COP 3330- Summer 2022

Project 1

1) This project can be completed in groups (max 3 per group) but submission on Webcourses is individual. Don't copy anybody's work nor give your code to anybody unless you **seriously** worked together on the project.

2) Please Submit **ONE** java file (Project1.java). It should look like this:

```
public class Project1 {

   public static void main(String[] args){
   //Test code goes here
   }

}

//---------------------------
class Abc{

}

//---------------------------
class Xyz{

}
```

3) Apply the Java naming conventions. See ***JavaNamingConventions.pdf*** posted on Webcourses

4) If you work with a classmate(s), your .java file must contain the following comment:

```
/*
- Project 1
- Names (first and last names) of all students who worked together on the project
- (optional) Add anything that you would like the TA to be aware of
*/
```

Example:

```
/*
- Project 1
- Joe Doe, Ericka Edwards and Jamal Dubois
*/


public class Project1 {

   public static void main(String[] args){
   //Test code goes here
   }

   }


//----------------------------
   class Abc{

   }

//----------------------------
   class Xyz{

   }
```

Students will not receive any credit if they don't submit the java file by the deadline!
Submissions by email will not be considered for a grade. Students must submit their projects on Webcourses by uploading the java file. It is the responsibility of the students to check their submissions to make sure that the file they submitted is indeed the right file and it is a readable file!

Project 1 statement

Please read this entire statement carefully before you start doing anything…

This project involves implementing a simple university personnel management program. The program contains at least three classes: ***Staff***, ***Student*** and ***Faculty***. Your program stores relevant information such as university ID, name, etc. Different information is stored depending on the type of the object. For example, a student has a GPA, a faculty has a title and department (professor, mathematics).

For each data member, your program must include a **getter** and a **setter**, and each class must include at least **two constructors**. The goal of this Project is to demonstrate the use of classes, inheritance, abstract classes, abstract methods, and method overriding.

For a student, we need a:

- full name
- id
- gpa
- Number of credit hours currently taken

For a faculty, we need a:

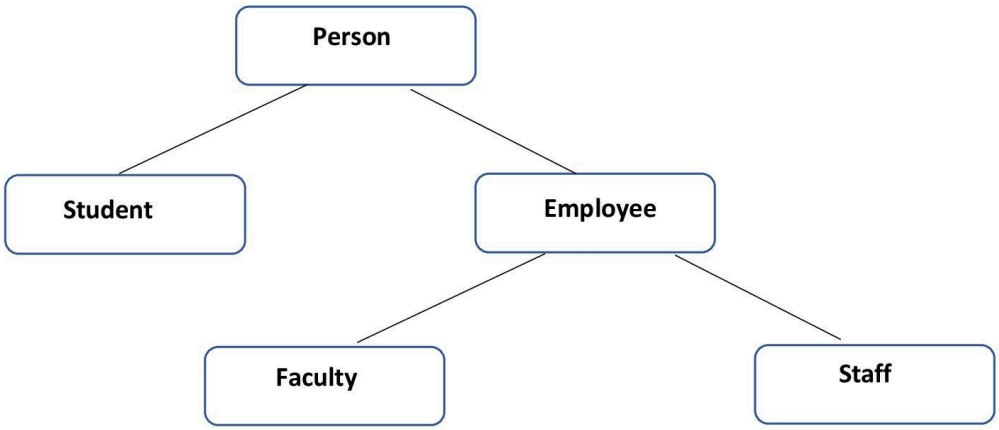
- full name
- id
- department (mathematics, engineering or english)
- Rank (professor or adjunct)

For a staff, we need a:

- full name
- id
- department (mathematics, engineering or english)
- status (part time or full time)

Students in this college pay $236.45 per credit hour in addition to a $52 administrative fee. Your code should generate a tuition invoice **(**a method within the class **Student**). Note that students get a 25% off total payment if their gpa is greater or equal to 3.85.

Your code should implement the following inheritance hierarchy:



Both classes Student and Employee inherit from the *abstract* class Person. The abstract class Person has what is common to a Student and an Employee (Faculty or Staff). The class Person must include: *public abstract void print();*

The abstract method *print* is overridden to:

- **print the fee invoice for a student.**
- **print the information of a faculty**
- **print the information of a staff**

It is left to you (the programmer) to come out with other abstract methods if you see fit(this is optional).

