Lab worksheet 4: Introduction

Instructions

1. Create a Java project in IntelliJ within your folder and name it using your student number in the following formats: **"OOP_CS_2022_XXX" or "OOP_ET_2022_XXX"**. (Eg: OOP_CS_2022_001, OOP_ET_20222_007)
2. Create distinct packages for each lab worksheet, naming them in the following format "**LW_XX**." (Eg: LW_01)
3. Create distinct classes for each question, naming them "**QX**." (Eg: Q1)
4. Upload your project files to your GitHub repository.

Questions

1. Imagine you are building a library system using Java to manage books and other library items. The system will involve the creation of classes for **BorrowableItems, Book, Person,** and **Student**. These classes will help you showcase OOP principles.

   Create an abstract class called **BorrowableItems** with an abstract method **displayInfo()**.

   Create a class named **Book** as a subclass of the **BorrowableItems** class with the following private attributes:
   - **title** (String)
   - **author** (String)
   - **ISBN** (String)
   - **available** (boolean)

   Provide public getter and setter methods for these attributes.
   Include a constructor to initialize the attributes.
   Implement the **displayInfo()** method that displays information about the book.

   Create a **Person** class with the following private attribute:
   - **name** (String).

   Create a constructor to set the **name**.

   Create a **Student** class that inherits from the **Person** class.

   Create a **Library** class with the following private attribute:
   - **BorrowableItemsList** (Use an ArrayList or another suitable data structure to store **BorrowableItems** objects. Refer to any internet resource for the syntax of the data structure.)

   Implement methods to:
   - Add library items to the collection.
   - Check out library items (mark them as available or unavailable).

- List all available library items in the library.
- Display information about the library, including the number of available items.

Create instances of the **Book** class representing library books. Use the following book information:

- Title: "Seena Maragatham", Author: "Sujatha Thilaka", ISBN: "9789553023975"
- Title: "Nuramakadya Bauthika Nuladanayakshanamaala", Author: "Seynamasasekaka", ISBN: "9789553548721"
- Title: "Island of a Thousand Mirrors", Author: "Nayomi Munaweera", ISBN: "9781616953623"

Create an instance of the **Library** class to manage the library system.

Add the created books to the library's collection using the **addLibraryItem** method of the **Library** class. For each book, call **addLibraryItem** to include them in the library's inventory.

Display information about the library, including the total number of items, by calling the **displayLibraryInfo** method.

List all available library items in the library using the **listAvailableItems** method of the **Library** class.

Demonstrate the process of checking out a library item by calling the **checkoutItem** method. For example, check out the book with the title "Island of a Thousand Mirrors" by setting its availability to false.

List available items in the library again using the **listAvailableItems** method. Observe that the book you checked out is now marked as unavailable.

2. 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.
  - 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**.

Lab worksheet 4: Introduction

- 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.
  - 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.

Lab worksheet 4: Introduction

- 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**.