

どうやってECMAScript 6
を学び始めるか



azu

azu

📍 Tokyo/Japan

✉ info@efcl.info

🌐 http://efcl.info/

🕒 Joined on 7 years ago

312

Followers

4k

Starred

204

Following

Organizations



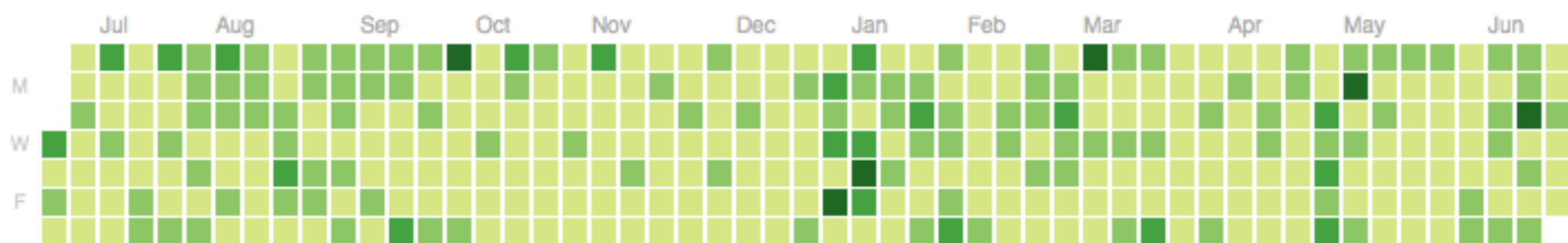
Popular repositories

promises-book JavaScript Promiseの本	298 ★
awesome-commit-english コミット英語についての記事まとめ	255 ★
NSDate-Escort NSDate utility library that is compatible with NSDate-Extensions API.	182 ★
github-reader [node-webkit] GitHub client app - Viewer for Notifications and News Feed.	92 ★
textlint The pluggable linting tool for text(plain text and markdown).	54 ★

Repositories contributed to

jser/jser.info JSer.infoのデータリポジトリです	80 ★
efcl/efcl.github.io Web Scratchブログのリポジトリ	24 ★
jser/jser.github.io JSer.info のプログラリポジトリ	17 ★
meta-ecmascript/es6-draft-revision ECMAScript 2015(ES6) revision repository.	5 ★
power-assert-js/espower-babel power-assert instrumentor for babel(6to5) + mocha.	43 ★

Contributions



Summary of Pull Requests, issues opened, and commits. [Learn more.](#)

Contributions in the last year

6,632 total

Jun 25, 2014 – Jun 25, 2015

Longest streak

598 days

November 6 – June 26

Current streak

598 days

November 6 – June 26

ECMAScript 5を知る

ES6 の前に ES5 大丈夫?

- Property Descriptor
- Object.create()
- Strict Mode
- [Thinking About ECMAScript 5 Parts - Tech.pro](#)

O'REILLY®



Speaking JavaScript

JavaScript

AN IN-DEPTH GUIDE FOR PROGRAMMERS

Dr. Axel Rauschmayer

なぜ ES5 が大事か？

- (恐らく)学ぶ過程でBabelのようなTranspilerに手を出してしまう
- TranspilerはES5の機能を使いきって実装されている
- JavaScript Transformation - JSConf 2015 // Speaker Deck

ES6の概要を知る

今すぐ活かす!

取新

vol.
87
2015

JavaScript

WEB+DB PRESS Vol.87

進化した仕様 ECMAScript 6 をまるごと解説


[速習] Google


世界一のインフラで
サービスを動かす!

Cloud Platform



ECMAScript 6 git.io/es6features

You can clone with HTTPS, SSH, or Subversion. 

 Clone in Desktop

 Download ZIP

Introduction

ECMAScript 6, also known as ECMAScript 2015, is the latest version of the ECMAScript standard. ES6 is a significant update to the language, and the first update to the language since ES5 was standardized in 2009. Implementation of these features in major JavaScript engines is underway now.

See the [ES6 standard](#) for full specification of the ECMAScript 6 language.

