

Ask Company



리액트와 함께 장고 부트캠프 / 오리엔테이션

# 리액트 소개

여러분의 파이썬/장고 페이스메이커가 되겠습니다.

이진석  
me@askcompany.kr

# 패러다임의 전환 : "웹 문서 → 웹 애플리케이션"

## 웹 문서

http://naver.com/ Go 12월 1월 4월 25 1998 1999 2000

1,794 captures 12 12월 98 - 17 7월 11

신문검색 | 국외검색 NAVER 마이네이버 | 서치센터

'2000년 표기 문제 New!  
'99년 기묘년을 맞으며...  
개인홈페이지 네이버에서는  
100% 등록

네이버 특별기획  
분야별 추천 사이트 -  
컴퓨터 New!

NAVER 검색  
분류에서 검색  
제목에서 검색 Quick Help 확장검색

NAVER 분류

- **건강/의학**  
건강관리, 의학, 병원, 의료상담
- **연예/오락**  
개인홈페이지, 연예인, 영화
- **교육**  
시험, 자격증, 대학, 유학
- **인문/예술**  
디자인, 패션, 인문과학, 박물관
- **뉴스/미디어**  
신문, 잡지, 최신뉴스, TV
- **자연과학**  
공학, 컴퓨터과학, 생물학
- **레크리에이션**  
게임, 여행, 스포츠, 레저
- **정부/공공기관**  
한국정부, 정치, 국제기구
- **사업/경제**  
취업, 기업, 투자, 재테크
- **지역정보**  
대한민국, 국가, 서울
- **사회과학**  
경제학, 사회학, 언어학
- **참고자료**  
도서관, 사람찾기, 사전, 논문
- **사회/문화**  
결혼, 국경일, 기념일, 기관/단체
- **컴퓨터**  
인터넷, 채팅, 통신, S/W

홈페이지등록 | 최신등록사이트 | 분야별추천사이트  
도움말 | 새소식 | 다른검색사이트 | 광고안내

Microsoft Internet Explorer

대  
격  
변

## 웹 애플리케이션

new york

뉴욕  
New York  
미국  
뉴욕

흐림 · 3°C  
오전 6:54

경로 저장 주변 휴대전화로 보 내기 공유

사진

간략한 정보

뉴욕은 미국 북동부, 뉴욕주의 남쪽 끝에 있는 도시이다. 미국에서 가장 인구가 많은 도시로, 세계에서 가장 인구가 많은 도시 중 하나이며 미국의 최대도시이다. 위키백과

지도 데이터 ©2020 Google 대한민국 학관 의견 보내기 5km

# SPA (Single Page Application) 개발 방법론

웹 문서의 기본 동작에서는 화면 전환 시에

1. 서버로부터 새로운 화면에 대한 HTML/CSS/JavaScript를 받아와서
2. 전체 화면을 새로 그립니다.

➔ 웹 문서에 적합한 방식

SPA 방식의 화면 전환

1. JavaScript를 통해 화면을 변경합니다. ➔ 화면 전환 느낌이 나도록.
2. 필요 시에 백그라운드에서 JavaScript로 서버와 통신을 합니다.

➔ 웹 애플리케이션에 적합한 방식

# jQuery로도 웹 애플리케이션 만들 수 있죠.



<https://youtu.be/vjjshcZoQxw?t=205>



# 웹 프론트엔드 3가지 언어

HTML

CSS

Heavy  
JavaScript

# Heavy JavaScript

최신 자바스크립트를 사용하고 싶어.

