Lab 02: Mastering Classes and Objects in Java

Objectives:

- Understand the concept of method parameters and how to pass arguments to methods.
- Practice creating and using objects from multiple classes.
- Implement methods that interact with objects of different classes.
- Design and develop a simple program using multiple classes to solve a real-world problem.

1. Classes and Objects:

- **Classes:** A blueprint or template that defines the properties (member variables) and functionalities (methods) of objects. Imagine a class as a cookie cutter used to create multiple cookies (objects) with the same shape and features.
- **Objects:** Instances of a class. Each object has its own set of variables and methods, allowing them to hold unique data and perform actions. Think of each cookie created from the cookie cutter as a separate object.

Example:

```
class Car {
  String model; // Member variable
  int year;   // Member variable

  public void startEngine() { // Method
    System.out.println("Engine started!");
  }
}

// Creating an object of the Car class
Car myCar = new Car();
myCar.model = "Corolla"; // Assigning value to a member variable
myCar.year = 2023;
myCar.startEngine(); // Calling a method on the object
```

2. Methods with Parameters:

- Methods can receive data as input using parameters. This allows for flexibility in how methods operate.
- Parameters are declared within the method's parentheses, specifying the data type and name.
- Inside the method, the parameter names are used to access the passed data.

Example:

```
class Greeting {
  public void sayHello(String name) { // Parameter 'name'
    System.out.println("Hello, " + name + "!");
  }
}
```

```
Greeting greeter = new Greeting();
greeter.sayHello("Ali"); // Passing argument to the parameter
```

3. Object Interaction:

- Objects of different classes can interact and collaborate by calling each other's methods. This allows for complex program logic and data exchange between different parts of your program.

Example (using the previous Car and a new Driver class):

```
class Driver {
  String name;

  public void drive(Car car) {  // Parameter of type Car
    System.out.println(name + " is driving a " + car.model + " (" + car.year + ").");
  }
}

Driver ali = new Driver();
ali.name = "Ali";
ali.drive(myCar);
```

Exercises

1. Create a class named Student with member variables for name, roll number, and marks in three subjects. Write a constructor to initialize these variables.
2. Create a class named Calculator with methods for addition, subtraction, multiplication, and division. Each method should take two numbers as parameters and return the result.
3. Create a class named Book with member variables for title, author, and genre. Write a method named displayDetails() that prints the book information.
4. Create a class named Shape with a member variable for color (default: "black"). Create a subclass named Circle that inherits from Shape and has a member variable for radius. Implement a method in Circle to calculate the area (π * radius^2).
5. Write a program that creates two Rectangle objects (assuming you have a Rectangle class similar to Circle) with different widths and heights. Calculate and print the area of each rectangle.

Challenging & Real-World Exercises

1. Grocery Store Inventory:

- Create a class named Product with member variables for name, price, and quantity (in stock).
- Create a class named GroceryStore that has an array of Product objects representing its inventory.
- Implement methods in GroceryStore to:
  - Add a new product to the inventory.
  - Sell a product (reduce quantity by 1, assuming sufficient stock).
  - Search for a product by name and display its details (if found).
  - Print a report showing all products and their current stock levels.

2. Online Shopping Cart:

- Create a class named Item with member variables for product name, price, and quantity (desired by the customer).
- Create a class named ShoppingCart that has an array of Item objects representing the customer's cart.
- Implement methods in ShoppingCart to:
  - Add an item to the cart (check if the product exists before adding).
  - Update the quantity of an existing item in the cart.
  - Remove an item from the cart.
  - Calculate the total bill amount considering item prices and quantities.
  - Print the cart contents with item names, quantities, and individual prices.

3. Social Media Post:

- Create a class named Post with member variables for author name, content (text), and number of likes.

- Create another class named SocialMediaPlatform that has an array of Post objects representing user posts.
- Implement methods in SocialMediaPlatform to:
    - Create a new post for a user (add it to the array).
    - Like a post by incrementing its likes count (check if the post exists).
    - Search for posts by a specific user and display them.
    - Display the post with the highest number of likes.

4. Restaurant Menu:

- Create a class named Dish with member variables for name, description (optional), and price.
- Create another class named RestaurantMenu that has an array of Dish objects.
- Implement methods in RestaurantMenu to:
    - Add a new dish to the menu.
    - Display all dishes with their names and prices.
    - Search for a dish by name and display its details (if found).
    - Categorize dishes based on a type (e.g., appetizer, main course) and display them separately (assuming a category member variable in Dish).

5. Music Playlist:

- Create a class named Song with member variables for title, artist, and genre.
- Create a class named MusicPlayer that has an array of Song objects representing a playlist.
- Implement methods in MusicPlayer to:
    - Add a song to the playlist (check for duplicates).
    - Remove a song from the playlist.
    - Shuffle the playlist (randomly reorder the songs).
    - Play a song by title (search and play the audio file - simulation using system commands).
    - Create playlists based on genre (create a separate playlist object) and add songs from the main playlist based on their genre.

Roll No: ____________________________________

Name: ______________________________________

Subject: _____________________________________

Teacher Name:________________________________