

+ (void) initialize 与 + (void) load 两个方法的比较

	+load	+initialize
调用时机	被添加 runtime 时	收到第一条消息时，也可能永远不调用
调用顺序	父类 -> 子类 -> 分类	父类 -> 子类
调用次数	1 次	系统执行 1 次，手动可以调用多次
是否需要显式调用父类实现	否	否
是否沿用父类的实现	否	是
分类中的实现	类和分类都执行	分类覆盖类中的实现

相同点：

- 1. 系统都执行一次。
- 2. 父类在子类之前被调用。

不同点：

- 1. load方法会在**加载类的时候**就被调用，也就是 ios 应用启动时就会加载所有的类，就会调用每个类的 +load 方法；initialize 方法会在**第一次初始化这个类之前**被调用，我们用它来初始化静态变量。
- 2. load会在 main() 函数之前调用，+initialize 则在类实例化或调用类方法时调用。load 顺序在 initialize 之前。
- 3. 如果子类中没有 +initialize 方法，则会再次调用父类的 +initialize 方法。
- 4. 类别会覆盖主类的 +initialize 方法，+load 方法则不会被覆盖。
- 5. +initialize 方法的调用看起来会更合理，通常在它里面写代码比在 + load 里写更好，因为它是懒调用的，是有可能完全不被调用的。
- 6. 类接收消息时，运行时会先**检查 + initialize 有没有被调用过**。如果没有，则会在消息被处理前调用。
- 7. initialize 最终是通过 objc_msgSend 来执行的，objc_msgSend 会执行一系列方法查找，并且 Category 的方法会覆盖类中的方法；load 是在被添加到 runtime 时开始执行，父类最先执行，然后是子类，最后是 Category。又因为是直接获取函数指针来执行，不会像 objc_msgSend 一样会有方法查找的过程。