

《一路编程》

图书基本信息

书名：《一路编程》

13位ISBN编号：9787121304783

出版时间：2017-1-1

作者：史蒂夫·富特 (Steven Foote)

页数：271

译者：佟达

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《一路编程》

内容概要

《一路编程》是一本编程入门书籍，然而，如果以书中所讲内容作为入门标准，估计十有八九的在职程序员都不能算已入门。现代软件开发，已经不仅仅是写出正确的代码这么简单，环境、依赖、构建、版本、测试及文档，每一项都对软件是否成功交付起到至关重要的作用，这些都是每一个程序员在开发软件过程中必备的技能。《一路编程》对于上述的每一种技能都做了简洁而精练的介绍，以满足最基本的日常软件开发。换句话说，《一路编程》实际上是为现代软件开发的入门设下了最基本的门槛。相信每一位本书的读者，不论是即将进入软件行业，还是已经在软件行业工作多年，都会获得收获。

强烈推荐刚刚或将要成为程序员的人及其朋友们阅读《一路编程》。

书籍目录

1 “Hello, World” 写下第一个程序	1
选择文本编辑器	1
核心功能	2
做出你的选择	4
Sublime Text	5
TextMate	5
Notepad++	5
Gedit	6
Vim	6
Eclipse	6
IntelliJ	7
Xcode	7
Visual Studio	7
创建项目目录	8
从小处着手：创建测试文件	8
HTML和JavaScript如何在浏览器中一起工作	10
小幅修改的意义	11
乘胜追击	13
在manifest.json中引用JavaScript	16
让它运行起来	17
能力越大，责任越大	18
总结	18
2 软件如何工作	19
什么是“软件”	

19	
软件生命周期	
20	
源代码——一切开始的地方	
21	
一组指令	
21	
编程语言	
22	
从源代码到0和1	
27	
编译型语言与解释型语言：源代码何时变成二进制码	
27	
运行时环境	
28	
处理器执行	
29	
输入和输出	
29	
输入让软件更实用（可重用）	
30	
输入从哪来	
31	
软件如何获得输入	
32	
输出类型	
32	
GIGO：垃圾进，垃圾出（Garbage In，Garbage Out）	
33	
状态	
34	
给kittenbook添加状态	
35	
内存和变量	
37	
变量	
37	
变量存储	
38	
有限的资源	
41	
内存泄漏	
41	
总结	
42	
3 认识你的计算机	
43	
计算机很笨	
43	

计算机有魔力	44
站在巨人的肩膀上	44
计算机内部	44
处理器	44
短期存储器	45
长期存储器	45
使用计算机	46
文件系统	46
命令行：取得控制权	48
总结	58
4 构建工具	59
（几乎）全部自动化	59
安装Node	60
安装Grunt	62
帮你创造软件的软件	65
避免错误	66
更快地工作	66
自动化的任务	67
编译	67
测试	68
打包	68
部署	68
构建你自己的构建过程	69
Gruntfile.js	69
使用Grunt插件	

69	
加载Grunt插件	
72	
注册任务	
73	
看好了	
74	
总结	
77	
5 数据（类型）、数据（结构）、数据（库）	
79	
数据类型	
79	
为什么存在不同数据类型	
80	
基本数据类型	
80	
组合数据类型	
85	
动态和静态类型语言	
92	
数据结构	
93	
集合	
96	
栈	
96	
树	
97	
图	
98	
如何选择高效的数据结构	
101	
数据库	
101	
长期（持久化）存储	
101	
关系型数据库	
101	
SQL简介	
103	
总结	
105	
6 正则表达式	
107	
Ctrl+F组合键：寻找模式	
107	
在JavaScript中使用正则表达式	
108	

重复
109
?
109
+
110
*
110
特殊字符和转义字符
111
{1, 10} : 创造属于你的超能力
111
匹配任意字符的 “.”
112
不要太贪婪
112
从[A-Za-z]理解方括号
113
字符列表
113
范围
114
排除
114
电话号码模式
115
我需要\s
117
方括号的快捷方式
118
限制条件
119
提取标签
123
高级查找和替换
124
(一行的) 开头和结尾
124
标记
125
全局匹配
125
忽略大小写
125
多行
125
什么时候会用到正则表达式
125
grep

125	
代码重构	
126	
校验	
127	
数据抽取	
127	
总结	
127	
7 何时使用if、for、while	
129	
操作符	
129	
比较操作符	
129	
逻辑操作符	
130	
一元操作符	
132	
二元操作符	
132	
三元操作符	
135	
“真”和“假”	
137	
“语法糖”	
139	
循环遍历一个数组	
140	
遍历图片	
140	
嵌套循环	
141	
你需要停下来	
142	
无限循环	
144	
再停一下	
145	
当你不知道什么时候停下	
145	
何时执行	
145	
事件	
145	
监听器	
146	
定时任务	
147	

超时	
147	
在事情出错前接住它	
148	
编写健壮的代码	
149	
总结	
149	
8 函数和方法	
151	
函数结构	
151	
定义	
152	
调用	
152	
参数	
153	
调用栈	
155	
代码封装	
156	
一次做好一件事	
156	
分而治之	
157	
物尽其用	
161	
代码重用	
161	
解决通用问题	
161	
用更少的代码做更多的事情	
161	
不要做重复的事 (DRY)	
163	
作用域	
164	
全局变量	
166	
本地变量	
166	
变量查找是怎么工作的	
167	
总结	
170	
9 编程标准	
171	
编码惯例	

