

# 33-35

## ▼ 33-个人性格

### ▼ 1 个人性格是否和本书主题无关

- 你无法提高自己的聪明程度，但性格在一定程度可以改进
- 个人性格对造就程序员高手具有决定性意义

### ▼ 2 聪明和谦虚

- 高智商与优秀程序之间没有密切的关系
- 精通编程的人是哪些了解自己头脑有多大局限的人，都很谦虚。
- 承认自己智力有限，并且通过学习来弥补。你会成为更好的程序员
- 越是谦虚，进步就越快
- 谦虚的程序员善于弥补其不足之处，编写的代码让自己和他人易于看懂，其中的错误也较少

### ▼ 3 求知欲

- 在开发过程中建立自我意识
- 实验
- 阅读解决问题的有关方法
- 在行动之前做分析和计划
- 学习成功项目的开发经验
- 阅读文档
- 阅读其他书本期刊
- 同专业人士交往

### ▼ 向专业开发看齐

- 技术带头人会为人写代码，而非机器
- 有人之所以写出难以看懂的代码，主要还是因为其代码质量太差

### ▪ 4 诚实

### ▪ 5 交流与合作

### ▪ 6 创造力和纪律

### ▼ 7 懒惰

- 拖延不喜欢的任务
- 迅速做完不喜欢的任务，摆脱之

- 编写工具来完成不喜欢的任务，以摆脱之（高级）

## ▼ 8 不如你想象中起作用的性格因素

### ▼ 坚持

- 写代码卡壳时，不妨设计新类。或者先放一边，后面再解决问题
- 善用潜意识，遇到遇难时设置底牌时间。到时间搞不完先停下来干点别的，让潜意识自己屡屡

### ▪ 经验

### ▼ 编程狂人

- 不推荐疯狂输出，要合理安排作息。打持久战方是王道

## ▼ 9 习惯

- 不能用没有习惯来代替坏习惯
- 努力培养好习惯，坏习惯自然而然就变少了

## ▼ 34-软件工艺的话题

- 1 征服复杂性
- 2 精选开发过程
- 3 首先是为人写程序，其次才是机器
- 4 深入一门语言去编程，而不是浮于表面
- 5 借助规范集中注意力

### ▼ 6 基于问题域编程

- 将程序划分为不同层次的抽象
- 问题域的低层技术

### ▪ 7 当心落石

### ▪ 8 迭代。反反复复，一次又一次

### ▼ 9 汝当分离软件与信仰

- 判断
- 折中主义

### ▼ 实验

- 试图没有错误是最大的错误，设计正是仔细的规划小错误以避免大错误的过程
- 实验应该贯穿于整个开发过程，并且要能基于实验结果改变思路。
- 保持开放的心态，这样才能有所收获

## ▼ 55-更多

- 拥有一套核心书籍，涉及到主要的软件开发活动：需求、设计、构建、管理、测试
- 1 关于软件构建的信息
- 2 构建之外的话题
- 3 期刊
- 4 软件开发者的阅读计划
- 5 参加专业组织