## **PRACTICAL 2:**

## **AIM:** Programs for polymorphism and Inheritance

**Program 1:**

Create a class OverLoadDemo and also create method test in it. Overload test in four ways. First version takes no parameter, the second takes one integer parameter, and the third takes two integer parameters and fourth takes one double parameter. Call all methods from main() function.

Source Code:

package lab2;

public class OverLoadDemo

{

int x,y,z;

double e;

public void test()

{

x=10;

System.out.println(x);

}

public void test(int a)

{

x=a;

System.out.println(x);

}

public void test(int b,int c)

{

y=b;

z=c;

System.out.println(y);

System.out.println(z);

}

public void test(double d) {

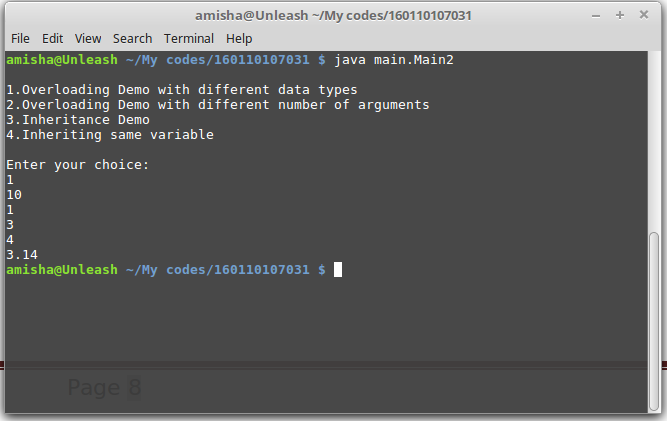
e=d;

System.out.println(e);

}

}

Output:



**Program 2:**

Create class ‘Point3D’. There are three constructors in a Class ‘Point3D’.The first form accepts one double argument. The second form accepts two double arguments. The third form accepts three double arguments. These are used to initialize all of the instance variables. Also prints instance variables. Demonstrate use of overloaded constructors in main function.

Source Code:

package lab2;

public class Point3D

{

double x,y,z,d,e,f;

public Point3D()

{

}

public Point3D(double x)

{

this.x=x;

System.out.println("x="+x);

}

public Point3D(double y,double z)

{

this.y=y;

this.z=z;

System.out.println("y="+y+" z="+z);

}

public Point3D(double d,double e,double f)

{

this.d=d;

this.e=e;

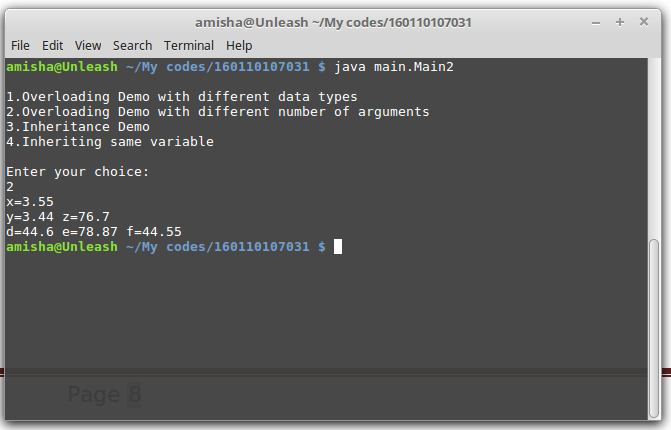
this.f=f;

System.out.println("d="+d +" e="+e +" f="+f);

}

}

Output:



**Program 3:**

Class M has two instance variables of type float and string. Class N extends M and has one Instance variable of type Double. Instantiate class N and Initialize and display its member’s value.

