

如何成为一名优秀的 **web** 前端工程师（前端攻城师）？

程序设计之道无远弗届，御晨风而返。—— 杰佛瑞 · 詹姆士

我所遇到的前端程序员分两种：

第一种一直在问：如何学习前端？

第二种总说：前端很简单，就那么一点东西。

我从没有听到有人问：如何做一名优秀、甚至卓越的 **WEB** 前端工程师。

何为：前端工程师？

前端工程师，也叫 **Web** 前端开发工程师。他是随着 **web** 发展，细分出来的行业。

Web 前端开发技术主要包括三个要素：**HTML**、**CSS** 和 **JavaScript**！

它要求前端开发工程师不仅要掌握基本的 **Web** 前端开发技术，网站性能优化、**SEO** 和服务端的基础知识，而且要学会运用各种工具进行辅助开发以及理论层面的知识，包括代码的可维护性、组件的易用性、分层语义模板和浏览器分级支持等。

随着近两三年来 **RIA**(Rich Internet Applications 的缩写，中文含义为：丰富的因特网应用程序)的流行和普及带来的诸如：**Flash/Flex**，**Silverlight**、**XML** 和服务端语言(**PHP**、**ASP.NET**，**JSP**、**Python**)等语言，前端开发工程师也需要掌握。

前端开发的入门门槛其实很低，与服务器端语言先慢后快的学习曲线相比，前端开发的学习曲线是先快后慢。

HTML 甚至不是一门语言，他仅仅是简单的标记语言！

CSS 只是无类型的样式修饰语言。当然可以勉强算作弱类型语言。

JavaScript 的基础部分相对来说不难，入手还算快。

也正因为如此，前端开发领域有很多自学成“才”的同行，但大多数人都停留在会用的阶段，因为后面的学习曲线越来越陡峭，每前进一步都很难。

Web 前端技术有一些江湖气，知识点过于琐碎，技术价值观的博弈也难分伯仲，即全局的系统的知识结构并未成体系，这些因素也客观上影响了“正统”前端技术的沉淀！而且各种“奇技淫巧”被滥用，前端技术知识的传承也过于泛泛，新人难看清时局把握主次。因此，前端技术领域，为自己觅得一个靠谱的师兄，重要性要盖过项目、团队、公司、甚至薪水。

另一方面，正如前面所说，前端开发是个非常新的职业，对一些规范和最佳实践的研究都处于探索阶段。

总有新的灵感和技术不时闪现出来，例如 CSS sprite、负边距布局、栅格布局等；

各种 JavaScript 框架层出不穷，为整个前端开发领域注入了巨大的活力；

浏览器大战也越来越白热化，跨浏览器兼容方案依然是五花八门。

为了满足“高可维护性”的需要，需要更深入、更系统地去掌握前端知识，这样才可能创建一个好的前端架构，保证代码的质量。

随着手持设备的迅猛发展，带动了 HTML5 行业标准的快速发展。web 领域的技术，大概有 10 年都没有大的更新了！

现在市场很需要优秀的、高级的前端工程师。

一方面是因为这是一个比较新的细分行业，而且前端程序员大都自学一部分，知识结构不系统；另一方面，大学里面没有这种课程，最重要的是：北大青鸟这类培训机构也没有专门的前端工程师的培训课程！！

吴亮在《JavaScript 王者归来》第一张的序里面说：大多数程序员认为 Javascript 过于简陋，只适合一些网页上面花哨的表现，所以不愿花费精力去学习，或者以为不学习就能掌握。

实际上，一门语言是否脚本语言，往往是她的设计目标决定，简单与复杂并不是区分脚本语言和非脚本语言的标准。

事实上，在脚本语言里面，**Javascript** 属于相当复杂的一门语言，他的复杂度即使放在非脚本语言中来衡量，也是一门相当复杂的语言！

Javascript 的复杂度不逊色于 Perl 和 Python！

如何学习前端知识？

我们生活在一个充满规则的宇宙里面。社会秩序按照规则运行，计算机语言几乎全部是规则的集合。计算机前辈们定义规则，规则约束我们，我们用规则控制数据。**大部分时候，对数据的合理控制，来自于你对规则的掌握。**

