|  |  |  |
| --- | --- | --- |
| **Ex. No. 5** | **Inheritance** | |
| Date of Exercise | 15.09.2020 |  |

# General Aim of the Exercise:

# To implement the given programs using Inheritance and polymorphism.

**Description:**

**Multilevel Inheritance:**

When there is a chain of inheritance, it is known as multilevel inheritance.

**Polymorphism:**

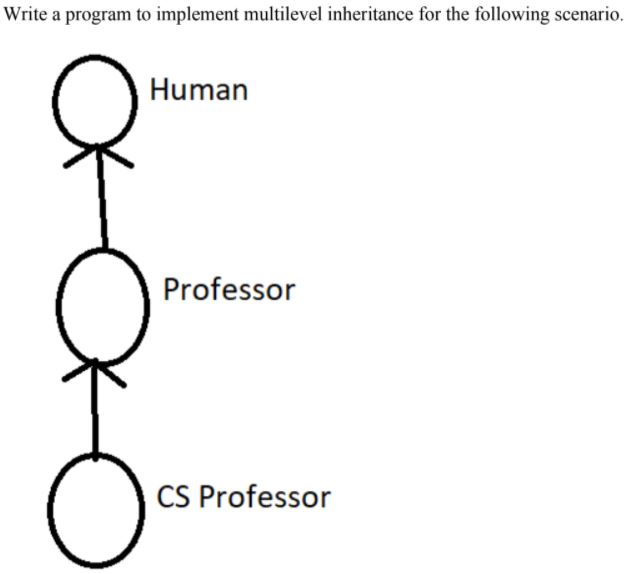
Polymorphism is the ability of an object to take on many forms. The most common use of polymorphism in OOP occurs when a parent class reference is used to refer to a child class object.

**Dynamic Method Dispatching**

Method overriding is one of the ways in which Java supports Runtime Polymorphism. Dynamic method dispatch is the mechanism by which a call to an overridden method is resolved at run time, rather than compile time.

* When an overridden method is called through a superclass reference, Java determines which version(superclass/subclasses) of that method is to be executed based upon the type of the object being referred to at the time the call occurs. Thus, this determination is made at run time.
* At run-time, it depends on the type of the object being referred to (not the type of the reference variable) that determines which version of an overridden method will be executed
* A superclass reference variable can refer to a subclass object. This is also known as upcasting. Java uses this fact to resolve calls to overridden methods at run time.

**P rogram 1:**



**Aim**

To implement multilevel inheritance for the given scenario

# Youtube Link: https://www.youtube.com/watch?v=ZCEClvXIQfg&feature=youtu.be

# Algorithm:

# 1.Start the program.

# 2.Declare 3 classes Human , Professor and CSProfessor .Create 2 constructors for each classes.One with parameters and other with no parameters.

# 3.Make the subclasses Professor and CSProfessor inherit the properties of base class Human by using super() and passing input variables as parameters.

# 4.Create an object for the subclasses and display it.Display the output.

# 5.Stop the program

# Source Code:

package com.Multilevel;  
class Human  
{  
 Human()  
 {  
 System.*out*.println(**"I am a human"**);  
 }  
 Human(String x)  
 {  
 System.*out*.println(**"I have a "**+x);  
 }  
}  
  
class Professor extends Human  
{  
 Professor()  
 {  
 System.*out*.println(**"I am an educated human"**);  
 }  
  
 Professor(String x,String y)  
 {  
 super(x);  
 System.*out*.println(**"I have a "**+y);  
 }  
}  
  
class CS\_Professor extends Professor  
{  
 CS\_Professor()  
 {  
 System.*out*.println(**"CS Professor:I am a educated human with CS degree"**);  
 }  
  
 CS\_Professor(String x,String y,String z)  
 {  
 super(x,y);  
 System.*out*.println(**"I have a "**+z);  
 //System.out.println("Totally I have a"+x+","+y+"and"+z);  
 }  
}  
public class Inheritance\_Multilevel {  
  
 public static void main(String[] args) {  
 Professor Joy = new Professor();  
 CS\_Professor Happy = new CS\_Professor(**"heart"** , **"soul"**, **"brain"**);  
 }  
}

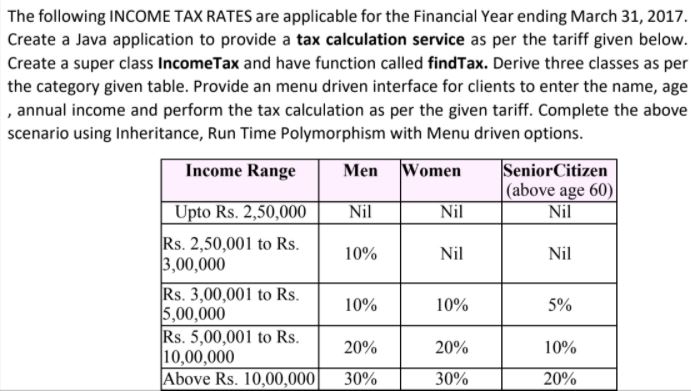
# Output:

# 

**Result:**

Thus the program to implement multilevel inheritance for the given scenario was executed and verified successfully.

**P rogram 2:**



**Aim:** To calculate Income tax using the given data in java programing (Polymorphism concept)

.

# Youtube Link: https://www.youtube.com/watch?v=ZCEClvXIQfg&feature=youtu.be

# Algorithm:

# 1.Start the program.

# 2.Declare 4 classes IncomeTax ,Men ,Women ,SeniorCitizen and create constructors for each of them.

# 3.Declare required data members and input to be got from the user in IncomeTax() .

# 4.Declare another function findTax() under void type and use the overriding method for the same in all the classes.

# 5.Make the subclasses inherit the property of the baseclass IncomeTax by using super() and passing input parameters in it.

