

Lab worksheet 5: Object Oriented Concepts

1. You are designing a class named **Constants** to store mathematical and scientific constants. Implement the class with the following requirements:

Declare a final double constant named **PI** and initialize it with the value 3.14159.

Declare a final int constant named **SPEED_OF_LIGHT** and initialize it with the value 299792458 (speed of light in meters per second).

Implement a constructor that takes a double parameter and initializes a final double constant named **GRAVITY** with the provided value.

2. Imagine you are developing a **library management system** in Java, and you want to create classes for handling books in different genres. You decide to organize these classes into packages to maintain a clean and structured codebase.

Create a package named **library.books** that will contain classes related to books.

Inside the **library.books** package, create a class named **FictionBook** that represents a fictional book. Include attributes such as **title**, **author**, and **genre**. Implement a method **displayInfo()** to display information about the fiction book.

Create another package named **library.utils** that will contain utility classes.

Inside the **library.utils** package, create a class named **BookUtils** that provides a utility method for processing books. Implement a static method named **printBookDetails** that takes a **FictionBook** object as a parameter and prints its details.

3. You are tasked with creating a simple class called **BankAccount** that represents a user's bank account. Implement the class with the following requirements:

Create a private variable **balance** to store the account balance.

Implement a constructor that initializes the **balance** to 0.

Create public methods **deposit** and **withdraw** to modify the account **balance**.

Ensure that the **withdraw** method cannot make the **balance** negative. If an attempt is made to **withdraw** more than the current **balance**, print an error message and do not perform the **withdrawal**.

Implement a public method **getBalance** to retrieve the current **balance**.

Lab worksheet 5: Object Oriented Concepts

In a separate class named **BankManagementSystem**, demonstrate the usage of the **BankAccount** class by performing the following steps:

Create an instance of **BankAccount**.
Deposit an amount into the account.
Withdraw an amount from the account.
Display the current **balance**.

Ensure that the **BankAccount** class uses appropriate access modifiers for encapsulation.

4. You are developing a utility class named **MathOperations** to perform common mathematical operations. Implement the class with the following requirements:

Declare a static variable named **pi** of type double and initialize it with the value 3.14159.

Implement a static method named **calculateCircleArea** that takes the **radius** as a parameter and returns the **area of a circle** using the formula: **area = pi * radius * radius**.

Implement a static method named **calculateSquareArea** that takes the **side length** as a parameter and returns the **area of a square** using the formula: **area = side * side**.

Ensure that the **pi** variable is accessible only within the **MathOperations** class.

Demonstrate the usage of the **MathOperations** class in a separate class named **GeometryCalculator** by performing the following steps:

Print the value of the static variable **pi** from the **MathOperations** class.
Calculate and print the **area of a circle** with a **radius** of 5 using the **calculateCircleArea** method.
Calculate and print the **area of a square** with a **side length** of 4 using the **calculateSquareArea** method.

5. You are tasked with implementing a program that calculates the area of a rectangle and a square. Utilize the concept of scope in Java to create a solution. Implement the following:

Create a class named **AreaCalculator**.

Declare a static method named **calculateRectangleArea** that takes two parameters (**length** and **width**) and calculates the **area of a rectangle**.

Lab worksheet 5: Object Oriented Concepts

Declare a static method named **calculateSquareArea** that takes one parameter (**sideLength**) and calculates the **area of a square**.

In the main method:

- Declare a variable **rectangleLength** and assign it the value 5.

- Declare a variable **rectangleWidth** and assign it the value 8.

- Declare a variable **squareSideLength** and assign it the value 4.

- Call the **calculateRectangleArea** method and print the result.

- Call the **calculateSquareArea** method and print the result.