《ES6 标准入门（第2版）》

图书基本信息

书名：《ES6 标准入门（第2版）》

13位ISBN编号：9787121276577

出版时间：2016-1

作者：阮一峰 著

页数：336

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.tushu000.com

《ES6 标准入门（第2版）》

内容概要

ES6（又名 ES2105）是 JavaScript 语言的新标准，2015 年 6 月正式发布后，得到了迅速推广，是目前业界超级活跃的计算机语言。《ES6标准入门（第2版）》是国内仅有的一本 ES6 教程，在前版基础上增补了大量内容——对标准进行了彻底的解读，所有新增的语法知识（包括即将发布的 ES7）都给予了详细介绍，并且紧扣业界开发实践，给出了大量简洁易懂、可以即学即用的示例代码。
《ES6标准入门（第2版）》为中级难度，适合对 JavaScript 语言或 ES5 已经有所了解的读者，用来提高水平，了解这门语言的最新发展；也可当作参考手册，查寻 ES6/ES7 新增的语法点。

《ES6 标准入门（第2版）》

书籍目录

第 1 章　ECMAScript 6简介 001
1.1 ECMAScript和JavaScript的关系 001
1.2 ECMAScript的历史 001
1.3 部署进度 002
1.4 Babel转码器 003
命令行环境 004
浏览器环境 005
Node.js环境 005
在线转换 006
1.5 Traceur转码器 006
直接插入网页 006
在线转换 007
命令行转换 008
Node.js环境的用法 008
1.6 ECMAScript 7 009
第 2 章　let和const命令 010
2.1 let命令 010
基本用法 010
不存在变量提升 011
暂时性死区 011
不允许重复声明 012
2.2 块级作用域 013
为什么需要块级作用域 013
ES6的块级作用域 013
2.3 const命令 015
2.4 跨模块常量 016
2.5 全局对象的属性 017
第 3 章　变量的解构赋值 018
3.1 数组的解构赋值 018
基本用法 018
默认值 020
3.2 对象的解构赋值 021
3.3 字符串的解构赋值 024
3.4 数值和布尔值的解构赋值 024
3.5 函数参数的解构赋值 024
3.6 圆括号问题 025
不能使用圆括号的情况 025
可以使用圆括号的情况 026
3.7 用途 026
第 4 章　字符串的扩展 029
4.1 字符的Unicode表示法 029
4.2 codePointAt() 030
4.3 String.fromCodePoint() 031
4.4 字符串的遍历器接口 031
4.5 at() 032
4.6 normalize() 032
4.7 includes()，startsWith()，endsWith() 033

4.8 repeat() 033
4.9 padStart()，padEnd() 034
4.10 模板字符串 034
4.11 实例：模板编译 037
4.12 标签模板 038
4.13 String.raw() 042
第 5 章　正则的扩展 043
5.1 RegExp 构造函数 043
5.2 字符串的正则方法 043
5.3 u 修饰符 043
5.4 y 修饰符 045
5.5 sticky 属性 047
5.6 flags 属性 048
5.7 RegExp.escape() 048
第 6 章　数值的扩展 049
6.1 二进制和八进制数值表示法 049
6.2 Number.isFinite()，Number.isNaN() 049
6.3 Number.parseInt()，Number.parseFloat() 051
6.4 Number.isInteger() 051
6.5 Number.EPSILON 051
6.6 安全整数和Number.isSafeInteger() 052
6.7 Math 对象的扩展 054
Math.trunc() 054
Math.sign() 054
Math.cbrt() 055
Math.clz32() 055
Math.imul() 056
Math.fround() 056
Math.hypot() 057
对数方法 057
三角函数方法 058
6.8 指数运算符 058
第 7 章　数组的扩展 059
7.1 Array.from() 059
7.2 Array.of() 061
7.3 数组实例的copyWithin() 062
7.4 数组实例的find() 和findIndex() 063
7.5 数组实例的fill() 063
7.6 数组实例的entries()、keys() 和values() 064
7.7 数组实例的includes() 064
7.8 数组的空位 065
7.9 数组推导 067
第 8 章　函数的扩展 069
8.1 函数参数的默认值 069
基本用法 069
与解构赋值默认值结合使用 070
参数默认值的位置 071
函数的length 属性 072
作用域 073

