

```
package com.singleton.type4;
```

```
public class SingletonTest04 {
```

```
    public static void main(String[] args) {  
        System.out.println("懒汉式2， 线程安全~");  
        Singleton instance = Singleton.getInstance();  
        Singleton instance2 = Singleton.getInstance();  
        System.out.println(instance == instance2); // true  
        System.out.println("instance.hashCode=" + instance.hashCode());  
        System.out.println("instance2.hashCode=" + instance2.hashCode());  
    }
```

```
}
```

```
// 懒汉式(线程安全，同步方法)
```

```
class Singleton {  
    private static Singleton instance;
```

```
    private Singleton() {}
```

```
    //提供一个静态的公有方法，加入同步处理的代码，解决线程安全问题
```

```
    //即懒汉式
```

```
    public static synchronized Singleton getInstance() {  
        if(instance == null) {  
            instance = new Singleton();  
        }  
        return instance;  
    }
```

```
}
```