ES6 includes the following new features:

- arrows
- classes
- enhanced object literals
- template strings
- destructuring
- default + rest + spread
- let + const
- iterators + for..of
- generators
- unicode
- modules
- module loaders

git.io/es6features

实行环境

実行環境を揃える

→ Browser(JavaScriptエンジン)

→ MSEdge、Firefox、Chrome、Safari(Webkit JSC)

→ Babel(Transpiler)

Browserへの実装状況

Please note that *some of these tests* represent existence, not functionality or full conformance. Sort by number of features? Show obsolete platforms

- Show unstable platforms?
- V8 SpiderMonkey JavaScriptCore Chakra Carakan KJS Other
- Useful feature (1 point) Significant feature (2 points) Landmark feature (4 points)

Feature name	Compilers/polyfills										Desktop brows				
	60%	59%	71%	31%	18%	25%	17%	4%	14%	63%	34%	60%	61%	62%	41%
	Current browser	Trident	Brave	Arcton	Chakra	Carakan	KJS	Other	Edge ^[3]	FF 31 ESR	FF 38	FF 39	FF 40	CH 43, OP 30 ^[4]	
Optimisation															
<input type="checkbox"/> proper tail calls (tail call optimisation)	0/2	0/2	1/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2
Syntax															
<input type="checkbox"/> default function parameters	5/6	3/5	5/6	4/6	0/6	3/5	0/6	0/6	0/6	0/6	3/6	3/6	3/6	3/6	0/6
<input type="checkbox"/> rest parameters	4/5	4/5	4/5	2/5	3/5	3/5	0/5	0/5	0/5	5/5	3/5	4/5	4/5	4/5	0/5
<input type="checkbox"/> spread (...) operator	12/12	12/12	12/12	2/12	1/12	2/12	0/12	0/12	0/12	10/12	8/12	12/12	12/12	12/12	0/12
<input type="checkbox"/> object literal extensions	6/6	6/6	6/6	4/6	5/6	5/6	0/6	0/6	0/6	6/6	0/6	6/6	6/6	6/6	3/6
<input type="checkbox"/> for..of loops	6/8	8/8	8/8	5/8	1/8	2/8	0/8	0/8	0/8	5/8	4/8	6/8	6/8	6/8	6/8
<input type="checkbox"/> octal and binary literals	4/4	2/4	4/4	2/4	0/4	2/4	0/4	0/4	0/4	4/4	2/4	4/4	4/4	4/4	4/4
<input type="checkbox"/> template strings	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	2/3	0/3	3/3	3/3	3/3	3/3
<input type="checkbox"/> RegExp "y" and "u" flags	1/2	1/2	1/2	0/2	0/2	0/2	0/2	0/2	0/2	1/2	1/2	1/2	1/2	1/2	0/2
<input type="checkbox"/> destructuring	23/32	26/32	29/32	16/32	12/32	21/32	0/32	0/32	0/32	0/32	17/32	23/32	23/32	23/32	0/32
<input type="checkbox"/> Unicode code point escapes	0/2	1/2	1/2	1/2	0/2	1/2	0/2	0/2	0/2	2/2	0/2	0/2	0/2	1/2	0/2

ECMAScript 6

compatibility table

Info choose Browsers

Use sql whildchars (%) / case sensitiv!

Android 4.3
 Chrome 45.0.2440
 Firefox 38.0
 Firefox 41.0
 IE 8.0
 IE 9.0
 IE 10.0
 IE 11.0
 IE / Edge 12
 Mob Safari 8.0
 Opera 12.17
 Safari 8.0.5
 All Browsers
 Specification
 Tests

ANGLE_instanced_arrays
 ActiveXObject
 AesGcmEncryptResult
 AnalyserNode
 Animation
 AnimationEffectReadOnly
 AnimationEffectTiming
 AnimationEvent
 AnimationPlaybackRate
 AnimationTimeline
 AnonXMLHttpRequest
 AppBannerPromptResult
 ApplicationCache
 ApplicationCacheErrorEvent
 AriaRequestEvent
 Array
 ArrayBuffer
 Atomics
 Attr
 Audio
 AudioBuffer
 AudioBufferSourceNode
 AudioContext

							o	f				+	?	
				f	f	f	u					+	?	
							o					+	?	
	f	f	f					f	o		o	+	?	
			f									+	?	
?	f	f	f									+	?	
?	f	f	f									+	?	
	f	f	f									+	?	
												+	?	
	f		f									+	?	
									f			+	?	
f	f	f	f	f	f	f	f	f	f	f	f	+	?	✓
f	f	f	f			f	f	f	f	f	f	+	?	✓
			o									+	?	
f	f	f	f	o	o	o	o	f	o	f	o	+	?	✓
f	f	f	f			f	f	f	o	f	o	+	?	
	f	f	f					f	o		o	+	?	✓
	f	f	f					f	o		o	+	?	
	f	f	f					f				+	?	

