

程序员练级攻略（2018）：正式入门

2018-06-05 陈皓



程序员练级攻略（2018）：正式入门

朗读人：柴巍 11'04" | 5.08M

学习了前面文章中的入门级经验和知识后，你可能会会有两种反应。

- 一种反应可能是，你对编程有一点的兴趣了，甚至有一点点小骄傲，可能还会四处炫耀。我想说，请保持这种感觉，但是你也要清醒一下，上面的那些东西，还不算真正的入门，你只是入门了一条腿。
- 另一种反应也可能是，你被吓着了，觉得太难了。感觉不是一般人能玩的，如果是这样的话，我想鼓励你一下 -- "无论你做什么事，你都会面对各式各样的困难，这对每个人都是一样的，而只有兴趣、热情和成就感才能让你不怕这些困难"。所以，你问问你自己，是否从中收获了成就感或是兴趣，如果没有的话，可能这条路并不适合你。如果有的话，哪怕一丁点儿，你也应该继续坚持下来。

这篇文章，我主要是让你成为更为专业的入门程序员。请注意，此时，你可能需要读一些比较枯燥的书，但我想说，这些是非常非常重要的。你一定要坚持住。

编程技能

在系统地学习编程技能之前，我希望你能先看一下 "[The Key To Accelerating Your Coding Skills](#)"，这篇文章会告诉你如何有效地快速提高自己的编程能力。

然后接下来是下面几大块内容，但还只是入门级的。

- 编程技巧方面 - 你可以开始看怎么把程序写好的书了，这里推荐的是《[代码大全](#)》。这本书好多年没有更新了，其中有一些内容可能有点过时，但还是一本非常好的书，有点厚，你不需要马上就看完。在你的编程路上，这本书可以陪你走很久，因为当你有更多的编程经验时，走过更多的坑后，再把这本书拿出来看看，你会有更多的体会。好的书和不好的书最大的不一样的是，好的书在你不同的阶段来读，你会有不同的收获，而且还会产生更多的深层次的思考！《代码大全》就是这样的一本书。
- 编程语言方面 - 这个阶段，你可以开始了解一下 Java 语言了，我个人觉得 Java 是世界上目前为止综合排名最好的语言。你一定要学好这门语言。推荐《[Java 核心技术（卷 1）](#)》，除了让你了解 Java 的语法，它还会让你了解面向对象编程是个什么概念（如果你觉得这本书有点深，那么，你可以降低难度看更为基础的《[Head First Java](#)》）。然后，既然开始学习 Java 了，那就一定要学 Spring，推荐看看《[Spring in Action](#)》或是直接从最新的 Spring Boot 开始，推荐看看《[Spring Boot 实战](#)》。关于 Spring 的这两本书，里面可能会有很多你从来没有听说过的东西，比如，IoC 和 AOP 之类的东西，能看懂多少就看懂多少，没事儿。
- 操作系统 - 这里你可以看看《[鸟哥的 Linux 私房菜](#)》，这本书会让你对计算机和操作系统，以及 Linux 有一个非常全面的了解，并能够管理或是操作好一个 Linux 系统。当然，这本书有很多比较专业的知识，你可能会看不懂，没关系，就暂时略过就好了。这本书的确并不适合初学者，你能看多少就看多少吧。
- 网络协议 - 你需要系统地了解一下 HTTP 协议，请到 MDN 阅读一下其官方的[HTTP 的文档](#)。你需要知道 HTTP 协议的几个关键点：1) HTTP 头，2) HTTP 的请求方法，3) HTTP 的返回码。还有，HTTP 的 Cookie、缓存、会话，以及链接管理，等等，在 MDN 的这个文档中都有了。对于 HTTP 协议，你不需要知道所有的东西，你只需要了解这个协议的最关键的那些东西就好了。
- 数据库设计 - 你需要系统地了解一下数据库设计中的那些东西，这里推荐慕课网的一个在线课程 -- "[数据库设计的那些事](#)"。每个小课程不过 5-6 分钟，全部不到 2 个小时，我相信你一定能跟下来。你需要搞清楚数据的那几个范式，还有 SQL 语句的一些用法。当然，你还要学习和使用一下数据库，这里推荐学习开源的 MySQL。你可以看官方文档，也可以看一下这本书《[MySQL 必知必会](#)》。

