

JavaScript & TypeScript

Kwestie techniczne

organizacja szkolenia

- Prowadzący szkolenie
 - Tomasz Sułkowski
 - Trener IT
 - FrontEnd Engineer
- Cel i plan szkolenia
- Obowiązki bieżące
- Pytania, dyskusje, potrzeby
- Elastyczność zagadnień



twitter.com/sulco

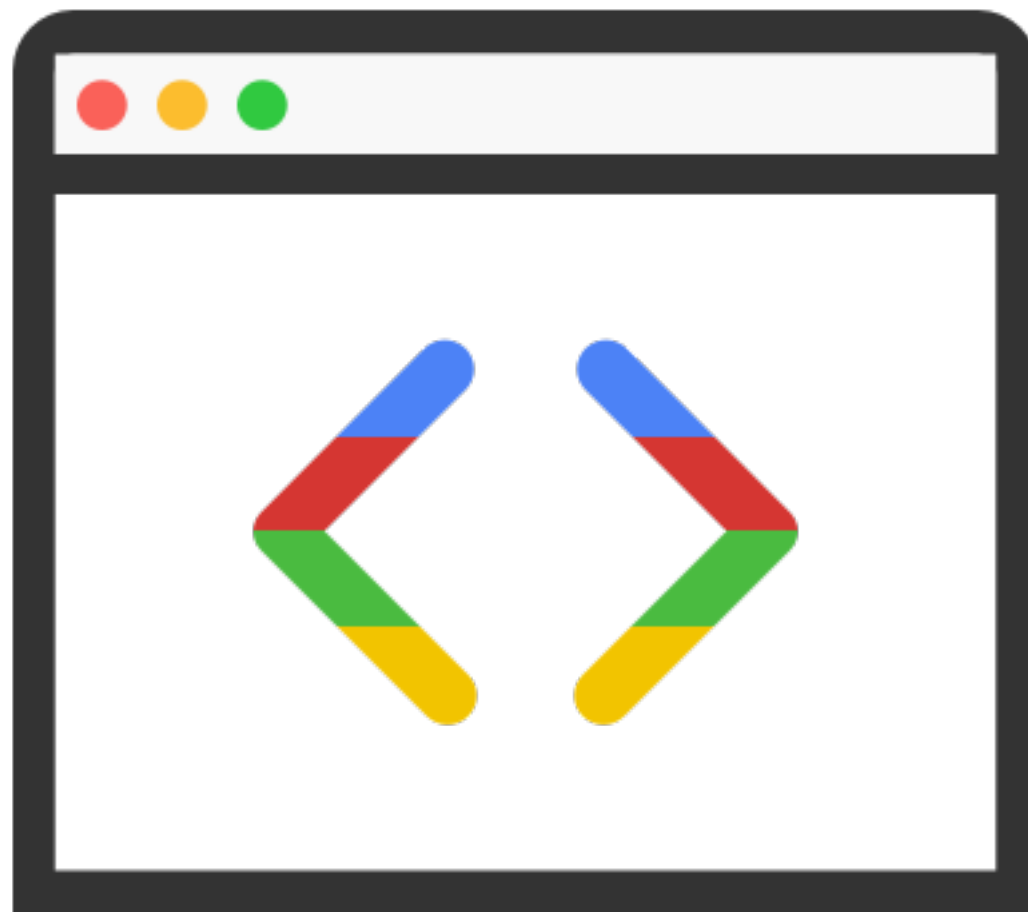
for tips & tricks



Narzędzia



In-browser Developer Tools



In-browser Developer Tools

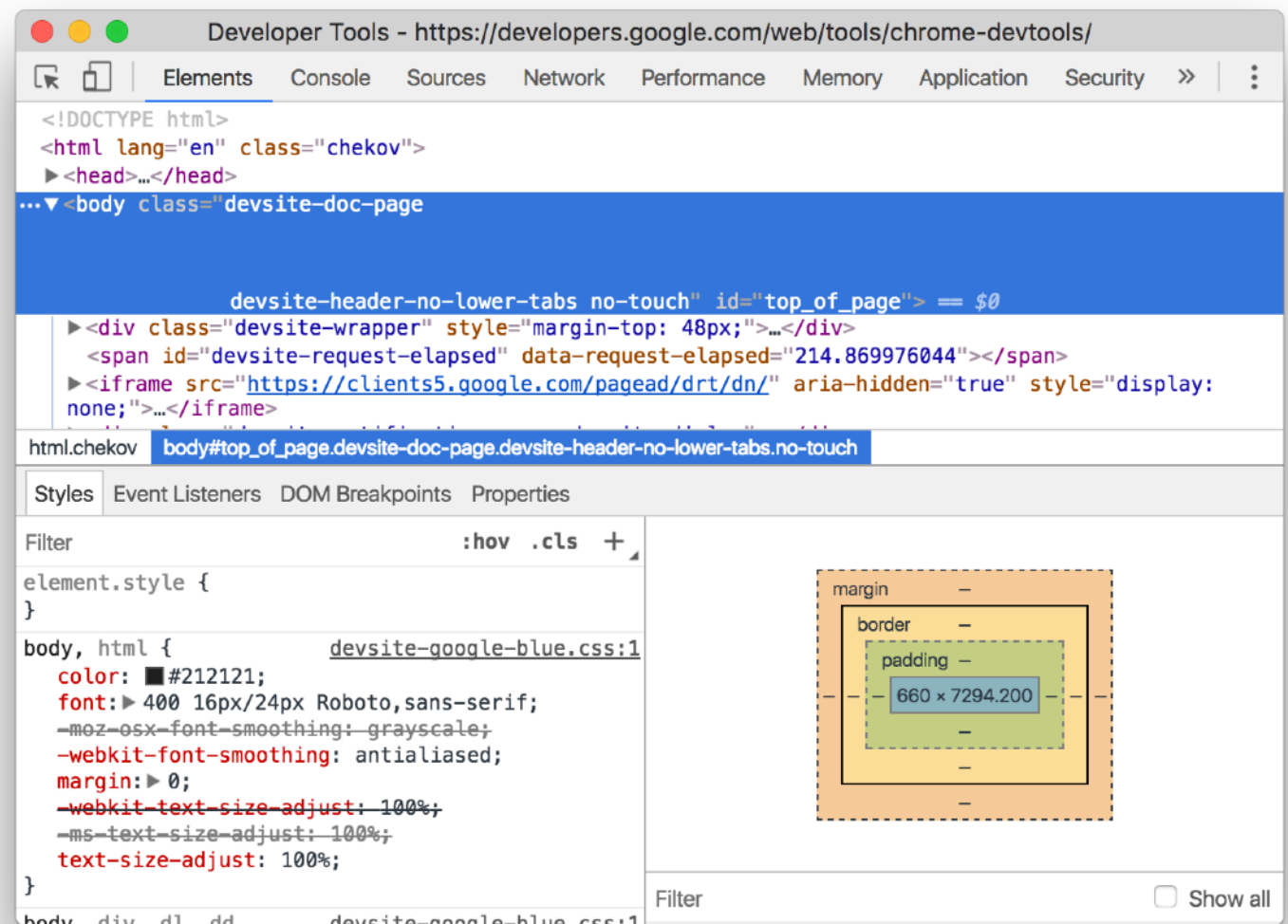
Narzędzia wbudowane we wszystkie współczesne przeglądarki, umożliwiają:

- Debugowanie poszczególnych warstw aplikacji
- Bezpośrednią interakcję z przeglądarką i aplikacją
