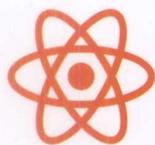


## 版权注意事项：

- 1、书籍版权归作者和出版社所有
- 2、本PDF仅限用于个人获取知识，进行私底下的知识交流
- 3、PDF获得者不得在互联网上以任何目的进行传播
- 4、如觉得书籍内容很赞，请购买正版实体书，支持作者
- 5、请于下载PDF后24小时内删除本PDF。



JavaScript ES6

控件应用

布局技术

网络技术

导航栈技术

项目实战

# React Native 全教程

## 移动端跨平台应用开发

• 张益琿 编著 •

实战React Native



清华大学出版社

## / 作者介绍 /



### 张益晖

(网名: 琿少)

工学学士，多年iOS开发经验，开发过iOS平台系列游戏疯狂越狱1-2、应用物通配货软件、VIPExam考试库、证券财经软件等，现就职于中国唯品会，主要从事移动端应用开发，对iOS开发和React Native跨平台开发拥有丰富经验，曾出版《iOS开发实战：从零基础到App Store上架》（清华大学出版社）。



## 内 容 简 介

本书由经验丰富的移动开发工程师精心编撰,全书从逻辑上可分为5个部分,循序渐进地向读者展示使用 React Native 开发跨平台移动应用的全流程,第1部分介绍 React Native 语言基础 JavaScript;第2部分介绍大量使用于 React Native 开发中的 ECMAScript 6 的新特性;第3部分介绍 React Native 开发技巧,包括独立组件应用、布局技术、网络技术、导航栈技术等;第4部分通过3个实战项目手把手地教读者开发完整的 React Native 应用;第5部分介绍 React Native 的一些高级技巧,比如和原生交互、嵌入原生应用、React Native 组件开发等。本书特别对 React Native 在开发 iOS 和 Android 跨平台应用时给出范例效果对比演示,现场感十足。

本书既适合想快速上手 React Native 的初学者、有 Android 和 iOS 开发基础想构建跨平台移动应用的开发人员使用,也可用作培训机构和大中专院校的教学参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

React Native 全教程:移动端跨平台应用开发/张益琿编著. —北京:清华大学出版社,2018  
ISBN 978-7-302-49813-1

I. ①R… II. ①张… III. ①移动终端—应用程序—程序设计—教材 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2018)第 037615 号

责任编辑:王金柱

封面设计:王翔

责任校对:闫秀华

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者:北京泽宇印刷有限公司

经 销:全国新华书店

开 本:190mm×260mm

印 张:25.5

字 数:653千字

版 次:2018年4月第1版

印 次:2018年4月第1次印刷

印 数:1~3000

定 价:79.00元

产品编号:075000-01

# 前 言

首先，笔者十分荣幸也十分高兴你选择本书来学习 React Native 的开发。由于笔者的个人能力有限，这本书可能并不是最完美最优秀的，但是相信无论你的基础如何，都可以随着本书的学习与练习，完完整整地独立开发出自己的 React Native 应用，因为笔者也是这么学过来的。因此，将本书定义为教程，不如称其为一本经验与总结的笔记，相信你在学习的过程中会深有体会。

说起来，JavaScript 的语法并不是这本书的核心，但是学习一种功夫之前，一定要有一把顺手的武器，因为依然有很多初学者或原生开发者对 JavaScript 语言的了解并不深入，所以本书的前 4 章着重对 JavaScript 语法以及 ES6 的新特性进行介绍，帮助你为后边的学习扫除基础障碍。

学习客户端编程，最重要的莫属界面、数据、逻辑这 3 部分，本书的第 5~8 章将向你介绍 React Native 中的基础界面组件、数据与网络技术以及用户交互管理技术等，学习一门技能就是在完成一张大拼图，每一个知识点都是这张拼图中的一块，学习完这 4 章内容，你将掌握 React Native 开发中所有的基础技能，后面就是对它们的组合和应用了。

本书第 9~11 章提供了 3 个完整的 React Native 实战练习，这 3 章的项目也将由简到难，帮助你熟练应用前面所学习的知识。

本书第 12 章为扩展章节，这一章节将更偏向介绍 React Native 的一些高级开发技术，比如和原生交互、嵌入原生应用、开发 React Native 组件等，如果你有兴趣，可以好好研究一下。

