Lab Exercise No:1

Exercise Objective(s): The concept of inheritance

Exercise: Create a class called Vehicle. Create subclasses like Truck, Bus, Car etc. Add common methods
in the base class and specific methods in the corresponding class. Create a class called Road and create objects for the Truck, Car, Bus etc and display the appropriate message.


Road.java

```
package com.oops;

 class Vehicle {
        String name;
        int wheels;
         Vehicle(String name,int wheels){
                this.name=name;
                this.wheels=wheels;
        }
          void displayInfo() {
                 System.out.println("Vehicle Name: " + name + ", Wheels: " + wheels);
         }


}
class Truck extends Vehicle{
        int loadcapacity;
        Truck(String name,int wheels,int loadcapacity){
                super(name,wheels);
                this.loadcapacity=loadcapacity;
        }
        void displayInfo() {
       super.displayInfo();
       System.out.println("Load Capacity: " + loadcapacity + " tons");
   }

}

class Bus extends Vehicle {
   int passengerCapacity;

   Bus(String name, int wheels, int passengerCapacity) {
      super(name, wheels);
      this.passengerCapacity = passengerCapacity;
   }
```

```
    void displayInfo() {
        super.displayInfo();
        System.out.println("Passenger Capacity: " + passengerCapacity + " people");
    }
}
 public class Road {
    public static void main(String[] args) {
        Truck truck = new Truck("Truck", 6, 10);
        Bus bus = new Bus("Bus", 4, 50);
        truck.displayInfo();
        System.out.println();
        bus.displayInfo();
        System.out.println();
    }
   }
```

Lab Exercise No:2

Write a Java program to Implement single inheritance

SingleInheritance.java

```
package com.oops;


class Animal{
        String name;
        Animal(String name){
                this.name=name;
        }
        void DispalyInfo() {
                System.out.println("Animal name: "+name);
        }
}
class Dog extends Animal{
        String breed;
        Dog(String breed,String name){
                super(name);
                this.breed=breed;
        }
        void DisplayInfo() {
                super.DispalyInfo();
                System.out.println("Breed Name: "+breed);
        }

}
```

```
public class SingleInheritance {
	public static void main(String[] args) {
		Dog dg=new Dog("rocky","husky");
		dg.DisplayInfo();
	}
}
```

Lab Exercise No:3

Write a Java program to based on the multilevel inheritance in Java

MultipleInheritance.java

```
package com.oops;


class Animals {
   String name;

   Animals(String name) {
      this.name = name;
   }

   void DispalyInfo() {
      System.out.println("Animal name: " + name);
   }
}

class Mamal extends Animals {
   boolean hasfur;

   Mamal(String name, boolean hasfur) {
      super(name);
      this.hasfur = hasfur;
   }

   void display() {
      super.DispalyInfo();
      System.out.println("Has Fur: " + hasfur);
   }
}

class Dogs extends Mamal {
```

```
    String breed;

    Dogs(String name, String breed, boolean hasfur) {
        super(name, hasfur);
        this.breed = breed;
    }

    void DisplayInfo() {
        super.display();
        System.out.println("Breed Name: " + breed);
    }
}

public class MultipleInheritance {
    public static void main(String[] args) {
        Dogs dog = new Dogs("Rocky", "husky", true);
        dog.DisplayInfo();
    }
}
```

Lab Exercise No:4

Create a class named 'Member' having the following members:
Data members
1 - Name
2 - Age
3 - Phone number
4 - Address
5 - Salary
It also has a method named 'printSalary' which prints the salary of the members.
Two classes 'Employee' and 'Manager' inherits the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same.

MemberExample.java

```
package com.oops;
```

```
class Member{
	String Name;
	int age;
	String phonenumber;
	String Address;
	double Salary;

	Member(String Name,
	int age,
	String phonenumber,
	String Address,
			double Salary) {
		this.Name = Name;
		this.age = age;
		this.phonenumber = phonenumber;
		this.Address = Address;
		this.Salary = Salary;
	}
	void printsalary() {
		System.out.println("Salary of members: "+Salary);
	}

}

class Employee extends Member{
	String Specialization;
	String Department;
```

```
    Employee(String Name, int age, String phonenumber, String Address, double Salary,
String Specialization){
            super(Name,age,phonenumber,Address,Salary);
            this.Specialization=Specialization;
      }
      void displayDetails() {
    System.out.println("Employee Details:");
    System.out.println("Name: " + Name);
    System.out.println("Age: " + age);
    System.out.println("Phone Number: " + phonenumber);
    System.out.println("Address: " + Address);
    System.out.println("Specialization: " + Specialization);
    printsalary();
  }
}

class Manager extends Member {
  String Department;

  Manager(String Name, int age, String phonenumber, String Address, double Salary, String
Department) {
    super(Name, age, phonenumber, Address, Salary);
    this.Department = Department;
  }

  void displayDetails() {
    System.out.println("Manager Details:");
    System.out.println("Name: " + Name);
```

```
        System.out.println("Age: " + age);
        System.out.println("Phone Number: " + phonenumber);
        System.out.println("Address: " + Address);
        System.out.println("Department: " + Department);
        printsalary();
    }
}

public class MemberExample {
    public static void main(String[] args) {
        Employee employee = new Employee("John Doe", 30, "1234567890", "123 Main St", 50000, "Software Development");
        Manager manager = new Manager("Jane Smith", 40, "0987654321", "456 Elm St", 75000, "IT Department");

        employee.displayDetails();
        System.out.println();
        manager.displayDetails();
    }
}
```