- Emulację różnych warunków działania

Funkcjonalności te można rozwijać za pomocą pluginów.

DevTools: Elements

- inspekcja/edycja DOM
- inspekcja/edycja styli
- praca ze zdarzeniami



DevTools: Console

- log komunikatów o błędach, ostrzerzeniach itp

- interakcja z JS za pomocą wiersza poleceń

- możliwość przeskoczenia do źródła komunikatu

- przvdadne metody:

```
console.log(myString, myObject)
```

```
console.error('')
```

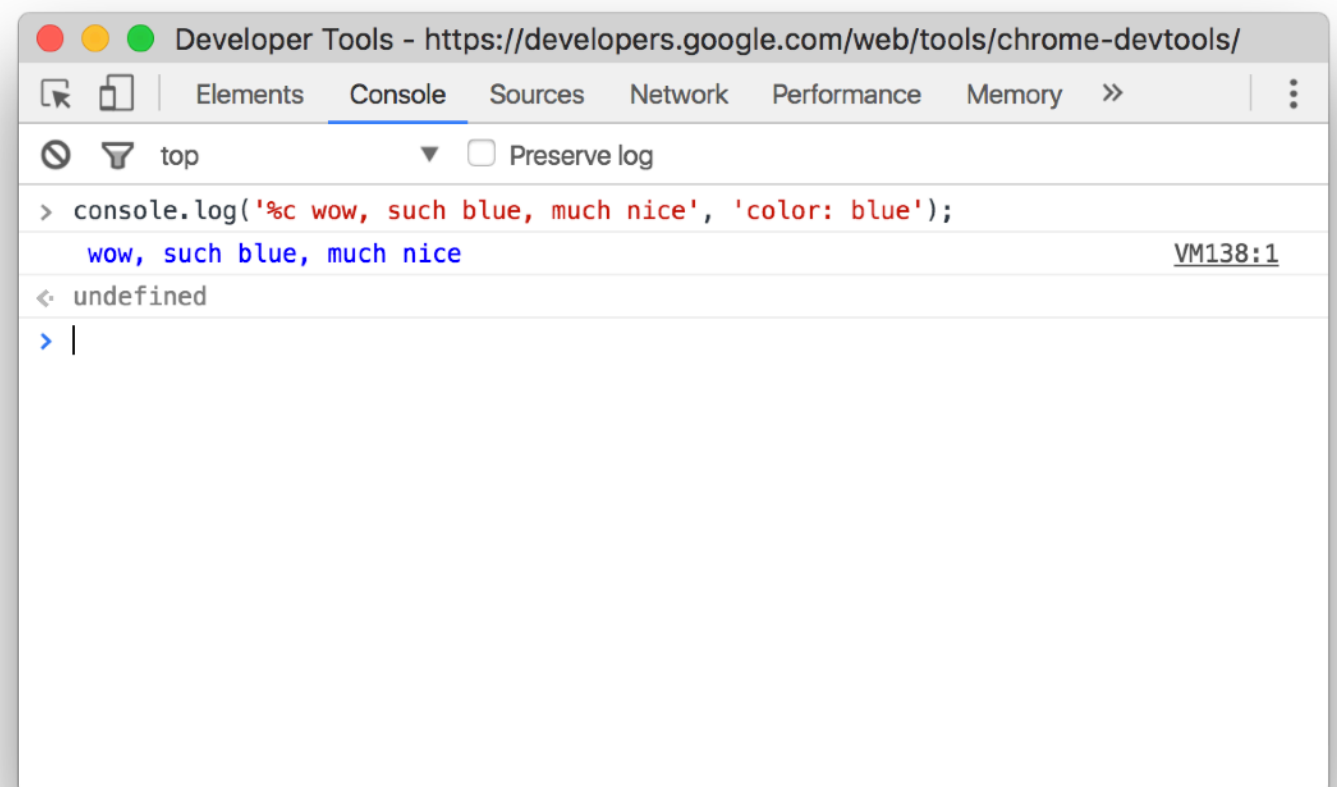
```
// ...
```

```
console.table(myArr)
```

```
console.time('my time')
```

```
console.timeEnd('my time')
```

```
copy()
```



