**Assignment 6**

**Problem:**

Create a class Shape having atmost two dimensions. Define two

subclasses circle and rectangle of Shape.

    i) Override a method area.

    ii)Show compile time and run time polymorphism(Dynamic  method dispatch).

    iii)Use a final method for display.

    iv) use super keyword.

**Solution :**

Overriding method area

Package tree:

Pack -----------> shape.java(super class)

|

|-------> p2 -----------> circle.java(sub class)

|

|------->rectangle.java(sub class)

|

|----->main.java

**Code ::**

**Shape.java**

package pack;

public class shape

{

int dim1=0,dim2=0;

public final float pival()

{

float pi = (float)3.14;

return pi;

}

public void dispra(int a,int b)

{

dim1=0;

dim2=0;

System.out.println("Dimentions are\nDim1 : "+dim1+"\nDim2 : "+dim2);

}

public void dispca(int a)

{

dim1=a;

dim2=0;

System.out.println("Dimensions are\nDim1 : "+dim1+"\nDim2 : "+dim2);

}

}

**Circle.java**

package pack.p2;

import pack.\*;

class circle extends shape

{

float p=pival();

public void dispca(int a)

{

System.out.println("Area of the circle with radious"+a+" is:"+(p\*a\*a));

}

}

**Rectangle.java**

package pack.p2;

import pack.\*;

class rectangle extends shape

{

public void dispra(int a,int b)

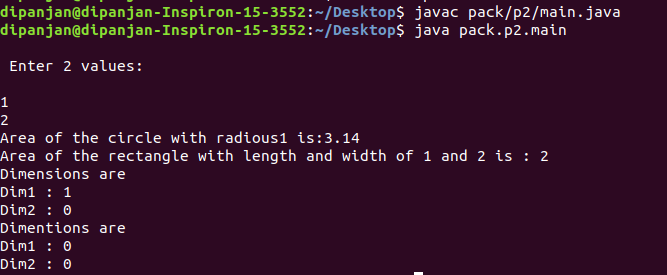
{

System.out.println("Area of the rectangle with length and width of "+a+" and "+b+" is : "+(a\*b));

}

}

**Overriding Output**



**Compile time polymorphism ---> Method overloading**

**Run time Polymorphism ---> Method overriding**

**Final method**

Codes

circle.java

package pack.p2;

import pack.\*;

class circle extends shape

{

float pival()

{

float pi = (float)3.00;

return pi;

}

float p=pival();

public void dispca(int a)

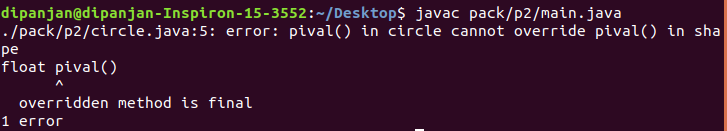
{

System.out.println("Area of the circle with radious"+a+" is:"+(p\*a\*a));

}

}

**OUTPUT**



**Usage of super keyword**

**Code**

circle.java

package pack.p2;

import pack.\*;

class circle extends shape

{

float p=pival();

public void dispca(int a)

{

super.dispca(a);

}}

**OUTPUT**

