## 简介

单例模式保证一个类有且只有一个实例对象，并提供一个访问该实例对象方法的解决方案。

## 问题

一个全局使用的类频繁的创建和销毁。

一个类如果存在多个实例会导致逻辑混乱。

## 要点

1. 构造器私有。
2. 拥有一个静态的、自身属性对象。
3. 一个访问自身对象的访问方法。

## 分类

1. 饿汉式

在自身属性后面直接实例化自身对象。不管用不用到单例，都会占用内存，但因为在加载类时已经初始化完成，所以第一次调用时会很快。

1. 懒汉式

在访问方法里面实例化自身对象。因为在访问时才实例化对象，如果工作较多，性能上会有损耗，第一次调用时会较慢，以后就会很快。

1. 双重锁

线程安全，第一次调用访问方法时才同步，避免了性能损耗。

1. 在方法上加同步synchronized

线程安全，但是每次都要同步，会影响性能。

1. 静态内部类

利用classload机制来保证初始化instance时只有一个线程，所以也是线程安全的，同时也没有性能损耗。

3、登记式