- Konsola może być uruchomiona pod każdą z innych zakładek - wystarczy

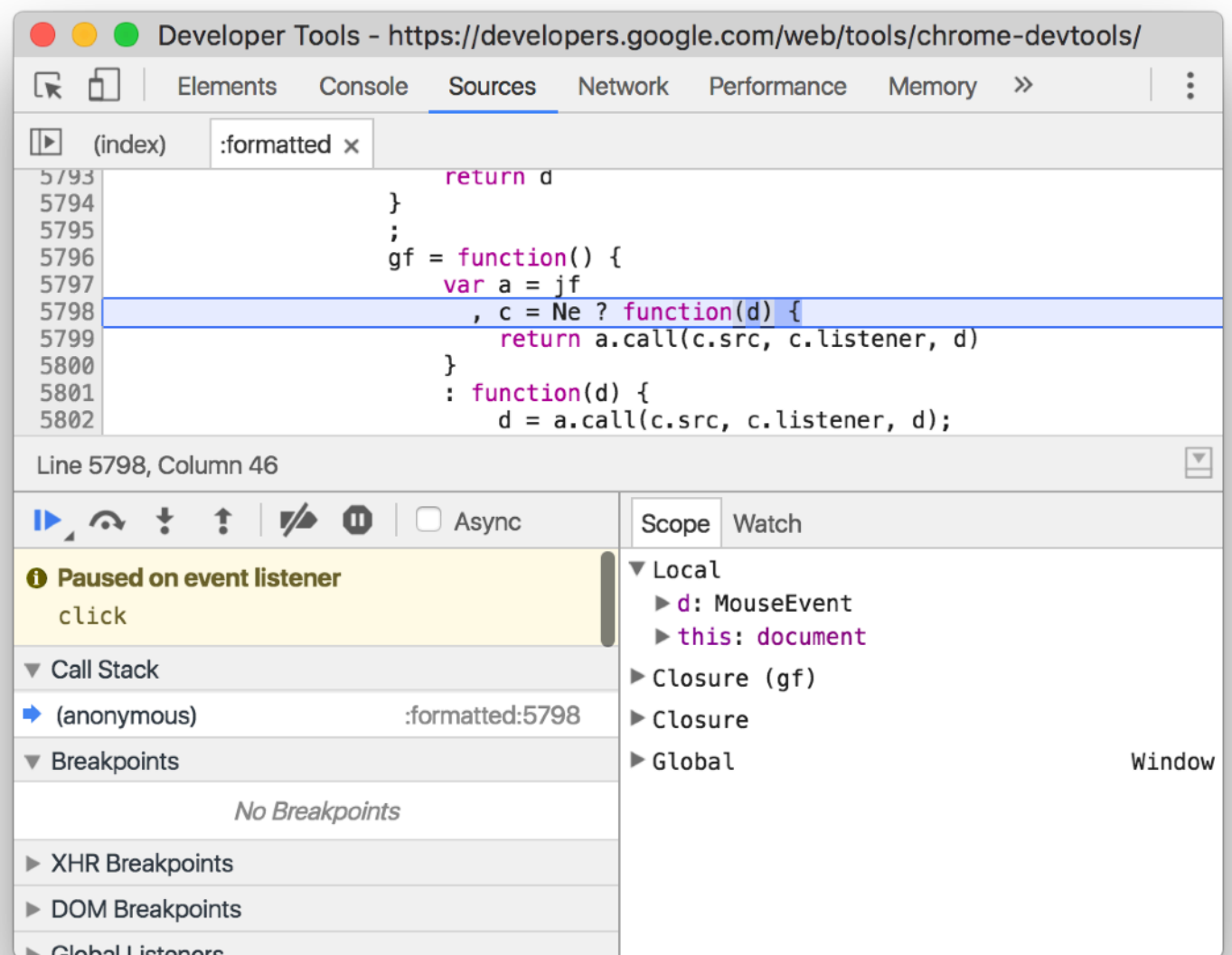


DevTools: Sources

- podgląd źródeł aplikacji
- ustawianie breakpoints i klasyczne możliwości debuggowania



- obsługa sourcemap
- możliwość zapisywania zmian na dysk
- i budowy kolekcji snippetów



A solid yellow rectangle with a subtle drop shadow, centered on a white background.

JavaScript

Funkcje

- Są jedynym ograniczeniem zasięgu zmiennych (**var**)
- Dwa rodzaje funkcji:

```
function foo(a) {  
  var b = 2;  
}
```

```
var bar = function (a) {  
  var b = 2;  
}
```

- *hoisting* działa inaczej w każdym z tych przypadków

IIFE

Immediately Invoked Function Expression

- Sposób na ograniczenie zakresu zmiennych

```
var a = 2;
```

```
(function foo(){
```

```
    var a = 3;
```

```
    console.log( a ); // 3
```

```
})();
```

```
console.log( a ); // 2
```

Closure

"domknięcia"

- Funkcja ma dostęp do swojego leksykalnego zasięgu – nawet, jeśli jest wykonywana poza nim

```
function foo() {  
  var a = 2;
```

```
  function bar() {  
    console.log( a );  
  }
```

```
  return bar;  
}
```

```
var baz = foo();
```

```
baz(); // 2
```

Obiekty

- sposób na organizację **zmiennych** i **funkcji** w logiczne grupy -> stają się wtedy "**własnościami**" i "**metodami**"
- wbudowane obiekty: *String, Date, Math, Boolean, Number, RegExp, Object, Array*
- podstawowe sposoby na stworzenie obiektu to
 - słowo kluczowe **new**
 - literał obiektowy **{ }**

Obiekty

```
var person = new Object();
```

```
person.name = 'Manvendra SK';
```

```
person.age = 23;
```

```
person.speakName = function () {  
    alert('Hello, I' m' + this.name);  
};
```

```
person.speakName();
```

Obiekty

```
var person = {  
  name: 'Manvendra SK',  
  age: 23,  
  speakName: function () {  
    alert('Hello, I' m' + this.name);  
  }  
};
```

```
person.speakName();
```


Constructor function

```
var Person = function (name, age) {  
    this.name = name;  
    this.age = age;  
};
```

```
Person.prototype.speakName = function () {  
    alert('Hello, I' m' + this.name);  
};
```

```
var manvendra = new Person('Manvendra', 23);  
manvendra.speakName();
```

this keyword

- wskazuje na kontekst wywołania
- różne wartości w zależności od sposobu wywołania funkcji

```
var a = {  
  name: 'Alpha',  
  sayMyName: function () {  
    console.log(this.name);  
  }  
}  
// 'Alpha'
```

```
var b = {  
  name: 'Beta',  
  sayMyName: function () {  
    setTimeout(function () {  
      console.log(this.name);  
    });  
  }  
}  
// undefined
```



```

function iIds(startAt, showSessionRoot, iNewNmVal, endActionsVal, iStringVal, seqProp, htmlEncodeRegex) {
  if (SbUtil.dateDisplayType === 'relative') {
    iRange();
  } else {
    iSelActionType();
  }
  iStringVal = notifyWindowTab;
  startAt = addSessionConfigs.sbRange();
  showSessionRoot = addSessionConfigs.elHiddenVal();
  var headerDataPrevious = function(tabArray, iNm) {
    iPredicateVal.SBDB.deferCurrentSessionNotifyVal(function(evalOutMatchedTabUrlsVal) {
