Introduction to Computer Science and Programming 1
CSCI120

Course Project

If it is a group assignment, please add the information here

# of Students in the Group:		
Student 1	*First name, last name*	*Student-ID*
Student 2	*First name, last name*	*Student-ID*
Student 3	*First name, last name*	*Student-ID*
Student 4	*First name, last name*	*Student-ID*

Requirements

- This is a group/individual assignment. Do it in a group of 3 or less.
  - o Note: If you are 4 people in the group, then doing the "bonus" part of the project is mandatory, otherwise your project will be evaluated based on 75%.
- There is a bonus part in the project too which gives you additional 3% in you do it completely
- Please use meaningful name for your variables and functions
- Try to reuse your solutions as much as possible.
- A group more than 4 people is not accepted!

Project Starter

- I have prepared a project starter for you and many of the classes you need have been already defined for you.
- Please download the project starter from the portal and read the code to make yourself familiar with the code.
- Then read the project description and then read the code in the project starter again.

Project Description

In this project you are going to design an object-oriented model for an educational portal using Python. Through this portal the students can register as a new user (student) and also login to the portal through a registration and login view respectively and perform some actions that are available to them.

Define and write a Python class for each of the following entities (Feel free to add more classes if needed):

- o Account
- o Student
- o StudentProfile
- o Transcript
- o GeneralTranscript (child of the Transcript)
- o CurrentSemesterTranscript (child of Transcript)
- o TakenCourse
- o CollegeCourse
- o Manager
- o Semester
- o Menu
- o Portal
- o PortalManager

- Draw the object diagram for each of the above entities.
- You need to define properties (instance or static properties) and behaviors (methods) for the above classes. Before you decide what properties or methods to consider for each object, read the rest of this problem and figure out what states and behaviors you can define for each of the classes.

The class **Account**: Each student is going to have one account. At the beginning the student should register with a username and a password to create an account.
- The username must be at least 6 characters long.
- The password must be at least 6 characters and at least one digit character in it.

The class **Student**: Each student is defined as following:
- A number, which shows the admission year of the student.
- A general transcript of type <u>GeneralTranscript</u>, which will show all the courses the student has taken so far since s/he has been in the school.
- A semester transcript of type <u>CurrentSemesterTranscript</u> which will show all the courses the student is taking in the current semester.
- A student's profile from type StudentProfile which contains some personal information about the student

The class **StudentProfile**: Once a new student registers, a profile is created for the student. The profile contains the following information:

- o StudentID
- o Firstname
- o Lastname
- o Gender
- o Address
- o Country of Origin
- o Age

The class **GeneralTranscript**: The general transcript shows all courses the student has taken. For each course, the name of the course, the code of the course, the grade the student had received in the course and number of units and the number of semester in which the course is taken is shown. The General Transcript include the courses the student has taken in the current semester.

The class **CurrentSemesterTranscript**: Similar to General Transcript but only contains the courses the student has taken in the current semester.

The class **CollegeCourse:** Each CollegeCourse, consists of the following information:
- Course's name
- Course's code
- Course's number of units. Each course has a unit which means the weight of the course in the GPA.

The class **TakenCourse**: Each course consists of the following information:
- CollegeCourse (which is the name, code and unit)
- Student's grade: if the course is not taken yet, the default value is -1.
- The number of the semester the course is taken in. If the course is not taken yet, this value is 0.

Note: You can suppose each school year is 2 semester. For instance, Semester 1 means, the year one and semester one. Semester 2 means year 1 and semester 2. Semester 3 means, year 2 and semester 1 and Semester 4 means, year 2 and semester 2 and so on.

The class **Semester** has the following properties:
- Year number: Shows the year number.
- Semester number: Shows the semester number in the year (could be 1 or 2)
- isCurrentSemester: If True, that means it is the current semester otherwise it is False.

The class **Manager**: There is one instance from the class **Manager** which represents the manager of the college. The Manager class, has the following properties:

- firstName
- lastName
- Title

The class **Portal**. There is one instance of the class **Portal**. The class portal has the following properties:
- List of all students
- List of all college courses

The class **Menu**. The class Menu is used to create the main Menu of the application. Please see the menu below. <u>The Menu class does not have any instance property</u> but it is used to show the menu to the user and receives the menu selection from the user (what option the user is selected)

Note1: You are not limited only to above classes. If needed, feel free to create your own classes too. But your application should at least include the above classes.

Note2: Your application does not need to keep any offline data (it does not need to write to the file and read from the file).

