

## 70 | 程序员练级攻略：零基础启蒙

2018-05-31 陈皓

左耳听风

[进入课程 >](#)



讲述：柴巍

时长 09:44 大小 4.46M



如果你从来没有接触过程序语言，这里给你两个方面的教程，一个偏后端，一个偏前端。对从零基础开始的人来说，最重要的是能够对编程有兴趣，而要对编程有兴趣，就要有成就感。而成就感又来自于用程序打造东西，所以，我推荐下面这份不错的入门教程。

第一份入门教程，主要是让你体会到编程是什么。

[《与孩子一起学编程》](#)，这本书以 Python 语言教你如何写程序，是一本老少咸宜的编程书。其中会教你编一些小游戏，还会和你讲基本的编程知识，相当不错。

两个在线编程入门的网站：[Codecademy: Learn Python](#) 和 [People Can Program](#)，你也可以在这两个网站上学习 Python，只不过是英文的。

然后，你可以到 [CodeAbbey](#) 上去做一些在线编程的小练习。

第二份入门教程，主要是让你做点实际有用的东西。嗯，做个网页吧。

[MDN 的 Web 开发入门](#)，MDN 全称是 Mozilla Developer Network，你可以认为是 Web 方面的官方技术网站。这个教程会带着你建立一个网站。然后，你可以把你的网页发布在 GitHub 上。

这两份教程都很不错，都是从零开始，带着你从环境准备开始，一点一点地从一些简单又有趣的东西入手，让你感觉一下编程世界是什么样的，相信这两个教程可以让零基础的你喜欢上编程。

## 编程入门

在这时，我们使用 Python 和 JavaScript 作为入门语言。Python 就不用多说了，语法比较简单，有大量的库和语法糖，是零基础的人学习编程的不二之选。而 JavaScript 则是前端的语言，为了让你更有编程的成就感，所以，这也成了一门要学习的语言。（注意：对于计算机专业的学生来说，一般会使用 Pascal 做为入门的编程语言，但我觉得编程入门还是要以培养兴趣为主，所以，还是选一些能让人有成就感的语言会更好）。

## 入门语言 Python

如果你想更为系统地学习一下 Python 编程，我强烈推荐你阅读下面这两本书。它们是零基础入门非常不错的图书，里面有大量的更为实用的示例和项目，可以快速给你正反馈。

[Python 编程快速上手](#)

[Python 编程：从入门到实践](#)

这两本书除了编程语法方面的讲述有所不同之外，其他都差不多，主要是通过书中的示例来强化你对编程的学习。第一本偏文本处理，包括处理 Word、Excel 和 PDF，第二本中有一些 Web 项目和代码部署方面的内容。如果可能的话，你可以把两本书中的示例都跑一遍。如果你时间有限的话，我推荐你看第二本。

[编辑植入：极客时间上也有 Python 入门的视频课程。]

## 入门语言 JavaScript

如果想入门学习 JavaScript，我主要推荐以下在线教程。

[MDN JavaScript 教程](#)，你可以认为这是最权威的 JavaScript 官方教程了，从初级到中级再到高级。

[W3School JavaScript 教程](#)，这个教程比较偏 Web 方面的编程。

[JavaScript 全栈教程（廖雪峰）](#)，这是廖雪峰的一个比较偏应用的教程，也是偏 Web 方面的编程，同时包括涉及后端的 Node.js 方面的教程。

## 操作系统入门 Linux

学习编程你还需要会玩 Linux，虽然 Windows 占据着更多的桌面市场，但是你还是要了解 Linux。这里，你可以看一下，W3CSchool 上的在线教程 [Linux 教程](#)。

## 编程工具 Visual Studio Code

这里主要推荐时下最流行也是最好用的 Visual Studio Code，这个工具潜力十足，用它开发 Python、JavaScript、Java、Go、C/C++ 都能得心应手（[教程](#) - 本文发布时还基本没怎么翻译完）。

## Web 编程入门

如果玩到这里，你觉得有趣的话，可以学习一下 Web 方面的入门知识。**为什么是 Web 而不是别的其他技术呢？因为你正身处于第三次工业革命的信息化浪潮中，在这个浪潮中，Web 互联网是其中最大的发明，所以，这是任何一个程序员都不能错过的。**

