

《Node.js权威指南》

图书基本信息

书名：《Node.js权威指南》

13位ISBN编号：9787111460782

出版时间：2014-4-1

作者：陆凌牛

页数：560

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.tushu111.com

内容概要

《Node.js权威指南》旨在成为Node.js领域最全面、最系统和最具实战性的一本书，供初学者系统学习和开发者工作时参考。本书作者是资深Web开发专家，不仅全面、细致地讲解了Node.js的所有功能、特性、使用方法和开发技巧，而且还介绍了与之相关的各种扩展功能和工具的使用。此外，它实战性强，不仅每个知识点都配有精心设计的小案例（具体的实现步骤、完整的实现代码、最终的实现效果，图文并茂），而且还有两个综合性的案例，能让读者迅速获得实战经验。本书所有实例代码都经调试运行成功，读者可以对这些代码进行修改，以便观察各种不同的效果，加深对实例代码的理解。

《Node.js权威指南》共16章，分三个部分：第一部分（第1~12章）从Node.js的特性、优势、适用场景、安装配置到各个模块的功能作用，再到开发的具体方法和技巧，非常详尽地讲解了Node.js开发者必备的各种基础知识；第二部分（第13~15章）介绍了在使用Node.js进行开发时极有可能用到的第三方开发包，探讨了如何在Node.js应用程序中使用关系型数据库及NoSQL型数据库，如何使用Express框架开发Web应用程序，以及如何使用Socket.IO类库实现WebSocket通信等；第三部分（第16章）讲解了两个综合案例，如何结合使用Node.js与Socket.IO类库制作一个聊天室应用程序的服务器端及客户端，以及如何结合使用Node.js与Express框架制作一个Web应用程序的服务器端及客户端。

作者简介

陆凌牛，资深Web开发工程师、软件开发工程师和系统设计师。从事Web开发多年，对各种常见的Web开发技术（包括前端和后端）都有非常深入的研究，经验极其丰富。JavaScript、HTML 5和CSS 3等新技术的先驱者和布道者，对这3种技术以及这3种技术下的主流框架都有非常深刻的认识，并且付诸了大量实践。同时，他还擅长微软与Java的相关技术，在C#、VB.NET、ASP.NET、SQL Server、Oracle、Java、Struts、Spring、Hibernate等方面也积累了大量实战经验。

此外，他还是一位颇有影响力的技术作家，著有多本著作：

《HTML 5与CSS 3权威指南》

这是他的处女作，凭借优质的内容获得了读者的广泛好评，被读者誉为“系统学习HTML 5与CSS 3的最佳选择”和“Web前端工程师案头必备图书之一”，如今已经出至第2版。

