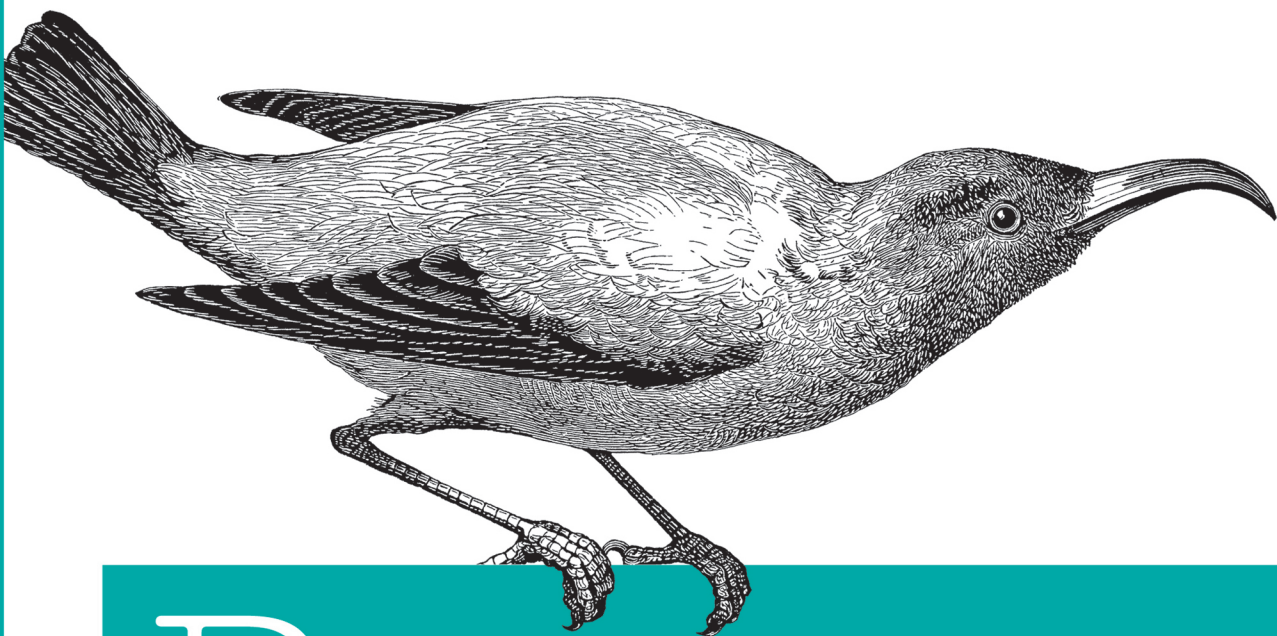


O'REILLY®



图灵程序设计丛书



# React

## 快速上手开发

React: Up & Running: Building Web Applications

来自Facebook的React入门与开发实践

[保] Stoyan Stefanov 著  
张俊达 译



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

---

提供各种书籍的pd电子版代找服务，如果你找不到自己想要的书的pdf电子版，我们可以帮您找到，如有需要，请联系QQ或微信2028969416.

PDF代找说明：

本人可以帮助你找到你要的PDF电子书，计算机类，文学，艺术，设计，医学，理学，经济，金融，等等。质量都很清晰，而且每本100%都带书签索引和目录，方便读者阅读观看，只要您提供给我书的相关信息，一般我都能找到，如果您有需求，请联系我微信或QQ2028969416。

本人已经帮助了上万人找到了他们需要的PDF，其实网上有很多PDF,大家如果在网上不到的话，可以联系我QQ，大部分我都可以找到，而且每本100%带书签索引目录。因PDF电子书都有版权，请不要随意传播，如果您有经济购买能力，请尽量购买正版。

