目标：

1. 掌握单例模式的应用场景
2. 掌握IDEA环境下的多线程调试方法
3. 掌握保证线程安全的单例模式侧罗
4. 掌握反射暴力攻击单例解决方案及原理分析
5. 序列化破坏单例的原理及解决方案
6. 掌握常见的单例模式写法。

单例模式（Singleton Pattern）：是指确保一个类在任何情况下都绝对只有一个实例，并提供一个全局访问点。

隐藏其所有的构造方法。属于创建型模式。

例如：ServletContext /sevletConfig /ApplicationContext / DBPool

单例模式的常见写法：

1. 饿汉式单例
2. 懒汉式单例
3. 注册式单例
4. ThreadLocal单例

**饿汉式单例**：在单例类首次加载时就创建实例。

隐藏构造、声明时创建、提供一个公共的获取实例的方法getInstance();

优点：执行效率高，性能高，没有任何的锁。

缺点：某些情况下，可能会造成内存浪费。

类加载顺序：先静态后动态、先上后下 、先属性后方法

**懒汉式单例**：被外部类调用时才创建实例。

**注册式单例**：将每一个实例都缓存到统一的容器中，使用唯一标识获取实例。