- 前端方面 - 前端的東西不算复杂，你需要学习几个东西。一个是和 JavaScript 相关的 [jQuery](#)，另一个是和 CSS 相关的 [Bootstrap](#)，学习这两个东西都不复杂，直接上其官网看文档就好了。最重要的是，你要学习一下，JavaScript 的 Ajax 请求后端的 API 接口的方式，而不是再像前面那样用后端来向前端返回 HTML 页面的形式了。这里，你需要学习一下，JavaScript 的 Promise 模式。[阮一峰翻译的 ES6 的教程中有相关的内容](#)。当然，你 Google 一下，也可以找到一堆学习资料。
- 字符编码方面 - 在你处理中文时有时会发现乱码出现，此时需要了解 ASCII 和 Unicode 这样的字符编码。这里推荐一篇文章 - "[关于字符编码，你所需要知道的 \(ASCII,Unicode,Utf-8,GB2312...\)](#) " 或是英文文章 "[The history of Character Encoding](#)" 以及 [Wikipedia - Character encoding](#)。还有 Github 上的这两个 Awesome 仓库：[Awesome Unicode](#) 和 [Awesome Code Points](#)。

为什么转成 Java 语言？

相信你可能会问，为什么之前学习的 Python 和 JavaScript 不接着学，而是直接切到 Java 语言上来，这样会不会切得太快了。这是一个好问题，这里需要说明一下，为什么我会切到 Java 这个语言上来，主要是有以下几方面考虑。

1. Java 是所有语言里面综合实力最强的，这也是为什么几乎所有大型的互联网或是分布式架构基本上都是 Java 技术栈。所以，这是一个工业级的编程语言（Python 和 JavaScript 还达不到这样的水准）。
2. 之所以，没有用 Java 来做入门语言而是用了 Python，这是因为编程是一件比较费脑子的事，一开始学习时，兴趣的培养很重要。Python 比较简单，容易上手，能够比较容易地提起兴趣，而用 Java 则可能比较难。
3. 在你有了一些编程语言的基础后，有了一些代码的逻辑后，切到工业级的编程语言上来，更为专业地学习编程，是非常有帮助的。像 Python 和 JavaScript 这样的动态语言用着是很爽，但是，只有像 C、C++ 和 Java 这样的静态语言才可以让你真正地进阶。
4. 对于一个合格的程序员，掌握几门语言是非常正常的事情。一方面，这会让你对不同的语言进行比较，让你有更多的思考。另一方面，这也是一种学习能力的培养，会让你对于未来的新技术学习得更快。很多时候，一些程序员只在自己熟悉的技术而不是合适的技术上工作，这其实并不好，这会让你的视野受限，而视野会决定你的高度。综上所述，这就是在入门的时候我故意让你多学几门语言的原因。

编程工具

编程工具方面，你需要开始学习使用下面这些工具了。

- 编程的 IDE。传统一点的，你可以使用 Eclipse ([教程](#))。当然，我推荐你使用 IntelliJ IDEA ([教程](#))。这两个工具都可以开发各种语言，但是主要用在 Java。如果你想玩得更时髦一些的话，使用 Visual Studio Code 也不错，这个工具潜力十足，用其开发 Python、JavaScript、Java、Go、C 和 C++ 都能得心应手 ([教程](#) - 本文发布时还基本没怎么翻译完)。
- 版本管理工具。版本管理工具是非常重要的编程工具。传统的有 P4、SVN、CVS 等，但都会被 Git 取代，所以，你就只用学习 Git 就好了。学习 Git 的教程网上有很多，这里我推荐非常系统的 [Pro Git 第二版](#)（如果你觉得 Pro Git 比较枯燥的话，备选[猴子都能懂的 Git 入门](#)），然后你要学会使用 GitHub。关于一些 Git 环境安装和准备以及 GitHub 使用，你可以自行 Google（比如：这篇[Github and Git 图文教程](#)或是这篇[Git 图文教程及详解](#)）。
- 调试前端程序。你需要学会使用 Chrome 调试前端程序，Google 一下会有很多文章，你可以看看 [超完整的 Chrome 浏览器客户端调试大全](#)。
- 数据库设计工具。你需要学会使用 MySQL WorkBench，这个工具很容易使用。相关的手册，你可以看一下[官方文档](#)。