**声明：本人只提供代找服务，每本100%索引书签和目录，因寻找pdf电子书有一定难度，仅收取代找费用。如因PDF产生的版权纠纷，本人概不负责，我们仅仅只是帮助你寻找到你要的pdf而已。**



图灵程序设计丛书

# React快速上手开发

React: Up & Running:  
Building Web Applications

[保] Stoyan Stefanov 著  
张俊达 译

O'REILLY®

*Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo*

O'Reilly Media, Inc. 授权人民邮电出版社出版

人民邮电出版社  
北 京

## 图书在版编目 (C I P) 数据

React快速上手开发 / (保加利亚) 斯托扬·斯特凡诺夫 (Stoyan Stefanov) 著 ; 张俊达译. — 北京 : 人民邮电出版社, 2017. 3

(图灵程序设计丛书)

ISBN 978-7-115-44773-9

I. ①R… II. ①斯… ②张… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2017)第018937号

## 内 容 提 要

本书是 React 入门书。前 3 章介绍如何从空白的 HTML 页面开始构建应用。第 4 章介绍 JSX 语法。从第 5 章开始, 你会学习到在实际开发中可能用到的一些附加工具。本书介绍的例子包括 JavaScript 打包工具、单元测试、语法检查、类型、在应用中组织数据流以及不可变数据。

本书适合有 JavaScript 基础的前端开发人员。

- 
- ◆ 著 [保] Stoyan Stefanov
  - 译 张俊达
  - 责任编辑 朱 巍
  - 执行编辑 杨 琳
  - 责任印制 彭志环
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
  - 邮编 100164 电子邮件 315@ptpress.com.cn
  - 网址 <http://www.ptpress.com.cn>
  - 北京 印刷
  - ◆ 开本: 800×1000 1/16
  - 印张: 13
  - 字数: 308千字 2017年3月第1版
  - 印数: 1—3 500册 2017年3月北京第1次印刷
  - 著作权合同登记号 图字: 01-2016-10061号
- 

定价: 49.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广字第 8052 号

# O'Reilly Media, Inc.介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 *Make* 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版、在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

## 业界评论

“O'Reilly Radar 博客有口皆碑。”

——*Wired*

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——*Business 2.0*

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——*CRN*

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——*Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——*Linux Journal*



---

# 目录

译者序 .....	xiii
前言 .....	xv
第 1 章 Hello World .....	1
1.1 设置 .....	1
1.2 Hello React World .....	2
1.3 刚才发生了什么 .....	4
1.4 <code>React.DOM.*</code> .....	4
1.5 特殊 DOM 属性 .....	7
1.6 React DevTools 浏览器扩展 .....	8
1.7 下一步：自定义组件 .....	9
第 2 章 组件的生命周期 .....	10
2.1 基础 .....	10
2.2 属性 .....	12
2.3 <code>propTypes</code> .....	13
2.4 <code>state</code> .....	16
2.5 带状态的文本框组件 .....	16
2.6 关于 DOM 事件的说明 .....	19
2.6.1 传统的事件处理 .....	20
2.6.2 React 的事件处理 .....	21
2.7 <code>props</code> 与 <code>state</code> .....	21
2.8 在初始化 <code>state</code> 时使用 <code>props</code> ：一种反模式 .....	22
2.9 从外部访问组件 .....	22
2.10 中途改变属性 .....	24

2.11	生命周期方法	25
2.12	生命周期示例：输出日志记录	26
2.13	生命周期示例：使用 <code>mixin</code>	28
2.14	生命周期示例：使用子组件	30
2.15	性能优化：避免组件更新	32
2.16	<code>PureRenderMixin</code>	34
第 3 章	Excel：一个出色的表格组件	37
3.1	构造数据	37
3.2	表头循环	38
3.3	消除控制台的警告信息	40
3.4	添加 <code>&lt;td&gt;</code> 内容	41
3.5	排序	44
3.6	排序的视觉提示	46
3.7	编辑数据	47
3.7.1	可编辑单元格	48
3.7.2	输入字段的单元格	50
3.7.3	保存	50
3.7.4	结论与虚拟 DOM Diff 算法	51
3.8	搜索	52
3.8.1	状态与界面	54
3.8.2	筛选内容	55
3.8.3	如何改进搜索功能	57
3.9	即时回放	58
3.9.1	如何改进回放功能	59
3.9.2	有另一种实现方法吗	59
3.10	下载表格数据	59
第 4 章	JSX	62
4.1	Hello JSX	62
4.2	转译 JSX	63
4.3	Babel	64
4.4	客户端	64
4.5	关于 JSX 转换	66
4.6	在 JSX 中使用 JavaScript	68
4.7	在 JSX 中使用空格	69
4.8	在 JSX 中使用注释	70
4.9	HTML 实体	71
4.10	展开属性	73
4.11	在 JSX 中返回多个节点	75



