Yasmine El Hattab

Yi Wong

Final Project: Deliverable 1

Sunday, April 20

Explanation project:
My project will feature a system that imitates the one used in school to keep all the information needed by the establishment to ensure its smooth operation. Thus, the students will be able to register and drop courses, the teachers will upload grades and calculate their average, and the administrator will keep track of the students' transcripts.

Design Paradigm: List the functionalities you plan to demonstrate.

User : 1.is the superclass
Teacher: 1. can add the grades of students
2. can calculate the average of each student
Student: 1.can register for a course
2. Can drop a course
3. can view and add their personal information
Administrator: 1.can read the transcript of the student

Expected Output:
The users are supposed to be able to use this system as a tool. The Student should be able to drop and register for courses as well as view their grade. The Teachers are supposed to upload grades and see the transcripts. The administrators are supposed to view and write the transcripts.

The hierarchies in the project
There is User that is the superclass of Teacher, Student and Administrator.
There is Course that is a superclass of MandatoryCourse and ComplementaryCourse

Explanation about the interface
The interface is named Enrollable. It's used to let the Student enroll for different classes. Inside, there is an abstract method called enroll(Student s).

***Method applying runtime-polymorphism**

1. A method called PrintDetails() will be overridden by Students and Teachers.
2. A method called CalculateFee () will be overridden by MandatoryCourse and ComplementaryCourse

*** Where TextIO will be implemented**

1. The Students class will use textIO to read their transcripts
2. Students will be able to write their information in their transcript using textIO

The classes where Comparable and Comparator will be implemented
Comparable will be implemented by Course to sort the code of the courses.

Comparator will be implemented by Student to sort by name ascendingly, then by email ascendingly and then by default it will sort Id ascendingly.

The part completed for the deliverable 2

For deliverable 2:

1. All the classes should be created
2. The methods using TextIO should be finished
3. The methods CalculateFee() and PrintDetails() will be finished and will have unit testing
4. The classes that implement the interface enrollable will have complete methods
5. The method CalculateAverage() in the teacher class and it's unit testing