实践项目

这回我们需要设计一个投票系统的项目。

业务上的需求如下：

- 用户只有在登录后，才可以生成投票表单。
- 投票项可以单选，可以多选。
- 其它用户投票后显示当前投票结果（但是不能刷票）。
- 投票有相应的时间，页面上需要出现倒计时。
- 投票结果需要用不同颜色不同长度的横条，并显示百分比和人数。

技术上的需求如下：

- 这回要用 Java Spring Boot 来实现了，然后，后端不返回任何的 HTML，只返回 JSon 数据给前端。
- 由前端的 JQuery 来处理并操作相关的 HTML 动态生成在前端展示的页面。
- 前端的页面还要是响应式的，也就是可以在手机端和电脑端有不同的呈现。这个可以用 Bootstrap 来完成。

如果你有兴趣，还可以挑战以下这些功能。

- 在微信中，通过微信授权后记录用户信息，以防止刷票。
- 可以不用刷页面，就可以动态地看到投票结果的变化。

- Google 一些画图表的 JavaScript 库，然后把图片表得风骚一些。

小结

上面那些书和知识你要看完，还要能理解并掌握，我估计你最少也要花 1-2 年左右的时间。如果你能够走到这里，把前面的那些知识都了解了，不用精通，能独立地做出上面的那些实践项目，那么，你就算是真正的入门了。

而且，你已经是一个 " 全栈工程师 " 的样子了，在这里我要给你一个大大的赞。如果这个时候，你对编程还有很大的热情，那么我要恭喜你了，你可能会是一个非常不错的程序员。加油啊！

上面的那些技术已经算是比较专业的了。如果你已经大致掌握了，我相信你可以找到至少年薪 20 万以上的工作了，而且你的知识面算是有不错的广度了。但是深度还不够，这个时候，是一个比较关键点了。

你可能已经沉醉在沾沾自喜的骄傲的情绪中，那么你也可以就此止步，加入一些公司，在那里按部就班地完成一些功能性的开发，成为一个搬砖的码农。你也可以开始选择一个方向开始深入。

我给你的建议是选择一个方向开始深入。因为你并不知道你未来会有多大的可能性，也不知道你会成为什么样的人，所以为什么不再更努力一把呢？

后面，我们就开始非常专业的程序员之路了。这也是一般程序员和高级程序员的分水岭了，能不能过去就看你的了。

下面是《程序员练级攻略（2018）》系列文章的目录（持续更新中）。

- [开篇词](#)
- 入门篇
 - [零基础启蒙](#)
 - [正式入门](#)
- 修养篇
 - [程序员修养](#)
- 专业基础篇
 - 编程语言
 - 理论学科
 - 系统知识
- 软件设计篇
- 高手成长篇



版权归极客邦科技所有，未经许可不得转载

精选留言



Gloria_前端程序媛

9

非常干货，大一学生党表示这波付费操作很值，希望耗子叔大神可以多分享前端UI方面的知识!嘻嘻☺我超喜欢前端滴

2018-06-05

作者回复

那要到最后了

2018-06-07



恩言

5

皓哥，我只能说我爱你了，“有了兴趣，热情，成就感才能走下去”