IT 领域日新月异，React Native 是一种移动端跨平台软件开发框架，可能并不是最优秀的，但是 Facebook 的长期维护和它优秀的设计思想无论如何都是值得我们学习的。有人说，每学习一门技术，每次离开自己所擅长的领域走向新的领域学习都是一种重生，这个过程可以让你感受到不同的思维模式，体验到不同圈子的乐趣。和你一样，笔者也是一名学习者，如果你愿意，可以随时和笔者交流，QQ：316045346。

为方便读者上机练习，本书提供了全书实例源代码，下载地址：

<https://pan.baidu.com/s/1msOpjsdGcoSRCN5K4qcTPQ>（注意区分数字和英文字母大小写）

如果你在下载过程中遇到问题，可发邮件至 [booksaga@126.com](mailto:booksaga@126.com)，邮件标题为“React Native 全教程：移动端跨平台应用开发下载资源”，获得帮助。

最后，再次感谢你选择了本书，笔者也真心地希望它可以帮助你到达自己的预定目标。这本书最终能呈现在你的面前，除了笔者的努力，还要感谢支持我的家人和朋友，尤其是王金柱编辑，在写作过程中他给了我巨大的帮助与鼓励。

琿 少

2018年2月

# 目 录

<b>第 1 章 从 JavaScript 开始</b> ..... 1	2.1.2 switch-case 分支结构.....35
1.1 学习环境的配置..... 1	2.2 循环结构.....37
1.1.1 使用浏览器进行 JavaScript 代码的调试..... 1	2.2.1 while 循环结构.....37
1.1.2 使用 Sublime Text 工具来编写 JavaScript 代码..... 3	2.2.2 for 循环结构.....38
1.1.3 安装 Sublime Text 插件 管理器 PackageControl..... 3	2.3 中断与跳转结构.....39
1.1.4 使用 PackageControl 进行 JavaScript 代码智能提示插件的安装..... 5	2.3.1 break 语句.....39
1.1.5 安装 JavaScript 代码格式化 插件..... 7	2.3.2 continue 语句.....41
1.1.6 在 Sublime Text 中运行 JavaScript 代码..... 7	2.4 异常捕获结构.....42
1.2 初识 JavaScript..... 8	2.4.1 使用 throw 语句抛出异常.....43
1.2.1 JavaScript 的语法特点..... 9	2.4.2 异常的捕获与处理.....44
1.2.2 JavaScript 中的变量..... 10	2.4.3 异常的传递.....46
1.3 JavaScript 中的数据类型..... 12	2.5 JavaScript 中的函数.....48
1.3.1 原始类型..... 13	2.5.1 使用函数语句定义函数.....48
1.3.2 引用类型..... 16	2.5.2 使用函数表达式定义函数.....50
1.4 JavaScript 中的运算符..... 18	2.5.3 使用 Function 构造函数.....51
1.4.1 算术运算符..... 18	<b>第 3 章 JavaScript 对象基础</b> ..... 52
1.4.2 赋值运算符..... 21	3.1 初识 JavaScript 对象.....52
1.4.3 关系运算符..... 22	3.1.1 在 JavaScript 中创建对象.....52
1.4.4 逻辑运算符..... 24	3.1.2 设置对象的属性和行为.....54
1.4.5 位运算符..... 26	3.2 JavaScript 中常用的内置对象.....55
1.4.6 特殊运算符..... 30	3.2.1 JavaScript 中的 Number 对象.....55
1.4.7 运算符的优先级与结合性..... 32	3.2.2 JavaScript 中的 String 对象.....57
<b>第 2 章 JavaScript 流程控制与函数</b> ..... 34	3.2.3 JavaScript 中的 Boolean 对象.....59
2.1 条件分支结构.....34	3.2.4 JavaScript 中的 Array 对象.....60
2.1.1 if-else 分支结构..... 34	3.2.5 JavaScript 中的 Date 对象.....64
	3.2.6 JavaScript 中的 Math 对象.....67
	3.2.7 JavaScript 中的 RegExp 对象.....69
	3.2.8 JavaScript 中的 Function 对象.....73
	3.3 深入 JavaScript 中的 Object 对象.....75
	3.3.1 为对象属性进行配置.....76

