

**Don Bosco Institute of
Technology,
Kurla(W)Department of Electroni
cs and Tele-**

Lab Number:	8
Student Name:	Sakshi Morey
Roll No :	34

Title:

1. To perform Multilevel Inheritance in JAVA. Create a Person class representing name, age and address. Inherit person class to employee class with emp ID and salary factor. Inherit the Employee class to programmer class with technical skills and hike attributes. Implement valid methods to input the details from the user in the main method and display for 3 programmers.
2. To perform Hierarchical Inheritance in JAVA. Create an Employee class with attributes EmpID and EmpSalary. Also create necessary methods/constructors to accept these values from the user. Create classes permanent Employee and Temporary Employee which will be derived classes of Employee. Mention hike attribute in these derived classes and calculate the total salary using generate_salary() method for respective types of employees. Objects of the derived classes should be created and salaries for the permanent and temporary employees should be calculated and displayed on the screen.

Learning Objective:

- Students will be able to perform multilevel inheritance using JAVA.
- Students will be able to perform hierarchical inheritance using JAVA

Learning Outcome:

- To understand how to use the private members using friend function and friend class.

Course Outcome:

ECL304.2	Comprehend building blocks of OOPs language, inheritance, package and interfaces.
-----------------	---

Theory:

- Explain in details about various inheritance types supported in JAVA.
Inheritance is a mechanism of deriving a new class from an existing class. The existing (old) class is known as base class or super class or parent class. The new class is known as a derived class or sub class or child class. The extends keyword indicates that you are making a new class that derives from an existing class. The meaning of "extends" is to increase the functionality.

**Don Bosco Institute of
Technology,
Kurla(W)Department of Electroni
cs and Tele-**

1. single inheritance: A sub-class is derived from only one super class. It inherits the properties and behavior of a single-parent class.
2. multi-level inheritance: A class is derived from a class which is also derived from another class is called multi-level inheritance. In simple words, we can say that a class that has more than one parent class is called multi-level inheritance. The classes must be at different levels. Hence, there exists a single base class and single derived class but multiple intermediate base classes.
3. hierarchical inheritance: If a number of classes are derived from a single base class, it is called hierarchical inheritance.
4. hybrid inheritance: It consists of more than one. Hybrid inheritance is the combination of two or more types of inheritance.
5. Java does not support multiple inheritance due to ambiguity.

Algorithm:	<p>STEP1: Start</p> <p>STEP2: Create class Person</p> <p>STEP3: Define attributes and method display() and getDetails() STEP</p> <p>P4: Create child class Employee</p> <p>STEP 5: Define attributes salary EmpID and methods display() & getDetails() STEP6: Create another child class Programmer</p> <p>STEP 7: Define attributes hike, skills and methods display() & getDetails() STEP8: In main class, create 3 objects for 3 programmers</p> <p>STEP9: Display output</p> <p>STEP10: Stop</p>
Program:	<pre> package inheritance; import java.util.Scanner; class Person{ Scanner in = new Scanner(System.in); String name; String address; int age; Person() { name = ""; address = ""; age = 0; } void display() { System.out.println("Name: "+name); System.out.println("Age: "+age); } } </pre>

```

        System.out.println("Address:"+address);
    }
    void getDetails()
    {
        System.out.println("Enter name :
        "); name =
        in.nextLine(); System.out.println("Enter
        address : "); address =
        in.nextLine(); System.out.println("Enter
        age:"); age = in.nextInt();
    }
}
class Employee extends
    Person{int empID; double
    salary; Employee()
    {
        empID= 0;
        salary=0.0;
    }
    void getDetails()
    {
        super.getDetails(); System.out.println("Ente
        r Employee ID : "); empID =
        in.nextInt(); System.out.println("Enter base
        salary : "); salary= in.nextDouble();
    }
    void display()
    {
        super.display(); System.out.println("Employee
        eID:"+empID);
        System.out.println("BaseSalary:Rs."+salary);
    }
}
public class Programmer extends Employee
{
    double hike; String
    skills; Programmer()
    {
        hike =
        0; skills="";
    }
    void getDetails()
    {
        super.getDetails(); System.out.println("Ente
        r salary hike : "); hike= in.nextDouble();
        System.out.println("Enter technical skills :
        "); in.nextLine();
        skills=in.nextLine();
    }
    void display()
    {
        super.display();
    }
}