      if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
        iPredicateVal.SBDB.normalizeTabList(function(appMsg) {
          if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
            iPredicateVal.SBDB.detailTxt(function(evalOrientationVal) {
              if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                iPredicateVal.SBDB.neutralizeWindowFocus(function(iTokenAddedCallback) {
                  if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                    iPredicateVal.SBDB.evalSessionConfig2(function(sessionNm) {
                      if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                        iPredicateVal.SBDB.iWindow2TabIdx(function(iURLsStringVal) {
                          if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                            iPredicateVal.SBDB.idx7Val(undefined, iStringVal, function(getWindowIndex) {
                              if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                                addTabList(getWindowIndex.rows, iStringVal, showSessionRoot && showSessionRoot.length > 0 ? showSessionRoot : startAt ? [startAt] : []);
                                if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                                  evalAllowLogin(tabArray, iCtrl, showSessionRoot && showSessionRoot.length > 0 ? showSessionRoot : startAt ? [startAt] : []);
                                  if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                                    BrowserAPI.getALLWindowAndTabs(function(iSession1Val) {
                                      if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                                        SbUtil.currentSessionSrc(iSession1Val, undefined, function(initCurrentSession) {
                                          if (!htmlEncodeRegex || htmlEncodeRegex == iContextTo) {
                                            addSessionConfigs.render(matchText(iSession1Val, iStringVal, evalRateActionQualifier, undefined, seqProp));
                                            id: -13,
                                            unfilteredWindowCount: initCurrentSessionCache,
                                            filteredWindowCount: iCtrl,
                                            unfilteredTabCount: parseTabConfig,
                                            filteredTabCount: evalRegisterValue5Val
                                          } : [], cacheSessionWindow, evalRateActionQualifier, undefined, seqProp);
                                        }
                                      }
                                    });
                                  }
                                });
                              }
                            });
                          }
                        });
                      }
                    });
                  }
                });
              }
            });
          }
        });
      }
    });
  }
}

