



# V.T Patel Department of Electronics & CommunicationEngineering

#### Part-VI Inheritance

#### **Practical No: 25**

**AIM:** Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get\_data() to initialize base class data members and another member function display\_area() to compute and display the area of figures.

#### **Solution**

#### Practical25.java

```
package com.jayshil.javaapp;
class Shape{
  double dimension 1 = 15.5, dimension 2 = 12.2;
  void get data() {
  void display area(){
  }
class Triangle extends Shape{
  @Override
  void display area() {
        double area = 0.5*dimension 1*dimension 2;
        System.out.println("Area of triangle is : "+area);
     }
  }
class Rectangle extends Shape{
  @Override
  void display area() {
     double area = dimension 1*dimension 2;
     System.out.println("Area of rectangle is : "+area);
  }
```





# V.T Patel Department of Electronics & CommunicationEngineering

```
public class Practical25 {
   public static void main(String[] args) {
     Rectangle obj_0 = new Rectangle();
     obj_0.display_area();
     Triangle obj_1 = new Triangle();
     obj_1.display_area();
}
}
```

#### Output

2



# CSpit Chandubhai S Patel Institute of Technology

# V.T Patel Department of Electronics & CommunicationEngineering

**Practical No: 26** 

**AIM:** Override the method 'area()', in the class 'Figure' for the following geometric shapes triangle and square.

#### Solution

Practical26.java

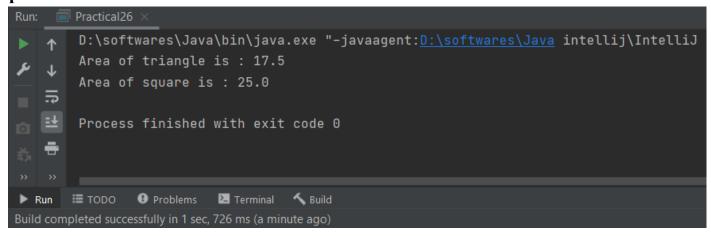
```
package com.jayshil.javaapp;
class Figure{
 double dimension 1 = 5.0, dimension 2 = 7.0;
 void area()
     System.out.println("No object created");
class triangle extends Figure{
 double area = 0.5*dimension 1*dimension 2;
 @Override
 void area() {
     System.out.println("Area of triangle is: "+area);
class square extends Figure{
 double area = dimension 1*dimension 1;
 @Override
 void area() {
     System.out.println("Area of square is : "+area);
public class Practical26 {
 public static void main(String[] args) {
     triangle triobj = new triangle();
     triobj.area();
     square sqobj = new square();
     sqobj.area();
```





# V.T Patel Department of Electronics & CommunicationEngineering

### Output





# V.T Patel Department of Electronics & CommunicationEngineering



**Practical No: 27** 

**AIM:** Use the concept of abstract classes, where the method area() is declared as abstract in the class 'Figure'.

#### Solution

Practical27.java

```
package com.jayshil.javaapp;
abstract class Figures{
   abstract void area(double dimension_1);
}

class subclass extends Figures{
   @Override
   void area(double dimension_1) {
       double area = dimension_1*dimension_1;
       System.out.println("The area of Square is : "+area);
   }
}

public class Practical27 {
   public static void main(String[] args) {
       subclass subobj = new subclass();
       subobj.area(4.0);
   }
}
```

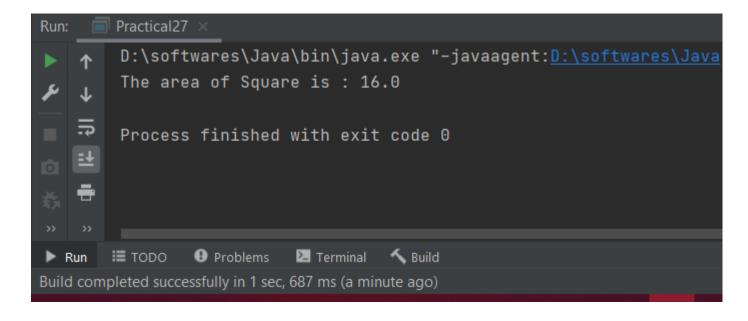




# V.T Patel Department of Electronics & CommunicationEngineering

#### Output

6





# Cspit Chandubhai S Patel Institute of Technology

# V.T Patel Department of Electronics & CommunicationEngineering

**Practical No: 28** 

**AIM:** Create a program with an abstract class named 'Test'. Which contains two methods named 'callme()' and 'callmetoo()'. Declare the method 'callme' as abstract and let the methods 'callmetoo()' have a body that is a simple message. Derive a subclass 'Test2' from the abstract class 'Test' which implements the methods 'callme()'. Let the object of the subclass 'Test2' calls the two methods. Analyse which version of the method 'callme()' is executed.

#### **Solution**

Practical4.java

```
package com.jayshil.javaapp;
abstract class Test{
   abstract void callme();
   void callmetoo(){
      System.out.println("Callmetoo from Test class");
   }
}
class Test2 extends Test{
   @Override
   void callme() {
      System.out.println("Callme from Test2 class");
   }
}
public class Practical28 {
   public static void main(String[] args) {
      Test2 obj1 = new Test2();
      obj1.callme();
      obj1.callmetoo();
   }
}
```





# V.T Patel Department of Electronics & CommunicationEngineering

### Output

8





# V.T Patel Department of Electronics & CommunicationEngineering



**Practical No: 29** 

**AIM:** Write a java program which explains the concept of super keyword.

#### **Solution**

Practica29.java

```
package com.jayshil.javaapp;

class Animals{
   String name = "This is a dog";
}

class pet extends Animals{
   String name = "This is a cat";
   void type_of_pet()
   {
      System.out.println("Without super keyword : "+name);
      System.out.println("With super keyword : "+super.name);
   }
}

public class Practical29 {
   public static void main(String[] args) {
      pet obj = new pet();
      obj.type_of_pet();
   }
}
```

#### Output





# V.T Patel Department of Electronics & CommunicationEngineering

**Practical No: 30** 

**AIM:** Write a program to do the find out the average of 4 different variables which has been declared and initialized in separate class. Using multilevel inheritance.

#### **Solution**

#### Practica30.java

```
package com.jayshil.javaapp;
class firstclass{
 int first var = 4;
class secondclass extends firstclass{
 int second var = 8;
class third class extends secondclass{
 int third var = 2;
class fourth class extends third class{
 int fourth var = 10;
public class Practical30 {
  public static void main(String[] args) {
     fourth class obj = new fourth class();
    int average = (obj.first var+obj.second var+
obj.third var+obj.fourth var) \sqrt{4};
    System.out.println("The average of numbers is:
"+average);
```





# V.T Patel Department of Electronics & CommunicationEngineering

## Output

11



18EC068 [Jayshil Patel] [Java Programing]