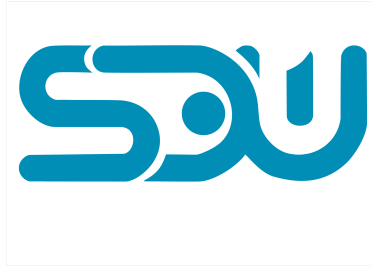


PROGRAMMING TECHNOLOGIES AND EDUCATIONAL PRACTICE



PROJECT ASSIGNMENT - 1

Project: Organizational Employee Management System (OOP)

100 points (5% from Total)

In this project, you will design a classical “Organizational Management System” using Object Oriented Programming (OOP) using Java Programming Language.

1. PROJECT DESCRIPTION

Employee Management System is a Java application designed to perform and handle employee management functions. Using this Object-oriented application, users can perform management of employees such as adding new employee details, editing/updating their details, removing them from the system, or viewing all employee details of their company.

Basically, the application is divided into four different operations/functionalities: (1) Addition of new User or employee (2) Removing existing User or employee (3) Updating User details in the system (4) Viewing information of existing users. (5) Exiting.

2. PROJECT FLOW

Write a java program to implement “Organizational Employee Management System”. In which you have to include given topics as per requirement of the program: Inheritance, Polymorphism, Abstract Classes, Interfaces, Exceptional Handling and I/O Fundamentals. Avoid copying the solutions from any student/website. Try to implement using the concept covered in the class.

Solution must be unique for each student.

For each of these features you have to create a separate class to maintain the decorum of the code. Also you can implement the concept of Switch Cases to automate the task and make it more useful and user friendly. Users need to just enter the numbers as per given and it will perform the task.

In this code you have to use the concept of Classes and Objects to solve the problem. The requirements are a menu for the user which consists of 6 different features.

1. To add an employee

2. Add an exception handler for every modifier if the opposite value is entered “*like int instead of String*”.

- a. (String)Employee name;
- b. (int)Employee age;
- c. (String)Employee email;
- d. (int)Employee position;
- e. (int)Employee ID;
- f. (float)Employee salary;
- g. (long)Employee contact;

3. To view details of an employee

- a. Implement `getInfo() method;`
- b. Ask user to enter details of employee
- c. If the file does not exist throw exception “FileNotFoundException:”

4. To remove “file” info of an employee

- a. Implement method `removeFile(int ID) .`
- b. Create a file when each employee information is entered by the user with the name of employee ID (“file” + ID + “.txt”). Each employee information should be saved into a file directly.
- c. If the employee file exists(`file.exists`) “delete employee”. Remove the txt file.
Print message: `System.out.println("\nEmployee has been removed Successfully");`

d. **else:** `System.out.println("\nEmployee does not exists :(");`

5. To update “file” info of an employee

- a. Press to update Employee details.
- b. Enter the ID of the employee which you want to update.
- c. Print the employee information from the file on the screen.
- d. Then ask user:

Please Enter the detail you want to Update :

For example:

If you want to Change the Name, then Enter Current Name and Press Enter. Then write the new Name then Press Enter. It will Update the Name.

>>name

Please Enter the Updated Info :**Name**

6. To exit the Portal.

- a. Press to exit, print message: `System.out.println(" Thank You For Sharing your details :)");`

7. In order to continue add “Press Enter to Continue each time”

Files to Create: Here is some information about creating classes and objects. Use the same logic for other classes too and try to brainstorm.

1. *EmployManagementSystem*

a. *Create 5 Switch Cases*

Case 1 -> Create file

Case 2 -> Enter employee ID, view file.

Case 3 -> Enter employee ID (To remove)

Case 4 -> Update employee. If the employee file ID does not exist, throw an exception.

Case 5 -> CodeExit.

2. *Employee_Add*

Usage in **EmployManagementSystem** class:

Employee_Add ep =new Employee_Add();

ep.createFile();