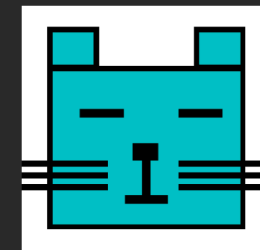
Big JS-Compatibility-Table

Status Page

- それぞれのブラウザが持つ実装ステータスページ
- [The WebKit Open Source Project - Web Platform Status](#)
- [Developer Resources : MS Edge Dev](#)
- [Chromium Dashboard](#)

Transpilerを使う前に

Do **not** use transpilers as a basis to learn
new language features



JavaScript Transformation - JSConf 2015

Transpiler is not Learning Tool

- 次の分類が何を基準にしてるか分からない場合は **Transpiler** で学ぶのは危険が伴う

A	B
Math Extension	SubClassing
spread (...) operator	Proxy
Map	WeakMap
Set	WeakSet
Promise	Symbol
Template Strings	Temporary Dead Zone
.....

ES6を学ぶ前に

- ES5がわからない
 - => ES5を学ぶ
- Transpilerの動作原理がわからない
 - => ブラウザが対応してる機能を優先して学ぶ
- Transpilerがではできないことも理解してる
 - => 容量用法を持って正しく使いましょう

ES6を学ぶ

Constants

[Constants](#)

Scoping

[Block-Scoped Variables](#)[Block-Scoped Functions](#)

Arrow Functions

[Expression Bodies](#)[Statement Bodies](#)[Lexical this](#)

Extended Parameter Handling

[Default Parameter Values](#)[Rest Parameter](#)[Spread Operator](#)

Template Strings

[String Interpolation](#)[Custom Interpolation](#)[Raw String Access](#)

Extended Literals

[Binary & Octal Literal](#)[Unicode String & RegExp Literal](#)

Enhanced Regular Expression

[Regular Expression Sticky Matching](#)

Enhanced Object Properties

[Property Shorthand](#)[Computed Property Names](#)

Constants

Constants

Support for constants (immutable variables).

ECMAScript 6 — syntactic sugar: [reduced](#) | [traditional](#)

```
const PI = 3.141593
PI > 3.0
```

es6-features.org

ECMAScript 6 — syntactic sugar: [reduced](#) | [traditional](#)

```
// only in ES6 though, the help of Object.prototype
// and only in global context and not in a block scope
Object.defineProperty(typeof global === "object" ? global : window, "PI", {
  value:      3.141593,
  enumerable: true,
  writable:   false,
  configurable: false
})
PI > 3.0;
```

*your JavaScript
start coding in ES6 now !!*

EXPLORING ES6

Upgrade to the next version of JavaScript

Exploring ES6



Dr. Axel Rauschmayer

Ecmanauten

getify/You-Dont-Know-

JS

ES6を書く

Tower of Babel

Learn ES6 features using babel

- » BABEL_SETUP [COMPLETED]
- » CLASS [COMPLETED]
- » CLASS_EXTEND
- » MODULES_WITH_NAME
- » MODULES_DEFAULT_EXPORT [COMPLETED]
- » BLOCK_SCOPE
- » ITERATOR_FOR_OF
- » GENERATOR
- » DESTRUCTURE
- » ARROW_FUNCTION [COMPLETED]
- » rest_and_spread [COMPLETED]

tower-of-babel

HELP

EXIT

書く

- Learn ES2015 · Babel
- ライブラリをES6で書いて公開する所から始めよう
- NW.jsやElectronで動くアプリを書く
 - io.js(V8)が入ってる