Source code:

>> Class M:

package lab2;

public class M

{

float a;

String b;

public M(float c,String d)

{

a=c;

b=d;

}

public float Initialize1()

{

return a;

}

public String Initialize2()

{

return b;

}

}

>> Class N inherited from M:

package lab2;

public class N extends M

{

double e;

public N(double f)

{

super(4.15f,"Anmol");

e=f;

}

public void display()

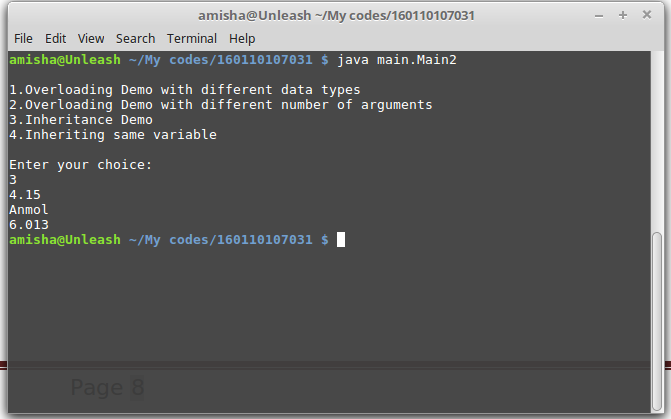
{

System.out.println(Initialize1()+"\n"+Initialize2()+"\n"+e);

}

}

Output:



**Program 4:**

Write an application that illustrates variable hiding. Class S declares an instance variable named X of type integer. Class T extends S and Declare an instance variable named X of type String. Instantiate both these classes. Initialize and display the variable named X in each of these objects.

Source code:

>>Class S:

package lab2;

public class S

{

int X;

public S(int x)

{

X=x;

System.out.println("S called");

}

}

>>Class T inherited from S:

package lab2;

public class T extends S

{

String X;

public T(String p,int x)

{

super(x);

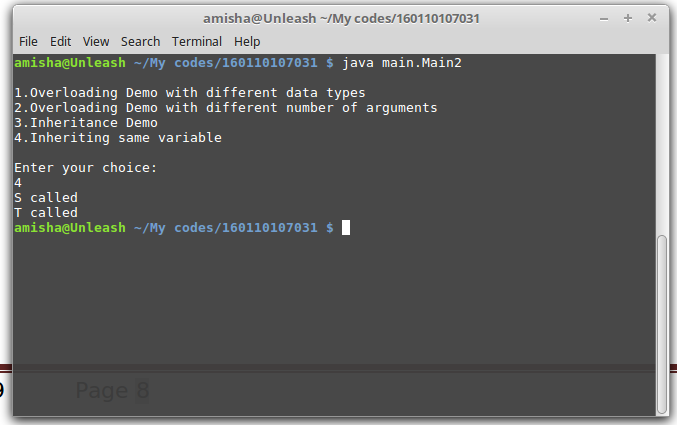
X=p;

System.out.println("T called");

}

}

Output:



Main class to run all programs according to choice:

package main;

import lab2.\*;

import test.\*;

import java.util.Scanner;

public class Main2

{

public static void Overload()

{

OverLoadDemo D = new OverLoadDemo();

D.test();

D.test(1);

D.test(3,4);

D.test(3.14);

}

public static void point3D()

{

Point3D ob =new Point3D(3.55);

Point3D ob1= new Point3D(3.44,76.7);

Point3D ob2= new Point3D(44.6,78.87,44.55);

}

public static void inherit()

{

N ob1=new N(6.013);

ob1.display();

}

public static void samevar()

{

T t=new T("hello",4);

}

public static void main(String a[])

{

int ch;

Scanner ob=new Scanner(System.in);

System.out.println("\n1.Overloading Demo with different data types\n2.Overloading Demo with different number of arguments\n3.Inheritance Demo\n4.Inheriting same variable\n");

System.out.println("Enter your choice: ");

ch = ob.nextInt();

switch(ch)

{

case 1:Overload();

break;

case 2:point3D();

break;

case 3:inherit();

break;

case 4:samevar();

break;

default:

System.out.println("enter a valid choice");

}

}

}