



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Experiment No. 3
Implement a program that demonstrates the concepts of class and objects
Date of Performance:
Date of Submission:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Aim: Implement a program that demonstrates the concepts of class and objects

Objective: To develop the ability of converting real time entity into objects and create their classes.

Theory:

A class is a user defined blueprint or prototype from which objects are created. It represents the set of properties i.e., members and methods that are common to all objects of one type. In general, class declarations can include these components, in order:

1. Modifiers: A class can be public or has default access.
2. class keyword: class keyword is used to create a class.
3. Class name: The name should begin with a initial letter (capitalized by convention).
4. Superclass (if any): The name of the class's parent (superclass), if any, preceded by the keyword extends. A class can only extend (subclass) one parent.
5. Interfaces (if any): A comma-separated list of interfaces implemented by the class, if any, preceded by the keyword implements. A class can implement more than one interface.
6. Body: The class body surrounded by braces, {}.

An OBJECT is a basic unit of Object-Oriented Programming and represents the real-life entities. A typical Java program creates many objects, which interact by invoking methods.

An object consists of:

1. State: It is represented by attributes of an object. It also reflects the properties of an object.
2. Behavior: It is represented by methods of an object. It also reflects the response of an object with other objects.
3. Identity: It gives a unique name to an object and enables one object to interact with other objects.

Code:

// A simple Java program to demonstrate classes and objects

```
class Rectangle
{
    int length;
    int width;

    // Constructor
    public Rectangle(int length, int width)
    {
        this.length = length;
        this.width = width;
    }

    // Method
    public int calculateArea()
    {
        return length * width;
    }
}

public class Main {
    public static void main(String[] args)
    {
        // Creating an object of the Rectangle class
        Rectangle rect1 = new Rectangle(5, 10);

        // Accessing the object's attributes and methods
        System.out.println("Length of rect1: " + rect1.length);
        System.out.println("Width of rect1: " + rect1.width);
        System.out.println("Area of rect1: " + rect1.calculateArea());
    }
}
```

Output -

```
Length of rect1: 5  
Width of rect1: 10  
Area of rect1: 50  
  
=== Code Execution Successful ===
```

Conclusion:

Creating classes and objects in Java is a fundamental aspect of object-oriented programming that enables developers to model real-world entities and behaviors effectively. By defining a class as a blueprint with attributes and methods, you can encapsulate data and functionality in a single structure. This not only promotes code organization and reusability but also enhances maintainability and readability.