应用 074
8.2 rest 参数 074
8.3 扩展运算符 075
含义 075
替代数组的apply 方法 076
扩展运算符的应用 077
8.4 name 属性 079
8.5 箭头函数 080
基本用法 080
使用注意点 081
嵌套的箭头函数 083
8.6 函数绑定 084
8.7 尾调用优化 085
什么是尾调用 085
尾调用优化 086
尾递归 087
递归函数的改写 088
8.8 函数参数的尾逗号 089
第 9 章　对象的扩展 090
9.1 属性的简洁表示法 090
9.2 属性名表达式 092
9.3 方法的name 属性 093
9.4 Object.is() 094
9.5 Object.assign() 095
9.6 属性的可枚举性 097
9.7 属性的遍历 098
9.8 __proto__ 属性，Object.setPrototypeOf()，Object.getPrototypeOf() 099
9.9 对象的扩展运算符 101
第10 章　Symbol 103
10.1 概述 103
10.2 作为属性名的Symbol 105
10.3 实例：消除魔术字符串 106
10.4 属性名的遍历 107
10.5 Symbol.for()，Symbol.keyFor() 109
10.6 内置的Symbol 值 110
Symbol.hasInstance 110
Symbol.isConcatSpreadable 110
Symbol.species 111
Symbol.match 111
Symbol.replace 112
Symbol.search 112
Symbol.split 112
Symbol.iterator 112
Symbol.toPrimitive 113
Symbol.toStringTag 113
Symbol.unscopables 114
第11 章　Proxy 和Reflect 116
11.1 Proxy 概述 116
11.2 Proxy 实例的方法 119

get() 119
set() 121
apply() 122
has() 123
construct() 123
deleteProperty() 124
defineProperty() 125
enumerate() 125
getOwnPropertyDescriptor() 126
getPrototypeOf() 126
isExtensible() 127
ownKeys() 127
preventExtensions() 128
setPrototypeOf() 128
11.3 Proxy.revocable() 129
11.4 Reflect 概述 129
11.5 Reflect 对象的方法 130
第12 章　二进制数组 133
12.1 ArrayBuffer 对象 134
概述 134
ArrayBuffer.prototype.byteLength 135
ArrayBuffer.prototype.slice() 135
ArrayBuffer.isView() 136
12.2 TypedArray 视图 136
概述 136
构造函数 136
数组方法 138
字节序 140
BYTES_PER_ELEMENT 属性 141
ArrayBuffer 与字符串的互相转换 142
溢出 142
TypedArray.prototype.buffer 143
TypedArray.prototype.byteLength，　TypedArray.prototype.byteOffset 143
TypedArray.prototype.length 143
TypedArray.prototype.set() 144
TypedArray.prototype.subarray() 144
TypedArray.prototype.slice() 144
TypedArray.of() 144
TypedArray.from() 145
12.3 复合视图 145
12.4 DataView 视图 146
12.5 二进制数组的应用 148
AJAX 148
Canvas 148
WebSocket 149
Fetch API 149
File API 149
第13 章　Set 和Map 数据结构 151
13.1 Set 151

基本用法 151
Set 实例的属性和方法 152
遍历操作 153
13.2 WeakSet 155
13.3 Map 157
Map 结构的目的和基本用法 157
实例的属性和操作方法 159
遍历方法 160
与其他数据结构的互相转换 162
13.4 WeakMap 164
第14 章　Iterator 和for...of 循环 166
14.1 Iterator 的概念 166
14.2 数据结构的默认Iterator 接口 168
14.3 调用Iterator 接口的场合 172
14.4 字符串的Iterator 接口 173
14.5 Iterator 接口与Generator 函数 174
14.6 遍历器对象的return()、throw() 175
14.7 for...of 循环 175
数组 175
Set 和Map 结构 177
计算生成的数据结构 177
类似数组的对象 178
对象 179
与其他遍历语法的比较 180
第15 章　Generator 函数 181
15.1 简介 181
基本概念 181
yield 语句 182
与Iterator 接口的关系 184
15.2 next 方法的参数 185
15.3 for...of 循环 187
15.4 Generator.prototype.throw() 189
15.5 Generator.prototype.return() 193
15.6 yield* 语句 194
15.7 作为对象属性的Generator 函数 199
15.8 Generator 函数的this 200
15.9 Generator 函数推导 201
15.10 含义 202
Generator 与状态机 202
Generator 与协程 202
15.11 应用 203
第16 章　Promise 对象 208
16.1 Promise 的含义 208
16.2 基本用法 208
16.3 Promise.prototype.then() 211
16.4 Promise.prototype.catch() 212
16.5 Promise.all() 215
16.6 Promise.race() 216
16.7 Promise.resolve() 216

