

Note:

1) Projects can be completed in groups (max 4 per group) but submission onto WebCourses is individual. Once again, 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. It should look like this:

```
public class Project2 {  
  
    public static void main(String[] args){  
        //Test code goes here  
    }  
  
}  
  
//-----  
class Abc{  
  
}  
  
//-----  
class Xyz{  
  
}
```

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

```
/*  
- Project 2  
- 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 2  
- Hatim Boustique, Ericka Edwards and Jamal Dubois  
  
*/  
  
public class Project3 {  
  
    public static void main(String[] args){  
        //Test code goes here  
    }  
  
}  
//-----  
interface Int1 {  
  
    }  
//-----  
  
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 in 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 java file!

Project 2 statement

Please read this entire statement carefully before you start doing anything...(Please don't assume things)

This project is a **continuation** to the previous one (You should be able to reuse most of the code of Project1!). It involves implementing a simple university personnel management program.

Note well the following:

- You may use any data structure to store the objects. That is, you may use an array, array list, ...etc.
- Your code should handle ALL exceptions. The bottom line is that your code should NOT crash!
- In addition, you code must have a user-defined exception called *IdException* that handles the ids entered by the user. Note that an person's ID must be of the form **LetterLetterDigitDigitDigitDigit** like e17894. See sample run below. **Just as we had in Project1, we require that ids are checked for duplication. That is: No two persons can have the same id. An exception of type *IdException* is thrown when the id entered is either duplicated or in wrong format.**
- When option 7 is chosen to exist the program, the user is presented with the option to create a report that has the most up-to-date data from the array (or ArrayList list...) saved into a text file (**report.txt**). A sample report is provided at the end of this document.
- Note that your report.txt must list the students sorted by either the gpa or number of credit hours. Use the Java **Comparator** interface to sort the students. For that, copy the Student objects into an ArrayList that you sort by gpa or by credit hours.

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: **j1254**

Invalid ID format. Must be LetterLetterDigitDigitDigitDigit

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**

Would you like to create the report? (Y/N): **y**

Would you like to sort your students by (1) gpa or (2) credit hours: **1**

Report created and saved on your hard drive!

Goodbye!

Sample Report (**report.txt**)

Report created on 11/15/2021

Faculty Members

1. John Miller
ID: jo7894
Professor,Engineering

Staff Members

1. Jama Kareem
ID: ja6980
Sciences, Full Time

Students (**Sorted by gpa**)

1. Julia Alvarez
ID: ju1254
Gpa: 3.26
Credit hours: 7

2. Matt Jones
ID: ma0258
Gpa: 2.78
Credit hours: 0