171	
设定标准	
172	
黑科技，用还是不用	
172	
立即付款还是先用后付款	
173	
写可维护的代码	
173	
代码格式化	
174	
保持一致	
175	
空白字符	
176	
规则不会自己出现：要制定规则	
177	
使用其他人的成果	
179	
更快地构建	
179	
开源软件	
179	
由社区建立	
180	
什么时候该自己写	
180	
最佳实践	
181	
文档	
181	
计划	
181	
测试	
181	
总结	
182	
10 文档	
183	
文档化意图	
184	
自文档代码	
185	
不要将显而易见的东西写入文档	
187	
过时文档的危险性	
188	
用文档来找bug	
189	

为自己写文档	189
你的记忆力有多好	189
为了学习而记录文档	190
超越注释的文档	190
给别人写的文档	194
记录你的决定	195
记录你的资源	195
为了教学而写文档	196
总结	196
11 计划	197
三思而后行	197
创建规格说明	198
设计架构	198
画示意图	199
尝试破坏你的系统	200
迭代式计划	201
为扩展设计	202
你的优先级是什么	202
用户体验	202
性能	203
安全	203
伸缩性	203
截止日期	204
平衡的艺术	204
识别并创建限制条件	

204	知道可以做什么，不可以做什么
204	总结
206	12 测试和调试
207	手工测试
207	边做边测
208	尝试些疯狂的事
208	吃你自己的狗粮
209	自动化测试
209	单元测试
210	给Kittenbook配置测试
213	失败时代
217	间谍喜欢我们（我们也喜欢间谍）
218	集成测试
221	尽早发现问题
221	调试
222	错误
222	日志
223	断点
224	查看、监控和控制台
228	单步执行代码
229	调用栈
231	找到根本原因
231	编码、测试、调试、不断重复
231	总结
232	

13 授人以渔：如何用一生学习编程

233

如何搜索

233

找到正确的关键字

234

以终为始

236

识别高质量资源

236

个人博客：隐藏的宝藏

237

什么地方、什么时候，以及怎么问编程问题

237

什么地方

237

什么时候

240

怎么问

241

通过教别人来学习

241

总结

242

14 构建你的技能

243

做你自己的kittenbook

243

给Facebook重新设计风格

243

添加新功能

244

分享你的kittenbook专属版本

245

找到你自己的项目

245

解决你自己的问题

246

志存高远

246

获得帮助，提供帮助

247

开源项目

247

GitHub

247

找项目

248

贡献的不同方式

248	
创建你自己的项目	
249	
免费在线教育	
249	
欧拉项目	
249	
Udacity	
250	
Coursera	
250	
codeacademy	
251	
Khan Academy (可汗学院)	
251	
教程	
251	
付费教育	
251	
读书	
252	
Udacity和Coursera	
252	
Treehouse	
253	
总结	
253	
15 高级主题	
255	
版本控制	
255	
为什么使用版本控制	
256	
和团队一起工作	
257	
Subversion	
260	
Git	
260	
OOP (面向对象编程)	
266	
类	
266	
继承	
267	
实例	
268	
设计模式	
268	

发布订阅

268

中间人

269

单例

270

总结

270

精彩短评

- 1、口水书，还是少费钱在这类书上吧
- 2、用JavaScript 讲编程知识，可能适合野生前端？但其实很多知识只能意会无法言传。
- 3、Steven从一个麻瓜的角度带入，给还没有进入编程世界的新人讲述每个看似毫无头绪但程序员习以为常的日常，本书以JavaScript编程为例，讲述了环境、依赖、构建、版本、测试及文档的配置及其重要性，给读者构造了一个程序开发的大概轮廓。授人以渔，书后部分的资源也相当受用，推荐给编程新人和自觉还没入门的人~

1、一上来就看到某人说：广度有余，深度不足。我倒不这么认为。窃以为，若说要有深度，只要上书店看看就知道，专业书籍琳琅满目，选择余地很大，有足够多的、复杂的专业书籍供你选择。这说明什么呢？恰恰说明我们的缺少的就是：这种从细节上提示你还有多少广度可以抬头看到。一个劲的在深度上专研，未必能找到最适合自己的捷径，这就好比，一个新手进了大区啥都不知道随便找了一个方向死命地加点，等到哪天停下了的时候才发现除了这一条道之外还有更适合自己的更轻松的路径，可是因为已经付出的太多，洗点代价太高，无法回头了……这本书就是告诉你，在走得太深之前，先多环视四周，看看哪里的哪条路最适合自己的，看看知识点如何合理有效的点在最需要的地方才更事半功倍。对于新手来说，如果在阅读这本书的时候，遇到某些专业术语，他觉得太过深奥，但是恰好告诉了他，哦，原来我这里还有欠缺，我知道了我需要再去补充哪方面的知识。对于高手而言，也是告诉他，除了他走的这条漫长深远的道路之外，还有什么开阔的空间或者便捷的小路可以包抄，这些都是这本书对于新人和老人不同的帮助，一本书的好坏，也取决于阅读ta的人如何寻找到其中帮助到自己的那个点。就好像一艘货轮上，装满了各式各样大家需要的物资，有的人需要食物，有的人需要衣服，有的人需要工具，可是装船的人不知道，他们只会把船尽可能装的慢慢的，船到岸之后，让需要的人自己去寻找自己的货物，总不至于为了一百种不同的需求，准备一百条不同的小船吧。且说若是新人一枚，要是看一本书遇到部分专业名词或者属于都不知道自己如何通过其他渠道寻找到答案，还是趁早放弃学习编程吧。且说若是老人一枚，若是自身都不能主动区分专业教课书籍和普通辅助书籍之间的差别，并把它们放在最合适的位置，而在这里指责书籍没有全按照个人的需要心思去写，我想说，也许你需要的不是一本书，而是一个专业的全面的个人导师，当然这样一位导师的价格绝对就不是65块钱那么便宜了。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com