National University of Computer and Emerging Sciences

Laboratory manual
For
Software Design and Architecture

Course Instructor	**Amir Iqbal**
Lab Instructor	**Muhammad Ahmed Khan** **Shumaila Arshad**
Email	
Section	**BSE-4C**
Date	**01-02-24**
Semester	**Spring 24**

Instructions for lab submission:
You have to submit source code (.java) files along with a word document on google classroom. In the word document you have to give the heading of each exercise/question, then paste your source code and output snippet. Save your word document in the following format: roll number-lab no-section i.e. 21I-0008-lab1-BSE4D.

Objective:

1. OOP recap
2. Java practice

Software for this lab:

1. Apache NetBeans IDE

Exercise 1:
Create a class named “Student” with private variables “name”, “age”, “degree” and a public variable “nationality”. Add setters and getters for this class. Add a public method “showDetails” which will print all student data.

Exercise 2: Employee Payroll System with Inheritance and Polymorphism
The objective of this exercise is to create a simple employee payroll system using inheritance and polymorphism. This exercise will involve designing a class hierarchy for different types of employees and utilizing polymorphism to calculate and display their monthly salary.

1. Define a base class named "Employee" with the following attributes and methods:
 - Attributes: `name` (string), `employee_id` (int)
 - Methods:
 - `public Employee(String name, int employee_id)`: Constructor to initialize the employee's name and ID.
 - `calculate_salary()`: Abstract method to be implemented by derived classes for calculating the monthly salary.

2. Create two derived classes: "HourlyEmployee," and "SalariedEmployee," inheriting from the "Employee" class.

3. Implement the following in each derived class:

- HourlyEmployee:
  - Attributes: `hourly_rate` (float), `hours_worked` (float)
  - Methods:
    - `HourlyEmployee(String name, int employee_id, float hourly_rate, float hours_worked)`: Constructor to initialize the hourly employee's information.
    - `calculate_salary()`: Override the base class method to calculate the monthly salary for an hourly employee i.e. hourly_rate * hours_worked.

- SalariedEmployee:
  - Attributes: `monthly_salary` (float)
  - Methods:
    - `SalariedEmployee(String name,int employee_id,float monthly_salary)`: Constructor to initialize the salaried employee's information.
    - `calculate_salary()`: Override the base class method to return the monthly salary for a salaried employee.

4. Create a function named `display_employee_information` that takes a list of employees and displays each employee's name, ID, and monthly salary using polymorphism.
5. In the main part of your program, create instances of each type of employee, add them to a list, and call `display_employee_information` with this list.

Exercise 3:
1. Create a derived class "CommissionedEmployee," inheriting from the "Employee" class.
Implement the following attributes and methods:
- Attributes: `base_salary` (float), `commission_rate` (float), `sales_amount` (float)
  - Methods:
    - `CommissionedEmployee(String name,int employee_id, float base_salary, float commission_rate, float sales_amount)`: Constructor to initialize the commissioned employee's information.
    - `calculate_salary()`: Override the base class method to calculate the monthly salary for a commissioned employee.

2. In the Main program, create instances of CommissionedEmployee and reuse the previously created “display_employee_information” method to display information of CommissionedEmployee.