4.12	JSX 和 HTML 的区别	77
4.12.1	class 和 for 属性不能用了吗	77
4.12.2	style 属性值是一个对象	77
4.12.3	闭合标签	78
4.12.4	用驼峰法命名属性	78
4.13	JSX 和表单	78
4.13.1	onChange 处理器	78
4.13.2	value 和 defaultValue 的区别	79
4.13.3	<textarea> 的值	79
4.13.4	<select> 的值	80
4.14	使用 JSX 实现 Excel 组件	82
第 5 章	为应用开发做准备	83
5.1	一个模板应用	83
5.1.1	文件和目录	84
5.1.2	index.html	85
5.1.3	CSS	86
5.1.4	JavaScript	86
5.1.5	更现代化的 JavaScript	86
5.2	安装必备工具	89
5.2.1	Node.js	90
5.2.2	Browserify	90
5.2.3	Babel	90
5.2.4	React 相关	91
5.3	开始构建	91
5.3.1	转译 JavaScript	91
5.3.2	打包 JavaScript	92
5.3.3	打包 CSS	92
5.3.4	大功告成	92
5.3.5	Windows 版本	93
5.3.6	在开发过程中构建	93
5.4	发布	94
5.5	更进一步	95
第 6 章	构建应用	96
6.1	Whinepad v. 0.0.1	96
6.1.1	基本设置	97
6.1.2	开始编写代码	97
6.2	组件	99
6.2.1	设置	99

6.2.2	组件发现工具 .....	100
6.2.3	<Button> 组件 .....	101
6.2.4	Button.css .....	102
6.2.5	Button.js .....	103
6.2.6	表单 .....	106
6.2.7	<Suggest> .....	106
6.2.8	<Rating> 组件 .....	109
6.2.9	<FormInput> “工厂组件” .....	112
6.2.10	<Form> .....	115
6.2.11	<Actions> .....	117
6.2.12	对话框 .....	119
6.3	应用配置 .....	121
6.4	<Excel>: 改进的新版本 .....	123
6.5	<Whinepad> .....	131
6.6	总结 .....	134
第 7 章	lint、Flow、测试与复验 .....	136
7.1	package.json .....	136
7.1.1	配置 Babel .....	137
7.1.2	脚本 .....	137
7.2	ESLint .....	138
7.2.1	安装 .....	138
7.2.2	运行 .....	138
7.2.3	规则列表 .....	140
7.3	Flow .....	140
7.3.1	安装 .....	141
7.3.2	运行 .....	141
7.3.3	注册类型检查 .....	141
7.3.4	修复 <Button> .....	142
7.3.5	app.js .....	144
7.3.6	关于 props 和 state 类型检查的更多内容 .....	145
7.3.7	导出 / 导入类型 .....	147
7.3.8	类型转换 .....	148
7.3.9	invariant .....	148
7.4	测试 .....	150
7.4.1	安装 .....	150
7.4.2	首个测试 .....	151
7.4.3	首个 React 测试 .....	152
7.4.4	测试 <Button> 组件 .....	153
7.4.5	测试 <Actions> 组件 .....	157

