5S哲学

整理（Seiri）

整顿（Seiton）

清楚（Seiso）

清洁（Seiketsu）

身美（Shitsuke）

1. ：整洁代码

为什么要整洁代码

糟糕的代码究竟有什么危害

混乱的代价

简单代码，依其重要顺序：

能通过所有测试

没有重复代码

体现系统中的全部设计理念

包括尽量少的实体，比如类、方法、函数等

1. ：有意义的命名

取个好名字的的规则

名副其实

（如果名称需要注释来补充，那就不算是名副其实）

避免误导

做有意义的区分

废话都是冗余

例子（variable一词永远都不应该出现变量名中，Table一词永远不当出现在表名中）

犯错的代码

getActivityAccount（）

getActivityAccounts（）

getActivityAccountInfo（）

如果缺少明确的约定，God knows调用哪个函数。要区分名称，就要以读者能够鉴别不同之处的方式来区分

2.5使用读得出来的名称

如果名称读不出来，那讨论的时候就会像个傻鸟

2.6使用可搜索的名称

单字母变量和数字常量有一个问题，就是很难在一大篇文字中找出来

单字母应该仅用于短方法中的本地变量，名称长短应与其作用域对应

2.7避免使用编码

2.7.1匈牙利语标记法

2.7.2成员前缀

也不必使用m\_前缀来标明成员变量。应当把类和函数做得足够小，消除对成员前缀的需求