SAMS Teach Yourself

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- “Read Less, Do More”（精读多练）的教学理念
- 以示例引导读者完成最常见的任务

每章内容针对初学者精心设计，1小时轻松阅读学习，
24小时彻底掌握关键知识

每章**案例与练习题**助你轻松完成常见任务，
通过**实践**提高应用技能，巩固所学知识

# Node.js

## 入门经典

[英] George Ornbo 著
傅强 陈宗斌 译

人民邮电出版社
POSTS & TELECOM PRESS

# Node.js
# 入门经典

[英] George Ornbo 著
傅强 陈宗斌 译

人民邮电出版社
北京

**图书在版编目（CIP）数据**

Node.js入门经典 / （英）奥尔波（Ornbo,G.）著；傅强，陈宗斌译. — 北京：人民邮电出版社，2013.4
ISBN 978-7-115-31107-8

Ⅰ. ①N… Ⅱ. ①奥… ②傅… ③陈… Ⅲ. ①JAVA语言—程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2013)第034588号

## 版 权 声 明




**Node.js 入门经典**

◆ 著 [英] George Ornbo
译 傅 强 陈宗斌
责任编辑 傅道坤

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
北京鑫正大印刷有限公司印刷

◆ 开本：787×1092 1/16
印张：22
字数：544 千字 2013 年 4 月第 1 版
印数：1－3 500 册 2013 年 4 月北京第 1 次印刷

著作权合同登记号 图字：01-2012-7073 号
ISBN 978-7-115-31107-8

定价：59.00 元

**读者服务热线：(010)67132692 印装质量热线：(010)67129223**
**反盗版热线：(010)67171154**
**广告经营许可证：京崇工商广字第 0021 号**

# 内容提要

Node.js 是一套用来编写高性能网络服务器的 JavaScript 工具包，从 2009 年诞生之日起，就获得了业内专家和技术社区的强烈关注。而本书采用直观、循序渐进的方法对如何使用 Node.js 来开发及具速度和可扩展性优势的服务器端应用程序进行了讲解。

本书分为 6 部分，第 1 部分介绍了 Node.js 的基本概念和特性；第 2 部分讲解如何借助 HTTP 模块和 Express Web 框架，使用 Node.js 创建基本的网站；第 3 部分介绍了调试和测试 Node.js 应用程序的工具，以及部署 Node.js 应用的方法；第 4 部分讲解了 Node.js 实现实时编程的能力以及 Socket.IO；第 5 部分介绍了 Node.js API 以及构建 Node.js 应用程序所使用的组件；第 6 部分则介绍了 CoffeeScript 这款 Java 预编译器的知识，以及如何在 Node.js 中使用中间件、Backbone.js 来创建单页面应用的知识。

本书内容循序渐进、深入浅出、步骤详尽，而且附有大量适合动手实践的示例，可帮助读者在最短的时间内掌握 Node.js。本书适合对 Node.js 感兴趣的零基础人员阅读，也适合对 Web 前端开发、后端开发感兴趣的技术人员阅读。

# 关于作者

George Ornbo 是英国的一位 JavaScript 和 Ruby 开发人员。他开发 Web 应用程序已有 8 年时间，一开始是以自由职业者的身份工作，最近则为伦敦的 pebble {code}工作。他的博客地址是 http://shapeshed.com，在网络中大多数常见的地方他都以@shapeshed 出现。

# 献辞

本书献给我的妻子 Kirsten。没有你的支持，本书就不可能问世。

# 致谢

感谢 Trina MacDonald 和 Pearson 团队给了我编写本书的机会。你们的鼓励和引导是无价之宝。

感谢本书的技术编辑 Remy Sharp。你找出了大量错误并监督了评审过程。我欠你一顿酒！本书中若还有错误，那肯定是我自己的过失。

感谢我在 pebble {code}的同事。在本书写作伊始，你们就是我的后盾。你们允许我在大型项目中灵活安排时间，让我得以完成本书，我深表感激。

# 前　言

Node.js 可以让开发人员在服务器上使用 JavaScript，这让熟悉 JavaScript 的开发人员又多了一种服务器端的开发技能，但 Node.js 并非仅限于此。它重新思考了在现代 Web 环境下的网络编程，在这个环境下，应用程序可能需要从许多不同的地方读写数据，也可能有上百万个并发用户。

在具有传统的计算机科学学位的开发人员眼中，JavaScript 就是一种玩具语言。但是，JavaScript 已经经历了无数次的挑战，而且如今在 Web 的浏览器和服务器端（借助于 Node.js）中已经不可或缺。现在是编写 JavaScript（尤其是在服务器上）的最好时节！

Node.js 表示一个开发平台，在创建适用于现在 Web 的应用程序时，Node.js 大有裨益，这些应用程序包括：

- 实时应用程序；
- 多人游戏；
- 单页面应用程序；
- 基于 JSON 的 API。

Node.js 专注于速度和可扩展性，而且在无需昂贵硬件的情况下，能处理上千个并发用户的需求。Node.js 项目最近成为 GitHub 上最受关注的项目，如今，eBay、LinkedIn 和 Microsoft 这样的公司已经开始使用它。

Node.js 绝不仅仅只是服务器上的 JavaScript。它是一个功能齐全的网络编程平台，能够针对现代 Web 编程的需求做出响应。

## 本书读者对象

本书并不要求读者一定具备编程经验，但是如果有基本的 JavaScript 使用经验会更好。由于 Node.js 主要从终端运行，因此知道什么是终端，以及如何运行基本的命令也会更有帮助。最后，由于 Node.js 主要是一个网络编程工具，因此，如果读者对 Internet 的运行机制略知一二，会更好。