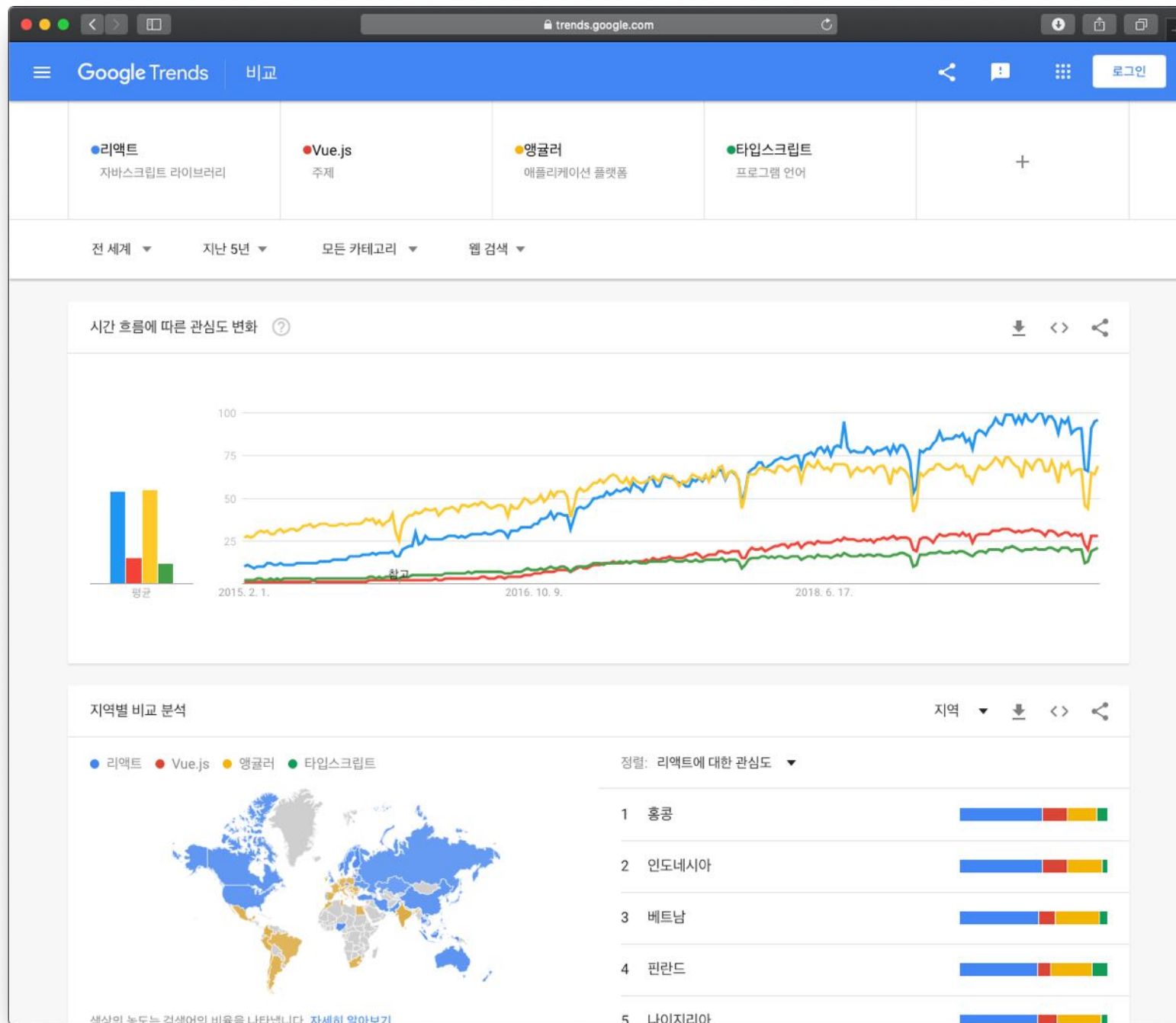
그런데, 브라우저 호환성을 생각하면 ES5를 써야해. class/async/await/generator/arrow function. 등이 없 ... ☹

→ module bundler (webpack 등)을 써서, 코드 변환 (Transpiling) 하면 OK.