```

Asynchronizność w JavaScript

Callbacks

```
Something.save(function(err, data) {  
    if (err) {  
        //error handling  
        return;  
    }  
    console.log('success', data);  
});
```

Promises

```
Something.save()  
  .then(function() {  
    console.log('success');  
  })  
  .then(updateOtherthing)  
  .then(logResults)  
  .catch(function() {  
    console.error('failure');  
  });
```

Generators

```
function* foo () {  
  var index = 0;  
  while (index < 2) {  
    yield index++;  
  }  
}  
  
var bar = foo();
```

```
console.log(bar.next());  
// { value: 0, done: false }  
console.log(bar.next());  
// { value: 1, done: false }  
console.log(bar.next());  
// { value: undefined, done: true }
```

async / await

```
async function save(Something) {  
  try {  
    await Something.save() //promise  
  } catch (ex) {  
    //error handling  
  }  
  console.log('success');  
}
```

Praca z tablicami

$[a, b, c] \Rightarrow [1, 3, 5]$ **.map**

You have an array and want a new array with every element transformed.

$[a, b, c] \Rightarrow () \Rightarrow \{\}$ **.forEach**

You want to run something (side-effects) for every element of your array.

$[a, b, c] \Rightarrow [a, c]$ **.filter**

You have an array and want to have a subset of this array (only keep some items).

$[a, b, c] \Rightarrow \text{true} \mid \text{false}$ **.some** **|** **.every**

You have an array and to assess something as true or false.

$[a, b, c] \Rightarrow \{...\}$ **.reduce**

You have an array and want to change it to whatever.

Document Object Model

DOM

- Mapuje strukturę HTML/XML na obiekty
- DOM składa się z węzłów ("Node")
- Najpopularniejsze typy (w sumie 12):
`ELEMENT_NODE (1)`, `ATTRIBUTE_NODE (2)`, `TEXT_NODE (3)`
- Typ węzła określa własność `.nodeType`

DOM

- Główny węzeł dokumentu HTML to `document`
- Podstawowe metody węzła elementu:

`.getElementById(...)`

`.getElementsByTagName(...)`

`.getElementsByClassName(...)`

`.querySelector(...)`

`.querySelectorAll(...)`

Zdarzenia

- Nasłuchiwanie zdarzeń w przeglądarce:

```
element.addEventListener('click', callbackFn);
```

```
element.removeEventListener('click', callbackFn);
```

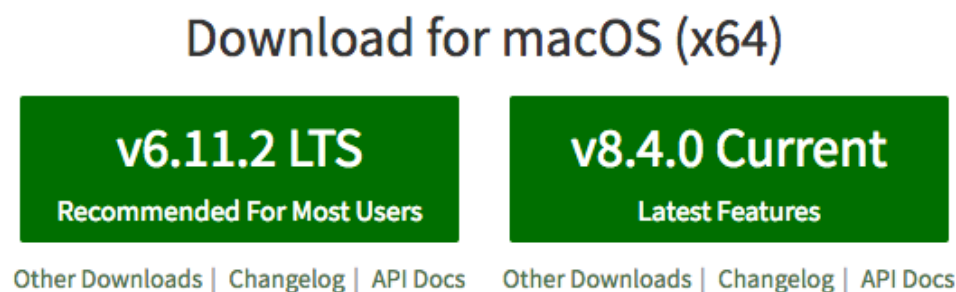
- Zdarzenia podlegają propagacji
(zdarzenie najpierw wywoływane jest na najgłębszym węźle, następnie na jego rodzicu itd)
- Funkcja callback otrzymuje jako parametr obiekt ze szczegółami zdarzenia
- Przydatne metody zdarzenia:

```
event.preventDefault();
```

```
event.stopPropagation();
```



Node.js - instalacja



Oficjalny instalator

- łatwa instalacja
- automatyczna konfiguracja
- update wymaga reinstalacji
- uciążliwa zmiana między wersjami