《HTML 5开发精要与案例详解》

不仅在内容上延续了前一本书的精彩，而且同样也取得了非常好的口碑和销量。

《Sencha Touch权威指南》

国内第一本关于Sencha Touch的专著，定位与本书相同，出版后得到了该领域读者的认可。

书籍目录

前言

第1章 Node.js介绍 / 1

1.1 Node.js概述 / 2

1.1.1 使用Node.js能够解决什么问题 / 2

1.1.2 实现高性能服务器 / 2

1.1.3 非阻塞型I/O及事件环机制 / 2

1.1.4 Node.js适合开发的应用程序 / 3

1.2 安装Node.js / 3

1.3 Node.js中的模块 / 4

1.4 一个简单的示例应用程序 / 6

1.5 小结 / 8

第2章 Node.js中的交互式运行环境——REPL / 9

2.1 REPL运行环境概述 / 10

2.2 在REPL运行环境中操作变量 / 10

2.3 在REPL运行环境中使用下划线字符 / 12

2.4 在REPL运行环境中直接运行函数 / 12

2.5 在REPL运行环境中定义并启动服务器 / 13

2.6 REPL运行环境中的上下文对象 / 13

2.7 REPL运行环境中的基础命令 / 14

2.8 小结 / 17

第3章 Node.js基础知识 / 18

3.1 Node.js中的控制台 / 19

3.1.1 console.log方法 / 19

3.1.2 console.error方法 / 20

3.1.3 console.dir方法 / 21

3.1.4 console.time方法与console.timeEnd方法 / 22

3.1.5 console.trace方法 / 22

3.1.6 console.assert方法 / 23

3.2 Node.js中的全局作用域及全局函数 / 23

3.2.1 Node.js中的全局作用域 / 23

3.2.2 setTimeout函数与clearTimeout函数 / 25

3.2.3 setInterval函数与clearInterval函数 / 25

3.2.4 定时器对象的unref方法与ref方法 / 27

3.2.5 与模块相关的全局函数及对象 / 28

3.3 __filename变量与__dirname变量 / 33

3.3.1 __filename变量 / 33

3.3.2 __dirname变量 / 34

3.4 事件处理机制及事件环机制 / 34

3.4.1 EventEmitter类 / 35

3.4.2 EventEmitter类的各个方法 / 35

3.4.3 获取指定事件的事件处理函数的数量 / 41

3.4.4 EventEmitter类自身所拥有的事件 / 42

3.4.5 事件环机制 / 44

3.5 在Node.js中使用调试器 / 45

3.5.1 在命令行窗口中使用调试器 / 45

3.5.2 观察变量值或表达式的执行结果 / 48

3.5.3 设置与取消断点 / 49

- 3.5.4 调试器中可以使用的其他实用命令 / 50
- 3.5.5 使用node-inspector调试工具 / 54
- 3.6 小结 / 55
- 第4章 模块与npm包管理工具 / 56
 - 4.1 核心模块与文件模块 / 57
 - 4.2 从模块外部访问模块内的成员 / 58
 - 4.2.1 使用exports对象 / 58
 - 4.2.2 将模块定义为类 / 58
 - 4.2.3 为模块类定义类变量或类函数 / 61
 - 4.3 组织与管理模块 / 61
 - 4.3.1 从node_modules目录中加载模块 / 61
 - 4.3.2 使用目录来管理模块 / 62
 - 4.3.3 从全局目录中加载模块 / 62
 - 4.4 模块对象的属性 / 63
 - 4.5 包与npm包管理工具 / 65
 - 4.5.1 Node.js中的包 / 65
 - 4.5.2 npm包管理工具 / 67
 - 4.6 小结 / 68
- 第5章 使用Buffer类处理二进制数据 / 69
 - 5.1 创建Buffer对象 / 70
 - 5.2 字符串的长度与缓存区的长度 / 72
 - 5.3 Buffer对象与字符串对象之间的相互转换 / 74
 - 5.3.1 Buffer对象的toString方法 / 74
 - 5.3.2 Buffer对象的write方法 / 75
 - 5.3.3 StringDecoder对象 / 75
 - 5.4 Buffer对象与数值对象之间的相互转换 / 77
 - 5.5 Buffer对象与JSON对象之间的相互转换 / 79
 - 5.6 复制缓存数据 / 80
 - 5.7 Buffer类的类方法 / 81
 - 5.7.1 isBuffer方法 / 81
 - 5.7.2 byteLength方法 / 81
 - 5.7.3 concat方法 / 82
 - 5.7.4 isEncoding方法 / 83
 - 5.8 小结 / 83
- 第6章 在Node.js中操作文件系统 / 84
 - 6.1 同步方法与异步方法 / 85
 - 6.2 对文件执行读写操作 / 86
 - 6.2.1 文件的完整读写 / 86
 - 6.2.2 从指定位置处开始读写文件 / 91
 - 6.3 创建与读取目录 / 97
 - 6.3.1 创建目录 / 97
 - 6.3.2 读取目录 / 98
 - 6.4 查看与修改文件或目录的信息 / 99
 - 6.4.1 查看文件或目录的信息 / 99
 - 6.4.2 检查文件或目录是否存在 / 101
 - 6.4.3 获取文件或目录的绝对路径 / 102
 - 6.4.4 修改文件访问时间及修改时间 / 103
 - 6.4.5 修改文件或目录的读写权限 / 104
 - 6.5 可以对文件或目录执行的其他操作 / 105

- 6.5.1 移动文件或目录 / 105
- 6.5.2 创建与删除文件的硬链接 / 106
- 6.5.3 创建与查看符号链接 / 107
- 6.5.4 截断文件 / 110
- 6.5.5 删除空目录 / 111
- 6.5.6 监视文件或目录 / 111
- 6.6 使用文件流 / 116
 - 6.6.1 流的基本概念 / 116
 - 6.6.2 使用ReadStream对象读取文件 / 119
 - 6.6.3 使用WriteStream对象写入文件 / 121
- 6.7 对路径进行操作 / 127
- 6.8 小结 / 134
- 第7章 实现基于TCP与UDP的数据通信 / 135
 - 7.1 使用net模块实现基于TCP的数据通信 / 136
 - 7.1.1 创建TCP服务器 / 136
 - 7.1.2 socket端口对象 / 142
 - 7.1.3 创建TCP客户端 / 151
 - 7.1.4 net模块中的类方法 / 164
 - 7.2 使用dgram模块实现基于UDP的数据通信 / 165
 - 7.2.1 创建UDP服务器与客户端 / 165
 - 7.2.2 实现广播与组播 / 172
 - 7.3 小结 / 175
- 第8章 创建HTTP与HTTPS服务器及客户端 / 176
 - 8.1 HTTP服务器 / 177
 - 8.1.1 创建HTTP服务器 / 177
 - 8.1.2 获取客户端请求信息 / 182
 - 8.1.3 转换URL字符串与查询字符串 / 184
 - 8.1.4 发送服务器端响应流 / 191
 - 8.2 HTTP客户端 / 202
 - 8.2.1 向其他网站请求数据 / 202
 - 8.2.2 向本地服务器请求数据 / 208
 - 8.2.3 制作代理服务器 / 210
 - 8.3 创建HTTPS服务器与客户端 / 211
 - 8.3.1 创建HTTPS服务器 / 211
 - 8.3.2 创建HTTPS客户端 / 216
 - 8.4 小结 / 223
- 第9章 进程与子进程 / 224
 - 9.1 Node.js中的进程 / 225
 - 9.1.1 进程对象的属性 / 225
 - 9.1.2 进程对象的方法与事件 / 227
 - 9.2 创建多进程应用程序 / 235
 - 9.2.1 使用spawn方法开启子进程 / 236
 - 9.2.2 使用fork方法开启子进程 / 243
 - 9.2.3 使用exec方法开启子进程 / 250
 - 9.2.4 使用execFile方法开启子进程 / 253
 - 9.3 在多个子进程中运行Node.js应用程序 / 254
 - 9.3.1 使用fork方法创建worker对象 / 254
 - 9.3.2 worker对象的方法与事件 / 262
 - 9.4 小结 / 270