仕様を読む

Standard ECMA-262

ECMAScript仕様書を読む

・ Issue #47 ・ azu/azu

ECMAScript 6ドラフトの
Diff検索用リポジトリを
作った | Web Scratch

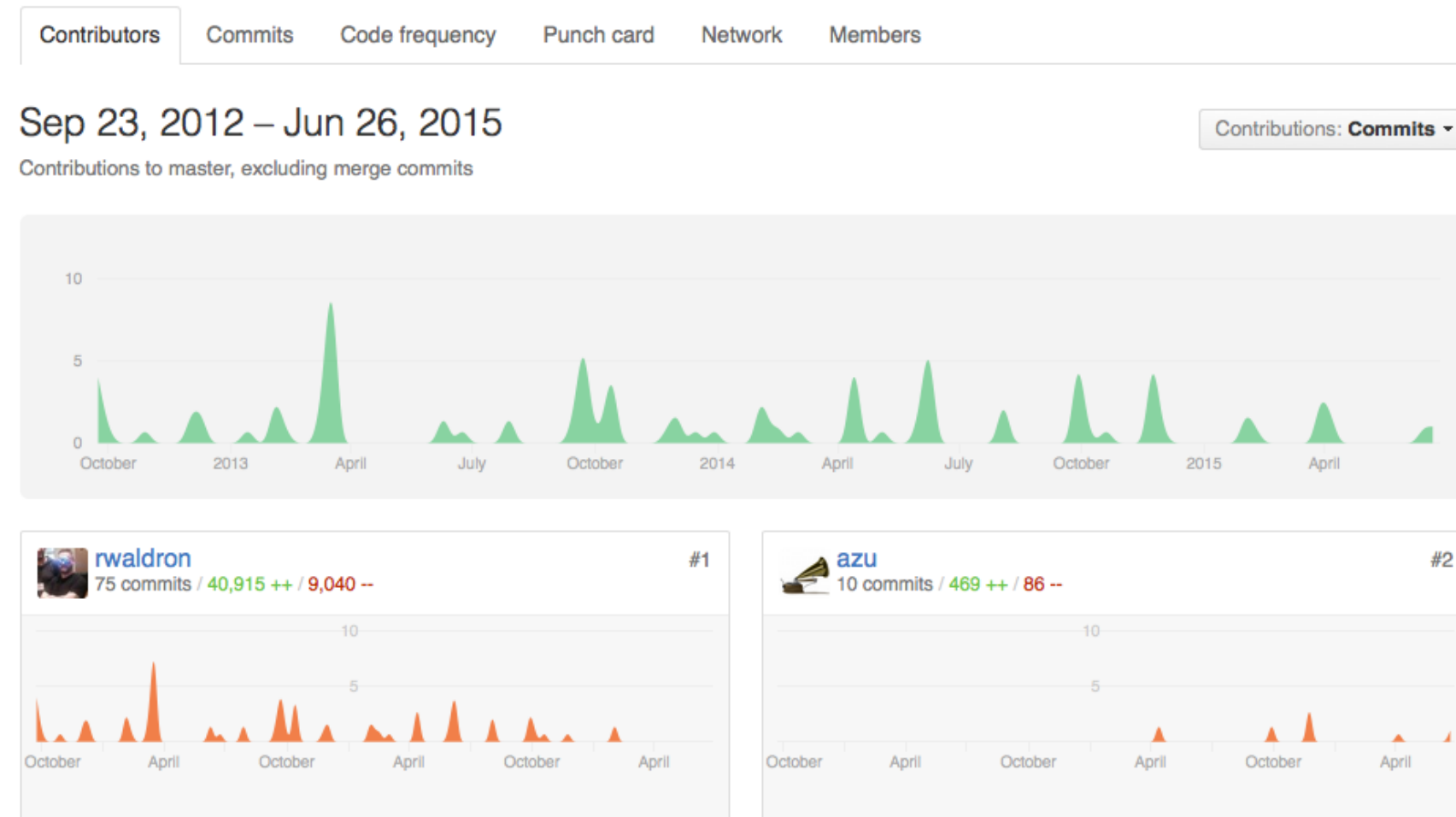
▶ ECMAScript Next

プロポーサラー一覧

- tc39/ecma262
- [Stage 0 Proposals](#)
- [ecmarkup](#)でプロポーサルを書き、GitHubに置く
- + [ES Discuss](#)
- 議論の場所を議論中: [Move es-discuss to discuss.webplatform.org?](#)

tc39/tc39-notes

- ECMAScriptを作成する委員会 (TC39)のミーティングノート
- 大体3ヶ月に1度開催
- #TC39MTGで一人読書会をやってます
- ミーティングに参加してないけどコミットしてる



明日には使えなくなるES7トーク

- 賞味期限 まで 後
- **ES7**以降の仕様策定の流れ
- **Stage**って何か?

まとめ

- WEB+DB PRESS Vol.87で概要を把握
- es6-features.orgで簡単にどんな機能が増えたのを見る
- tower-of-babelで実際に書いてみる
- もっと深く知りたい場合はExploring ES6を読む