CSIS-1410 - Assignment Students

Learning Objectives :

This assignment is intended to review and refresh a number of concepts taught in CSIS-1400, like:

- declaring classes
- accessing class members of other classes
- using an ArrayList
- using a static field
- interpreting a UML class diagram
- reading in user input from the keyboard
- providing user choices with a menu
- using a do-while loop
- using a switch statement
- using a for-each loop (a.k.a. enhanced for loop)
- overloading constructors
- overriding toString

Description:

On this assignment you can choose whether you prefer to work on your own or with a partner.

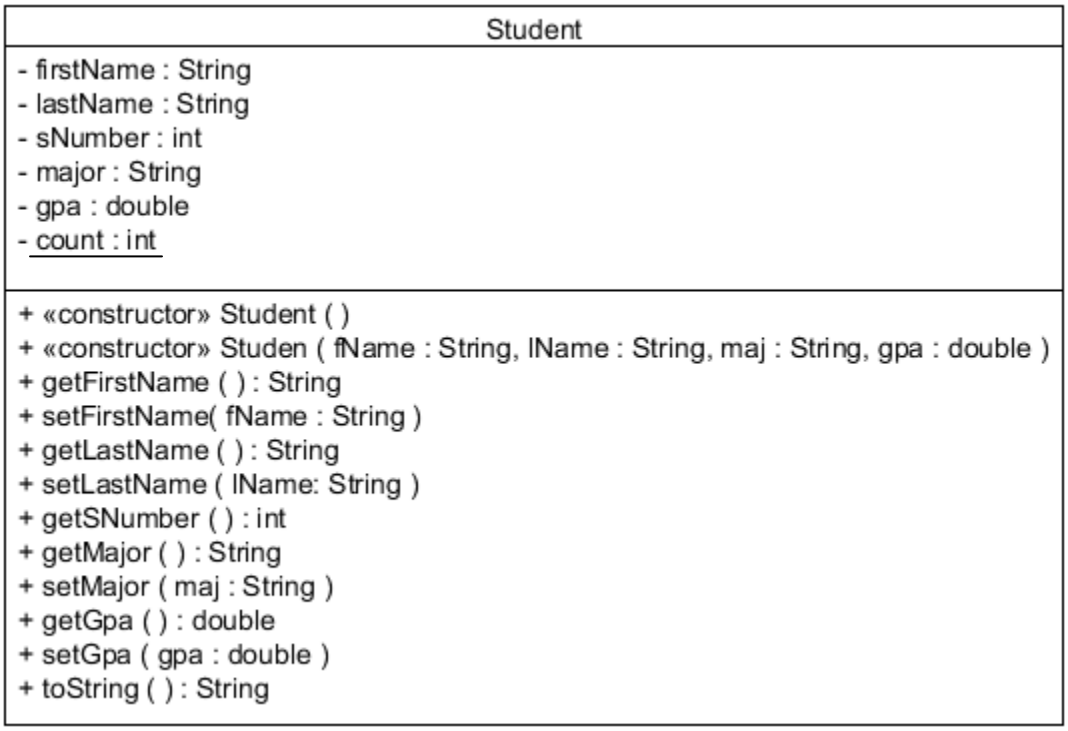
I encourage you to do whatever helps you best to refresh concepts taught in CSIS-1400.

You will write a program that keeps track of a list of students and exposes a given set of choices with a menu.

Do so by implementing at least two classes: `Student` and `StudentApp`

ad Student :

The class Student represents a student. Notice that neither count nor sNumber are passed to the constructor. These two fields also have no set method (mutator).



Hint: http://stackoverflow.com/questions/7221691/is-there-a-way-to-automatically-generate-getters-and-setters-in-eclipse

Sample Output :

```
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 4
ï
S1234567 John Smith (CS) gpa:3.6
S1234568 Lauren Edwards (CS) gpa:3.8
S1234569 Alex Taylor (EE) gpa:3.2
ï
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 1
ï
First name: Rob
Last name: Hill
Major: ME
GPA: 3.4

1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 2
ï
Find student with sNumber S1234568
S1234568 Lauren Edwards (CS) gpa:3.8
ï
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 3
ï
Delete student with sNumber S1234569
S1234569 Alex Taylor has been deleted
ï
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 2
ï
Find student with sNumber S1234569
Student could not be found
```

ad count:

Notice that count is underlined. This indicated that count is a static field.

Count keeps a running count of the Student objects that have been created.

It is used to create a unique sNumber for each student

ad constructors:

The generated field sNumber needs to be initialized in both constructors.

Create a unique 7 digit student number for each student based on the static field count. (e.g. 1234567 + count++)

The parameterized constructor uses the values passed to initialize the fields.

ad StudentApp:

StudentApp includes the main method .

Use private methods to structure your code.

If you want to add additional classes that is fine, too.

ad main

Create an ArrayList of students that is initialized with 3 different students.

Use a do-while loop and a switch statement to display a menu with choices and to respond to the user selections.

Here is how the menu should look like:

1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display number of students in list
6. Exit

The user should not be allowed to enter a student number because it is auto-generated.

If a user tries to find or delete a student based on a student number that doesn't exist, an appropriate message should be displayed. (see output)

If a student is actually found or deleted corresponding student data should be displayed as part of the response (see output)

```
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 4
ï
S1234567 John Smith (CS) gpa:3.6
S1234568 Lauren Edwards (CS) gpa:3.8
S1234570 Rob Hill (ME) gpa:3.4
ï
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 5

Number of Students: 3
ï
1. Add a student
2. Find a student
3. Delete a student
4. Display all students
5. Display the total number of students
6. Exit
Enter your selection: 6
ï
Good bye
```

Turning in :

Create a runnable jar file that includes the source code and submit it via Canvas

If you worked with a partner do the following:

- **Only one student** submits the runnable jar file with the source code.
  Make sure that each Java file includes the names of all students that contributed to the code
- **Each student** writes a brief submission note describing the team experience. This is a great opportunity to give kudos to a strong partner that helped you, it is also a place where you can let me know if you experienced difficulties.