

Create a class named 'Shape' with a method to print "This is This is shape". Then create two other classes named 'Rectangle', 'Circle' inheriting the Shape class, both having a method to print "This is rectangular shape" and "This is circular shape" respectively. Create a subclass 'Square' of 'Rectangle' having a method to print "Square is a rectangle". Now call the method of 'Shape' and 'Rectangle' class by the object of 'Square' class.

```
class Shape{

    public void print_shape(){

        System.out.println("This is shape");

    }

}

class Rectangle extends Shape{

    public void print_rect(){

        System.out.println("This is rectangular shape");

    }

}

class Circle extends Shape{

    public void print_circle(){

        System.out.println("This is circular shape");

    }

}

class Square extends Rectangle{

    public void print_square(){

        System.out.println("Square is a rectangle");

    }

}

public class Test{

    public static void main(String[] args){
```

```
Square sq = new Square();
```

```
sq.print_shape();
```

```
sq.print_rect();
```

```
}
```

```
}
```

Inbox (67) - priyankakilledar99@ Online Java Compiler - Online Java Compiler -

jdoodle.com/online-java-compiler/

Sponsored: DigitalOcean Run your scripts on a scalable compute platform with no surprise bills. [Get \\$100 Free Credit](#)

```
1 class Shape{
2     public void print_shape(){
3         System.out.println("This is shape");
4     }
5 }
6 class Rectangle extends Shape{
7     public void print_rect(){
8         System.out.println("This is rectangular shape");
9     }
10 }
11 class Circle extends Shape{
12     public void print_circle(){
13         System.out.println("This is circular shape");
14     }
15 }
16 class Square extends Rectangle{
17     public void print_square(){
18         System.out.println("Square is a rectangle");
19     }
20 }
21 public class Test{
22     public static void main(String[] args){
23         Square sq = new Square();
24         sq.print_shape();
25         sq.print_rect();
26     }
27 }
28
29
30
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

☐ Interactive

Stdin Inputs

CommandLine Arguments

[Execute](#) [...](#) [Download](#)

Result

CPU Time: 0.16 sec(s), Memory: 32000 kilobyte(s) compiled and executed in 0.808 sec(s)

```
This is shape
This is rectangular shape
```

Type here to search

ENG 11:13 AM 6/24/2020