关于 Web 编程，有下面几个方向你要学习一下。

**前端基础。**要系统地学习一下前端的知识，也就是 CSS、HTML 和 JavaScript 这三个东西。这里还是给出 MDN 的相关的技术文档页面 [CSS 文档](#) 和 [HTML 文档](#)。**文档很大，你要学习的并不是所有的东西，而是了解 CSS 和 HTML 是怎么相互作用来展示数据的，然后，不用记忆文档中的内容，这两个文档是用来查找知识的。**另外，你可以简单地学习使用 JavaScript 操纵 HTML。理解 DOM 和动态网页（可以参看 [W3Schools 的 JavaScript HTML DOM 的教程](#)）。

**后端基础。**如果你想省点事，不想再学一门新的语言了，那么你可以直接用 Python 或者 Node.js，这两个技术在前面提到的廖雪峰的那个教程里提到过。当然，如果你想试试另外一种脚本型的也是比较主流的编程语言，那么可以搞搞 PHP，它也是很快就可以上手的语言。学习 PHP 语言，你可以先跟着 [W3School 的 PHP 教程](#) 玩玩（其中有连接数据

库的 MySQL 的教程)。然后，以 [PHP 的官网文档](#) 作为更全的文档来学习或查找相关的技术细节。

下面是一些学习要点：

学习 HTML 基本语法。

学习 CSS 如何选中 HTML 元素并应用一些基本样式。

学会用 Firefox + Firebug 或 Chrome 查看你觉得很炫的网页结构，并动态修改。

在一台 Linux 机器上配置 LEMP - Ubuntu/Nginx/PHP/MySQL 这个环境。

学习 PHP，让后台 PHP 和前台 HTML 进行数据交互，对服务器相应浏览器请求形成初步认识。实现一个表单提交和反显的功能。

把 PHP 连接本地或者远程数据库 MySQL ( MySQL 和 SQL 现学现用够了 )。

这里，你可能会问我，入门时有三个后端语言，一个是 Python，一个是 Node.js，一个是 PHP，你对这三门语言怎么看？老实说，Python 我还看好一些，PHP 次之，Node.js 最后。原因是：

Python 语言的应用面还是很广的。（当然，性能可能会有一些问题，但是用于一些性能不敏感的和运维或是一些小工具相关的，还是非常好用的。另外，Python 的应用场景其实还是很多的，包括机器学习和 AI 也有 Python 的身影。用 Python 来做一些爬虫、简单的中间件、应用或是业务服务也是很不错的。）

PHP 也是一个比较主流的简单的语言（PHP 在目前来说还是一个比较主流的语言，但其发展潜力有限，虽然可以让你找得到工作，但是一般玩玩就行了）。

Node.js 号称 JavaScript 的后端版，但从目前发展来说，在后端的世界里，并不能承担大任，而且问题很多。一些前端程序员用它来做后端的粘合层，我个人觉得这样做只是掩盖前后端配合有问题，或是接口设计思维上的懒惰，我还是希望前端程序员应该认真学习一门真正的后端语言。

当然，这里只是让你感觉一下，**Web 前端编程的感觉，只是为了入门而已。所以，对于这些语言你也不用学得特别精通，感觉一下这几个不同的语言就可以了，然后知道相关的文档和知识在哪里，这样有助于你查阅相应的知识点。**

## 实践项目

无论你用 Python，还是 Node.js，还是 PHP，我希望能做一个非常简单的 Blog 系统，或是 BBS 系统，需要支持如下功能：

用户登录和注册（不需密码找回）。

用户发贴（不需要支持富文本，只需要支持纯文本）。

用户评论（不需要支持富文本，只需要支持纯文本）。

你需要从前端一直做到后端，也就是说，从 HTML/CSS/JavaScript，到后面的 PHP（Python/Node.js），再到数据库。这其中需要你查阅很多的知识。

