

别死写代码，这 25 条比涨工资都重要

JavaGuide 2019-10-12 12:03

来源: <http://rrd.me/eucMA>

如果每个程序开发人员都只是周而复始地写代码，想必编程的工作就会让很多程序员发狂。那么如何在枯燥的工作中寻找乐趣，我认为我们要对工作保持激情和兴奋感、始终学习的态度，还需要掌握一些实用高效的编程技巧，这样对于提升工作乐趣和效率大有裨益。

今天给大家分享一下 25 个实用编程小技巧，希望可以给开发编程人员的工作和学习带来更多的思考。

- 1、不要毫无计划地写代码，思考、调研、计划、编码、测试、修改，一个都不能少；**
- 2、不要写代码前过度计划，在一头钻进代码前做点计划是好事，但是即便是好事，也可能物极必反。
- 3、请勿低估代码质量的重要性，如果你只能够关注你所写的代码的一个方面，那么肯定是可读性。
- 4、使用实现功能的最简单方案，作为专业的程序员，你的职责不是找出问题的一个解决方案，而是找出问题的最简单的解决方案；
- 5、适时放弃，当你开始怀疑一个解决方案的时候，你就应该考虑抛弃它，并且重新思考这个问题。不管你已经在这个解决方案中投入了多少精力。像 GIT 这样的版本控制系统能够帮助你分开管理和尝试多种不同的解决方案，把它利用起来吧；
- 6、正确使用 Google，除非你正在使用一种极其前沿的技术，否则当你遇到一个问题时，很可能别人早就遇到过同样的问题了，并且也找到了解决方案了。给自己省点时间，先 Google 一下；
- 7、做好封装，基本的想法就是你想你的代码高内聚和低耦合，意思是说保持相关的代码在一起（在一个类中），降低不同类之间的相互依赖；
- 8、做好规划，写好需求再写代码，尽可能编写目前正在实现的方案所需的最少量代码；**
- 9、要懂算法，使用合适的数据结构；**
- 10、不要写重复性代码，要用好配置文件，不要使用没必要的条件语句和临时变量；
- 11、做好代码注释，但是不要给傻子都知道的代码写注释；
- 12、一定要写好测试，如果可能的话，甚至在开始写代码实现需求之前，你就应该开始预估和设计需要测试校验的情况了。测试驱动开发（Testing-driven development, TDD）不是什么花俏的炒作，它是会实实在在会对你思考功能特性、寻找更好的设计方案产生积极影响的。**
- 13、不要觉得代码运行起来就是正确的，有些时候代码的 bug 可能并不是显而易见的；
- 14、要能够质疑既有代码，作为一个初学者，总是应该假定那些你读不懂的、且没有文档注释的代码很可能就是糟糕的代码。质疑之，询问之，使用 git blame 揪出罪魁祸首！

15、不要过度迷恋最佳实践，我觉得 "最佳实践" 其实是害人的，它暗示着你不需要深入研究它，这就是有史以来最佳实践，不用质疑！

16、不要过度迷恋性能优化，如果你在运行代码之前就在优化它了，那很可能你就是在过早优化代码了，也很可能你正在费时费力做的优化是完全没必要的。

17、以用户体验为目标，要站在最终用户的角度看问题。专业的开发者要考虑这个特定功能的用户需要什么、怎样使用，要想方设法使得这个功能容易让用户发现和使用，而不是想方设法在应用中用最便捷添加这个功能，毫不考虑这个功能的可发现性和可用性。

18、为你的开发任务挑选合适的工具，你可以使用最原始的工具建造房子，然后享受甜蜜时光。你也可以花费一些时间和金钱去了解先进的工具、更快地建造更好的房子。工具在不断地改进中，你要乐意去学习它们、使用它们。

19、要理解好代码问题和数据问题之间的关系，即使是程序中最小的 bug 也会导致它所管理的数据去到一种不可预测的状态。尤其是当所有数据校验都完全在这个有 bug 的程序中进行时。

20、切勿重复造轮子，使用好现有的轮子和各种开源库，会让你事半功倍。当然，不要仅仅为了使用一两个函数就引入一整个代码库，在 JavaScript 中的典型例子就是 lodash 代码库；

21、对代码审查保持正确的态度，应该把每一次代码复审当作是学习的机会，欢迎他们、感激他们、从中学习，最重要的，当你从你的代码复审人员那里学习到东西的时候，要感谢他们；

22、用好版本控制工具和系统，新手往往低估了一个好的版本控制系统的威力，我这里所说的好的版本控制系统其实就是指 Git；

23、不要过度使用共享状态，一个新手可能会尝试使用定时器来解决这个共享变量的竞态条件问题，特别是当他们必须处理一个数据锁的问题时。这是危险的标志，别这么做，注意它，在代码复审中指出它，永远也不要接受这样的代码。

24、正视 Error，Error 是好东西。Error 意味着你在进步，意味着你可以通过简单的后续修改就获得更多的进步。专业程序员喜爱 Error。新手则痛恨 Error；

25、学会休息，任何人的大脑都需要休息，身体也需要休息。

推荐阅读

[新手也能看懂的 SpringBoot 异步编程指南](#)

[JWT 身份认证优缺点分析以及常见问题解决方案](#)

[架构设计--互联网架构演化](#)

[值得开发者关注的 Java 8 后时代的语言特性](#)

[【原创】关于Spring中的参数校验的一点思考](#)

[Spring常见问题总结（补充版）](#)

[分布式ID生成方案总结](#)

[入职一个月的职场小白,谈谈自己这段时间的感受](#)

[Code Review最佳实践](#)

[后端开发必备的 RestFul API 知识](#)

[技术面试复习大纲](#)

[如何给老婆解释什么是RPC](#)

[干货收藏 | Java程序员必备的一些流程图](#)

[十分钟搞懂Java效率工具Lombok使用与原理](#)

[看完这篇文章，别说自己不会用Lambda表达式了！](#)

[一些有助于你拿Offer的文章](#)

[面试官：“谈谈Spring中都用到了那些设计模式？”。](#)

如果大家想要实时关注我更新的文章以及分享的干货的话，可以关注我的公众号。

欢迎关注 点个再看

喜欢此内容的人还喜欢

面试官：为什么忘记密码要重置而不是告诉你原密码？

JavaGuide



美团面试：如何检测和避免线程死锁？

JavaGuide



技术面通过，HR面挂了！难崩！

JavaGuide