3.3.2	Object 构造方法对象中的常用函数	77	4.8	使用 class 定义类	119
3.3.3	Object 实例对象中的常用方法	82	4.8.1	使用 class 定义类	119
3.4	面向对象编程技术	82	4.8.2	class 类的继承	120
3.4.1	JavaScript 中模拟类的方式	83	4.9	模块引入	121
3.4.2	在 JavaScript 中实现继承机制	86	4.9.1	export 关键字	121
<b>第 4 章</b>	<b>ECMAScript 6 新特性</b>	<b>91</b>	4.9.2	import 关键字	122
4.1	ECMAScript 6 的块级作用域	91	4.9.3	默认导出与导入	122
4.1.1	let 关键字	92	<b>第 5 章</b>	<b>React Native 开发环境的搭建</b>	<b>124</b>
4.1.2	const 关键字	94	5.1	iOS 开发环境的搭建	124
4.2	解构赋值	95	5.1.1	申请 AppleID 账号	124
4.2.1	数组的解构赋值	95	5.1.2	安装 Xcode 开发工具	125
4.2.2	对象的解构赋值	96	5.2	Android 开发环境的搭建	126
4.2.3	字符串与函数参数的解构赋值	98	5.2.1	下载 Android Studio 开发工具	126
4.3	箭头函数	99	5.2.2	安装相关 SDK 和模拟器	127
4.3.1	箭头函数的基本用法	99	5.3	React Native 开发环境配置	130
4.3.2	箭头函数中 this 的固化	100	5.3.1	安装 React Native 构建环境	130
4.4	Set 与 Map 数据结构	102	5.3.2	运行你的第一个 React Native 应用	131
4.4.1	Set 集合结构	102	<b>第 6 章</b>	<b>React Native 独立组件基础篇</b>	<b>134</b>
4.4.2	Map 字典结构	104	6.1	Text 文本组件的应用	134
4.5	Proxy 代理	106	6.1.1	文字风格设置	134
4.5.1	使用 Proxy 代理对对象的属性读写进行拦截	106	6.1.2	Text 组件属性的设置	138
4.5.2	Proxy 代理处理器支持的拦截操作	108	6.1.3	Text 组件的嵌套	140
4.6	Promise 承诺对象	110	6.1.4	React Native 程序的调试	141
4.6.1	Promise 对象执行异步任务	110	6.2	Button 按钮组件的应用	142
4.6.2	Promise 任务链	112	6.2.1	Button 组件的简单使用	142
4.6.3	Promise 对象组合	113	6.2.2	小应用：屏幕霓虹灯	144
4.7	Generator 生成器与 yield 语句	115	6.3	Image 图像组件的应用	145
4.7.1	Generator 函数应用	115	6.3.1	渲染图像的方式	145
4.7.2	Generator 任务参数的传递	117	6.3.2	Image 组件的风格自定义	148

6.3.3 Image 组件的属性和方法 解析.....	151	6.16 ListView 列表组件的应用.....	193
6.4 Switch 开关组件的应用.....	154	6.16.1 使用 DataSource 渲染 ListView 视图.....	193
6.5 Slider 滑块组件的应用.....	156	6.16.2 ListView 属性方法解析.....	197
6.6 ActivityIndicator 指示器组件的 应用.....	159	6.17 高性能列表组件 FlatList.....	199
6.7 TextInput 用户输入组件的应用.....	160	6.17.1 创建一个简单的 FlatList 列表 视图.....	199
6.8 StatusBar 状态栏组件的应用.....	165	6.17.2 FlatList 中常用方法解析.....	202
6.9 Picker 选择器组件的应用.....	167	6.18 分区列表组件 SectionList 的 应用.....	202
6.10 Modal 模态视图组件的应用.....	169	6.19 RefreshControl 刷新组件的 应用.....	205
6.11 KeyboardAvoidingView 组件的 应用.....	171	<b>第 7 章 React Native 独立组件 高级篇.....</b>	<b>208</b>
6.12 WebView 网页组件的应用.....	174	7.1 时间选择器 DatePickerIOS 组件的 应用.....	208
6.12.1 WebView 常用属性解析.....	174	7.2 DrawerLayoutAndroid 抽屉组件的 应用.....	209
6.12.2 WebView 加载过程监听相关 属性.....	177	7.3 进度条组件的应用.....	211
6.12.3 React Native 与 WebView 交互.....	178	7.3.1 通过文件名分平台加载组件.....	212
6.13 View 视图组件的应用.....	179	7.3.2 ProgressBarAndroid 组件常用 属性.....	213
6.13.1 View 组件 Style 属性的 解析.....	180	7.3.3 ProgressViewIOS 组件常用 属性.....	214
6.13.2 View 组件基础属性的 解析.....	182	7.4 SegmentedControlIOS 组件的 应用.....	214
6.14 Touchable 相关交互组件的 应用.....	183	7.5 Android 平台上的工具条组件.....	215
6.14.1 TouchableWithoutFeedback.....	184	7.6 Navigator 导航控制器.....	218
6.14.2 TouchableOpacity.....	185	7.6.1 Navigator 牛刀小试.....	219
6.14.3 TouchableNativeFeedback.....	186	7.6.2 Navigator 属性配置.....	220
6.14.4 TouchableHighlight.....	188	7.6.3 Navigator 实例方法解析.....	221
6.15 ScrollView 滚动视图组件的 应用.....	189	7.7 iOS 平台的导航控制器 NavigatorIOS 组件.....	222
6.15.1 ScrollView 的基础用法.....	189	7.7.1 使用 NavigatorIOS 组件.....	222
6.15.2 ScrollView 常用属性解析.....	190		
6.15.3 手动设置 ScrollView 组件的 滚动位置.....	192		

