

Lab worksheet 4: Object Oriented Concepts

1. Imagine you are building a University Management System using Java to manage all the academic and administrative activities of the university. The system will involve the creation of classes like **Person**, **Lecturer**, **Student**, **Degree**, **Department**, **Course**, and **UniversityManagementSystem**. These classes will help you showcase OOP principles.

Implement the following classes:

- **Person Class:**

- Create an abstract class named **Person**.
- Add a private String variable **name**.
- Include an abstract method **displayInfo()**.
- Implement getter and setter methods for the name variable.
- Provide a constructor that initializes the name variable.

- **Lecturer Class:**

- Create a class named **Lecturer** that extends the **Person** class.
- Add a private String variable **position**.
- Include a variable **department** of type **Department**.
- Add a private ArrayList variable **coursesTeaching** to store **Course** objects.
- Implement a constructor to initialize all variables.
- Implement the **displayInfo()** method to display the **Lecturer** information.
- Implement getter and setter methods for the **position** variable.
- Implement a setter method for the **department** variable.
- Implement **displayDepartmentInfo()** to display information about the **department**.
- Implement **addCourse()** method to add a **Course** to the **coursesTeaching** ArrayList.
- Implement **removeCourse()** method to remove a **Course** from the **coursesTeaching** ArrayList.
- Implement **listCoursesTeaching()** method to print the details of the **Course** objects in the **coursesTeaching** ArrayList.

- **Student Class:**

- Create a class named **Student** that extends the **Person** class.
- Add private String variables **studentID** and **year**.
- Include a variable **degree** of type **Degree**.
- Add a private ArrayList variable **coursesEnrolled** to store **Course** objects.
- Implement a constructor to initialize all variables.

Lab worksheet 4: Object Oriented Concepts

- Implement the **displayInfo()** method to display the **Student** information.
 - Implement getter and setter methods for **studentID** and **year**.
 - Implement **registerDegree()**, **displayDegreeInfo()**, **enrollCourse()**, **unenrollCourse()**, and **listCoursesEnrolled()** methods.
- **Degree Class:**
 - Create a class named **Degree**.
 - Add a private String variable **name**, a private Integer variable **numberOfStudents**, and a private ArrayList variable **coursesOffering** to store **Course** objects.
 - Implement a constructor to initialize these variables.
 - Implement the **displayInfo()** method to display **Degree** information.
 - Implement getter and setter methods for **name** and **numberOfStudents**.
 - Implement **offerCourse()**, **withdrawCourse()**, and **listCoursesOffering()** methods.
- **Department Class:**
 - Create a class named **Department**.
 - Add private String variable **name**.
 - Include a variable **departmentHead** of type **Lecturer**.
 - Add private ArrayList variables **coursesOffering** and **lecturersBelongsTo** to store **Course** and **Lecturer** objects.
 - Implement a constructor to initialize these variables.
 - Implement the **displayInfo()** method to display **Department** information.
 - Implement getter and setter methods for **name**.
 - Implement **appointDepartmentHead()**, **displayDepartmentHeadInfo()**, **offerCourse()**, **withdrawCourse()**, **addLecturer()**, and **removeLecturer()** methods.
- **Course Class:**
 - Create a class named **Course**.
 - Add private String variables **name**, **enrollType**, and private Integer variable **numberOfStudentsEnrolled**.
 - Include variables **lecturerInCharge** of type **Lecturer** and **degreeBelongsTo** of type **Degree**.
 - Implement a constructor to initialize these variables.
 - Implement the **displayInfo()** method to display **Course** information.

Lab worksheet 4: Object Oriented Concepts

- Implement getter and setter methods for **name**, **enrollType**, and **numberOfStudentsEnrolled**.
- Implement **addLecturerInCharge()**, **removeLecturerInCharge()**, **addDegreeBelongsTo()**, and **removeDegreeBelongsTo()** methods.
- **UniversityManagementSystem** Class:
 - Create a class named **UniversityManagementSystem** with the main method.
 - Inside the main method, create a sample scenario with the following steps:
 - Create one or more **Department** object(s) (e.g., "Software Engineering").
 - Create one or more **Degree** object(s) (e.g., "Computer Science").
 - Create one or more **Course** object(s) (e.g., "Object-Oriented Programming").
 - Create one or more **Lecturer** object(s).
 - Create one or more **Student** object(s).
 - Add all the missing information to the objects using the methods provided in their respective class.
 - Display information about the **Lecturer**, the **Student**, the **Department**, the **Course**, and the **Degree**.