**Anudip Foundation – AJP Lab 4**

**PROGRAM:**

class Vehicle {

    String brand;

    int speed;

    Vehicle(String brand, int speed) {

        this.brand = brand;

        this.speed = speed;

    }

    void displayInfo() {

        System.out.println("Brand: " + brand);

        System.out.println("Speed: " + speed + " km/h");

    }

}

class Car extends Vehicle {

    int numDoors;

    Car(String brand, int speed, int numDoors) {

        super(brand, speed);

        this.numDoors = numDoors;

    }

    void displayInfo() {

        super.displayInfo();

        System.out.println("Number of Doors: " + numDoors);

    }

}

class Bike extends Vehicle {

    String type;

    Bike(String brand, int speed, String type) {

        super(brand, speed);

        this.type = type;

    }

    void displayInfo() {

        super.displayInfo();

        System.out.println("Bike Type: " + type);

    }

}

class VehicleTest {

    public static void main(String[] args) {

        Car car = new Car("Toyota", 180, 4);

        Bike bike = new Bike("Yamaha", 120, "Sports");

        System.out.println("Car Details:");

        car.displayInfo();

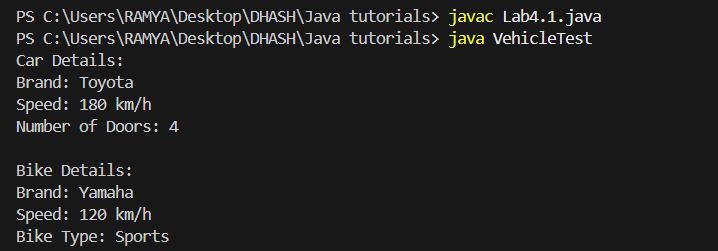
        System.out.println("\nBike Details:");

        bike.displayInfo();

    }

}

**OUTPUT:**



**PROGRAM:**

class Employee {

    String name;

    int id;

    double salary;

    Employee(String name, int id, double salary) {

        this.name = name;

        this.id = id;

        this.salary = salary;

    }

    void showDetails() {

        System.out.println("Name: " + name);

        System.out.println("ID: " + id);

        System.out.println("Salary: " + salary);

    }

}

class Manager extends Employee {

    String department;

    Manager(String name, int id, double salary, String department) {

        super(name, id, salary);

        this.department = department;

    }

    @Override

    void showDetails() {

        super.showDetails();

        System.out.println("Department: " + department);

    }

}

class Developer extends Employee {

    String programmingLanguage;

    Developer(String name, int id, double salary, String programmingLanguage) {

        super(name, id, salary);

        this.programmingLanguage = programmingLanguage;

    }

    @Override

    void showDetails() {

        super.showDetails();

        System.out.println("Programming Language: " + programmingLanguage);

    }

}

class EmployeeTest {

    public static void main(String[] args) {

        Manager manager = new Manager("Alice", 101, 75000, "HR");

        Developer developer = new Developer("Bob", 102, 85000, "Java");

        System.out.println("Manager Details:");

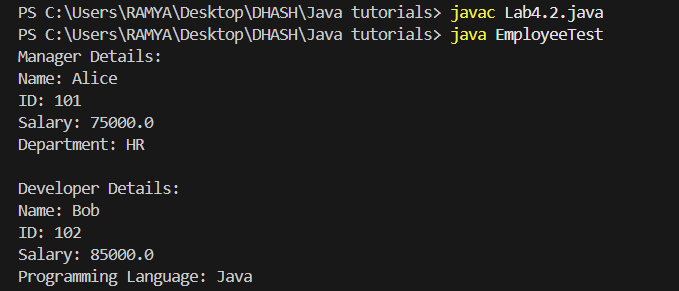
        manager.showDetails();

        System.out.println("\nDeveloper Details:");

        developer.showDetails();

    }

}

**OUTPUT:** 

**PROGRAM:**

class Person {

    String name;

    int age;

    Person(String name, int age) {

        this.name = name;

        this.age = age;

    }

    void display() {

        System.out.println("Name: " + name);

        System.out.println("Age: " + age);

    }

}

class Student extends Person {

    String course;

    int rollNo;

    Student(String name, int age, String course, int rollNo) {

        super(name, age);

        this.course = course;

        this.rollNo = rollNo;

    }

    void display() {

        super.display();

        System.out.println("Course: " + course);

        System.out.println("Roll No: " + rollNo);

    }

}

class Professor extends Person {

    String subject;

    int employeeId;

    Professor(String name, int age, String subject, int employeeId) {

        super(name, age);

        this.subject = subject;

        this.employeeId = employeeId;

    }

    void display() {

        super.display();

        System.out.println("Subject: " + subject);

        System.out.println("Employee ID: " + employeeId);

    }

}

class CollegeSystem {

    public static void main(String[] args) {

        Student student = new Student("John", 20, "CSE", 1101);

        Professor professor = new Professor("Dr. Smith", 45, "Physics", 2001);

        System.out.println("Student Details:");

        student.display();

        System.out.println("\nProfessor Details:");

        professor.display();

    }

**OUTPUT:**

