

```

tab fice

```

```

*****

```

```

*****

```

```

*prep MAPCS*

```

```

drop subcamp sasid sex citizen hispanic rwhite rblack rasian raian rnhpi entrm

```

```

drop entyr birthdt georg mdres milstat sector segment num_unique sicfice-runknown

```

```

drop coltm colyr tssn tssn2 idt campusid degdate degree sic source_file

```

```

*next drop duplicate records in MAPCS for testing MAPCS flag*

```

```

*mapcs*

```

```
sort fice idn
egen map_unique=tag(fice idn)
tab map_unique
tab fice if map_unique==1
```

```
drop if map_unique==0
drop map_unique
```

```
*****
```

```
*merge files for analysis*
```

```
*****
```

```
use "M:\AnnKellogg\Data Validation Analysis_Ann Kellogg\Core Collections\DIS-MAPCS Merged
Data Files\ MAPCS2018_ready for merge2.dta saved"
merge 1:1 fice idn using "M:\AnnKellogg\Data Validation Analysis_Ann Kellogg\Core Collections\DIS-
MAPCS Merged Data Files\DIS2018_ready for merge2.dta"
```

```
*save data set*
```

```
*tab number of students in DIS NOT in MAPCS*
*these are potential missing records from MAPCS as these*
*students graduated with a HEGIS code and degree date*
*that make them potential MAPCS candidates*
tab fice if _merge==2
```

```
*records in MAPCS with flag set to DIS=Yes
tab fice if mapcsflg==1
```

```
*record in MAPCS with DIS flag=Yes but no matching*
*record found in DIS. Either flag is wrong, records is missing from DIS*
*or IDNs are not consistent across DIS and MAPCS to support merge*
tab fice mapcsflg if _merge==1 & mapcsflg==1
```

```
*tab number of students with MAPCS Flag=Yes that found a match in DIS*
tab fice mapcsflg if _merge==3 & mapcsflg==1
```

```
*number in mapcs*
tab fice if mapcsflg!=.
```

```
*tab number of students in DIS with MAPCS Flag=No*
*if any records are present then mapcsflg is incorrectly set*
tab fice mapcsflg if _merge==3 & mapcsflg==2
```