

深入理解JavaScript特性



[深入理解JavaScript特性_下载链接1](#)

著者:[阿根廷] 尼古拉斯 · 贝瓦夸

出版者:人民邮电出版社

出版时间:2019-5

装帧:平装

isbn:9787115510402

本书旨在让读者轻松学习JavaScript的新进展，包括ES6及后续更新。书中提供了大量实用示例，以循序渐进的方式讲解了异步函数、对象解构、动态导入及异步生成器等内容。本书不仅介绍了箭头函数、解构、模板字面量以及其他语法方面的新元素，还全面

展示了ES6引入的流程控制机制，以及如何高效地简化自己的代码。本书的讨论还涉及ES6内置的新集合类型、使用代理控制属性访问、ES6中内置API的改进、CommonJS与ECMAScript模块的互用性等方面。

作者介绍:

作者简介:

尼古拉斯·贝瓦夸 (Nicolás Bevacqua)

知名JavaScript布道师，来自阿根廷的JavaScript编程高手，目前是Elastic公司用户界面工程师。另著有《JavaScript Web应用开发》一书。

译者简介:

李松峰

360前端开发资深专家、前端TC委员、W3C AC代表，奇舞团Web字体服务“奇字库”作者。

刘冰晶

毕业于北京邮电大学，现为奇舞团前端开发工程师，专注于数据可视化以及前端动画领域。

高峰

硕士，毕业于中国科学技术大学软件学院。现为奇虎360前端开发工程师。

审校简介:

黄小璐

毕业于华中科技大学计算机学院。现为奇虎360软件开发工程师。参与翻译了《高性能HTML5》《移动Web手册》《大型JavaScript应用最佳实践指南》《Web开发权威指南》等书。

欧雪雯

前端开发工程师，多年从事科技、前端技术领域业余翻译工作。

目录: 序 xi

前言 xiii

第1章 ECMAScript和JavaScript的未来 1

1.1 JavaScript标准简史 1

1.2 持续迭代的ECMAScript 3

1.3 浏览器支持和辅助工具 5

1.3.1 Babel转译器简介 5

1.3.2 使用ESLint提高代码质量和一致性 9

1.4 ES6特性 11

1.5 JavaScript的未来 12

第2章 ES6基础 14

- 2.1 对象字面量 14
 - 2.1.1 属性值简写 14
 - 2.1.2 可计算属性名 15
 - 2.1.3 方法定义 17
- 2.2 箭头函数 18
 - 2.2.1 词法作用域 19
 - 2.2.2 箭头函数的写法 20
 - 2.2.3 优点和用例 21
- 2.3 解构 22
 - 2.3.1 对象的解构 22
 - 2.3.2 数组的解构 25
 - 2.3.3 函数参数的默认值 25
 - 2.3.4 函数参数的解构 26
 - 2.3.5 解构的用例 28
- 2.4 剩余参数和扩展运算符 29
 - 2.4.1 剩余参数 29
 - 2.4.2 扩展运算符 30
- 2.5 模板字面量 32
 - 2.5.1 字符串插值 32
 - 2.5.2 多行模板字面量 33
 - 2.5.3 标签模板 35
- 2.6 let和const声明 37
 - 2.6.1 块级作用域和let声明 38
 - 2.6.2 暂时性死区 39
 - 2.6.3 const声明 41
 - 2.6.4 const和let的优势 42
- 第3章 类、符号、对象和装饰器 44
 - 3.1 类 44
 - 3.1.1 使用类 44
 - 3.1.2 类的属性和方法 47
 - 3.1.3 类的继承 49
 - 3.2 符号 51
 - 3.2.1 本地符号 51
 - 3.2.2 符号的实际用法 53
 - 3.2.3 全局符号注册表 55
 - 3.2.4 众所周知的符号 57
 - 3.3 对象的内置改进 58
 - 3.3.1 使用Object.assign扩展对象 58
 - 3.3.2 使用Object.is进行对象比较 61
 - 3.3.3 Object.setPrototypeOf 62
 - 3.4 装饰器 63
 - 3.4.1 初识JavaScript装饰器 63
 - 3.4.2 装饰器叠加及不变性提醒 64
 - 3.4.3 用例：C#中的特性 64
 - 3.4.4 在JavaScript中装饰属性 66
- 第4章 迭代与流程控制 68
 - 4.1 Promise 68
 - 4.1.1 快速理解Promise 68
 - 4.1.2 Promise的延续与连缀 72
 - 4.1.3 创建Promise 76
 - 4.1.4 Promise的状态 78
 - 4.1.5 Promise#finally提案 79
 - 4.1.6 Promise.all和Promise.race 81
 - 4.2 迭代器协议与可迭代协议 83

