

```
import java.util.Arrays;

public class Singleton {

    public static class LazyHolder {
        //final只能被实例化一次
        private static final Singleton instance = new Singleton();
    }

    private Singleton() {

    }

    public static final Singleton getInstance() {
        return LazyHolder.instance;
    }

    public static void main(String [] args) {
        Singleton singleton1 = Singleton.getInstance();
        Singleton singleton2 = Singleton.getInstance();
        if(singleton1 == singleton2)
            System.out.println("same");
    }
}
```

因为一个类只能被一个线程初始化，因此利用内部类的形式可以实现多线程下单例的形式。而只有当某个线程使用单例的时候才被初始化，所以实现了懒加载