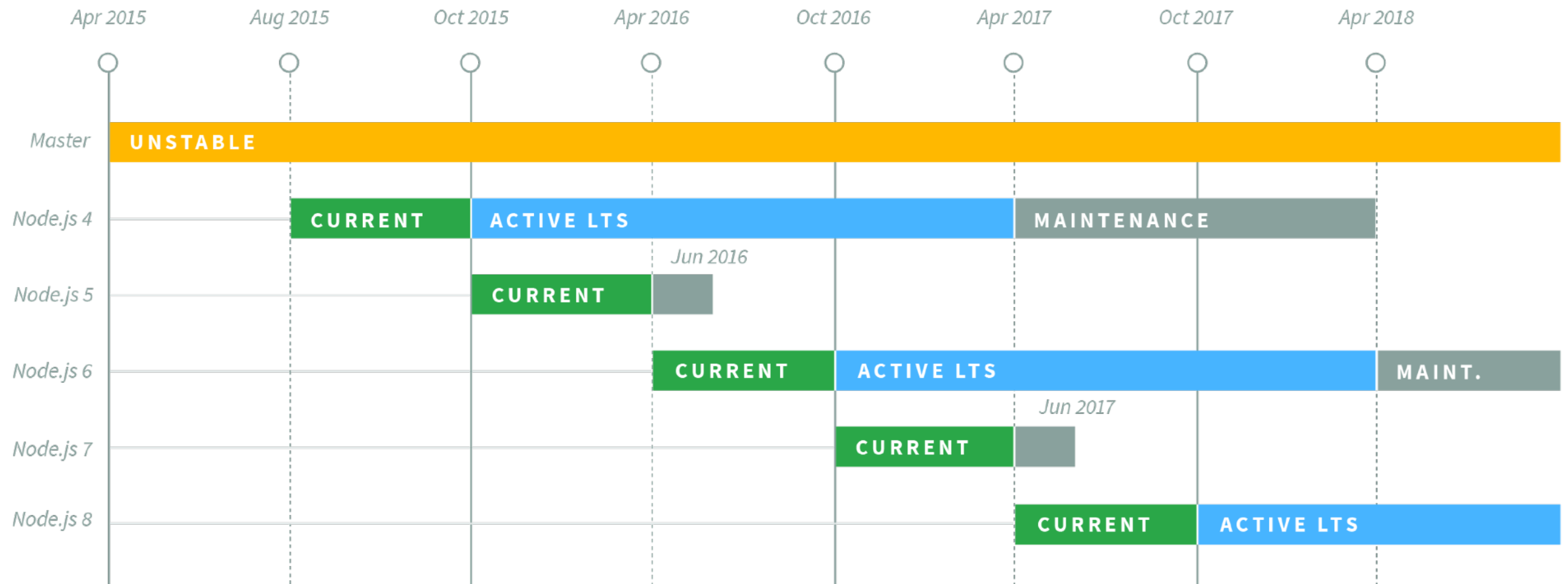
Node Version Managers

- łatwa instalacja dowolnej wersji
- szybkie zmiany wersji
- brak wizarda
- konieczność reinstalacji globalnych modułów

 **creationix / nvm**

 **coreybutler / nvm-windows**

Node.js Long Term Support (LTS) Release Schedule



Moduły w Node.js

"CommonJS"

```
// file: app/greet.js  
module.exports = function myFunction() {  
    console.log('Welcome!');  
};
```

```
// file: app/index.js  
var greet = require('./greet');  
greet();
```

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


Wbudowany moduł http

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

```
var server = http.createServer(handleRequests);  
server.listen(3000, '127.0.0.1');
```

```
function handleRequests(req, res) {  
    var headers = {'Content-Type': 'text/plain'};  
    res.writeHead(200, headers);  
    res.end('We need to move to express soon ...');  
}
```

Zarządzanie pakietami



„Node Packaged Modules”

- system zarządzania zależnościami "dla JS"
- zależności opisywane z dokładnością do wersji w pliku **package.json**
- **npm install** - instaluje pakiety, których jeszcze nie ma w projekcie
- **npm update** - sprawdza, czy istnieją nowsze wersje pakietów + instaluje
- **npm install nazwapakietu --save-dev**
npm install nazwapakietu --save
- instaluje pakiet, dodaje go do listy
- **npm install nazwapakietu -g**
- instaluje pakiet globalnie (zewnętrzne narzędzia)

NPM - przydatne polecenia



„Node Packaged Modules”

`npm init` – inicjalizuje projekt

`npm uninstall nazwa --save`

`npm run zadanie` – uruchamia zadanie z sekcji "scripts" z `package.json`

- skróty:

`npm init -y` → [domyślne wartości]