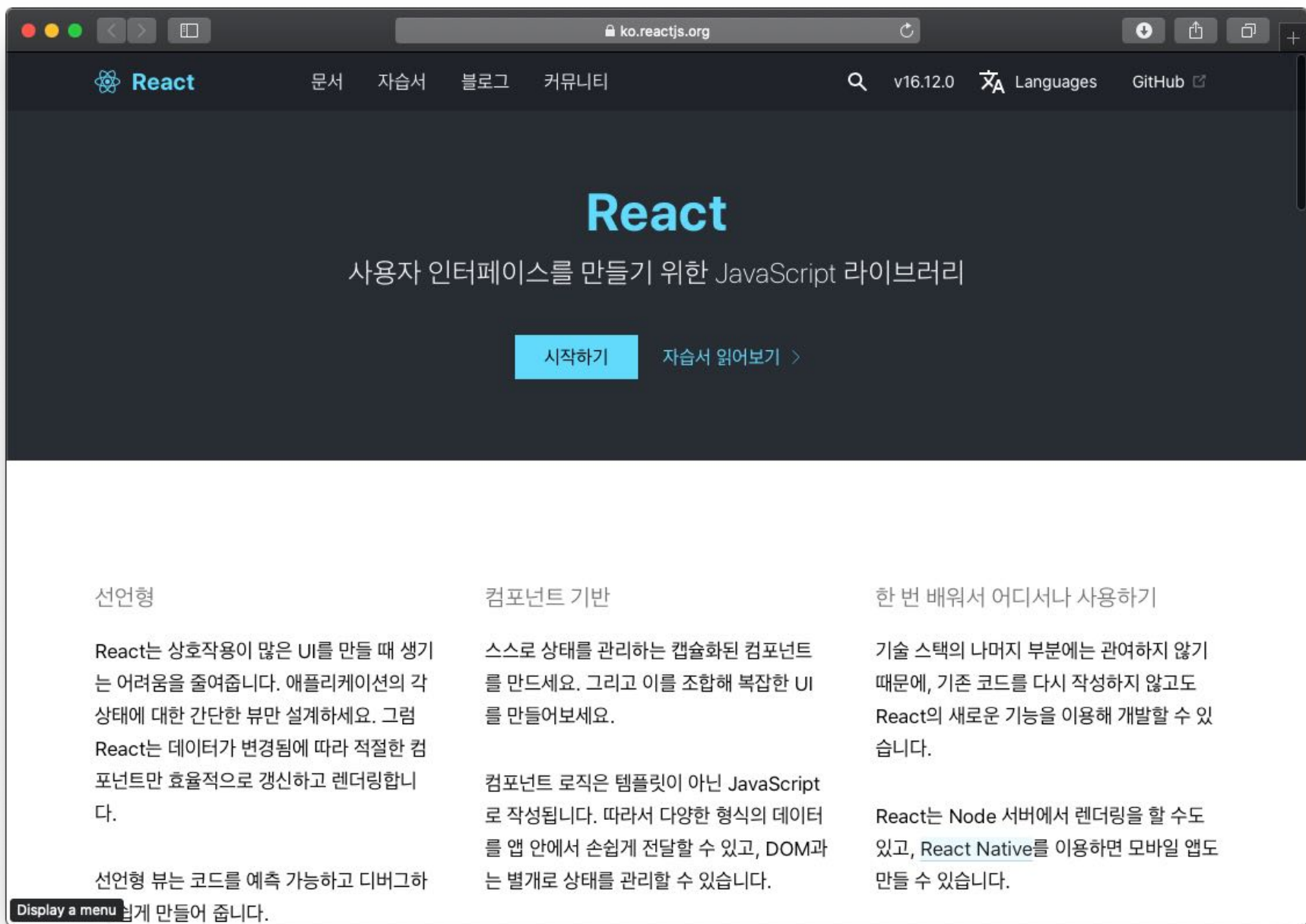
C/C++로 UI 애플리케이션 개발하진 않잖아. JavaScript ...

UI 프레임워크/라이브러리가 필요해.

→ React.js, Vue.js, Angular를 쓰세요.



<http://bit.ly/2U90o0W>





# 다양한 자바스크립트 런타임

## 거의 모든 웹브라우저에서 HTML/CSS/JavaScript 지원

- 웹 브라우저 단에서 구동
- 웹브라우저/버전마다 지원하는 JavaScript Spec이 상이

## 데스크탑에서는 NodeJS 활용

- 데스크탑/서버 단 개발이 가능
- 윈도우에서의 Microsoft의 전폭적인 지지

nodejs로 구동되는 간단 웹서버

```
'use strict';
```

```
const http = require('http');
```

```
const port = process.env.PORT || 8888;
```

```
http.createServer((req, res) => {  
  res.writeHead(200, {'Content-Type': 'text/plain'});  
  res.end('Hello World : ');  
}).listen(port);
```

# 다양한 자바스크립트 버전

ES3 (1999)

ES5 (2009)

**ES6** (2015) → [git.io/es6features](https://git.io/es6features)

ES7 (2016)

ES8 (2017) : async/await 지원

ES9 (2018) : object rest/spread 지원

ES10 (2019)

# 각 브라우저 ECMAScript 6 호환성

<https://kangax.github.io/compat-table/es5/>  
<https://kangax.github.io/compat-table/es6/>