- 4.2.1 迭代的原理 83
- 4.2.2 无穷序列 85
- 4.2.3 迭代对象以生成键/值对 88
- 4.2.4 打造多功能播放列表 90
- 4.3 生成器函数与生成器对象 93
 - 4.3.1 生成器基础 93
 - 4.3.2 手工迭代生成器 96
 - 4.3.3 将生成器混入可迭代对象 97
 - 4.3.4 使用生成器遍历树 99
 - 4.3.5 传递生成器函数 101
 - 4.3.6 处理异步流 102
 - 4.3.7 在生成器上抛出错误 103
 - 4.3.8 代替生成器返回 104
 - 4.3.9 基于生成器的异步I/O 105
- 4.4 异步函数 109
 - 4.4.1 各种异步代码 109
 - 4.4.2 使用async/await 111
 - 4.4.3 并发异步流 112
 - 4.4.4 错误处理 113
 - 4.4.5 深入理解异步函数 114
- 4.5 异步迭代 119
 - 4.5.1 异步迭代器 119
 - 4.5.2 异步生成器 120
- 第5章 巧妙使用ES中的集合 122
 - 5.1 使用ES6 map 124
 - 5.1.1 初识ES6 map 124
 - 5.1.2 散列映射和DOM元素 128
 - 5.2 理解和使用WeakMap 130
 - 5.3 ES6中的Set 131
 - 5.4 ES6 WeakSet 133
- 第6章 使用代理控制属性访问 136
 - 6.1 了解代理 136
 - 6.1.1 捕获get访问 137
 - 6.1.2 捕获set访问 138
 - 6.1.3 通过代理实现模式验证 140
 - 6.2 可撤销代理 142
 - 6.3 代理捕获器 143
 - 6.3.1 has捕获器 143
 - 6.3.2 deleteProperty捕获器 144
 - 6.3.3 defineProperty捕获器 146
 - 6.3.4 ownKeys捕获器 148
 - 6.4 高级代理捕获器 150
 - 6.4.1 getOwnPropertyDescriptor捕获器 150
 - 6.4.2 apply捕获器 151
 - 6.4.3 construct捕获器 154
 - 6.4.4 getPrototypeOf捕获器 156
 - 6.4.5 setPrototypeOf捕获器 157
 - 6.4.6 preventExtensions捕获器 158
 - 6.4.7 isExtensible捕获器 159
- 第7章 ES6中内置API的改进 161
 - 7.1 数字 161
 - 7.1.1 二进制和八进制字面量 161
 - 7.1.2 Number.isNaN 162
 - 7.1.3 Number.isFinite 164

- 7.1.4 Number.parseInt 165
- 7.1.5 Number.parseFloat 166
- 7.1.6 Number.isInteger 166
- 7.1.7 Number.EPSILON 167
- 7.1.8 Number.MAX_SAFE_INTEGER和Number.MIN_SAFE_INTEGER 167
- 7.1.9 Number.isSafeInteger 168
- 7.2 Math 170
 - 7.2.1 Math.sign 171
 - 7.2.2 Math.trunc 171
 - 7.2.3 Math.cbrt 171
 - 7.2.4 Math.expm1 172
 - 7.2.5 Math.log1p 172
 - 7.2.6 Math.log10 173
 - 7.2.7 Math.log2 173
 - 7.2.8 三角函数 174
 - 7.2.9 Math.hypot 174
 - 7.2.10 按位计算助手 174
- 7.3 字符串和Unicode 175
 - 7.3.1 String#startsWith 175
 - 7.3.2 String#endsWith 176
 - 7.3.3 String#includes 177
 - 7.3.4 String#repeat 177
 - 7.3.5 字符串填充和去空白 179
 - 7.3.6 Unicode 180
 - 7.3.7 String.prototype[Symbol.iterator] 181
 - 7.3.8 有关分割字形段的提案 182
 - 7.3.9 String#codePointAt 183
 - 7.3.10 String.fromCodePoint 184
 - 7.3.11 Unicode-Aware字符串反转 185
 - 7.3.12 String#normalize 185
- 7.4 正则表达式 186
 - 7.4.1 粘连修饰符/y 186
 - 7.4.2 Unicode修饰符/u 187
 - 7.4.3 具名捕获组 188
 - 7.4.4 Unicode属性转义 190
 - 7.4.5 后行断言 191
 - 7.4.6 新的/s (dotAll) 修饰符 193
 - 7.4.7 String#matchAll 193
- 7.5 Array 196
 - 7.5.1 Array.from 196
 - 7.5.2 Array.of 198
 - 7.5.3 Array#copyWithin 199
 - 7.5.4 Array#fill 200
 - 7.5.5 Array#find和Array#findIndex 201
 - 7.5.6 Array#keys 201
 - 7.5.7 Array#values 202
 - 7.5.8 Array#entries 202
 - 7.5.9 Array.prototype[Symbol.iterator] 202
- 第8章 JavaScript模块 204
 - 8.1 CommonJS 204
 - 8.2 JavaScript模块 209
 - 8.2.1 严格模式 209
 - 8.2.2 export语句 209
 - 8.2.3 import语句 213

8.2.4 动态import() 215
8.3 ES模块的实践考量 216
第9章 实用建议 219
9.1 变量声明 219
9.2 模板字面量 223
9.3 简写及对象解构 227
9.4 剩余参数和扩展运算符 229
9.5 函数偏好 231
9.6 类和代理 235
9.7 异步代码流 238
9.8 复杂性蠕变、抽象及约定 241
• • • • • ([收起](#))

[深入理解JavaScript特性_下载链接1](#)

标签

JavaScript

前端开发

es6

力荐

编程

programming

评论

值得一读

书是好书，不过翻译水平有待提高

简单明了的介绍了ES6的新特性， 以及使用场景。 比较适合中级小白来深入学习。

[深入理解JavaScript特性_下载链接1](#)

书评

[深入理解JavaScript特性_下载链接1](#)