CS 241 Final Project

Purpose

This project is designed to do several things:

1. Introduce you to a number of concepts that will help you throughout the rest of your school career
2. Explore different ways of doing things using C

Assignment

You are to create a program that creates and manipulates a school database. You are not expected to create a full relational database; this is covering basic concepts, so we will be creating a flat-file database.

You will need to create 5 “tables” for our flat-file database:

1. Students
2. Classes
3. Enrollment
4. Assignments
5. Grades

You will need to create structs for each of the above. They are defined as follows:

1. Students have a first name of 30 characters, a last name of 30 characters, and a social security number of 9 characters.
2. Classes have an id which is an integer and a title of 30 characters
3. Enrollment has a class id which is an integer and a student social security number of 9 characters
4. Assignments have an assignment id which is an integer, a title of 30 characters, a point value which is an integer, and a class id which is an integer
5. Grades have a class id which is an integer, a student social security number of 9 characters, and a grade which is an integer

You will need to interact with your users in 3 different ways:

1. Menu based interactions
2. Custom command line environment
3. CLI arguments

Menu Based Interactions

Main Menu	
1. Add Data	Loads Add Menu
2. Edit Data	Loads Edit Menu
3. Delete Data	Loads Delete Menu
4. View Data	Loads View Menu
5. Use Command Line Environment	Close Menu and Load Command Line Environment
6. Exit	Exit Program
Add Data Menu	
1. Add Student	Loads add student prompts
2. Add Class	Loads add class prompts
3. Add Assignment	Loads add assignment prompts
4. Add Grade	Loads add grade prompts
5. Enroll Student	Loads enroll student prompts
6. Return to Main Menu	Close add menu and load main menu
Edit Data Menu	
1. Edit Student	Loads edit student prompts
2. Edit Class	Loads edit class prompts
3. Edit Assignment	Loads edit assignment prompts
4. Edit Grade	Loads edit grade prompts
5. Return to Main Menu	Close edit menu and load main menu
Delete Data Menu	
1. Delete Student	Loads delete student prompts
2. Delete Class	Loads delete class prompts
3. Delete Assignment	Loads delete assignment prompts
4. Delete Grade	Loads delete grade prompts
5. Drop Student	Loads drop student prompts
6. Return to Main Menu	Close delete menu and load main menu
View Data Menu	
1. View Students	Displays all students
2. View Classes	Displays all classes
3. View Assignments	Loads view assignments prompts
4. View Grades	Loads view grades menu
5. View Enrollment	Loads view enrollment menu
6. Return to Main Menu	Close view menu and load main menu
View Grades Menu	
1. View Class Average Grade	Loads avg grade class selection prompts
2. View Student Average Grades	Loads avg grade student selection prompts
3. View Class Assignment Average Grades	Loads avg assignment class selection prompts
4. View Class Assignment Grades	Loads class and assignment selection prompts
5. Return to Main Menu	Close view grades menu and return to view menu

The menu prompts are defined as follows:

Add Student
- Enter First Name
- Enter Last Name
- Enter Student SSN

Add Class
- Enter Class Title

Add Assignment
- Enter Class ID or (-1 for Class List)
- Enter Assignment Title
- Enter Point Value

Add Grade
- Enter Class ID or (-1 for Class List)
- Enter Assignment ID or (-1 for Assignment List)
- Enter Student SSN
- Enter Earned Points

Enroll Student
- Enter Class ID or (-1 for Class List)
- Enter Student SSN

Edit Student
- Enter Student SSN
- Enter New First Name (or leave blank for no change)
- Enter New Last Name (or leave blank for no change)

Edit Class
- Enter Class ID or (-1 for Class List)
- Enter New Class Title (or leave blank for no change)

Edit Assignment
- Enter Class ID or (-1 for Class List)
- Enter Assignment ID or (-1 for Assignment List)
- Enter New Title (or leave blank for no change)
- Enter New Point Value (or leave blank for no change)

Edit Grade
- Enter Class ID or (-1 for Class List)
- Enter Assignment ID or (-1 for Assignment List)
- Enter Student SSN
- Enter New Earned Points (or leave blank for no change)

Delete Student
    Enter SSN
    Are you sure you wish to delete [display info] (Y/N)?
Delete Class
    Enter Class ID or (-1 for Class List)
    Are you sure you wish to delete [display info] (Y/N)?
Delete Assignment
    Enter Class ID or (-1 for Class List)
    Enter Assignment ID or (-1 for Assignment List)
    Are you sure you wish to delete [display info] (Y/N)?
Delete Grade
    Enter Class ID or (-1 for Class List)
    Enter Assignment ID or (-1 for Assignment List)
    Enter Student SSN
    Are you sure you wish to delete [display info] (Y/N)?
Drop Student
    Enter Class ID or (-1 for Class List)
    Enter Student SSN
    Are you sure you wish to delete [display info] (Y/N)?