7.4.6 更多模拟交互 .....	159
7.4.7 测试完整的交互 .....	160
7.4.8 代码覆盖率 .....	163
<b>第 8 章 Flux</b> .....	<b>165</b>
8.1 理念 .....	166
8.2 回顾 Whinepad .....	166
8.3 Store .....	167
8.3.1 Store 事件 .....	169
8.3.2 在 <Whinepad> 中使用 Store .....	170
8.3.3 在 <Excel> 中使用 Store .....	173
8.3.4 在 <Form> 中使用 Store .....	174
8.3.5 界定 .....	174
8.4 Action .....	175
8.4.1 CRUD Action .....	175
8.4.2 搜索与排序 .....	176
8.4.3 在 <Whinepad> 中使用 Action .....	178
8.4.4 在 <Excel> 中使用 Action .....	179
8.5 Flux 回顾 .....	181
8.6 immutable .....	182
8.6.1 immutable 存储数据 .....	183
8.6.2 immutable 数据操作 .....	184
<b>关于作者</b> .....	<b>187</b>
<b>关于封面</b> .....	<b>187</b>



---

# 译者序

这是一本关于 React 的入门书。

React 在近年来广受关注，其声明式 UI、组件化的思想让开发和维护具有复杂交互性的界面变得更容易。目前，前端社区中不断涌现出各种构建 React 应用的方案，各种前端技术会议也多次提及与 React 相关的主题。可以说，React 生态圈正朝着越来越好的方向发展。在翻译本书时，我和我的开发团队正在使用 React、Redux 和 Webpack 作为主要的前端技术栈。我们最直观的感受就是开发过程非常愉悦，思路也更为清晰。

互联网上已经有不少关于 React 的学习资料，但是本书的两个特点让其脱颖而出。一是本书作者 Stoyan 就职于 Facebook，对自家开发的技术自然了解得更加透彻；二是本书把讨论重点放在 React 本身，避免初学者因为 React 相关技术栈过于庞大而望而却步。有一种常见的误解是：你需要花费大量时间在配置工具上，然后才能开始学习 React；但其实真相并非如此。在本书中，我们将从零开始构建一个 React 应用。读者可以在掌握 React 基础后，进行下一步的学习。

感谢稀土掘金的江昇老师引荐，让我得到翻译本书的机会。感谢迅雷云存储前端团队在日常工作和学习中给予我的支持。感谢以下好友阅读本书不同版本的译稿并提出宝贵意见：徐沛文、陈超超、石鑫、黄碧龙。当然还要感谢图灵公司各位编辑老师在本书翻译过程中给予的支持与帮助。

还要特别感谢团子与亭子。

总而言之，这本书的顺利出版离不开大家的努力和帮助。由于水平有限，文中纰漏在所难免，恳请广大读者批评指正。如果对于本书有任何意见或想法，欢迎发送邮件至 [bttzjd@gmail.com](mailto:bttzjd@gmail.com)。

张俊达

2016 年 11 月于深圳



---

# 前言

大约 2000 年，洛杉矶。这又是一个温暖舒适的加利福尼亚之夜，淡淡的海风轻轻拂来，舒爽惬意。我正准备使用 FTP 把我新建的站点 CSSsprites.com 传送到服务器并向全世界发布。在发布的前几个晚上，我一直在思考一个问题：“到底为什么只把 20% 的工作量放在解决应用的主要问题，却把剩下的 80% 花费在努力克服用户界面的问题上呢？”如果能把所有调用 `getElementById()` 和考虑应用状态（用户上传是否完成？如果上传出错，上传对话框是否要继续显示？）的时间节约出来，我能利用这部分时间完成多少其他的工具呢？为什么界面开发这么耗时？如何处理不同浏览器之间的差异？想到这些，我的大好心情荡然无存。

