

Practical 04

1. public class TEmployee

```
{
    public static void main(String[] args)
    {
        Employee ep1 = new Employee();
        ep1.setEpID(101);    ep1.setEpName("Mr.
Bogdan");    ep1.setEpDesignation("Software
Engineer");

        Employee ep2 = new Employee();
        ep2.setEpID(102);
ep2.setEpName("Ms. Bird");
        ep2.setEpDesignation("HR Manager");

        System.out.println("Employee 1 Details:");
        System.out.println("ID: " + ep1.getEpID());
        System.out.println("Name: " + ep1.getEpName());
        System.out.println("Designation: " + ep1.getEpDesignation());

        System.out.println("\nEmployee 2 Details:");
        System.out.println("ID: " + ep2.getEpID());
        System.out.println("Name: " + ep2.getEpName());
        System.out.println("Designation: " + ep2.getEpDesignation());
    }
}
```

```
public class Employee {
    private int epID;    private
String epName;
    private String epDesignation;

    public int getEpID() {
        return epID;
    }

    public void setEpID(int epID) {
        this.epID = epID;
    }
}
```

```

    public String getEpName() {
        return epName;
    }

    public void setEpName(String empName) {
this.epName = empName;
    }

    public String getEpDesignation() {
        return epDesignation;
    }

    public void setEpDesignation(String epDesignation) {
this.epDesignation = epDesignation;
    }
}

```

```

2. class SuperB { int
    y;

    void setIt(int n) {
y = n;
    }

    void increase() {
        y = y + 1;
    }

    void triple() {
        y = y * 3;
    }

    int returnIt() {
        return y;
    }
}

class SubC extends SuperB {
    void triple() {
        y = y + 3;
    }
}

```

```

    void quadruple() {
        y= y* 4; // new method
    }
}

public class TInheritance
{
    public static void main(String[] args)
    {
        SupB b = new SupB();
        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());
    }
}

```

3. // Person class class

```

Person
{
    private String name;
    private int id;

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {

```

```
        this.id = id;
    }
}
```

```
// Student class
class Student
extends Person {
    private
    String course;
    public String
    getCourse() {
        return course;
    }

    public void setCourse(String course) {
        this.course = course;
    }
}
```

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

    public String getProgramme() {
        return programme;
    }

    public void setProgramme(String programme) {
        this.programme = programme;
    }
}
```

```
// Test Class
public
class TestPerson
{
    public static void main(String[] args)
    {
        Student s= new Student();
        S.setName("John Doe");
        s.setId(12345);
        s.setCourse("Computer Science");

        Lecturer l = new Lecturer();
        l.setName("Dr. Jane Smith");
    }
}
```

```

        l.setId(98765);
        l.setProgramme("Software Engineering");
        System.out.println("Student Details:");
        System.out.println("Name: " + s.getName());
        System.out.println("ID: " + s.getId());
        System.out.println("Course: " + s.getCourse());

        System.out.println("\nLecturer Details:");
        System.out.println("Name: " + l.getName());
        System.out.println("ID: " + l.getId());
        System.out.println("Programme: " + l.getProgramme());
    }
}

```

4. public class Animal {} public class
Mammal extends Animal {}
public class Reptile extends Animal {}

```

public class Dog extends Mammal {
    public static void main(String args[]) {
        Animal an = new Animal();
        Mammal ma= new Mammal();
        Dog do = new Dog();
        System.out.println(ma instanceof Animal);
        System.out.println(do instanceof Mammal);
        System.out.println(do instanceof Animal);
    }
}

```