Note3: Try to use all principles of Object-Oriented Design in this project:
- Encapsulation
- Polymorphism
- Inheritance
- Public Interface

The Flow of the application: Here you are going to create a text-based application. No graphical UI is needed for this project.

The application contains several views and a main menu. (all text-based) as following:

Login View: Once you run the program, the following view is shown to the user:
****
Please enter your account to login:
****
Username:
Password:

Not registered yet? Type "Register" and press enter to start the registration process!

If the username and password were correct then the program prints the following message and wait until the user press any key and then prints the main menu as following. If the username or password were wrong then the program informs the user using the following error message and asks the user to try again.

If username and password were <u>correct:</u>
****
Welcome to Columbia College!
****

If the username or password was <u>wrong:</u>
****
Your account does not exist. Please try again!
****

If the user enters "Register" and press enter key, then the registration process is shown to the user as following. As a result of registration, a new account and a profile is generated for the student and the student should be able to login, next time without having to register again. Upon registration, you should create a random 8 digit number as the studentID.

****
Welcome to Columbia College: Please Register
****

Please enter your first name:
Please enter your last name:
Please enter your gender [M/F/O]:
Please enter your country of origin:
Please enter the year of admission:

Please enter your age:

Please enter a username [At least 6 characters]:
Please enter a password [At least 6 characters with at least one digit]:

Thanks, your account and profile has been created successfully. Welcome Aboard <u>[The name of the student]</u>

Once the user successfully entered his/her username and password the program shows the above welcome message and then wait for the user to enter any key and then prints the following main menu:

****
Select from the options:
****
——[1] Print my enrolment certificate
——[2] Print my courses
——[3] Print my transcript
——[4] Print my GPA
——[5] Print my ranking among all students in the college
——[6] List all available courses
——[7] List all students
——[8] Show My Profile
——[9] Logout
——[10] Exit
——[11] Bonus
****
Enter the number corresponding to each item to proceed:

Note: If the student enters 10, the application will terminate otherwise it will perform the corresponding action and will print the main menu again as long as the student has not entered 10 (which is Exit).

The user enters a number between 1 and 9 every time to perform the corresponding functionality.

- <u>Print my enrolment certificate:</u> If the user entered '1', the program prints the following information from the user in the following format. And then prints the main menu again.

   Dear Sir/Madam,

   This is to certify that Mr. Peter Brown with student id 7813007 is a student at semester 1 at Columbia College. He was admitted to our college in 2011 and has taken 1 course(s) so far. Currently he resides at 850 West Vancouver, Vancouver.

   If you have any question, please don't hesitate to contact us.
   Thanks,
   [Manager: Peter Jackson]

- <u>Print my courses:</u> If the user entered '2', the program prints all the courses the student has taken in the following format. And then prints the above main menu again.

   Hi Mr. Peter Brown,
   You have taken the following courses so far:
   1) CSCI101: Python
   2) CSCI202: Project Management
   3) CSCI301: Java Programming
   4) CSCI 401: Android Programming [Current semester]

- <u>Print my transcript:</u> If the user entered '3', then the program prints the transcript of the user in the following format and then prints the above menu.

Hi Mr. Peter Brown,
Here is your general transcript:

   1) CSCI101: Python: 80
   2) CSCI202: Project Management: 45
   3) CSCI301: Java Programming: 64
   4) CSCI401: Android Programming: 70 [Current semester]
      YOUR GPA IS: 64.75

   Here is your current semester transcript:
   1) CSCI401: Android Programming: 70 [Current semester]
      YOUR Current Semester GPA IS: 70

Note: Please notice to calculate the GPA you need to take into account the number of units of a course. For instance if you got 80 in a course which is 4 units and you got 70 in a course which is 3 units, then your GPA is calculated using the following formula:

GPA = (80*4 + 70*3) / (3+4) = 75.7

- Print my GPA: If the user entered '4', then the program prints the GPA of the student in the following format and then prints the above main menu.

  Hi Mr. Peter Brown,
  Your overall GPA is 64.75
  Your current semester's GPA is 70

- Print my ranking among all students in the college: If the user enters '5', the program will find the rank of the student based on his/her gpa and print it and then print the above main menu. (rank is the based on your gpa with respect to other students)

  Hi Mr. Peter Brown,
  Your overall GPA is 64.75 and therefore your rank is 3.

