

封装为java三大特性之一。顾名思义，就是将需要的变量以及方法封装起来，外界无法直接操作和修改，只能用public修饰符指定的变量及方法来访问。增强其代码的安全性。

封装

- 使用private修饰符来修饰成员变量或者方法。
- 使用public修饰符来修饰成员变量或方法来提供共有接口，以供访问其private的私有数据。
- 使用this来指定其实例变量，将实例变量与参数设为同名。
- 使用构造方法来初始化private私有数据，或者通过public修饰的变量及方法来初始化private私有数据。

Cloth.java

```
1 package com.test;
2 public class Cloth {
3     private int size;
4     private String color;
5
6     public Cloth(){}
7     public Cloth(int size,String color) { //利用构造方法来初始化变量
8         this.size = size; //方法被哪个对象调用，this就指向那个对象。这里是实例变量size.
9         this.color = color;
10    }
11    public void setSize(int size) {
12        this.size = size;
13    }
14    public int getSize() {
15        return size;
16    }
17    public void setColor(String color) {
18        this.color = color;
19    }
20    public String getColor() {
21        return color;
22    }
23 }
```

执行下列命令，在当前目录创建一个包与Cloth类。

```
1 javac -d . Cloth.java
```

Test.java

```
1 //package com.test; //放在同一包中，可以直接访问包
2 package com.test2;
3 import com.test.*; //不同包，用import来访问
4 class Test {
5     public static void main(String []args) {
6         System.out.println("-----init data-----");
7         Cloth test = new Cloth(30, "red");
8         System.out.println("size="+test.getSize()+" , color="+test.getColor());
```

```
9      System.out.println("----use method init data---");
10     Cloth test2 = new Cloth();
11     test2.setSize(40);
12     test2.setColor("green");
13     System.out.println("size="+test2.getSize()+", color="+test2.getColor());
14 }
15 }
```

运行结果

```
jzzh@jizizihe:~/workspace/java/capsulation$ javac -d . Test.java
jzzh@jizizihe:~/workspace/java/capsulation$ java com.test2.Test
-----init data-----
size=30, color=red
----use method init data--
size=40, color=green
```