这里有几个技术点你需要关注一下。

1. 用户登录时的密码不应该保存为明文，应该用 MD5+Salt 来保存（关于这个是什么，希望你能自行 Google）。
2. 用户登录后，对于用户自己的贴子可以有“重新编辑”或“删除”的功能，但是无权编辑或删除其它用户的贴子。
3. 数据库的设计，你需要三张表：用户表、文章表和评论表，它们之间是怎么关联的，你需要学习一下。这里有个 PHP 的 blog 教你怎么建表，你可以 [前往一读](#)。

如果你有兴趣，你可以顺着这个小项目，研究一下下面这几个事。

图片验证码。

上传图片。

阻止用户在发文章或评论时输入带 HTML 或 JavaScript 的内容。

防范 SQL 注入。参看[PHP 官方文档](#)或[微软官方文档](#)，或者你自己 Google 一下。

上面这些东西，不是什么高深的东西，但是可以让你从中学到很多。相信你只需要自己 Google 一下就能搞定。

## 小结

接下来，我总结下今天的内容。首先，我推荐了 Python 和 JavaScript 作为入门语言，让你尽快上手，获得成就感，从而激发你想持续学习的热情。随后介绍了 Linux 操作系

统、Visual Studio Code 编程工具、Web 编程入门等方面的学习资料，并给出了学习要点。最后，我给出了一个实践项目，帮助你理解和巩固今天所学的内容。

消化好了今天的内容，就准备好精力迎接后面的挑战吧。下篇文章中，我们将正式入门学习该如何编程。

下面是《程序员练级攻略》系列文章的目录。

## [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

#### [软件设计](#)

### 高手成长篇

#### [Linux 系统、内存和网络（系统底层知识）](#)

#### [异步 I/O 模型和 Lock-Free 编程（系统底层知识）](#)

#### [Java 底层知识](#)

#### [数据库](#)

#### [分布式架构入门（分布式架构）](#)

#### [分布式架构经典图书和论文（分布式架构）](#)

#### [分布式架构工程设计（分布式架构）](#)



[微服务](#)

[容器化和自动化运维](#)

[机器学习和人工智能](#)

[前端基础和底层原理（前端方向）](#)

[前端性能优化和框架（前端方向）](#)

[UI/UX 设计（前端方向）](#)

[技术资源集散地](#)



# 左耳朵耗子

全年独家专栏《左耳听风》

20000 名程序员的练级攻略

陈皓

资深技术专家  
骨灰级程序员



新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 69 | 程序员练级攻略：开篇词

下一篇 71 | 程序员练级攻略：正式入门

## 精选留言 (55)

写留言



star001007

2018-05-31

52

我大PHP竟然就这么算了

展开 ∨



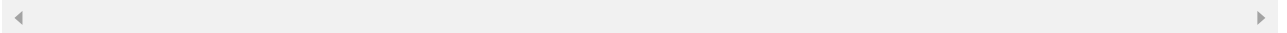
段誉

2018-05-31

👍 19

攻略很实用，基础很重要，在人工智能诱惑下，不知道多少人还在坚持做一些基础的东西。

作者回复: 花2-3年打实基础，坚持学习基础原理，别看他们一开始跑的快，但是没有后劲，这跟长跑和登山一样，后半程随便秒他们。



高鸿祥

2018-05-31

👍 16

零基础自学，学三门语言，一个操作系统，一个数据库。确定不会凌乱吗？确定不是从入门到放弃？



蛮骨

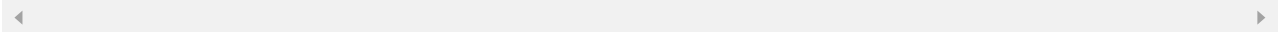
2018-05-31

👍 12

说实话，我一直以为nodejs会比python强点 🤖

展开 ∨

作者回复: 这是幻觉



lianleven

2018-05-31

👍 9

第一篇下来，感觉用下班后的时间就要全力学习一个月

展开 ∨



echo

2018-06-02

👍 8

有一种毕业九年，还没启蒙的感觉 🤖 🤖 回想起来如果早点能接触到皓哥2011年写的攻



略，这些年也会有个努力的方向。从启蒙开始，加油！

展开 ▾



杨雪峰

2018-05-31

👍 8

用go替换java可以吗？

展开 ▾

作者回复: 用go可以替C/C++，Java写业务代码，目前还取代不掉，除非业务简单。



ASCE1885

2018-05-31

👍 7

现在的计算机专业应该不会学pascal了吧 😊

展开 ▾



叶小落

2018-06-14

👍 6

不太赞同耗子叔对node的理解，node还是有他存在的价值的。

node的定位不在于传统定位的后端，应该定位在于前后端分离，微服务架构中的消费者，通过node使后端关注抽象的业务数据，前端关注展示数据。并且node在前端的工程化起到了巨大作用。