3. *Employee_Remove implements Remove*

4. *Employee_Show*

5. *Employee_Update*

6. *EmployDetail(abstract class)*

a. 1 inheritor -> *Employee_Add class.*

b. usage in *Employee_Add class.*

```
EmployDetail emp = new EmployDetail()
```

7. *MainMenu(each time user “Presses enter to continue ” go to MainMenu.)*

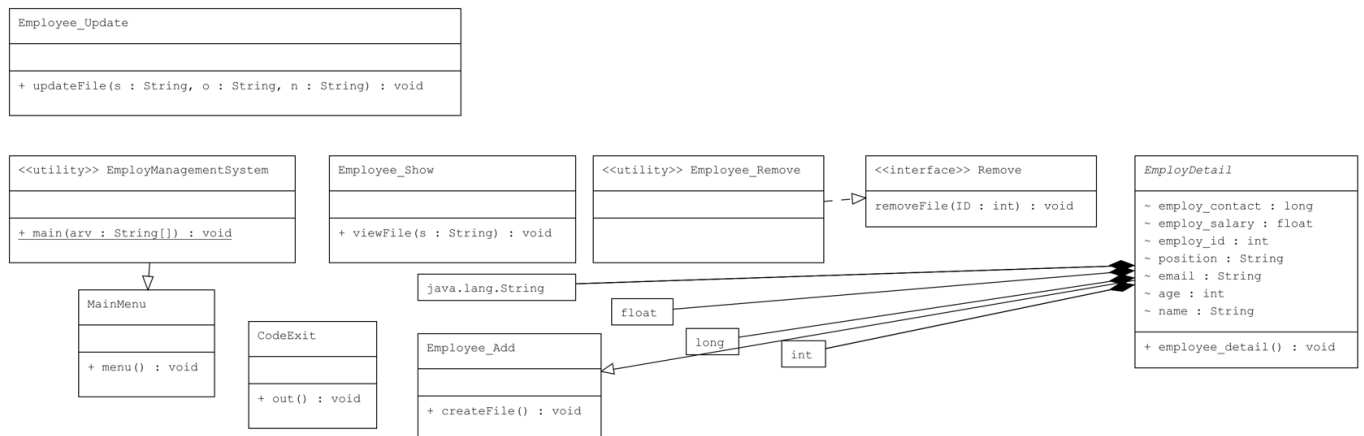
a. public void menu() -> 5 usages in “EmployManagementSystem”

```
MainMenu obj1 = new MainMenu();
```

```
obj1.menu();
```

8. *CodeExit*

UML DIAGRAM - Take a look at the Diagram for the method and interface infrastructure.



FLOW OF THE EMPLOYEE INFRASTRUCTURE

1. Home Page - Class “Main Menu”

```
*****
      Organizational Management System
*****

Press 1 : Add an Employee Details
Press 2 : See an Employee Details
Press 3 : Remove an Employee
Press 4 : Update Employee Details
Press 5 : Exit the EMS Portal

Please Enter choice :|
```

2. Ex: Press 1 to add Employee Details.

```
*****
      Organizational Management System
*****

Press 1 : Add an Employee Details
Press 2 : See an Employee Details
Press 3 : Remove an Employee
Press 4 : Update Employee Details
Press 5 : Exit the EMS Portal

Please Enter choice :1
Enter Employee's name -----: Name Surname
Enter Employee's Age -: Age
Enter Employee's ID -----: 112233
Enter Employee's Email ID ----: 123@mail
Enter Employee's Position ----: Position
Enter Employee contact Info --: 12345
Enter Employee's Salary -----: 1500$

Employee has been Added :)
```

3. Ex: Press 2 to see Employee details

```
*****
Organizational Management System
*****

Press 1 : Add an Employee Details
Press 2 : See an Employee Details
Press 3 : Remove an Employee
Press 4 : Update Employee Details
Press 5 : Exit the EMS Portal

Please Enter choice : 2

Please Enter Employee's ID : 112233
Employee ID:112233
Employee Name      :Name Surname
Age               :Age
Employee Contact   :12345
Email Information  :123@mail
Employee position  :Position
Employee Salary    :1500$

Press Enter to Continue..|
```

4. Ex: Press 3 to Remove Employee txt file information.

```
Press Enter to Continue...

*****
Organizational Management System
*****

Press 1 : Add an Employee Details
Press 2 : See an Employee Details
Press 3 : Remove an Employee
Press 4 : Update Employee Details
Press 5 : Exit the EMS Portal

Please Enter choice : 3

Please Enter Employee's ID : 112233

Employee has been removed Successfully

Press Enter to Continue..|
```

5. Ex: Press 4 to Update Employee Details

```
Please Enter choice :4

Please Enter Employee's ID :112233
Employee ID:112233
Employee Name      :Name Surname
Age               :Age
Employee Contact   :12345
Email Information   :123@mail
Employee position   :Position
Employee Salary     :1500$
Please Enter the detail you want to Update :
For Example :
If you want to Change the Name, then Enter Current Name and Press Enter. Then write the new Name then Press Enter. It will Update the Name.

Employee Name
Please Enter the Updated Info :New Name Surname

Press Enter to Continue...|
```

6. Ex: Press 5 to Exit Portal

```
*****
      Organizational Management System
*****

Press 1 : Add an Employee Details
Press 2 : See an Employee Details
Press 3 : Remove an Employee
Press 4 : Update Employee Details
Press 5 : Exit the EMS Portal

Please Enter choice :5

*****
Thank You For Sharing your details :)
*****
```

NOTES ON GRADING

Make sure that every class does exactly what is described in the project requirements. Do NOT share your code with anyone. It would be considered academic dishonesty, and strictly penalized. All the works submitted shall be inspected by a special program and reviewed by the instructors. Any kind of similarity, or not being able to answer questions on the project gives the instructor full right to penalize the work, and even cancel the results that have already been graded because

of cheating issues. In simple words, the fact that you did not cheat yourself, or that you showed your work and it has already been graded, does not help you. All the sides involved in cheating (which is a crime) will be penalized. Again, do not share your code under any conditions. Although you are strongly encouraged to ask questions and have discussions on Google Classroom, make sure you do not share your code there as well, please.