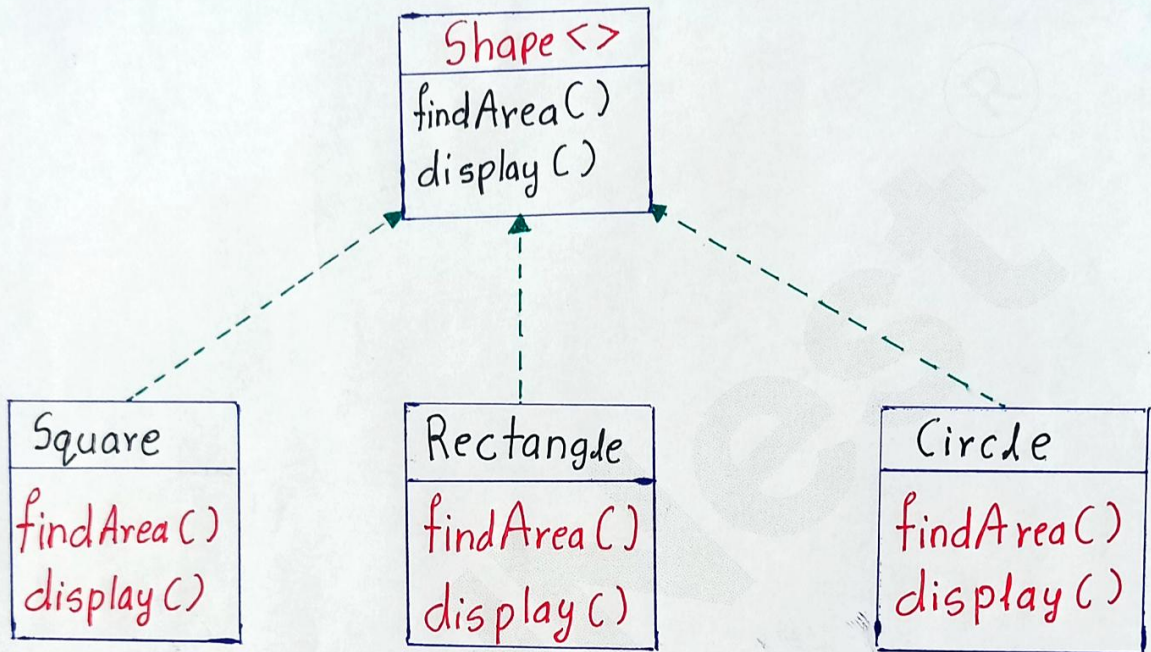


# Achieving Abstraction Using Interfaces.



```
public interface Shape
{
    void findArea();
    void display();
}
```



```
public class Square implements Shape {
```

```
    public void findArea()  
{
```

```
        int e=10;
```

```
        int ar= e * e;
```

```
        System.out.println("Area of square="+ar);
```

```
    }
```

```
    public void display()  
{
```

```
        System.out.println("This shape is a square");
```

```
    }
```

```
}
```

```
public class Rectangle implements Shape {
```

```
    public void findArea()  
{
```

```
        int l= 10;
```

```
        int b= 5;
```

```
        int ar= l * b;
```

```
        System.out.println("Area of rectangle="+ar);
```

```
    }
```





```
public void display()
{
    System.out.println("This shape is a rectangle");
}
```

```
public class Circle implements Shape {
    public void findArea()
    {
        int r = 10;
        double ar = 3.14 * r * r;
        System.out.println("Area of circle = " + ar);
    }

    public void display()
    {
        System.out.println("This shape is a circle");
    }
}
```



```
public class ShapeApp {
```

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

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

```
        Rectangle re = new Rectangle();
```

```
        Circle ci = new Circle();
```

```
        fun(sq);
```

```
        fun(re);
```

```
        fun(ci);
    }
```

```
    public static void fun (Shape s) {
```

```
        s.findArea();
```

```
        s.display();
    }
```

```
}
```





## Output

Area of square = 100

This shape is a square

Area of rectangle = 50

This shape is a rectangle

Area of circle = 314.0

This shape is a circle

For achieving abstraction in java we can go for abstract classes as well as interfaces both are going to promote polymorphism.