第10章 Node.js中的错误处理与断言处理 / 271

- 10.1 使用domain模块处理错误 / 272
 - 10.1.1 domain模块概述 / 272
 - 10.1.2 创建并使用Domain对象 / 274
 - 10.1.3 隐式绑定与显式绑定 / 276
 - 10.1.4 绑定回调函数与拦截回调函数 / 279
 - 10.1.5 domain堆栈的弹出与推入 / 280
 - 10.1.6 Domain对象的销毁 / 286
- 10.2 Node.js中的断言处理 / 286
 - 10.2.1 equal方法与notEqual方法 / 287
 - 10.2.2 strictEqual方法与notStrictEqual方法 / 288
 - 10.2.3 assert方法与ok方法 / 288
 - 10.2.4 deepEqual方法与notDeepEqual方法 / 289
 - 10.2.5 throws方法与doesNotThrow方法 / 290
- 10.3 小结 / 293

第11章 加密与压缩 / 294

- 11.1 加密与解密处理 / 295
 - 11.1.1 crypto模块概述 / 295
 - 11.1.2 散列算法 / 296
 - 11.1.3 HMAC算法 / 297
 - 11.1.4 公钥加密 / 298
- 11.2 压缩与解压缩处理 / 305
 - 11.2.1 创建各种用于压缩及解压缩的对象 / 305
 - 11.2.2 zlib模块中的各种方法 / 310
- 11.3 小结 / 311

第12章 Node.js中的其他模块 / 312

- 12.1 使用dns模块解析域名 / 313
 - 12.1.1 使用resolve方法将域名解析为DNS记录 / 313
 - 12.1.2 使用lookup方法查询IP地址 / 315
 - 12.1.3 使用reverse方法反向解析IP地址 / 316
 - 12.1.4 dns模块中的各种错误代码 / 317
- 12.2 使用punycode模块转换punycode编码 / 318
- 12.3 使用os模块获取操作系统信息 / 320
- 12.4 使用readline模块逐行读取流数据 / 323
 - 12.4.1 创建Interface对象 / 323
 - 12.4.2 Interface对象所拥有的各种方法与事件 / 327
- 12.5 使用util模块中提供一些实用方法 / 335
- 12.6 使用vm模块改变脚本运行环境 / 346
 - 12.6.1 在独立环境中运行JavaScript代码 / 346
 - 12.6.2 创建并使用Script对象 / 350
- 12.7 自定义REPL运行环境 / 352
- 12.8 小结 / 358

第13章 数据库访问 / 359

- 13.1 在MongoDB数据库中存取数据 / 360
 - 13.1.1 MongoDB概述 / 360
 - 13.1.2 安装MongoDB数据库 / 360
 - 13.1.3 安装MongoDB包 / 361
 - 13.1.4 连接MongoDB数据库 / 361
 - 13.1.5 在MongoDB数据库中插入数据 / 364

- 13.1.6 在MongoDB数据库中查询数据 / 369
- 13.1.7 在MongoDB数据库中更新与删除数据 / 384
- 13.1.8 使用Mongoose类库 / 393
- 13.2 在MySQL数据库中存取数据 / 395
 - 13.2.1 建立连接与关闭连接 / 395
 - 13.2.2 执行数据的基本处理 / 399
 - 13.2.3 执行存储过程 / 404
 - 13.2.4 执行多表结合查询 / 406
 - 13.2.5 以数据流的方式处理查询数据 / 409
 - 13.2.6 创建连接池 / 411
- 13.3 小结 / 413
- 第14章 使用Express构建Web应用程序 / 414
 - 14.1 Express概述 / 415
 - 14.1.1 安装Express / 415
 - 14.1.2 使用Express开发一个简单的示例应用程序 / 415

精彩短评

- 1、有很多错误呀，不知道怎么审稿的。讲解有点混乱
- 2、对API的说明详细但有不少需要勘误或更新的内容，初学时一定要动手编码（查源码、官方文档次之）结果不同的地方（譬如数据库的包本书介绍的旧版本就与新版本的函数输出不同）标注出来别想当然把本书当做权威工具书照搬。
- 3、国人写的书还是不够严谨，代码也不没有引人深思和模仿的能力，不过最为一部NodeJS功能大全和API小手册来看看还可以，还是要多看源码和官方API文档比较有效。这本书不推荐收藏。
- 4、作为一本入门书籍是很不错的选择
- 5、不错的书，就是自己用这个技术的机会不多。讲的全面，到位。

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.tushu111.com