```
package com.singleton.type7;


public class SingletonTest07 {

  public static void main(String[] args) {
    System.out.println("使用静态内部类完成单例模式");
    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());

  }

}

// 静态内部类完成， 推荐使用
class Singleton {
  private static  Singleton instance;

  //构造器私有化
  private Singleton() {}

  //写一个静态内部类,该类中有一个静态属性 Singleton
  private static class SingletonInstance {
    private static final Singleton INSTANCE = new Singleton();
  }

  //提供一个静态的公有方法，直接返回SingletonInstance.INSTANCE

  public static  Singleton getInstance() {

    return SingletonInstance.INSTANCE;
  }
}
```