时间快进到 2015 年 3 月。在当时召开的 Facebook F8 开发者大会上，我所在的团队准备公布两个完全重写的 Web 应用：一个第三方评论模块和一个配套的评论审核工具。和我的小应用 CSSsprites.com 相比，这两个应用非常成熟，功能也复杂得多，并且流量非常大。虽然如此，其开发过程依然令人愉悦。团队中的新成员（甚至包括刚接触 JavaScript 和 CSS 的新手）都能很快地融入其中，轻松高效地贡献功能特性并改进现有代码。团队中的一个成员说：“现在我发现这就是自己热爱的一切！”

在这段时间里发生了什么？答案是：React 诞生了。

React 是一个 UI 库，让你只需定义一次用户界面，就可以将其用在多个地方。之后，当应用的状态（state）发生变化时，React 将会自动作出反应、更新界面，你无需做其他任何工作。毕竟你已经定义了用户界面。尽管说是定义，其实代码更加偏向声明式，你可以使用可管理的小型组件构造出一个强大的应用。你再也不需要在函数里花费一半的代码量寻找 DOM 节点了，而是可以只维护应用的状态（通过常规的 JavaScript 对象），把剩下的工作都交给 React 帮你完成。

学习 React 非常划算。一旦学会这个库，便可以使用它构建以下类型的应用：

- Web 应用
- 原生 iOS 和 Android 应用
- Canvas 应用
- TV 应用
- 原生桌面应用

你可以使用与构造组件和用户界面相同的思路，创建具有原生应用性能和控制能力的原生应用（真正的原生控制，而不仅仅是看起来像原生）。这并不是指“一次编写，到处运行”（我们的技术尚未实现这一点），而是“一次学习，到处使用”。

简而言之，学习 React 可以帮你节省 80% 的时间，使你可以把精力集中在主要问题上（比如你的应用存在的真正目的）。

## 关于本书

本书从 Web 开发的角度介绍如何学习 React。在前 3 章，你将从一个空白的 HTML 页面开始构建应用。这使得你可以将关注点放在 React 本身，无需了解任何新语法或者辅助工具。

第 4 章介绍 JSX。这是一项单独、可选的技术，通常会同 React 一起使用。

从第 5 章开始，你将学习在实际开发中可能用到的一些附加工具。介绍的例子包括 JavaScript 打包工具（Browserify）、单元测试（Jest）、语法检查（ESLint）、类型（Flow）、在应用中组织数据流（Flux）以及不可变数据（Immutable.js）。所有关于这些辅助技术的讨论都会力求简化，让你依然将精力放在 React 上。你会很快熟悉这些工具的使用，并能根据具体情况选择使用哪些工具。

祝你在学习 React 的过程中一切顺利，大有收获！

## 排版约定

本书使用下列排版约定。

- 等宽字体 (*Constant width*)  
表示广义上的计算机编码，包括变量或函数名、数据库、数据类型、环境变量、语句和关键字。
- 等宽粗体 (***Constant width bold***)  
表示应该由用户按照字面输入的命令或其他文本。
- 等宽斜体 (*Constant width italic*)  
表示应该由用户替换或取决于上下文的值。





该图标表示提示或建议。



该图标表示一般说明。



该图标表示警告或提醒。

## 代码示例

补充材料（包括代码示例、练习题等）可以从 <https://github.com/stoyan/reactbook> 下载。

本书旨在帮助你做好工作。一般来说，你可以在程序和文档中使用本书的代码。除非你使用了很大一部分代码，否则无需联系我们获取许可。例如，使用来自本书的几段代码编写一个程序不需要许可。销售和分发 O'Reilly 书中用例的光盘需要许可。通过引用本书用例和代码来回答问题不需要许可。把本书中的大量用例代码并入你的产品文档需要许可。

我们很希望但不强求注明信息来源。一条信息来源通常包括书名、作者、出版社和 ISBN。例如：“*React: Up & Running* by Stoyan Stefanov (O'Reilly). Copyright 2016 Stoyan Stefanov, 978-1-491-93182-0”。

