Practical 04

Student ID: **27607**

Student Name: **PGWickramasuriya**

O1.

```
// Employee class class Employee
{     private int empID;     private
String empName;     private String
empDesignation;


    // Getters and setters for empID
public int getEmpID() {         return
empID;
    }


    public void setEmpID(int empID) {
this.empID = empID;
    }


    // Getters and setters for empName
public String getEmpName() {
return empName;
    }
```

```
   public void setEmpName(String empName) {        this.empName = empName;
   }

   // Getters and setters for empDesignation
public String getEmpDesignation() {
return empDesignation;
   }

   public void setEmpDesignation(String empDesignation) {
this.empDesignation = empDesignation;
   }
}

// Test class to invoke the Employee class
public class EmployeeTest {     public
static void main(String[] args) {        //
Creating two Employee objects
      Employee mrBogdan = new Employee();
      Employee msBird = new Employee();

      // Setting values for Mr. Janas using setters
mrJanas.setEmpID(101);        mrJanas.setEmpName("Mr.
Janas");        mrJanas.setEmpDesignation("Software
Engineer");
```

```
        // Setting values for Ms. Suni using setters
msSuni.setEmpID(102);         msSuni.setEmpName("Ms. Suni");
msSuni.setEmpDesignation("Data Scientist");


        // Printing the details of Mr.Janas using getters
        System.out.println("Employee ID: " + mrJans.getEmpID());
        System.out.println("Employee Name: " + mrJanas.getEmpName());
        System.out.println("Employee Designation: " + mrJanas.getEmpDesignation());


        // Printing the details of Ms. Suni using getters
        System.out.println("\nEmployee ID: " + msSuni.getEmpID());
        System.out.println("Employee Name: " + msSuni.getEmpName());
        System.out.println("Employee Designation: " + msSuni.getEmpDesignation());
    }
}
```

03.

Person class (Superclass):

```
class Person {     private
String name;

    private int id;
    // Constructor for Person     public
Person(String name, int id) {
this.name = name;

        this.id = id;
    }

    // Getters and setters for name and 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 Student extends Person {

private String course;



    // Constructor for Student     public Student(String

name, int id, String course) {         super(name, id);

this.course = course;

    }

    // Getter and setter for course

public String getCourse() {

return course;

    }


    public void setCourse(String course) {

this.course = course;

    }

}
```

Lecturer

```
class Lecturer extends Person {

private String programme;
```

```
   // Constructor for Lecturer     public Lecturer(String
name, int id, String programme) {        super(name, id);

      this.programme = programme;

   }


   // Getter and setter for programme
public String getProgramme() {
return programme;

   }


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

Student and Lecturer classes by creating one object of each:

```
public class TestPerson {     public static
void main(String[] args) {

      // Creating a Student object

      Student student = new Student("Sara", 101, "Computer Science");

      System.out.println("Student Name: " + student.getName());

      System.out.println("Student ID: " + student.getID());

      System.out.println("Student Course: " + student.getCourse());


      // Creating a Lecturer object
```

```
        Lecturer lecturer = new Lecturer("Janith ", 201, "Mathematics");

        System.out.println("\nLecturer Name: " + lecturer.getName());

        System.out.println("Lecturer ID: " + lecturer.getID());

        System.out.println("Lecturer Programme: " + lecturer.getProgramme());

    }

}
```