Practical-04

1.

Main class

```
public class Practical41 {


    public static void main(String[] args) {
        Employee bi=new Employee();
        Employee bo=new Employee();


        bi.setempID(987);
        bi.setempName("Mrs.Bird");
        bi.setempDesignation("Manager");


        bo.setempID(987);
        bo.setempName("Mrs.Bond");
        bo.setempDesignation("Manager");


        System.out.println("ID= "+bi.getempID());
        System.out.println("Name= "+bi.getempName());
        System.out.println("Designation= "+bi.getempDesignation());


        System.out.println();


        System.out.println("ID= "+bo.getempID());
        System.out.println("Name= "+bo.getempName());
        System.out.println("Designation= "+bo.getempDesignation());
}
}
```

Employee class

```
public class Employee {
   private int empID;
   private String empName,empDesignation;

   public void setempID(int a){empID=a;}
   public void setempName(String b){empName=b;}
   public void setempDesignation(String c){empDesignation=c;}

   public int getempID(){return empID;}
   public String getempName(){return empName;}
   public String getempDesignation(){return empDesignation;}
   }
```

2.

SuperB class

```
class SuperB {
   int x;
   void setIt (int n) { x=n;}
   void increase () { x=x+1;}
   void triple () {x=x*3;};
   int returnIt () {return x;}
}
```

SubC class

```
class SubC extends SuperB {
   void triple () {x=x+3;} // override existing method
   void quadruple () {x=x*4;} // new method
}
```

Main class

```
public class TestInheritance {
    public static void main(String[] args) {
        SuperB b = new SuperB();
        b.setIt(2);
        b.increase();
        b.triple();
        System.out.println( b.returnIt() );
        SubC c = new SubC();
        c.setIt(2);
        c.increase();
        c.triple();
        System.out.println( c.returnIt() ); }
}
```

Output

```
9
6
```

3.

Main class

```
public class Practical43 {

    public static void main(String[] args) {
      Student s=new Student();
      s.setName("Anne");
      System.out.println("Student name= "+s.getName());
      s.setID("28224");
      System.out.println("Student ID= "+s.getID());
      s.setCourse("Java");
      System.out.println("Course= "+s.getCourse());



      Lecturer l=new Lecturer();
      l.setName("Peter");
      System.out.println("Lecturer name= "+l.getName());
      l.setID("1005");
      System.out.println("Lecturer ID= "+l.getID());
      l.setProg("Web development");
      System.out.println("Programme= "+l.getProg());

    }
}
```

Person class

```
public class Person {
    private String name,id;

    public void setName(String name)
    {
        this.name=name;
    }
    public void setID(String id)
    {
        this.id=id;
    }
    public String getID()
    {
        return id;
    }
     public String getName()
    {
        return name;
    }
}
```

Student class

```
public class Student extends Person {
   private String course;
   public void setCourse(String course)
    {
        this.course=course;
    }
    public String getCourse()
    {
        return course;
    }
}
```

Lecturer class

```
public class Lecturer extends Person{
    private String programme;

    public void setProg(String programme)
    {
        this.programme=programme;
    }
    public String getProg()
    {
        return programme;
    }
}
```

4.

In the below code,

- Animal is the superclass.
- Mammal is a subclass of Animal.
- Reptile is another subclass of Animal.
- Dog is a subclass of Mammal.

```
public class Animal{}
public class Mammal extends Animal{}
public class Reptile extends Animal{}
public class Dog extends Mammal{}
```

<u>Main class</u>

```
  public static void main(String[] args) {
    {
    Animal a = new Animal();
    Mammal m = new Mammal();
    Dog d = new Dog();
    /*This checks whether the object m of class Mammal is an instance of Animal.
    Since Mammal is a subclass of Animal, the answer is true.*/
    System.out.println(m instanceof Animal);
    /*This checks whether the object d of class Dog is an instance of Mammal.
    Since Dog is a subclass of Mammal, the answer is true.*/
    System.out.println(d instanceof Mammal);
    /*This checks whether the object d of class Dog is an instance of Animal.
    Since Dog is a subclass of Mammal, and Mammal is a subclass of Animal, the answer is also true.*/
    System.out.println(d instanceof Animal);
  }
}
}
```

Output

```
true
true
true
```