7.7.2	NavigatorIOS 属性与方法 解析	225	8.13	进行用户位置获取	266
7.8	标签栏 TabBarIOS 组件	226	8.14	数据持久化技术	267
<b>第 8 章</b>	<b>React Native 技能进阶</b>	<b>230</b>	8.15	剪贴板工具的应用	270
8.1	React Native 布局技术	230	8.16	获取设备网络状态	271
8.1.1	布局中的主轴与次轴	231	8.17	React Native 动画技术	273
8.1.2	精准定义组件的尺寸	234	8.17.1	创建单值驱动的动画	273
8.1.3	相对定位与绝对定位	237	8.17.2	使用 timing 方法执行平滑 过渡动画	275
8.2	React Native 中的颜色定义	240	8.17.3	深入理解 easing	276
8.3	警告弹窗的应用	242	8.17.4	二维动画对象与衰减动画	278
8.3.1	Alert 组件的应用	243	8.17.5	弹簧动画	280
8.3.2	iOS 平台专用警告框 AlertIOS	245	8.17.6	Interpolation 插值动画	281
8.4	ActionSheetIOS 抽屉视图的应用	247	8.17.7	聚合动画值	282
8.4.1	普通功能列表抽屉	247	8.17.8	组合动画	283
8.4.2	分享视图抽屉	248	8.17.9	循环动画	285
8.5	自定义组件的属性与使用 样式表	250	8.17.10	布局动画	286
8.5.1	自定义组件的属性	250	8.17.11	自定义组件动画	287
8.5.2	通过 StyleSheet 样式表定义 组件的风格	251	8.18	调用设备振动模块	288
8.6	Android 平台的时间选择器	252	8.19	封装滑动手势	289
8.7	Android 平台悬浮提示信息 Toast 的 应用	254	8.20	获取屏幕尺寸信息	292
8.8	监听与控制 Android 设备返回键的 行为	255	8.21	特定平台代码	293
8.9	监听程序运行状态	257	8.22	定时器的简单应用	294
8.10	跨平台的分享功能	258	<b>第 9 章</b>	<b>实战项目：汇率转换器</b>	<b>296</b>
8.11	监听键盘事件	260	9.1	搭建汇率转换器项目主界面	297
8.12	React Native 网络技术	262	9.2	显示屏面板的初步开发	299
8.12.1	使用 fetch 方法进行网络 请求	262	9.3	货币类型切换功能开发	302
8.12.2	使用 XMLHttpRequest 进行 网络请求	264	9.4	键盘界面设计	306
			9.5	实现汇率转换器核心功能	310
			<b>第 10 章</b>	<b>实战项目：微信热门精选</b>	<b>315</b>
			10.1	申请免费的 API 服务	315
			10.2	搭建项目网络模块	317
			10.3	搭建文章列表界面	319