16.8 Promise.reject() 217
16.9 两个有用的附加方法 218
done() 218
finally() 218
16.10 应用 219
加载图片 219
Generator 函数与Promise 的结合 219
16.11 async 函数 220
第17 章　异步操作和async 函数 221
17.1 基本概念 221
异步 221
回调函数 221
Promise 222
17.2 Generator 函数 223
协程 223
Generator 函数的概念 223
Generator 函数的数据交换和错误处理 224
异步任务的封装 224
17.3 Thunk 函数 225
参数的求值策略 225
Thunk 函数的含义 226
JavaScript 语言的Thunk 函数 226
Thunkify 模块 227
Generator 函数的流程管理 228
Thunk 函数的自动流程管理 229
17.4 co 模块 230
基本用法 230
co 模块的原理 230
基于Promise 对象的自动执行 231
co 模块的源码 232
处理并发的异步操作 233
17.5 async 函数 234
含义 234
async 函数的实现 235
async 函数的用法 236
注意点 236
与Promise、Generator 的比较 238
第18 章　Class 240
18.1 Class 基本语法 240
概述 240
constructor 方法 243
实例对象 243
name 属性 244
Class 表达式 245
不存在变量提升 245
严格模式 246
18.2 Class 的继承 246
基本用法 246
类的prototype 属性和__proto__ 属性 247

extends 的继承目标 248
Object.getPrototypeOf() 249
super 关键字 249
实例的__proto__ 属性 250
18.3 原生构造函数的继承 250
18.4 Class 的取值函数（getter）和存值函数（setter） 253
18.5 Class 的Generator 方法 254
18.6 Class 的静态方法 254
18.7 Class 的静态属性 255
18.8 new.target 属性 256
18.9 Mixin 模式的实现 258
第19 章　修饰器 259
19.1 类的修饰 259
19.2 方法的修饰 261
19.3 为什么修饰器不能用于函数 262
19.4 core-decorators.js 263
@autobind 263
@readonly 263
@override 263
@deprecate（别名@deprecated） 264
@suppressWarnings 265
19.5 使用修饰器实现自动发布事件 265
19.6 Mixin 266
19.7 Trait 267
19.8 Babel 转码器的支持 269
第20 章　Module 270
20.1 严格模式 271
20.2 export 命令 271
20.3 import 命令 272
20.4 模块的整体加载 273
20.5 module 命令 274
20.6 export default 命令 274
20.7 模块的继承 276
20.8 ES6 模块加载的实质 277
20.9 循环加载 278
CommonJS 模块的加载原理 279
CommonJS 模块的循环加载 279
ES6 模块的循环加载 280
20.10 ES6 模块的转码 282
ES6 module transpiler 282
SystemJS 282
第21 章　编程风格 284
21.1 块级作用域 284
let 取代var 284
全局常量和线程安全 285
严格模式 285
21.2 字符串 285
21.3 解构赋值 285
21.4 对象 286

21.5 数组 288
21.6 函数 288
21.7 Map 结构 289
21.8 Class 290
21.9 模块 291
21.10 ESLint 的使用 292
第22 章　读懂ECMAScript 规格 293
22.1 概述 293
22.2 相等运算符 293
22.3 数组的空位 295
22.4 数组的map 方法 296
第23 章　参考链接 299
23.1 官方文件 299
23.2 综合介绍 299
23.3 let 和const 302
23.4 解构赋值 302
23.5 字符串 303
23.6 正则 304
23.7 数值 304
23.8 数组 304
23.9 函数 305
23.10 对象 306
23.11 Proxy 和Reflect 306
23.12 Symbol 308
23.13 二进制数组 308
23.14 Set 和Map 309
23.15 Iterator 310
23.16 Generator 311
23.17 Promise 对象 312
23.18 Class 314
23.19 Decorator 314
23.20 Module 315
23.21 工具 316

《ES6 标准入门（第2版）》

精彩短评

1、不得不说，这本书拍本真的太差了
2、javascript缝补的还是不太行啊
3、作为一名Python农民工，ES6的语法看着还是很舒服的。但这本书一般，定价略高，介绍顺序略混乱，前面各种用的很欢但是留到很后面才介绍对应的语用法。。。
4、写的一般，建议看一下这篇文章，比书写的好很多。https://zhuanlan.zhihu.com/p/20233021
5、细节与参引的翻译都很到位
6、缺少大一点的例子
7、阮老师这本书写得很仔细，看完了个大概，对ES6的语法基本了解了，代码示例也很详细，看来是时候全部改用ES6了。提到的ES7的知识也很有用~
8、读完了第一遍，适合于精神发散，看mdn文档想睡觉的人去阅读。等精神极好的时候，花一个上午再看一遍，再用一个下午写完读书笔记，写写react的demo应该能ES6初步入门了
9、两年多没写代码了，赶紧补补课...
10、ES6的入门级教材，讲的还算详尽，慢慢看，先扫读了一遍，觉得挺好的。
11、看到generator和promise，我选择跳过
12、恶补基础
13、作为一个伪前端，标准是要读的。
还得再补充一下ES6的转换到普通浏览器的流程。
用上ES6，写Javascript 666。
14、学习es6新特性就是从这本书开始的
15、初步入门ES6还是挺合适的
16、粗读一遍，前面几章关于基本语法及编码规范的对初学者帮助比较大，后面内容暂时用不到
17、es6很强大。书写的简明易懂，但要想真正明白还得动手练
18、已经离不开babel了...

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.tushu000.com