- List all available courses: If the user entered '6', the program will print the list of all available courses in the college in the following format and then print the menu.
  The following courses are offered in Columbia College:

  1) **CSCI101: Python: 3 units [Not taken]**
  2) **CSCI102: Object-Oriented Programming: 2 units [Not taken]**
  3) **CSCI201: Problem Solving: 1 units [Not taken]**
  4) **CSCI202: Project Management: 3 units [Not taken]**
  5) **CSCI301: Java Programming: 3 units [Taken at semester 1]**
  6) **CSCI302: Web Development: 2 units [Taken at semester 2]**
  7) **CSCI401: Android Programming: 2 units [Taken at semester 4]**
  2) **CSCI402: iOS Applications: 3 units [Taken at semester 3]**

- List all students in the college: If the user enters '7', the program will print the list of all students in the college in the following format and then print the menu.
  There are 4 students in Columbia College as following:
  1) **Peter Brown: 7813007**
  2) **Joseph Rod: 812345**
  3) **Cristina Li: 8012333**
  4) **Adams Wang: 7812999**

- Show My Profile: If the user entered '8', the program will print the profile of the student in the following format:
  Name: Mr. Peter Brown
  StudentID: 7813007
  Gender: Male
  Address: Vancouver
  Country of Origin: CANADA
  Age: 21
  Year of Admission: 2016
  Overall GPA: 64.75
  Courses Taken So far: CSCI101: Python, CSCI202: Project Management, CSCI301: Java
  Programming , CSCI401: Android Programming [Current semester]

- Logout: If the user entered '9', the program will print the login menu and let the user login again with the same or different account.
- Exit: If the user entered '10', the program terminates.

- Bonus: If the user enters the application shows another menu as following:
- ****Welcome to the extra features of the application****
- **—-[1] Print the list of all students based on their GPA (Ascendingly)**
- **—-[2] Print the list of names of all students alphabetically**
- **—-[3] Print the list of all Male students**
- **—-[4] Print the list of all Female students**
- **—-[5] List of top (highest GPA) male and female students**
- **—-[6] Back to the previous menu**

Testing the Application

This is the testing scenario:

Imagine there are 5 students on the portal:

Student	StudentID	Firstname	Lastname	Gender	Address	Country of Origin	Age	Admission Year
student1	8012321	Peter	Sand	M	Vancouver	Irland	21	2019
student2	8014525	Sheila	Rogers	F	Surrey	India	19	2018
student3	8011111	Edward	Richards	M	Burnaby	China	20	2019
student4	8033344	Souzan	Robson	F	Surrey	India	20	2019
student5	8012322	Jeff	Cooper	M	Vancouver	England	21	2018

And there are the following courses in the college

Course Code	Course name	Number of course unit
CSCI101	Python	3
CSCI102	Object-Oriented Programming	2
CSCI201	Problem Solving	1
CSCI202	Project Management	3
CSCI301	Java Programming	3
CSCI302	Web Development	2
CSCI401	Android Programming	2
CSCI402	iOS Applications	3

Imagine each student has taken some courses and received some marks. The table below show what course the student has taken and the grades.

Student	Course	Semester the course is taken	Grade
student1			
	Python	2019-term1	80
	Object-Oriented Programming	2019-term2	76
	Problem Solving	2020-term1 (Current)	67
	Project Management	2019-term1	82
	Java Programming	2019-term2	73
student2			
	Python	2018-term1	65
	Object-Oriented Programming	2018-term2	67

	Problem Solving	2018-term2	85
	Project Management	2019-term1	56
	Java Programming	2019-term1	75
	Web Development	2019-term2	76
	Android Programming	2019-term2	80
	iOS Applications	2020-term1 (current)	74
student3			
	Problem Solving	2019-term1	78
	Project Management	2019-term1	87
	No Course	2019-term2	56
	Web Development	2020-term1 (current)	77
student4			
	Project Management	2019-term1	89
	Java Programming	2019-term1	87
	Web Development	2019-term2	71
	Android Programming	2019-term2	69
	iOS Applications	2020-term1 (current)	75
student5	Web Development	2018-term1	78
	Android Programming	2018-term2	56
	iOS Applications	2018-term2	89
	Project Management	2019-term1	66
	Java Programming	2019-term1	77
	Object-Oriented Programming	2019-term2	87
	Problem Solving	2019-term2	67
	No Course	2020-term1 (current)	78

Each student has a username and password. Imagine the username and password of the above students is as following

Student	username	password
student1	"Student1"	"111111"
student2	"Student2"	"222222"
student3	"Student3"	"333333"
student4	"Student4"	"444444"
student5	"Student5"	"555555"

Good Luck 😌