10.4	文章目录视图与首页导航栏完善.....	322
10.5	文章详情页面的开发.....	326
10.6	为文章列表页添加下拉刷新与上拉加载更多功能.....	329
<b>第 11 章</b>	<b>实战项目：掌上新闻</b> .....	<b>332</b>
11.1	应用结构搭建.....	332
11.2	完善标题栏组件.....	335
11.3	进行网络模块的开发.....	338
11.4	使用列表展示数据.....	339
11.5	完善新闻目录列表.....	341
11.6	标题栏与页面联动开发与优化加载逻辑.....	344
11.7	使用导航进行页面跳转.....	348
11.8	完善下拉刷新与上拉加载更多功能.....	351
11.9	完善导航栏.....	353
11.10	添加收藏夹功能.....	356
11.11	优化方向与应用图标设置.....	361
<b>第 12 章</b>	<b>React Native 高级技巧</b> .....	<b>363</b>
12.1	直接操作组件的属性.....	363
12.2	对 React Native 版本进行升级.....	365
12.3	React Native 的更多调试技巧.....	366
12.4	React Native 插件开发.....	367
12.4.1	构建 iOS 工程的原生模块.....	367
12.4.2	构建 Android 工程的原生模块.....	371
12.4.3	深入了解原生模块的函数参数.....	373
12.5	封装原生 UI 组件.....	375
12.5.1	封装 iOS 平台的原生 UI 组件.....	375
12.5.2	开发 Android 跑马灯组件.....	382
12.6	在原生工程中嵌入 React Native 模块.....	387
12.6.1	将 iOS 工程的某个模块进行 React Native 化.....	387
12.6.2	将 Android 工程的某个模块进行 React Native 化.....	391
12.7	在真机上运行 React Native 工程.....	397

# 第 1 章

## 从 JavaScript 开始

JavaScript 是一门流行的脚本语言，同时，它也是开发 React Native 跨平台应用的基础语言。本书前几章会安排了较多篇幅来介绍 JavaScript，以帮助读者零门槛地进行 React Native 开发。当然，如果你是熟练的 Web 开发者，相信 JavaScript 对你来说并不能成为障碍，但是依然建议你阅读第 4 章的内容——ES6 的一些新特性，因为 ES 在 React Native 框架中随处可见。

本章，将简单地对 JavaScript 作全面的了解，同时配置好开发环境，以便于开启你的学习之旅。

### 1.1 学习环境的配置

编程是一种手脑结合的工作，我们在学习编程时，动手远比读书重要得多。因此在 JavaScript 语言学习正式开始之前，首先需要搭建完善、方便、易用的开发环境。有了称手的工具，在学习的路上，才能披荆斩棘，一往无前。

#### 1.1.1 使用浏览器进行 JavaScript 代码的调试

JavaScript 语言最初设计时用于作为浏览器的脚本语言，因此所有浏览器都可以对 JavaScript 代码进行解释与运行。MacOS 的 Safari 浏览器、Google 的 Chrome 浏览器都带有强大的开发者工具。以 Chrome 浏览器为例（在百度上搜索 Chrome，可以十分容易地下载到 Chrome 的最新版本），可以借助 HTML 文件来进行 JavaScript 代码的运行，先来编写一段简单的 HTML 代码，如下：

```
<html>
<head>
  <title></title>
```

```
<script type="text/javascript">
  console.log("Hello World");
</script>
</head>
<body>
</body>
</html>
```

用任何文本编辑器来写这段 HTML 代码都没有问题，这里推荐使用 Sublime Text 编辑器——一款十分优秀的跨平台代码编写软件，最后将此 HTML 文件的后缀保存为 html。

使用 Chrome 浏览器打开此 HTML 文件，会发现页面上一片空白，这没什么好奇怪的，这里只是编写了一个空的网页模板，其中嵌入了一句 JavaScript 脚本代码，除此之外，我们什么也没有做，那么如何来验证 JavaScript 代码的执行情况呢？打开 Chrome 浏览器的开发者工具，如图 1-1 所示。