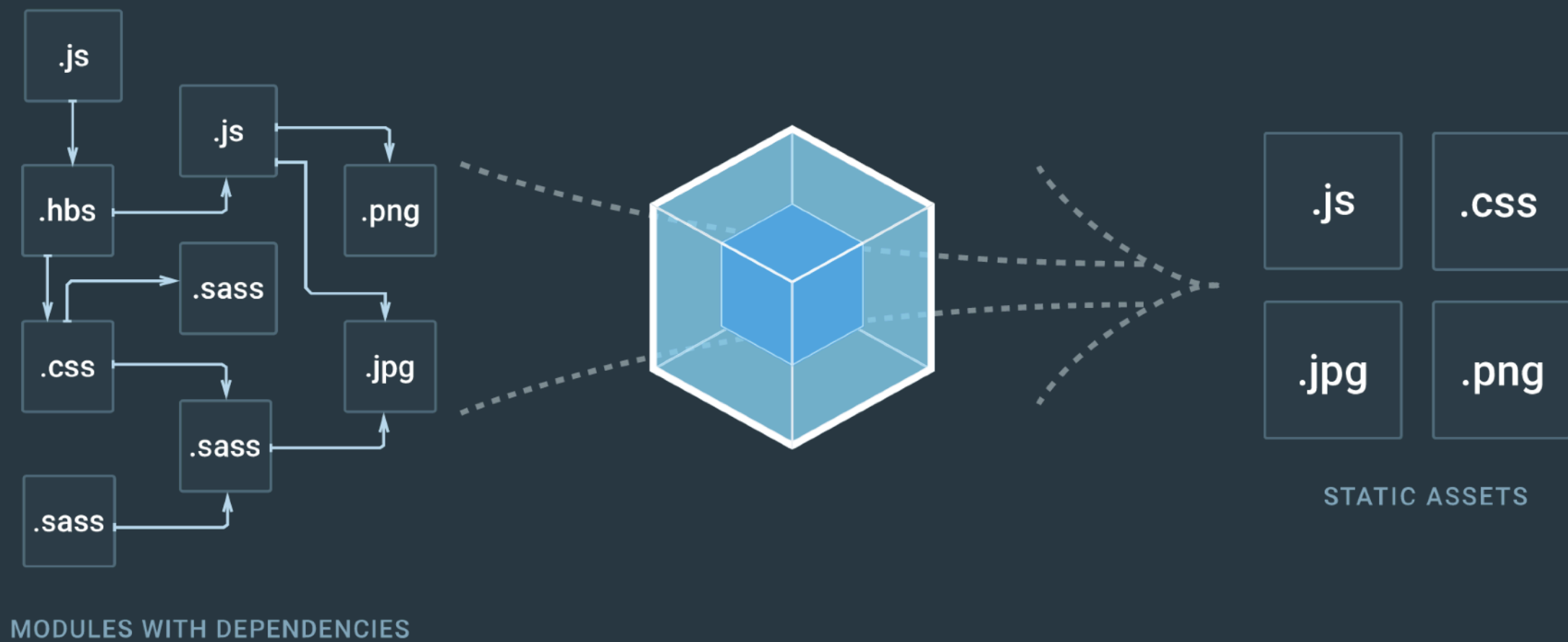
`npm install` → `npm i`

`--save` → `-S`

`--save-dev` → `-D`

(also, consider  **yarn**)

webpack





webpack: intro

- module bundler
- obsługuje wiele formatów modułów: ES2015, AMD, CommonJS (npm)
- traktuje *wszystko* jak moduły (np. scss, html, grafiki)
- React Hot Loader
- dobrze współpracuje z popularnymi task runnerami (Gulp, Grunt)
- de facto standard w środowisku React, popularny również poza

```
// instalacja webpack-cli, globalnie  
npm install webpack -g
```

```
// oraz instalacja lokalna  
npm install webpack --save-dev
```

webpack: struktura

```
module.exports = {  
  entry: {  
    main: './path/to/my/entry/file.js'  
  },  
  output: {  
    filename: 'bundle.js'  
  },  
  plugins: [],  
  module: {  
    rules: [  
      { test: /\.ts$/, use: 'ts-loader' }  
    ]  
  },  
  devtool: 'source-map'  
};
```

~~ECMAScript 6~~

ECMAScript 2015

ECMAScript 2015

- Kolejna (po ECMAScript 5) wersja standardu JavaScript
- Największe dotychczas rozwinięcie tego języka
- Od 2015 aktualizacje standardu będą **wydawane co roku**
- ECMAScript 2016 jest znacznie mniejszą ewolucją
- **5 etapów** do implementacji propozycji:

Strawman (0) → Proposal (1) → Draft (2) → Candidate (3) → Finished (4)

