

## 习惯一：对事不对人

不要谴责，不要评判，简单地表达自己的观点。

负面的评论和态度扼杀了创新。

团体决策：设定最终期限；逆向思维；设立仲裁人；支持已经做出的决定。

对“最好”的含义达成共识。只有更好，没有最好。

## 习惯二：跟踪变化

迭代和增量式学习；了解最新行情；参加本地的用户组活动；参加研讨会议；如饥似渴地阅读；跟踪技术变化。

做决策之前，评估新技术的优势。

不需要精通所有的技术，但是要清楚行业的动向，从而规划项目和职业生涯。

## 习惯三：让设计指导而不是操纵开发

CRC卡片：类名，职责（做什么），协作者（与什么对象一起工作）。

好的设计是正确的，不是精确的。

设计是没有价值的，但设计的过程是必不可少的。

## 习惯四：提早实现自动化部署

一开始就实现自动化部署应用。

系统的安装或部署应该简单、可靠、可重复。

## 习惯五：度量真实的进度

使用待办事项度量剩下的工作量。

关注功能，而不是日程表。

## 习惯六：用代码沟通

使用有意义的命名。

注释（目的，输入，输出，异常），不能代替优秀的代码。

## 习惯七：编写内聚的代码

让类的功能尽量集中，让组件尽量小。

## 习惯八：根据契约进行替换

通过替换遵循接口契约的类，来添加并改进功能特性，多使用委托而不是继承。

## 习惯九：报告所有的异常

处理或是向上传播所有的异常。

## 习惯十：做代码复查

通宵复查；捡拾游戏；结对编程。

代码复查提升代码质量，降低错误率。