		Compilers/polyfills								Desktop browsers																	
		98%	56%	71%	71%	72%	50%	69%	17%	5%	11%	96%	96%	98%	98%	98%	98%	98%	98%	98%	98%	98%	98%	99%	99%	99%	99%
Feature name	Current browser	Traceur	Babel 6 + core-js 2	Babel 7 + core-js 2	Babel 7 + core-js 3	Closure 2019.07	Type-Script + core-js 3	es6-shim	Konq 4.14 <sup>[1]</sup>	IE 11	Edge 17	Edge 18	FF 60 ESR	FF 67	FF 68 ESR	FF 69 Beta	FF 70 Nightly	CH 74, OP 61	CH 75, OP 62	CH 76, OP 63	CH 77, OP 64	SF 12	SF 12.1	SF 13 Beta	SF TP	WK	
Optimisation																											
proper tail calls (tail call optimisation)	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	2/2	2/2	2/2	2/2	2/2	2/2
Syntax																											
default function parameters	7/7	4/7	4/7	4/7	4/7	5/7	5/7	0/7	0/7	0/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7
rest parameters	5/5	4/5	3/5	3/5	3/5	2/5	4/5	0/5	0/5	0/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5
spread syntax for iterable objects	15/15	15/15	13/15	13/15	14/15	11/15	14/15	0/15	0/15	0/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15
object literal extensions	6/6	6/6	6/6	6/6	6/6	5/6	6/6	0/6	0/6	0/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6
for...of loops	9/9	9/9	9/9	9/9	9/9	6/9	9/9	0/9	0/9	0/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9	9/9
octal and binary literals	4/4	2/4	4/4	4/4	4/4	2/4	4/4	2/4	0/4	0/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4
template literals	7/7	6/7	6/7	6/7	6/7	5/7	5/7	0/7	0/7	0/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	6/7	6/7	7/7	7/7	7/7	7/7
RegExp "y" and "u" flags	6/6	4/6	4/6	4/6	4/6	0/6	0/6	0/6	0/6	0/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6
destructuring declarations	22/22	20/22	21/22	21/22	21/22	20/22	21/22	0/22	0/22	0/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22	22/22
destructuring assignment	24/24	23/24	24/24	24/24	24/24	22/24	24/24	0/24	0/24	0/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24
destructuring parameters	24/24	19/24	21/24	21/24	21/24	20/24	21/24	0/24	0/24	0/24	23/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24
Unicode code point escapes	2/2	1/2	1/2	1/2	1/2	1/2	1/2	0/2	0/2	0/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2
new.target	2/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2
Bindings																											
const	18/18	16/18	16/18	16/18	16/18	16/18	16/18	0/18	2/18	14/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18	18/18
let	14/14	12/14	12/14	12/14	12/14	12/14	12/14	0/14	0/14	12/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14	14/14
block-level function declaration <sup>[18]</sup>	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Functions																											
arrow functions	13/13	11/13	9/13	9/13	9/13	10/13	9/13	0/13	0/13	0/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13
class	24/24	17/24	19/24	19/24	19/24	14/24	19/24	0/24	0/24	0/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24
super	8/8	7/8	4/8	4/8	4/8	7/8	7/8	0/8	0/8	0/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
generators	27/27	24/27	24/27	24/27	24/27	19/27	22/27	0/27	0/27	0/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27	27/27
Built-ins																											
typed arrays	46/46	0/46	45/46	45/46	46/46	0/46	46/46	0/46	8/46	16/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46	46/46
Map	19/19	14/19	19/19	19/19	19/19	15/19	19/19	15/19	0/19	8/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19
Set	19/19	14/19	19/19	19/19	19/19	15/19	19/19	15/19	0/19	8/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19	19/19
WeakMap	12/12	6/12	12/12	12/12	12/12	9/12	12/12	0/12	0/12	6/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12	12/12
WeakSet	11/11	5/11	11/11	11/11	11/11	8/11	11/11	0/11	0/11	0/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11
Proxy <sup>[23]</sup>	34/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34	34/34
Reflect <sup>[24]</sup>	20/20	0/20	15/20	15/20	15/20	13/20	15/20	14/20	0/20	0/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20	20/20
Promise	8/8	4/8	8/8	8/8	8/8	6/8	8/8	7/8	0/8	0/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8

# 트랜스파일링 (Transpiling) (1/2)

최신 데스크톱 브라우저와 모바일 브라우저의 ES6 지원율이 90% 이상

EVERGREEN 브라우저의 경우, ES6 코드는 별도의 트랜스파일링 과정이 필요 X

최신 JavaScript의 Features를 사용하더라도

낮은 버전의 JS를 지원하는 브라우저 지원을 위해

낮은 버전의 JS코드로 변환하는 트랜스파일링이 필요

→ babel을 활용하여 변환

<https://d2.naver.com/helloworld/7495331>

# 트랜스파일링 (Transpiling) (2/2)

```
설치> npm install --global babel-cli  
      npm install --save babel-preset-es2015
```

t.js : ES6 문법의 JavaScript 코드

```
const add = (x=10, y=20) => x + y + 10  
console.log(add(1));
```

Tip: babel-preset-env를 통해  
2015/2016/2017을 한 번에 대응할 수 있습니다.