学习 **HTML, CSS** 应该先跟着书仔细、扎实的学一遍。然后就需要做大量的练习，做各种常规的、奇怪的、大量的布局练习来巩固、理解自己的知识。

而学习 **Javascript** 首先要知道这门语言可以做什么，不能做什么，擅长做什么，不擅长做什么！

如果你只想当一个普通的前端程序员，你只需要记住大部分 **Javascript** 函数，做一些练习就可以了。

如果你想当深入了解 **Javascript**，你需要了解 **Javascript** 的原理，机制。需要知道他们的本源，需要深刻了解 **Javascript** 基于对象的本质。

还需要 深刻了解 浏览器宿主 下 的 **Javascript** 的行为、特性。

因为历史原因，**Javascript** 一直不被重视，有点像被收养的一般！所以他有很多缺点，各个宿主环境下的行为不统一、内存溢出问题、执行效率低下等问题。

作为一个优秀的前端工程师还需要深入了解、以及学会处理 **Javascript** 的这些缺陷。

那么一名优秀的、甚至卓越的前端开发工程师的具备什么条件？

首先，优秀的 **Web** 前端开发工程师要在知识体系上既要有广度和深度！做到这两点，其实很难。所以很多大公司即使出高薪也很难招聘到理想的前端开发工程师。技术非黑即白，只有对和错，而技巧则见仁见智。

在以前，会一点 **Photoshop** 和 **Dreamweaver** 的操作，就可以制作网页。

现在，只掌握这些已经远远不够了。无论是开发难度上，还是开发方式上，现在的网页制作都更接近传统的网站后台开发，所以现在不再叫网页制作，而是叫 **Web** 前端开发。

Web 前端开发在产品开发环节中的作用变得越来越重要，而且需要专业的前端工程师才能做好。

Web 前端开发是一项很特殊的工作，涵盖的知识面非常广，既有具体的技术，又有抽象的理念。简单地说，它的主要职能就是把网站的界面更好地呈现给用户。

其次，优秀的 **Web** 前端开发工程师应该具备快速学习能力。**Web** 发展的很快，甚至可以说这些技术几乎每天都在变化！如果没有快速学习能力，就跟不上 **Web** 发展的步伐。前端工程师必须不断提升自己，不断学习新技术、新模式；仅仅依靠今天的知识无法适应未来。**Web** 的明天与今天必将有天壤之别，而前端工程师的工作就是要搞清楚如何通过自己的 **Web** 应用程序来体现这种翻天覆地的变化。

说到这里，我想起了一个大师说过的一句话：对于新手来说，新技术就是新技术。

对于一个高手来说，新技术不过是就技术的延伸。

再者，优秀的前端工程师需要具备良好的沟通能力，因为前端工程师至少都要满足四类客户的需求。

1、产品经理。这些是负责策划应用程序的一群人。他们会想出很多新鲜的、奇怪的、甚至是不可实现的应用。一般来说，产品经理都追求丰富的功能。

2、UI 设计师。这些人负责应用程序的视觉设计和交互模拟。他们关心的是用户对什么敏感、交互的一贯性以及整体的好用性。一般来说，**UI** 设计师于流畅靓丽、但并不容易实现的用户界面，而且他们经常不满前端工程师造成 **1px** 的误差。

3、项目经理。这些人负责实际地运行和维护应用程序。项目管理的主要关注点，无外乎正常运行时间、应用程序始终正常可用的时间、性能和截止日期。项目经理追求的目标往往是尽量保持事情的简单化，以及不在升级更新时引入新问题。

4、**最终用户**。指的是应用程序的主要消费者。尽管前端工程师不会经常与最终用户打交道，但他们的反馈意见至关重要。最终用户要求最多的就是对个人有用的功能，以及竞争性产品所具备的功能。

Yahoo 公司，YUI 的开发工程师 Nicholas C. Zakas 认为：

前端工程师是计算机科学职业领域中最复杂的一个工种。绝大多数传统的编程思想已经不适用了，为了在多种平台中使用，多种技术都借鉴了大量软科学的知识 and 理念。成为优秀前端工程师所要具备的专业技术，涉及到广阔而复杂的领域，这些领域又会因为你最终必须服务的各方的介入而变得更加复杂。专业技术可能会引领你进入成为前端工程师的大门，但只有运用该技术创造的应用程序以及你跟他人并肩协同的能力，才会真正让你变得优秀。

