Alex Young
Marc Harter

Foreword by
Ben Noordhuis

Node.js

in Practice

Includes 115 Techniques

MANNING

Node.js in Practice

ALEX YOUNG
MARC HARTER

MANNING
SHELTER ISLAND

For online information and ordering of this and other Manning books, please visit www.manning.com. The publisher offers discounts on this book when ordered in quantity. For more information, please contact

Special Sales Department
Manning Publications Co.
20 Baldwin Road
PO Box 761
Shelter Island, NY 11964
Email: orders@manning.com

©2015 by Manning Publications Co. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by means electronic, mechanical, photocopying, or otherwise, without prior written permission of the publisher.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in the book, and Manning Publications was aware of a trademark claim, the designations have been printed in initial caps or all caps.

♾ Recognizing the importance of preserving what has been written, it is Manning's policy to have the books we publish printed on acid-free paper, and we exert our best efforts to that end. Recognizing also our responsibility to conserve the resources of our planet, Manning books are printed on paper that is at least 15 percent recycled and processed without the use of elemental chlorine.

Manning Publications Co.
20 Baldwin Road
PO Box 761
Shelter Island, NY 11964

Development editor:	Cynthia Kane
Technical development editor:	Jose Maria Alvarez Rodriguez
Copyeditor:	Benjamin Berg
Proofreader:	Katie Tennant
Typesetter:	Gordan Salinovic
Cover designer:	Marija Tudor

ISBN 9781617290930
Printed in the United States of America
1 2 3 4 5 6 7 8 9 10 – EBM – 19 18 17 16 15 14

brief contents

PART 1 NODE FUNDAMENTALS 1

1 ■ Getting started 3

2 ■ Globals: Node's environment 15

3 ■ Buffers: Working with bits, bytes, and encodings 39

4 ■ Events: Mastering EventEmitter and beyond 64

5 ■ Streams: Node's most powerful and misunderstood feature 82

6 ■ File system: Synchronous and asynchronous approaches to files 114

7 ■ Networking: Node's true "Hello, World" 136

8 ■ Child processes: Integrating external applications with Node 174

PART 2 REAL-WORLD RECIPES 197

9 ■ The Web: Build leaner and meaner web applications 199

10 ■ Tests: The key to confident code 260

11 ■ Debugging: Designing for introspection and resolving issues 293

12 ■ Node in production: Deploying applications safely 326

PART 3 WRITING MODULES 359

13 ■ Writing modules: Mastering what Node is all about 361

contents

foreword xiii
preface xv
acknowledgments xvi
about this book xviii
about the cover illustration xx

PART 1 NODE FUNDAMENTALS 1

1 *Getting started 3*

1.1 Getting to know Node 4

Why Node? 4 ▪ Node's main features 6

1.2 Building a Node application 8

Creating a new Node project 9 ▪ Making a stream class 9
Using a stream 10 ▪ Writing a test 12

1.3 Summary 13

2 *Globals: Node's environment 15*

2.1 Modules 16

TECHNIQUE 1 Installing and loading modules 16
TECHNIQUE 2 Creating and managing modules 17

Technique 3 Loading a group of related modules 19
Technique 4 Working with paths 21

2.2 Standard I/O and the console object 22

Technique 5 Reading and writing to standard I/O 22
Technique 6 Logging messages 24
Technique 7 Benchmarking a program 25

2.3 Operating system and command-line integration 27

Technique 8 Getting platform information 27
Technique 9 Passing command-line arguments 28
Technique 10 Exiting a program 29
Technique 11 Responding to signals 31

2.4 Delaying execution with timers 32

Technique 12 Executing functions after a delay with setTimeout 32
Technique 13 Running callbacks periodically with timers 34
Technique 14 Safely managing asynchronous APIs 35

2.5 Summary 38

3 *Buffers: Working with bits, bytes, and encodings 39*

3.1 Changing data encodings 40

Technique 15 Converting buffers into other formats 40
Technique 16 Changing string encodings using buffers 41

3.2 Converting binary files to JSON 44

Technique 17 Using buffers to convert raw data 44

3.3 Creating your own binary protocol 58

Technique 18 Creating your own network protocol 58

3.4 Summary 63

4 *Events: Mastering EventEmitter and beyond 64*

4.1 Basic usage 65

Technique 19 Inheriting from EventEmitter 65
Technique 20 Mixing in EventEmitter 68

4.2 Error handling 69

Technique 21 Managing errors 69
Technique 22 Managing errors with domains 71

4.3 Advanced patterns 73

Technique 23 Reflection 73
Technique 24 Detecting and exploiting EventEmitter 75
Technique 25 Categorizing event names 77

4.4 Third-party modules and extensions 78

TECHNIQUE 26 Alternatives to EventEmitter 78

4.5 Summary 80

5 *Streams: Node's most powerful and misunderstood feature 82*

5.1 Introduction to streams 83

Types of streams 83 ▪ When to use streams 84 ▪ History 85 Streams in third-party modules 85 ▪ Streams inherit from EventEmitter 87

5.2 Built-in streams 88

TECHNIQUE 27 Using built-in streams to make a static web server 88

TECHNIQUE 28 Stream error handling 90

5.3 Third-party modules and streams 91

TECHNIQUE 29 Using streams from third-party modules 91

5.4 Using the stream base classes 94

TECHNIQUE 30 Correctly inheriting from the stream base classes 94

TECHNIQUE 31 Implementing a readable stream 96

TECHNIQUE 32 Implementing a writable stream 99

TECHNIQUE 33 Transmitting and receiving data with duplex streams 101

TECHNIQUE 34 Parsing data with transform streams 103

5.5 Advanced patterns and optimization 105

TECHNIQUE 35 Optimizing streams 105

TECHNIQUE 36 Using the old streams API 108

TECHNIQUE 37 Adapting streams based on their destination 109

TECHNIQUE 38 Testing streams 111

5.6 Summary 113

6 *File system: Synchronous and asynchronous approaches to files 114*

6.1 An overview of the fs module 115

POSIX file I/O wrappers 115 ▪ Streaming 117 ▪ Bulk file I/O 117 File watching 118 ▪ Synchronous alternatives 118