View Assignments
    Enter Class ID or (-1 for Class List)
View Grades
    Enter Class ID or (-1 for Class List)
    Enter Assignment ID or (-1 for Assignment List)
View Enrollment
    Enter Class ID or (-1 for Class List)

View Class Average Grade
    Enter Class ID or (-1 for Class List)
View Student Average Grades
    Enter Student SSN
View Class Assignment Average Grades
    Enter Class ID or (-1 for Class List)
View Class Assignment Grades
    Enter Class ID or (-1 for Class List)
    Enter Assignment ID or (-1 for Assignment List)

Custom Command Line Environment

The custom command line environment will read in information from the user and parse them. Your environment must handle the following commands:

add student [first] [last] [ssn]

add class [title]

add assignment [class id] [title] [point value]

add grade [class id] [assignment id] [student ssn] [points]

enroll [class id] [student ssn]

edit student [student ssn] [new first] [new last]

edit class [class id] [new title]

edit assignment [class id] [assignment id] [new title] [new point value]

edit grade [class id] [assignment id] [student ssn] [new point value]

delete student [student ssn]

delete class [class id]

delete assignment [class id] [assignment id]

delete grade [class id] [assignment id] [student ssn]

drop [class id] [student ssn]

view students

view classes

view assignments [class id]

view grades [class id] [assignment id]

view grades [student ssn] [class id]

view average grade [class id] [assignment id]

CLI Arguments

The CLI arguments will run one command per program call. All arguments after –[action] (e.g. –add-student) may be in any order. The following two commands will do the same thing

-add-student –f Keith –l Weber –s 123456789

-add-student –l Weber –s 123456789 –f Keith

You will need to support the following commands

-add-student –f [first] –l [last] –s [ssn]

-add-class –t [title]

-add-assignment –cid [class id] –t [title] –p [point value]

-add-grade –cid [class id] –aid [assignment id] –s [student ssn] –p [points]

-enroll –cid [class id] –s [student ssn]

-edit-student –s [student ssn] –f [new first] –l [new last]

-edit-class –cid [class id] –t [new title]

-edit-assignment –cid [class id] –aid [assignment id] –t [new title] –p [new point value]

-edit-grade –cid [class id] –aid [assignment id] –s [student ssn] –p [new point value]

-delete-student –s [student ssn]

-delete-class –cid [class id]

-delete-assignment –cid [class id] –aid [assignment id]

-delete-grade –cid [class id] –aid [assignment id] –s [student ssn]

-drop –cid [class id] –s [student ssn]

-view-s

-view-c

-view-a –cid [class id]

-view-g –cid [class id] –aid [assignment id]

-view-g –s [student ssn] –cid [class id]

-view-g –avg –cid [class id] –aid [assignment id]

Database Requirements

Database Storage

Your database will be stored in several flat files called students.db, classes.db, assignments.db, enrollment.db, grades.db

You will need to read from and write to these files regularly during the running of your program.

Your grade will be determined by the contents of these database files after the conclusion of the tests.

Deleting Data

One of the greatest challenges in dealing with databases is making sure that data is valid all the time. An example of this is what is called orphaned data. If you have data dependent on other data that is missing, you end up with the database version of a memory leak.

For example:

> We add a student Keith Weber 123456789
>
> We add a class called Underwater Basket Weaving
>
> We add an assignment called Basket1 worth 100 points
>
> We add a grade for Keith of 85 points
>
> We delete the Underwater Basket Weaving class.

In the example above, if we do not delete the assignment and grade when we delete the Underwater Basket Weaving class, we will not be able to access them again.

To account for this, you will need to delete all dependent data when deleting data from the database.