如果你感到对示例代码的使用超出了正当引用或者这里给出的许可范围，请随时通过 [permissions@oreilly.com](mailto:permissions@oreilly.com) 联系我们。

## Safari® 在线图书



Safari Books Online (<http://www.safaribooksonline.com>) 是应运而生的数字图书馆。它同时以图书和视频的形式出版世界顶级技术和商务作家的专业作品。技术专家、软件开发人员、Web 设计师、商务人士和创意专家等，在开展调研、解决问题、学习和认证培训时，都将 Safari Books Online 视作获取资料的首选渠道。

对于组织团体、政府机构和个人，Safari Books Online 提供各种产品组合和灵活的定

价策略。用户可通过一个功能完备的数据库检索系统访问 O'Reilly Media、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、Focal Press、Cisco Press、John Wiley & Sons、Syngress、Morgan Kaufmann、IBM Redbooks、Packt、Adobe Press、FT Press、Apress、Manning、New Riders、McGraw-Hill、Jones & Bartlett、Course Technology 以及其他几十家出版社的上千种图书、培训视频和正式出版之前的书稿。要了解 Safari Books Online 的更多信息，我们网上见。

## 联系我们

请把对本书的评价和问题发给出版社。

美国：

O'Reilly Media, Inc.  
1005 Gravenstein Highway North  
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室（100035）  
奥莱利技术咨询（北京）有限公司

O'Reilly 的每一本书都有专属网页，你可以在那里找到本书的相关信息，包括勘误表、示例以及其他信息。本书的网站地址是：

<http://shop.oreilly.com/product/0636920042266.do>

对于本书的评论和技术性问题，请发送电子邮件到：

[bookquestions@oreilly.com](mailto:bookquestions@oreilly.com)

要了解更多 O'Reilly 图书、培训课程、会议和新闻的信息，请访问以下网站：

<http://www.oreilly.com>

我们在 Facebook 的地址如下：

<http://facebook.com/oreilly>

请关注我们的 Twitter 动态：

<http://twitter.com/oreillymedia>

我们的 YouTube 视频地址如下：

<http://www.youtube.com/oreillymedia>

# 致谢

感谢在本书组稿的不同阶段阅读本书并给我发送反馈和修正意见的所有人：Andreea Manole、Iliyan Peychev、Kostadin Ilov、Mark Duppenthaler、Stephan Alber 和 Asen Bozhilov。

感谢 Facebook 每天开发或者使用 React 以及回答我问题的所有同事。此外还要感谢为 React 提供工具和库以及撰写文章和使用说明的 React 开发者社区。

非常感谢 Jordan Walke。

感谢 O'Reilly 的所有人，本书的出版离不开你们的努力：Meg Foley、Kim Cofer、Nicole Shelby 等。

感谢 Yavor Vatchkov 为本书中的示例应用设计了用户界面（可以在 [whinepad.com](http://whinepad.com) 体验）。



---

提供各种书籍的pd电子版代找服务，如果你找不到自己想要的书的pdf电子版，我们可以帮您找到，如有需要，请联系QQ或微信2028969416.

PDF代找说明：

本人可以帮助你找到你要的PDF电子书，计算机类，文学，艺术，设计，医学，理学，经济，金融，等等。质量都很清晰，而且每本100%都带书签索引和目录，方便读者阅读观看，只要您提供给我书的相关信息，一般我都能找到，如果您有需求，请联系我微信或QQ2028969416。

本人已经帮助了上万人找到了他们需要的PDF，其实网上有很多PDF,大家如果在网上不到的话，可以联系我QQ，大部分我都可以找到，而且每本100%带书签索引目录。因PDF电子书都有版权，请不要随意传播，如果您有经济购买能力，请尽量购买正版。

**声明：本人只提供代找服务，每本100%索引书签和目录，因寻找pdf电子书有一定难度，仅收取代找费用。如因PDF产生的版权纠纷，本人概不负责，我们仅仅只是帮助你寻找到你要的pdf而已。**