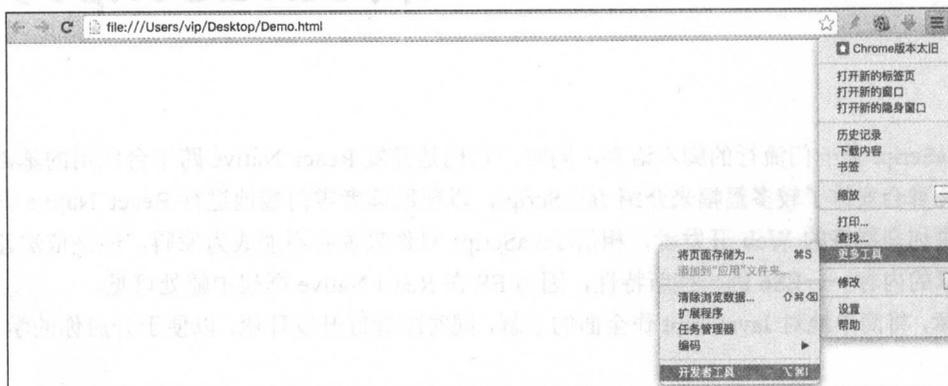


图 1-1 打开 Chrome 浏览器的开发者工具

之后可以看到，界面被分成了 3 部分，如图 1-2 所示。其中，网页区将显示实时的网页效果，功能区可以查看 HTML 标签、所加载的网络数据、加载耗时信息等。打印调试区中可以查看打印信息，也可以进行 JavaScript 代码的调试。如图 1-2 所示，打印区中打印出了我们在 HTML 代码中嵌入的“Hello World”语句，这就是 JavaScript 版的 HelloWorld 工程！

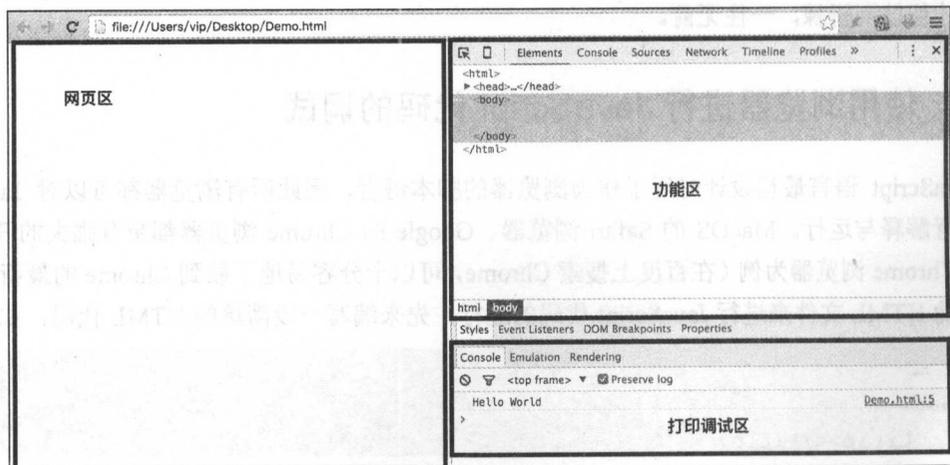


图 1-2 Chrome 浏览器的开发者工具

每次在 HTML 文件中编写一些 JavaScript 测试代码都要重新刷新浏览器，这对开发者来说是一件无法忍受的事情，有时，我们只是想知道某一行代码的执行结果，可以直接在开发者工具的打印调试区进行代码的编写与运行。例如，在其中输入如图 1-3 所示的代码。

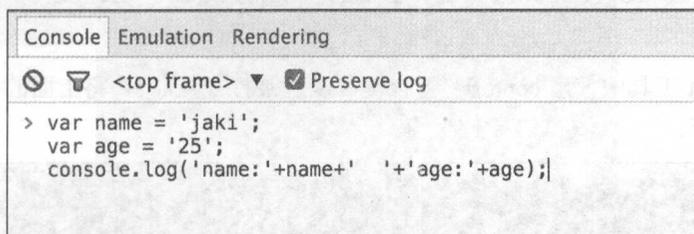


图 1-3 直接在 Chrome 浏览器中进行 JavaScript 代码的调试

### 抽丝剥茧

默认情况下，当在 Chrome 开发工具的调试区输入一行代码后，如果敲击 Enter 键会直接运行此行代码，若需要多行输入，需要使用 Shift+Enter 组合键来进行换行。