ES5로 변환 예시

```
셸> babel --presets=es2015 t.js
```

babel-preset-env를 지정할 시에는 --presets=env 를 사용

```
"use strict";
```

```
var add = function add() {  
  var x = arguments.length > 0 && arguments[0] !== undefined ? arguments[0] : 10;  
  var y = arguments.length > 1 && arguments[1] !== undefined ? arguments[1] : 20;  
  return x + y + 10;  
};
```

```
console.log(add(1));
```



# 참고) Transcript (1/2)

## 파이썬 코드를 자바 스크립트 코드로 트랜스파일링

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
<script type="module">
  import * as jquery_demo from "../__target__/jquery_demo.js";
  $(document).ready(jquery_demo.start);
</script>
```

```
# jquery/demo/jquery_demo.py
```

```
__pragma__ ('alias', 'S', '$')
```

```
def start ():
    def changeColors ():
        for div in S__divs:
            S (div) .css ({
                'color': 'rgb({}, {}, {})'.format (* [int (256 * Math.random ()) for i in range (3)]),
            })

    S__divs = S ('div')
    changeColors()
    window.setInterval(changeColors, 500)
```

[https://transcrypt.org/live/transcrypt/demos/jquery\\_demo/jquery\\_demo.html](https://transcrypt.org/live/transcrypt/demos/jquery_demo/jquery_demo.html)

# 참고) \_\_target\_\_/jquery\_demo.js - Transcript (2/2)

// Transcript'ed from Python, 2018-09-30 11:52:30

```
import {AssertionError, AttributeError, BaseException, DeprecationWarning, Exception, IndexError, IterableError, KeyError, NotImplementedError, RuntimeError, StopIteration, UserWarning, ValueError, Warning, __JsIterator__, __PyIterator__, __Terminal__, __add__, __and__, __call__, __class__, __envir__, __eq__, __floordiv__, __ge__, __get__, __getcm__, __getitem__, __getslice__, __getsm__, __gt__, __i__, __iadd__, __iand__, __idiv__, __ijsmod__, __ilshift__, __imatmul__, __imod__, __imul__, __in__, __init__, __ior__, __ipow__, __irshift__, __isub__, __ixor__, __jsUsePyNext__, __jsmod__, __k__, __kwargtrans__, __le__, __lshift__, __lt__, __matmul__, __mergefields__, __mergekwargtrans__, __mod__, __mul__, __ne__, __neg__, __nest__, __or__, __pow__, __pragma__, __proxy__, __pyUseJsNext__, __rshift__, __setitem__, __setProperty__, __setslice__, __sort__, __specialattrib__, __sub__, __super__, __t__, __terminal__, __truediv__, __withblock__, __xor__, abs, all, any, assert, bool, bytearray, bytes, callable, chr, copy, deepcopy, delattr, dict, dir, divmod, enumerate, filter, float, format, getattr, hasattr, input, int, isinstance, issubclass, len, list, map, max, min, object, ord, pow, print, property, py_TypeError, py_iter, py_metatype, py_next, py_reversed, py_typeof, range, repr, round, set, setattr, sorted, str, sum, tuple, zip} from './org.transcript.__runtime__.js';
```

```
var __name__ = "__main__";
export var start = function() {
  var changeColors = function() {
    for (var div of $divs) {
      $(div).css(
        dict({
          color: "rgb({}, {}, {})".format(
            ...(function() {
              var __accu0__ = [];
              for (var i = 0; i < 3; i++) {
                __accu0__.append(int(256 * Math.random()));
              }
              return __accu0__;
            })()
          )
        })
      );
    }
  };
  var $divs = $("div");
  changeColors();
  window.setInterval(changeColors, 500);
};
```

Life is short.  
You need Python and Django.

I will be your pacemaker.

