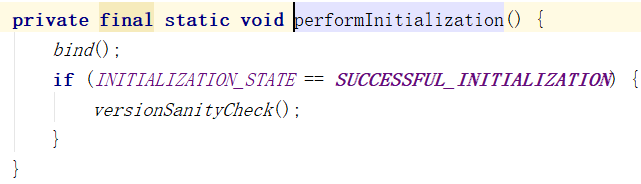
1. 如何使用
2. 分析

当我们调用getILoggerFactory()函数的时候，会先执行初始化操作：



而performInitialization()函数主要做了两件事，第一是执行绑定操作，第二就是在绑定操作完成后执行校验操作：



绑定操作的具体逻辑如下：

1. 通过LoggerFactory的classLoader来查找**org.slf4j.impl.StaticLoggerBinder.class**类，如果找到了多个该类则抛出错误，否则查找步骤通过。另外如果是Android平台则会跳过此步查找。
2. 调用StaticLoggerBinder.*getSingleton*();这个调用能够帮助具体实现完成一些初始化工作。实际上在第1步的时候有人会有个疑问，如果找不到**org.slf4j.impl.StaticLoggerBinder.class**类呢，实际上这一步就会抛出异常，因为找不到StaticLoggerBinder类，实际上这个异常会被捕获，然后会将初始化的结果置位NOP\_FALLBACK\_INITIALIZATION 。如果这一步正常执行完毕，就会将初始化的结果置为SUCCESSFUL\_INITIALIZATION

3、执行下面两个函数：

SubstituteLogger*fixSubstituteLoggers*();  
*replayEvents*();

前者主要是对代理Logger工厂的代理Logger设置需要代理的对象。