## 1.1.2 使用 Sublime Text 工具来编写 JavaScript 代码

Chrome 浏览器提供的开发者工具虽然强大，但是只适用于已完成项目的检查与调试，无法用来进行项目的开发。可以编写 JavaScript 代码的编辑器十分多，例如专门开发大型 Web 项目的 WebStorm 工具、小巧的用于开发移动端网页的 HBuilder 工具、通用编辑器 Sublime Text 工具等。对于 JavaScript 语法部分的学习，强烈建议大家使用 Sublime Text 工具。首先，Sublime Text 十分轻量，占内存极小，运行极快。其次，Sublime Text 有大量的插件支持，能够很好地提供代码高亮、智能补全、代码格式化等高级开发工具所提供的功能。最重要的是，Sublime Text 可以配置编译系统，有了它，我们就不再需要依赖浏览器来进行 JavaScript 代码的运行与调试了。

在如下网站可以下载到 Sublime Text 工具的最新版本：<http://www.sublimetext.com/>。

下载安装完成 Sublime Text 工具后，其已经自带了代码高亮的功能，可以进行 JavaScript 代码的编写，但是并没自动补全、代码格式化与运行 JavaScript 代码的功能，这些将会在后面一一解决，把 Sublime Text 工具武装成一款强大的 JavaScript 编辑器。

## 1.1.3 安装 Sublime Text 插件管理器 PackageControl

Sublime Text 工具的插件十分丰富，但是如何快速地找到并安装所需要的插件依然不容易。PackageControl 工具可以帮助我们解决这些问题。

在 Sublime Text 中有两种方式可以进行 PackageControl 插件的安装。

(1) 第一种方式是直接在 Sublime Text 的控制台输入如下代码，之后按 Enter 键来进行 PackageControl 工具的安装：

```
import urllib.request,os; pf = 'Package Control.sublime-package'; ipp =
sublime.installed_packages_path();urllib.request.install_opener( urllib.reques
t.build_opener( urllib.request.ProxyHandler() ) ); open(os.path.join(ipp, pf),
'wb').write(urllib.request.urlopen( 'http://sublime.wbond.net/' + pf.replace(' ',
'%20')).read())
```

在 Sublime Text 工具中使用 `ctrl+r` 组合键可以直接打开控制台，将上面的代码直接复制进去即可，如图 1-4 所示。

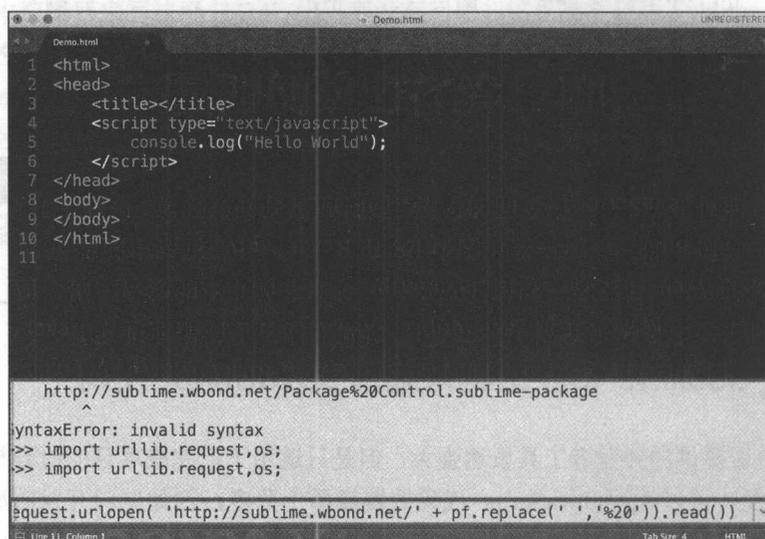


图 1-4 使用代码的方式进行 PackageControl 工具的安装

由于网络环境与 Sublime Text 版本更新有太多的不可控性，使用代码进行 PackageControl 工具的安装不一定会成功，我们也可以直接下载 PackageControl 插件进行安装。可以从如下地址下载 PackageControl 插件：<http://sublime.wbond.net/Package%20Control.sublime-package>。

下载完成后，如图 1-5 所示，打开 Sublime Text 工具的插件目录 Preferences → Browse Packages。将下载到的 PackageControl 安装文件放入 Installed Packages 文件夹中，如图 1-6 所示。