关于书籍：

HTML、CSS 类别书籍，都是大同小异，在当当网、卓越网搜索一下很多推荐。如果感觉学的差不多了，可以关注一下《CSS 禅意花园》，这个很有影响力。

Javascript 的书籍 推荐看老外写的，国内很多 Javascript 书籍的作者对 Javascript 语言了解的都不是很透彻。

这里推荐几本 Javascript 书籍：

初级读物：

《**JavaScript 高级程序设计**》：一本非常完整的经典入门书籍，被誉为 JavaScript 圣经之一，详解的非常详细，最新版第三版已经发布了，建议购买。

《**JavaScript 王者归来**》百度的一个 Web 开发项目经理写的，作为初学者准备的入门级教程也不错。

中级读物：

《**JavaScript 权威指南**》：另外一本 JavaScript 圣经，讲解的也非常详细，属于中级读物，建议购买。

《**JavaScript.The.Good.Parts**》：Yahoo 大牛，JavaScript 精神领袖 Douglas Crockford 的大作，虽然才 100 多页，但是字字珠玑啊！强烈建议阅读。

《**高性能 JavaScript**》：《JavaScript 高级程序设计》作者 Nicholas C. Zakas 的又一大作。

《**Eloquent JavaScript**》：这本书才 200 多页，非常短小，通过几个非常经典的例子(艾米丽姨妈的猫、悲惨的隐士、模拟生态圈、推箱子游戏等等)来介绍 JavaScript 方方面面的知识和应用方法。

高级读物：

《**JavaScript Patterns**》：书中介绍到了各种经典的模式，如构造函数、单例、工厂等等，值得学习。

《**Pro.JavaScript.Design.Patterns**》：Apress 出版社讲解 JavaScript 设计模式的书，非常不错。

《**Developing JavaScript Web Applications**》：构建富应用的好书，针对 MVC 模式有较为深入的讲解，同时也对一些流程的库进行了讲解。

《**Developing Large Web Applications**》：不仅有 JavaScript 方面的介绍，还有 CSS、HTML 方面的介绍，但是介绍的内容却都非常不错，真正考虑到了一个大型的 Web 程序下，如何进行 JavaScript 架构设计，值得一读。

要做优秀的前端工程师，还需要继续努力：《**高性能网站建设指南**》、《**Yahoo 工程师的网站性能优化的建议**》、**"YSLOW"**性能优化建议、《**网站重构**》、《**Web 开发敏捷之道**》、**"jQuery 库"**、**"前端框架"**、**"HTML5"**、**"CSS3"**。。。这些都要深入研究！

万事开头难！如果你能到这个境界，剩下的路自己就可以走下去了。

人们常说：不想当裁缝的司机，不是个好厨师。

如果单纯只是学习前端编程语言、而不懂后端编程语言(PHP、ASP.NET, JSP、Python)，也不能算作是优秀的前端工程师。

在成为一个优秀的前端工程师的道路上，充满了汗水和辛劳。

—— 王子墨 写于 Hong Kong,Kowloon

2012 年 10 月 28 日 20:18

本文用到的参考资料:

- 1、[What makes a good front end engineer?](#)
- 2、[Best Practices for Speeding Up Your Web Site](#)

建议继续学习:

1. [如何成为一名优秀的前端工程师](#) (阅读: 4617)
2. [Web 前端工程师编程能力飞升之路](#) (阅读: 4365)
3. [也谈谈前端，架构，框架与库](#) (阅读: 2950)
4. [Nicholas C. Zakas 谈怎样才能成为优秀的前端工程师](#) (阅读: 2588)
5. [前端性能优化的方向](#) (阅读: 2273)
6. [淘宝网：前端安全须知](#) (阅读: 2227)
7. [2009 年前端技术领域回顾](#) (阅读: 2215)
8. [我做前端一年半](#) (阅读: 2085)
9. [前端的横向发展](#) (阅读: 1853)
10. [用于前端的模板引擎](#) (阅读: 1814)