TECHNIQUE 39 Loading configuration files 119

TECHNIQUE 40 Using file descriptors 120

TECHNIQUE 41 Working with file locking 121

TECHNIQUE 42 Recursive file operations 125

TECHNIQUE 43 Writing a file database 128
TECHNIQUE 44 Watching files and directories 132

6.2 Summary 134

7 *Networking: Node's true "Hello, World" 136*

7.1 Networking in Node 137

Networking terminology 137 ▪ Node's networking modules 141
Non-blocking networking and thread pools 142

7.2 TCP clients and servers 143

TECHNIQUE 45 Creating a TCP server and tracking clients 143
TECHNIQUE 46 Testing TCP servers with clients 145
TECHNIQUE 47 Improve low-latency applications 147

7.3 UDP clients and servers 149

TECHNIQUE 48 Transferring a file with UDP 149
TECHNIQUE 49 UDP client server applications 153

7.4 HTTP clients and servers 156

TECHNIQUE 50 HTTP servers 156
TECHNIQUE 51 Following redirects 158
TECHNIQUE 52 HTTP proxies 162

7.5 Making DNS requests 165

TECHNIQUE 53 Making a DNS request 165

7.6 Encryption 167

TECHNIQUE 54 A TCP server that uses encryption 167
TECHNIQUE 55 Encrypted web servers and clients 170

7.7 Summary 173

8 *Child processes: Integrating external applications with Node 174*

8.1 Executing external applications 175

TECHNIQUE 56 Executing external applications 176
Paths and the PATH environment variable 176 ▪ Errors when executing external applications 177

TECHNIQUE 57 Streaming and external applications 178
Stringing external applications together 179

TECHNIQUE 58 Executing commands in a shell 180
Security and shell command execution 181

TECHNIQUE 59 Detaching a child process 182
Handing I/O between the child and parent processes 183 ▪ Reference counting and child processes 184

8.2 Executing Node programs 185

TECHNIQUE 60 Executing Node programs 185
TECHNIQUE 61 Forking Node modules 186
TECHNIQUE 62 Running jobs 188
Job pooling 190 ▪ Using the pooler module 191

8.3 Working synchronously 192

TECHNIQUE 63 Synchronous child processes 192

8.4 Summary 194

PART 2 REAL-WORLD RECIPES 197

9 *The Web: Build leaner and meaner web applications 199*

9.1 Front-end techniques 200

TECHNIQUE 64 Quick servers for static sites 200
TECHNIQUE 65 Using the DOM in Node 204
TECHNIQUE 66 Using Node modules in the browser 207

9.2 Server-side techniques 209

TECHNIQUE 67 Express route separation 209
TECHNIQUE 68 Automatically restarting the server 212
TECHNIQUE 69 Configuring web applications 215
TECHNIQUE 70 Elegant error handling 219
TECHNIQUE 71 RESTful web applications 222
TECHNIQUE 72 Using custom middleware 231
TECHNIQUE 73 Using events to decouple functionality 236
TECHNIQUE 74 Using sessions with WebSockets 238
TECHNIQUE 75 Migrating Express 3 applications to Express 4 242

9.3 Testing web applications 246

TECHNIQUE 76 Testing authenticated routes 246
TECHNIQUE 77 Creating seams for middleware injection 248
TECHNIQUE 78 Testing applications that depend on remote services 250

9.4 Full stack frameworks 256

9.5 Real-time services 257

9.6 Summary 258

10 *Tests: The key to confident code* 260

10.1 Introduction to testing with Node 261

10.2 Writing simple tests with assertions 262

TECHNIQUE 79 Writing tests with built-in modules 263
TECHNIQUE 80 Testing for errors 265
TECHNIQUE 81 Creating custom assertions 268

10.3 Test harnesses 270

TECHNIQUE 82 Organizing tests with a test harness 270

10.4 Test frameworks 273

TECHNIQUE 83 Writing tests with Mocha 273
TECHNIQUE 84 Testing web applications with Mocha 276
TECHNIQUE 85 The Test Anything Protocol 280

10.5 Tools for tests 282

TECHNIQUE 86 Continuous integration 283
TECHNIQUE 87 Database fixtures 285

10.6 Further reading 291

10.7 Summary 292

11 *Debugging: Designing for introspection and resolving issues* 293

11.1 Designing for introspection 294

Explicit exceptions 294 ▪ *Implicit exceptions* 295 ▪ *The error event* 295 ▪ *The error argument* 296

TECHNIQUE 88 Handling uncaught exceptions 296
TECHNIQUE 89 Linting Node applications 299

11.2 Debugging issues 300

TECHNIQUE 90 Using Node's built-in debugger 300
TECHNIQUE 91 Using Node Inspector 306
TECHNIQUE 92 Profiling Node applications 308
TECHNIQUE 93 Debugging memory leaks 311
TECHNIQUE 94 Inspecting a running program with a REPL 316
TECHNIQUE 95 Tracing system calls 322

11.3 Summary 325

12 *Node in production: Deploying applications safely* 326

12.1 Deployment 327

TECHNIQUE 96 Deploying Node applications to the cloud 327
TECHNIQUE 97 Using Node with Apache and nginx 332