



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.



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Code:

```
import java.io.*;
import java.util.*;
class Car {
    String model;
    String color;
    int year;

    void start() {
        System.out.println(model + " is starting.");
    }

    void drive() {
        System.out.println(model + " is driving.");
    }
}

public class Main {
    public static void main(String[] args) {
        Car car1 = new Car();
        car1.model = "Tesla Model S";
        car1.color = "Red";
        car1.year = 2020;

        Car car2 = new Car();
        car2.model = "BMW M3";
        car2.color = "Black";
        car2.year = 2019;

        car1.start();
        car1.drive();

        car2.start();
        car2.drive();
    }
}
```

output:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
Tesla Model S is starting.  
Tesla Model S is driving.  
BMW M3 is starting.  
BMW M3 is driving.  
  
=== Code Execution Successful ===
```

Conclusion:

Creating a class template and its objects in Java is a foundational aspect of object-oriented programming. The class serves as a reusable blueprint, allowing developers to define data and behavior in a structured way. By instantiating objects from a class, you can create multiple entities with similar characteristics while maintaining their own state. This encapsulation and reusability enhance code organization, maintainability, and scalability in software development.