

# GOF 设计模式小白教程之单例模式

作者: [valarchie](#)

原文链接: <https://ld246.com/article/1566998770528>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

# 单例模式 (Singleton)

## 定义:

该模式只涉及到一个类，该类负责创建自己的对象，同时确保只有唯一一个对象被创建。这个类提供一种访问其唯一的对象的方式，可以直接访问，不需要实例化该类的对象。

## 通俗解释:

好比以前的计划生育一对夫妻只能有一个孩子，因为生育抚养孩子的成本太高了。所以限制只能有一个。单例模式也是如此，为了节省创建对象的成本或节省对象占用的内存空间而实现的模式。

单例模式最重要的三点特征如下:

- 单例类只有一个实例对象;
- 该单例对象必须由单例类自行创建;
- 单例类对外提供一个访问该单例的全局访问点;

## 代码:

双重检测加volatile的单例模式

```
public class Singleton {  
    private volatile static Singleton instance = null;  
  
    private Singleton() {}  
  
    public static Singleton getInstance() {  
        if (instance == null) {  
            synchronized (Singleton.class) {  
                if (instance == null) {  
                    instance = new Singleton();  
                }  
            }  
        }  
        return instance;  
    }  
}
```

## 解析:

1. 在内存里只有一个实例，减少了内存的开销，尤其是频繁的创建和销毁实例。
2. 避免对资源的多重占用。