从技术的角度看，Nodejs 也并非一点也不适合纯后端的场景，只是node使用者大部分...

展开 ▾

作者回复: 有价值是一回事，有多少价值是另一回事。一条内裤一张卫生纸都有价值。

但是，在当今工业界内，严谨的工业级的架构中，后端从来没有见过Node.js，这种语言根本不适合后端架构（尤其是分布式架构）。我的经验来自于事实和数据，这是一篇严肃的不能误人的文章，我不能把Node.js说成后端语言。对不起！



海川

2018-05-31

👍 6

正在做个宠物领养的公益小程序，从前到后，Python+flask，可以当做入门作业了



Robert

2018-06-17

👍 5

很喜欢你的文章，但是作为前端，并不能认同你对node 的看法。  
你说 node 是前端 yy 后端的 产物。  
强调并请纠正：node 不是yy后端的产物，而是主动承担的产物。请纠正与正视。

当然，承认后端在整个产品中的重要地位。对整个互联网的环境、知识体系认知的全面...  
展开 ∨

作者回复: 谢谢回复。我思考了一下，我觉的你说的对，我表达不当，向你道歉。我修改一下：“Node.js 号称 JavaScript 的后端版，但从目前发展来说，在后端的世界里，并不能承担大任，而且问题很多。一些前端程序员用它来做后端的粘合层，我个人觉得这样做只是掩盖前后端配合有问题，或是接口设计思维上的懒惰，我还是希望前端程序员应该认真学习一门真正的后端语言。”



【技术人成...

2018-05-31

👍 5

皓哥，我在一个小城市做开发，看到这边有一些公司全部用ASP .NET开发，然后每接到一个项目，开发完就买一台虚拟机部署。这种做法是不是太老旧了？我觉得应该有一个统一的后端架构，每当有一个新的项目，也是往这个架构里塞，就类似于云平台的多租户一样，隔离就好了，这样能复用的东西可能多一些，运维也统一起来。又结合现在的很多容器服务，我甚至觉得用Docker部署应用的话，一般中小型企业，每个公司服务器端只需...  
展开 ∨

作者回复: 当然，你比他们牛多了！很多很多！你有架构师的潜质！



icy

2018-08-15

👍 4

买了《父与子的编程之旅》，简单翻了一下，比较适合小朋友看。  
有需要的朋友，出邮费，可以寄给你。

展开 ∨



没伞的孩子...  
2018-06-13

👍 4

没有运维选手吗？py已上手，准备入坑go,看耗子叔推荐的go，准备开始喽。

---



刘-阿-伟  
2018-05-31

👍 4

已经工作三年，工作中不会涉及到前端的东西，要学？

展开 ▾

---



junty  
2018-07-05

👍 3

不建议看 W3school 里的php教程，里面连接数据库 MySQL的教程已过时，看菜鸟runoob 的 php 比较好。

展开 ▾

---



lovedebug  
2018-06-01

👍 3

Node创始人都搞了二胎。用node做了几个项目后的感受是：不想用第二次🤔，不适合团队开发

---



kite  
2018-06-14

👍 2

我对nodejs的印象就是充满了各种bug，耗子哥对nodejs的评价我很认可，并且觉得typescript可能会有更多可能

展开 ▾

---



yao  
2018-07-10

👍 1

从学习ruby作为启蒙，现在正在死磕java。谢谢您的分享，我会把这个攻略作为我成为牛逼程序员的路径规划。

展开 ▾

---



小星星  
2018-06-19

👍 1

路漫漫其修远兮，吾将上下而求索。

展开 ∨