## 学习 Node.js 的理由

如果读者对创建有许多用户、处理联网数据或者有实时要求的应用程序感兴趣，那么 Node.js 是完成这些任务的极佳工具。此外，如果为浏览器创建应用程序，Node.js 可以让服务器是 JavaScript 的，这可以简化服务器和客户端之间的数据共享。Node.js 是现代 Web 的现代工具箱。

## 本书组织结构

本书首先讲解了 Node.js 的基础知识，包括运行你的第一个 Node.js 程序以及使用 npm（Node 包管理器），然后介绍了网络编程，以及 Node.js 使用 JavaScript 回调来支持异步编程风格的方法。

在本书第 2 部分，我们将学习如何通过使用 HTTP 模块和 Express（一个 Node.js 的 Web 框架），并借助 Node.js 创建基本的网站。我们还将学习如何使用 MongoDB 来让数据持久化。

第 3 部分介绍用于调试和测试 Node.js 应用程序的工具，其中介绍了许多用来支持开发的调试工具和测试框架。我们还将学习如何将 Node.js 应用程序部署到许多第三方服务上，包括 Heroku 和 Nodester。

第 4 部分讲解 Node.js 的实时能力并介绍 Socket.IO。我们将学习如何在浏览器和服务器之间发送消息，并构建一个完整的聊天服务器示例和一个实时的 Twitter 客户端。最后我们将学习如何使用 Node.js 创建 JSON API。

第 5 部分以 Node.js API 为主，并讲解用于创建 Node.js 应用程序的构件（buiding block）。我们将学习进程、子进程、事件、缓冲区和流。

第 6 部分介绍的是读者可能想了解的一些高级主题。我们将学习 CoffeeScript 这个 JavaScript 预编译器，Node.js 如何使用中间件，以及如何使用 Backbone.js 与 Node.js 一起创建单页面应用程序。第 22 章将介绍如何使用 npm 编写并发布你自己的 Node.js 模块。

## 代码示例

本书每章都带有几个代码示例。这些示例旨在帮助读者更好地理解 Node.js。读者可从 http://bit.ly/nodejsbook-examples 下载这些代码，也可从 https://github.com/shapeshed/nodejsbook.io.examples 的 GitHub 库下载。

# 目 录

## 第 1 部分 入门

# 第1部分　入门

# 第 1 章

# Node.js 介绍

在本章中你将学到：

- Node.js 是什么，为什么要创建它；
- 使用 Node.js 能创建的应用程序示例；
- 创建并运行第一个 Node.js 程序。

## 1.1 什么是 Node.js

2009 年 11 月 8 日，Ryan Dahl 在 jsconf.eu 这个关于 JavaScript 的会议上进行了一次演讲，将 Node.js 介绍给了 JavaScript 社区。他对于许多程序设计语言难以实现并发（同时做多件事情）并且经常导致糟糕的性能的问题颇为苦恼。他希望能够更容易地编写出快速的、支持许多用户并且高效地使用内存的联网软件，于是他创建了 Node.js。

在 Ryan Dahl 探索解决该问题的方法的同时，诸如 Apple 和 Google 这样的公司也开始在浏览器技术上大举投入。

鉴于 Google 依赖于浏览器来交付诸如 Gmail 这样的产品，Google 的工程师们创建了 V8，这是个给 Google Chrome 浏览器编写的 JavaScript 引擎，它也是专门为 Web 而设计的经过高度优化的软件。Google 希望 V8 能够发扬光大，于是将其以 BSD（Berkeley Software Distribution）协议开源。

Ryan Dahl 决定使用 V8 引擎来创建 JavaScript 服务器端环境，这样做具有如下理由。

- V8 引擎极快。
- V8 专注于 Web，所以在处理超文本传输协议（HTTP）、域名系统（DNS）和传输控制协议（TCP）等事务上驾轻就熟。

- JavaScript 在 Web 上人尽皆知，所以大多数开发人员都能使用它。

从核心上说，Node.js 是个事件驱动的服务器端 JavaScript 环境。也就是说，我们可以像使用 PHP、Ruby 和 Python 语言那样，使用 JavaScript 创建服务器端的应用程序。对于网络以及创建与网络交互的软件，它尤为专注。

## 1.2 使用 Node.js 能做什么

Node.js 是个程序设计平台，只要有想法和足够的编程技艺，它就无所不能。它既可以创建对文件系统进行操作的小段脚本，也可以创建大规模的 Web 应用程序来运行整个业务。由于 Node.js 的独特设计，它非常适合于多人游戏、实时系统、联网软件和具有上千个并发用户的应用程序。

以下是一些使用 Node.js 的公司。

- LinkedIn
- eBay
- Yahoo!
- Microsoft

能使用 Node.js 创建的应用程序有：

- 实时多人游戏；
- 基于 Web 的聊天客户端；
- 将网络中的数据源进行合并的混搭软件（Mashup）；
- 单页面浏览器应用程序；
- 基于 JSON 的 API。

**注意：什么是服务器端的 JavaScript?**

*大多数 Web 开发人员都熟悉于使用 JavaScript 操纵浏览器中的 Web 页面并与之交互。这就是通常所称的客户端 JavaScript，因为它发生在浏览器或者客户端。服务器端 JavaScript 发生在把页面发送给浏览器之前的服务器上。当然，使用的是同样的语言！*

By the Way

## 1.3 安装并创建第一个 Node.js 程序

说得够多的了！现在来看看运行中的 Node.js 并编写你的第一个 Node.js 程序。首先得安装 Node.js。用于 Windows 和 OSX 的安装程序可以在 Node.js 的主页下载：http://nodejs.org/。要想在这些平台上安装 Node.js，只需下载相关文件并双击安装程序即可。如果使用 Linux 或者想手动编译 Node.js，请在 https://github.com/joyent/node/wiki/installation 上找操作指南。