Hands-on Activity
People in School

Objective:

At the end of the activity, the students should be able to:

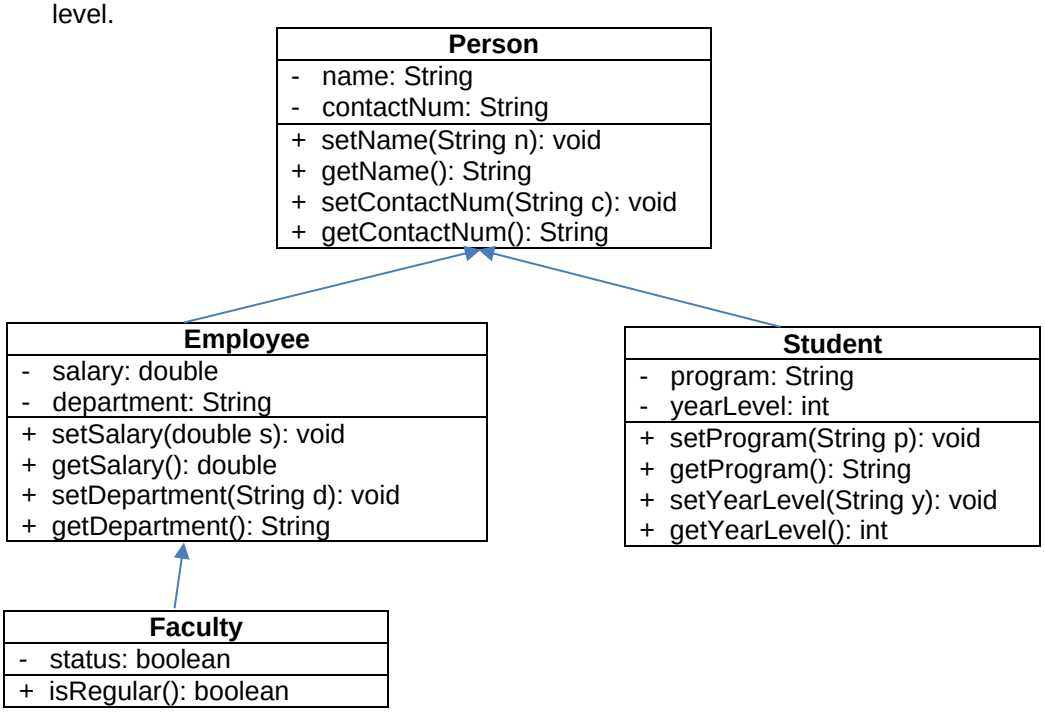
- Create a program that exhibits inheritance.

Software Requirements:

- Latest version of NetBeans IDE
- Java Development Kit (JDK) 8

Procedure:

1. Write a simple information system that will store and display the complete information of a student, faculty, or employee.
2. Create four (4) no-modifier classes named **Person**, **Student**, **Faculty**, and **Employee**.
3. Create a public class named **CollegeList**. This class shall contain the main method.
4. Refer to the UML Class Diagram for the names of the variables and methods. The (-) symbol represents private variables, while (+) represents public method. This should be the sequence of the program upon execution:
    a. Prompt the user to select among Employee, Faculty, or Student, by pressing **E**, **F**, or **S**.
    b. Ask the user to type the name and contact number.
    c. For Employee, ask the user to type the employee's monthly salary and the department where he/she belongs to *(Ex. Registrar)*. Then, display name, contact number, salary, and department.
    For Faculty, ask the user to press **Y** if the faculty member is regular/tenured or **N** if not. Then, display name, contact number, salary, department, and status.
    For Student, ask the user to type the student's enrolled program *(Ex. BSIT, BSTM)* and his/her year level (integers 1 to 4). Then, display name, contact number, program, and year level.



Sample Output:

```
Press E for Employee, F for Faculty, or S for Student: e
Type employee's name, contact number, salary, and department.
Press Enter after every input.
Mairo Berlin P. Reyes
09198765432
24500
Registrar
------------------------------
Name: Mairo Berlin P. Reyes
Contact Number: 09198765432
Salary: 24500.0
Department: Registrar
```

GRADING RUBRIC (100 points):

Criterion	Description	Max Points
Correctness	The code produces the expected result.	40
Logic	The code meets the specifications of the problem.	40
Efficiency	The code is concise without sacrificing correctness and logic.	10
Syntax	The code adheres to the rules of the programming language.	10