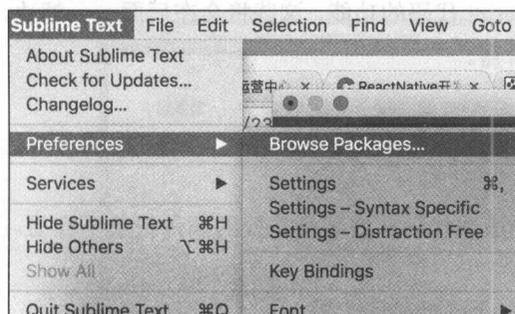


图 1-5 打开 Sublime Text 的插件目录

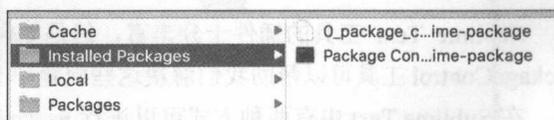


图 1-6 将 PackageControl 安装文件放入 Installed Packages 文件夹中

之后将 Sublime Text 完全关掉，重启 Sublime Text，如果可以在 Preferences 中看到 Package Control 项目，说明安装已经成功，如图 1-7 所示。

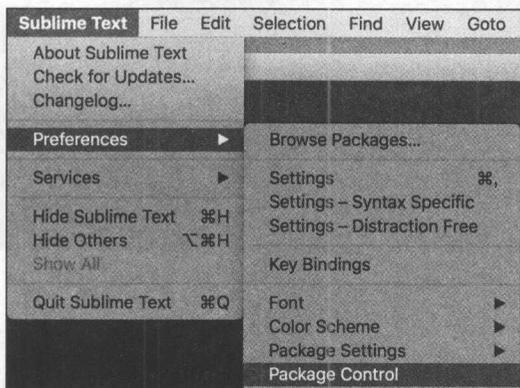


图 1-7 安装成功 PackageControl 工具

### 抽丝剥茧

上面提供的下载 PackageControl 安装包的地 址服务器在国外，国内访问时常会出现波动，如果你无法从上面的地址下载 PackageControl 安装包，可以尝试如下地址：  
<http://zyhshao.github.io/file/Package%20Control.sublime-package>。

## 1.1.4 使用 PackageControl 进行 JavaScript 代码智能提示插件的安装

对代码的智能提示是高级编辑工具必备的一项功能。SublimeCodeIntel 是一个全功能的代码自动提示插件，其支持众多流行语言的代码智能提示，例如 JavaScript、HTML、CSS、Python、PHP 等。

使用 PackageControl 可以十分方便地进行 SublimeCodeIntel 插件的安装，首先打开 Sublime Text 编辑工具，Mac 电脑使用 Command+Shift+P 组合键来打开命令行，在其中输入“package control:”来检索 PackageControl 工具所提供的命令，也可以通过 Sublime Text→Preferences→Package Control 直接打开 PackageControl 命令行，如图 1-8 所示。

PackageControl 工具中提供了许多易用的命令，如安装插件、查看已安装的插件列表、删除插件等，如图 1-9 所示。

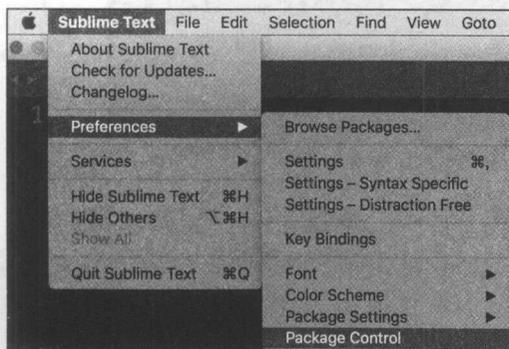


图 1-8 打开 PackageControl 命令行

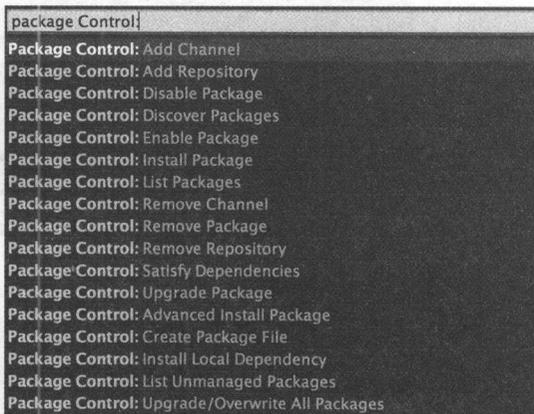


图 1-9 PackageControl 工具提供的命令