The class Employee should also be abstract, and it is supposed to include what is common to a staff and a faculty object.

Test your code with ONE array of size 100 of type Person (It must be an array, not an arraylist, linked list , hash map or anything else…). The sample run below should give you a clear idea about how your code should run. The user's entry is marked in bold so you can tell what your code should display to the screen and what the user enters.

Please note well that:

1) Your code should run exactly as shown on the sample run below (However, the TA will not deduct points because you skipped two lines instead of three or your tuition invoice has 56 hyphens instead of 63!).
2) When asked to enter the faulty's department, **matheMatics** and **MathematiCs** are considered to be the same. Your program should display ***Mathematics*** if faculty information is to be displayed to the screen. However, if the user enters ***Mathematics department***, then this is an invalid entry. Consider these departments only: Mathematics, Engineering and Sciences. As for the rank of a faculty, consider these ranks only: Professor and Adjunct.
3) The university ID has no required form so you may choose to enter anything to be the ID.
4) Your code should handle up to 5000 people (faculty, staff and students combined)

Sample Run: (Below is how your code should run)

```
                                    Welcome to my Personal Management Program


Choose one of the options:

1- Enter the information a faculty
2- Enter the information of a student
3- Print tuition invoice for a student
4- Print faculty information
5- Enter the information of a staff member
6- Print the information of a staff member
7- Exit Program

     Enter your selection: 2


Enter the student info:

           Name of Student: Julia Alvarez

           ID: ju1254

           Gpa: 3.26

           Credit hours: 7

Student added!


1- Enter the information a faculty
2- Enter the information of a student
3- Print tuition invoice for a student
```

```
4- Print faculty information
5- Enter the information of a staff member
6- Print the information of a staff member
7- Exit Program

     Enter your selection: 2



Enter the student info:

          Name of Student: Matt Jones

          ID: ma0258

          Gpa: 2.78

          Credit hours: 0

Student added!



1- Enter the information of the faculty
2- Enter the information of the two students
3- Print tuition invoice
4- Print faculty information
5- Enter the information of the staff member
6- Print the information of the staff member
7- Exit Program

     Enter your selection: A


Invalid entry- please try again


   1. Enter the information of a faculty
   2. Enter the information of a students
   3. Print tuition invoice
   4. Print faculty information
```

5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **1**

Enter the faculty info:

Name of the faculty: **John Miller**

ID: **jo7894**

Rank: **Instructor**

"Instructor" is invalid

Rank: **Assistant Professor**

"Assistant Professor" is invalid

Rank: **Professor**

Department: **Engineering**

Faculty added!

1. Enter the information of a faculty
2. Enter the information of a students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **3**

Enter the student's is: **ju1254**

Here is the tuition invoice for Julia Alvarez :

Julia Alvarez ju1254

Credit Hours:7 ($236.45/credit hour)

Fees: $52

Total payment (after discount): $1,707.15 ($0 discount applied)

---

1. Enter the information of a faculty
2. Enter information of a students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **3**

Enter the student's is: **ja1954**

No student matched!

1. Enter the information of the faculty
2. Enter information of the two students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of the staff member
6. Print the information of the staff member
7. Exit Program

Enter your selection: **4**

Enter the Faculty's id: jo7894

John Miller jo7894

Engineering Department, Professor

---

1. Enter the information of a faculty
2. Enter information of a students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **6**

Enter the Staff's id: **ha5879**

No Staff member matched!

1. Enter the information of a faculty
2. Enter information of a students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **5**

Name of the staff member: **Jamal Kareem**

Enter the id: **ja6980**

Department: **Sciences**

Status, Enter P for Part Time, or Enter F for Full Time: **f**

Staff member added!

1. Enter the information a faculty
2. Enter information of a students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **6**

Enter the Staff's id: **ja6980**

Jamal Kareem ja6980

Sciences Department, Full Time

---

1. Enter the information a faculty
2. Enter information of a students
3. Print tuition invoice
4. Print faculty information
5. Enter the information of a staff member
6. Print the information of a staff member
7. Exit Program

Enter your selection: **7**

Goodbye!