```

**Don Bosco Institute of
Technology,
Kurla(W)DepartmentofElectroni
csandTele-**

	<pre> System.out.println("SalaryHike:Rs."+hike); System.out.println("Totalsalary:Rs."+(salary+hike)); System.out.println("Technicalskills:"+skills); } publicstaticvoidmain(Stringargs[]) { System.out.println("Enterdetailsfor1stprogrammer"); Programmerobj1=newProgrammer(); obj1.getDetails(); System.out.println("Enterdetailsfor2ndprogrammer"); Programmerobj2=newProgrammer(); obj2.getDetails(); ; System.out.println("Enterdetailsfor3rdprogrammer"); Programmerobj3=newProgrammer(); obj3.getDetails(); System.out.println("\nDetailsof1stprogrammer"); obj1.display(); System.out.println("\nDetailsof2ndprogrammer"); obj2.display(); System.out.println("\nDetailsof3rdprogrammer"); obj3.display(); } } </pre>
Inputgiven:	<pre> Enter details for 1st programmerEnter name: Raks Enter address: abc Enter age: 12 Enter EmployeeID: 11 Enter basesalary: 1000 Enter salaryhike: 12 Entertechnicalskills: none Enterdetailsfor2ndprogrammer Enter name: ram Enter address: abcd Enter age: 23 Enter EmployeeID: 12 Enter basesalary: 1000 Enter salaryhike: </pre>

**Don Bosco Institute of
Technology,
Kurla(W)Department of Electroni
cs and Tele-**

	<pre> 12 Enter technical skills :none Enter details for 3rd programmerEnter name: raj Enter address: asbd Enter age: 33 Enter Employee ID :13 Enter base salary :1000 Enter salary hike :12 Enter technical skills :none </pre>
OutputScreenshot:	<pre> Details of 1st programmer Name : Raks Age : 12 Address : abc Employee ID : 11 Base Salary : Rs.1000.0 Salary Hike : Rs.12.0 Total salary : Rs.1012.0 Technical skills : none Details of 2nd programmer Name : ram Age : 23 Address : abcd Employee ID : 12 Base Salary : Rs.1000.0 Salary Hike : Rs.12.0 Total salary : Rs.1012.0 Technical skills : none Details of 3rd programmer Name : raj Age : 33 Address : asbd Employee ID : 13 Base Salary : Rs.1000.0 Salary Hike : Rs.12.0 Total salary : Rs.1012.0 Technical skills : none </pre>

2.

Algorithm :	STEP1:Start STEP2:createclass employee1,defineattributesandmethodssetdetails() STEP3: createchild classesPermanentEmpand TemporaryEmp STEP4:defineattributesandmethodgeneratesalary() inboththeclasses STEP5:Create main function STEP 6:Give the user 2 choices of permanent or temporary employee STEP 7:create object in main function according to the case selected STEP 8: print the output STEP9: Stop
Program:	<pre> import java.util.Scanner; class employee1 { int EmpID; float salary; void setdetails() { Scanner t = new Scanner(System.in); System.out.println("Enter your ID ="); EmpID = t.nextInt(); System.out.println("Enter your Salary ="); salary = t.nextFloat(); } } class PermanentEmp extends employee1{ double hike = 0.5; void generatesalary() { System.out.println("Salary of permanent employee is Rs." + (salary + (salary * hike))); } } class TemporaryEmp extends employee1{ double hike = 0.2; void generatesalary() { System.out.println("Salary of temporary employee is Rs." + (salary + (salary * hike))); } } public class employee { </pre>

**Don Bosco Institute of
Technology,
Kurla(W)DepartmentofElectroni
csandTele-**

	<pre> public static void main(String args[]) { Scanner in = new Scanner(System.in); System.out.println("Enter 1 for Permanent Employee and 2 for Temporary Employee"); int choice = in.nextInt(); switch(choice) { case 1: PermanentEmp p = new PermanentEmp(); p.setdetails(); p.generateSalary(); break; case 2: TemporaryEmp t = new TemporaryEmp(); t.setdetails(); t.generateSalary(); break; default: System.out.println("Invalid choice"); } } } </pre>
Input given:	<p>Enter 1 for Permanent Employee and 2 for Temporary Employee</p> <p>1</p> <p>Enter your ID =</p> <p>100</p> <p>Enter your Salary =</p> <p>50000</p>
Output Screenshot:	 <pre> Enter 1 for Permanent Employee and 2 for Temporary Employee 1 Enter your ID = 100 Enter your Salary = 50000 Salary of permanent employee is Rs.75000.0 </pre>