

题目：设计一个类，然后只能生成该类的一个实例

```
1 public class Test02{
2     /*单例模式，饿汉式，线程安全*/
3     public static class Singleton{
4         private final static Singleton INSTANCE = new Singleton();
5         private Singleton(){
6
7         }
8         public static Singleton getInstance(){
9             return INSTANCE;
10        }
11    }
12
13    /*懒汉式*/
14    public static class Singleton2{
15        private static Singleton2 INSTANCE = null;
16        private Singleton2(){}
17        public static Singleton2 getInstance()
18        {
19            if (INSTANCE == null){
20                INSTANCE = new Singleton2();
21            }
22            return INSTANCE;
23        }
24    }
25    /*懒汉式，线程安全，多线程环境下效率不高*/
26    public static class Singleton3{
27        private static Singleton3 INSTANCE = null;
28        private Singleton3(){
29
30        }
31        public static synchronized Singleton3 getInstance(){
32            if(instance == null){
33                instance = new Singleton3();
34            }
35        }
36    }
37 }
```

```
35     return instance;
36 }
37 }
38 /*懒汉式，变种*/
39 public static class Singleton4{
40     private static Singleton4 instance = null;
41     static{
42         instance = new Singleton4();
43     }
44     private Singleton4(){
45
46     }
47     public static Singleton4 getInstance(){
48         return instance;
49     }
50 }
51 /*使用静态内部类*/
52 public static class Singleton5{
53     private final static class SingletonHolder{
54         private static final Singleton5 INSTANCE = new Singleton5();
55     }
56     private Singleton5{}
57     public static Singleton5 getInstance(){
58         return SingletonHolder.INSTANCE;
59     }
60 }
61 /*静态内部类，采用双重校验锁，线程安全,两个if循环保证在多线程条件下
只创建一个实例*/
62 public static class Singleton6{
63     private Singleton6(){ }
64     private volatile static Singleton6 instance = null;
65     public static Singleton6 getInstance(){
66         if(instance == null){
67             synchronized(Singleton6.class)
68             {
```

```
69  if(instance == null){
70  instance = new Singleton6();
71  }
72  }
73  }
74  return instance;
75  }
76  }
77
78 }
```