# 6.Using if statement check whether the income obtained from the user is within the given range.Based on several range as per the given data , calculate the income tax.

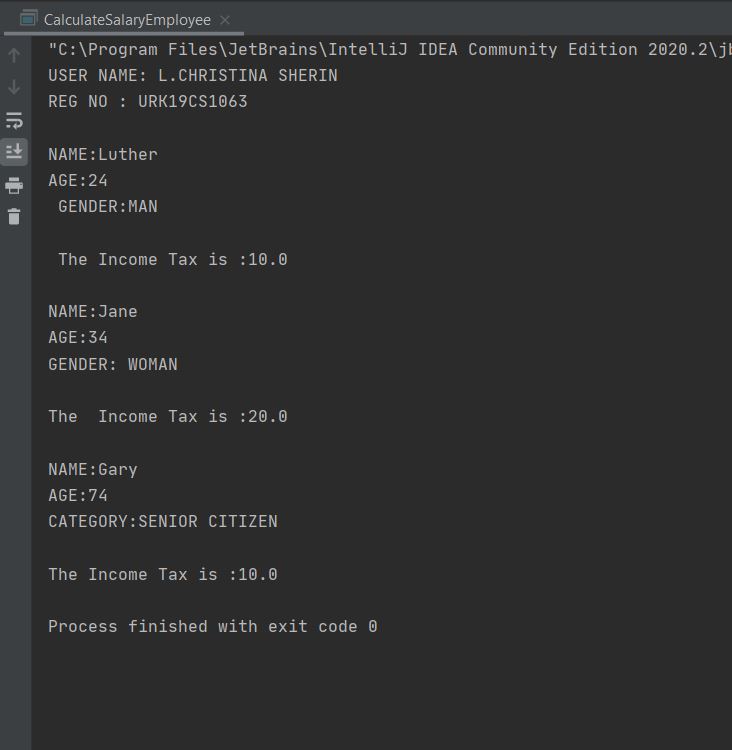
# 7.Declare a public class and introduce objects for the base classes.Using objects call the findTax() function.Pass the input in it and display the result.

# 8.Stop the program.

# Source Code:

package com.dyno;  
import java.util.Scanner;  
  
class IncomeTax{  
 int age;  
 double income;  
 String name;  
 IncomeTax() {  
  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.println(**"ENTER NAME: "**);  
 name = sc.next();  
 System.*out*.println(**"ENTER AGE: "**);  
 age = sc.nextInt();  
 System.*out*.println(**"ENTER ANNUAL\_INCOME: "**);  
 income = sc.nextDouble();  
 }  
  
 IncomeTax(String x,int y,double z){  
 name=x;  
 age=y;  
 income=z;  
 }  
  
 void findTax(){  
  
 }  
}  
  
class Men extends IncomeTax{  
 Men(String x,int y,double z)  
 {  
 super(x,y,z);  
 }  
   
 void findTax()  
 {  
 if(income<250000)  
 income=0;  
 if(income >= 250001 && income<=300000)  
 income=10%income;  
 if (income >= 300001 && income <=500000)  
 income=10%income;  
 if(income>=500001 && income<=1000000)  
 income=20%income;  
 if(income>=1000000)  
 income=30%income;  
 System.*out*.println(**"**\n**NAME:"**+name+**"**\n**AGE:"**+age+**"**\n **GENDER:MAN"**);  
 System.*out*.println(**"**\n **The Income Tax is :"**+income);  
 }  
  
 }  
  
  
  
class Women extends IncomeTax{  
 Women(String x,int y,double z)  
 {  
 super(x,y,z);  
 }  
  
 void findTax()  
 {  
  
 if(income<=300000)  
 income=0;  
 if (income >= 300001 && income <=500000)  
 income=10%income;  
 if(income>=500001 && income<=1000000)  
 income=20%income;  
 if(income>=1000000)  
 income=30%income;  
 System.*out*.println(**"**\n**NAME:"**+name+**"**\n**AGE:"**+age+**"** \n**GENDER: WOMAN"**);  
 System.*out*.println(**"**\n**The Income Tax is :"**+income);  
 }  
}  
  
  
class SeniorCitizen extends IncomeTax {  
  
 SeniorCitizen(String x,int y,double z)  
 {  
 super(x,y,z);  
 }  
  
 void findTax()  
 {  
 if(income<300000)  
 income=0;  
 if (income >= 300001 && income <=500000)  
 income=5%income;  
 if(income>=500001 && income<=1000000)  
 income=10%income;  
 if(income>=1000000)  
 income=20%income;  
 System.*out*.println(**"**\n**NAME:"**+name+**"**\n**AGE:"**+age+**"** \n**CATEGORY:SENIOR CITIZEN"**);  
 System.*out*.println(**"**\n**The Income Tax is :"**+income);  
 }  
 }  
public class CalculateSalaryEmployee{  
 public static void main(String[] args)  
 {  
 System.*out*.println(**"USER NAME: L.CHRISTINA SHERIN"**);  
 System.*out*.println(**"REG NO : URK19CS1063"**);  
 IncomeTax obj;  
 obj = new Men(**"Luther"**,24,300002.00);  
 obj.findTax();  
 obj = new Women(**"Jane"**,34,500002.00);  
 obj.findTax();  
 obj = new SeniorCitizen(**"Gary"**,74,600002.00);  
 obj.findTax();  
  
 }  
 }

# Output:

****

**Result:**

Thus the program to calculate income tax using the given data was executed and verified successfully.