2018-05-31



刘强

4

为什么我学的时候没有这啊 苍天啊

2018-06-06



辉哥

3

耗子老师，既然是云计算的时代，可不可以学习golang，而不学java呢？

2018-06-05



云学

3

这样的文章对于想转型的程序员就是雪中送炭，太棒了

2018-06-05





WTF

👍 2

可以作为java web开发的一个自我检视的checklist，亦可作为面试的技术面试问题大纲，一文二鸟。

2018-05-31



黑猫大侠

👍 1

多code，工作五年了，上面的都会，很熟练了。想多看看高负载，分布式开发部署，日志系统的架构。

2018-06-09



jony

👍 1

快工作两年了，这时能够遇到皓哥的专栏是不是很幸运啊

2018-06-05

| 作者回复

当然是

2018-06-07



瓶子

👍 1

啥也不说了，撸起袖子，就是干！~多谢耗子哥，干货十足，接地气！非常感谢！

2018-05-31



林子

👍 1

工作了六七年，都是干货分享，赞赞赞！

2018-05-31



法正

👍 0

目标是全栈工程师吗？一定要学Java吗，或者说一定要这么全面吗。如果只是对某一方面感兴趣是不是可以不用全都学，只是深入自己喜欢的那个。

2018-06-10



Silence-0°C

👍 0

我想向架构师发展，但是目前的工作不太切合，自己想努力，没有个清晰的方向，还望指教

2018-06-10



D瓜哥

👍 0

慕课网有个关于IDEA的视频教程，也非常棒。感兴趣的可以看一看。

<https://www.imooc.com/learn/924> [IntelliJ IDEA神器使用技巧-慕课网]

2018-06-10



metalmac.kyle

👍 0

皓哥，有个问题，这慢慢练级路固然很重要，不过有些我熟悉而且我兴趣点很明显倾向于网络以及操作系统底层，那能否跳过一些步骤直接切入呢？还是最好按您推荐的认认真真梳理

一遍打怪升级更重要呢？这个疑虑基于两点，客观说个人基础确实不太扎实经验也不如太多牛人丰富基础不牢固就考虑是否这样梳理更有效呢？二对技术的选择也有些偏好（比如不太喜欢Java web前端那些）担心个人广度不够会对自己受限

2018-06-10

作者回复

1) 入门并不是所有的可以跳过（比如程序员修养什么的），专业基础篇不能跳过。

2) 网络和操作系统底层，我个人觉得并不难，只是相对比较枯燥，也比较单一，就是个Know How的事，掌握也就是时间长短的问题，不必把自己限制在这个地方。

3) 你不了解Java，你也无法了解语言的发展，Java是最具生产力，也是最工业化的语言之一。你可以不学不喜欢，但是你的视野会被限制。

2018-06-11



宋桓公

0

感觉就Java没怎么学，it is time to learn java?

2018-06-10



Silence-0°C

0

谢谢，能够看到这么干货的文章，我是计算机专业毕业，工作却是在一个普通的省城工作，周围没有高大上的公司，空闲下来，自己也有在思考如何进一步发展，却没有自成体系，视野受限，有幸拜读了你的文章，茅塞顿开，跟随着你的脚步继续学习！

2018-06-09



奔向大海的山贼

0

有些还是有幫助的 这个系列我感觉是从全局来说的 或者说作为过来人总结的 因为每个人切入程序点不一样 就拿我来说 我是从c入行 然后到c++ 没事的时候正在学习python web技术就是想练手约一下这个脚本，拓展下技术栈 还犹豫过要不要学习java不过看来还是要后期了解下 其实学习顺序感觉每个人都不一样 都是知识点或者说技术栈 方面说的还是蛮到位的 本人入行四年

2018-06-08



周文童

0

《代码大全》这书太牛逼了

2018-06-08



Kenmmm

0

智能合约刚写完投票，这里又来投票(□♀)

2018-06-08



Lyle

0

当年没有这样的干货文章指导太可惜了。向耗子哥学习！

2018-06-07