ES2015 - wsparcie

Desktop browsers												Servers/runtimes												Mobile									
15%	60%	79%	90%	63%	85%	90%	90%	90%	93%	98%	98%	10%	21%	53%	99%	99%	11%	10%	18%	50%	56%	93%	67%	97%	55%	11%	15%	29%	10%	21%			
IE 11	Edge 12 ^[4]	Edge 13 ^[4]	Edge 14 ^[4]	FF 38 ESR	FF 45 ESR	FF 46	FF 47	FF 48	FF 49	CH 51, OP 38 ^[1]	CH 52, OP 39 ^[1]	SF 6.1, SF 7	SF 7.1, SF 8	SF 9	SF TP	WK	KQ 4.14 ^[5]	PJS	Node 0.12 ^[6]	Node 4 ^[6]	Node 5 ^[6]	Node 6 ^[6]	Echo JS	XS6	JXA	AN 4.4	AN 5.0	AN 5.1	iOS 7	iOS 8	iOS 9		
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	0/2	0/2	0/2	0/2	0/2	0/2	0/2	2/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2		
0/7	0/7	0/7	7/7	3/7	4/7	4/7	4/7	4/7	4/7	7/7	7/7	0/7	0/7	0/7	7/7	7/7	0/7	0/7	0/7	0/7	0/7	7/7	4/7	7/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7		
0/5	5/5	5/5	5/5	4/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	0/5	0/5	0/5	5/5	5/5	0/5	0/5	0/5	0/5	0/5	5/5	3/5	5/5	0/5	0/5	0/5	0/5	0/5	0/5	0/5		
0/15	12/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	15/15	0/15	5/15	9/15	15/15	15/15	0/15	0/15	0/15	0/15	15/15	15/15	10/15	15/15	11/15	0/15	0/15	0/15	0/15	5/15	9/15		
0/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	0/6	1/6	5/6	6/6	6/6	0/6	0/6	0/6	6/6	6/6	6/6	5/6	6/6	5/6	0/6	0/6	0/6	0/6	1/6	5/6		
0/9	6/9	7/9	7/9	7/9	7/9	7/9	7/9	7/9	7/9	9/9	9/9	0/9	2/9	8/9	9/9	9/9	0/9	0/9	7/9	7/9	7/9	7/9	9/9	8/9	0/9	0/9	7/9	0/9	2/9	8/9	8/9		
0/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	0/4	0/4	4/4	4/4	4/4	0/4	0/4	0/4	4/4	4/4	4/4	2/4	4/4	4/4	0/4	0/4	0/4	0/4	0/4	0/4		
0/5	4/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	0/5	0/5	5/5	5/5	5/5	0/5	0/5	0/5	5/5	5/5	5/5	4/5	5/5	5/5	0/5	0/5	0/5	0/5	0/5	5/5		
0/5	2/5	4/5	4/5	2/5	2/5	5/5	5/5	5/5	5/5	5/5	5/5	0/5	0/5	0/5	5/5	5/5	0/5	0/5	0/5	0/5	0/5	5/5	2/5	2/5	0/5	0/5	0/5	0/5	0/5	0/5	0/5		
0/22	0/22	0/22	21/22	19/22	19/22	19/22	21/22	21/22	21/22	22/22	22/22	0/22	9/22	19/22	22/22	22/22	0/22	0/22	0/22	0/22	0/22	21/22	12/22	21/22	19/22	0/22	0/22	0/22	0/22	9/22	19/22		
0/24	0/24	0/24	23/24	20/24	21/24	21/24	23/24	23/24	23/24	24/24	24/24	0/24	12/24	21/24	24/24	24/24	0/24	0/24	0/24	0/24	0/24	23/24	14/24	24/24	21/24	0/24	0/24	0/24	0/24	12/24	21/24		
0/23	0/23	0/23	22/23	18/23	18/23	18/23	19/23	19/23	19/23	23/23	23/23	0/23	10/23	18/23	23/23	23/23	0/23	0/23	0/23	0/23	0/23	22/23	12/23	23/23	18/23	0/23	0/23	0/23	0/23	10/23	18/23		
0/2	2/2	2/2	2/2	0/2	1/2	1/2	1/2	1/2	1/2	2/2	2/2	0/2	0/2	2/2	2/2	2/2	0/2	0/2	0/2	2/2	2/2	2/2	2/2	2/2	2/2	0/2	0/2	0/2	0/2	0/2	2/2		
0/2	0/2	1/2	2/2	0/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	0/2	0/2	0/2	2/2	2/2	0/2	0/2	0/2	0/2	2/2	2/2	2/2	2/2	2/2	0/2	0/2	0/2	0/2	0/2	0/2		
0/16	12/16	12/16	16/16	10/16	12/16	12/16	12/16	12/16	12/16	16/16	16/16	1/16	1/16	1/16	16/16	16/16	2/16	1/16	1/16	8/16	8/16	16/16	8/16	16/16	10/16	1/16	1/16	1/16	1/16	1/16	1/16		
0/12	10/12	10/12	12/12	0/12	10/12	10/12	10/12	10/12	10/12	12/12	12/12	0/12	0/12	0/12	12/12	12/12	0/12	0/12	0/12	6/12	6/12	12/12	8/12	12/12	0/12	0/12	0/12	0/12	0/12	0/12	0/12		
Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	Yes	Yes	No	No	Flag	Yes	Yes	Yes	No	Yes	No	No	No	No	No	No	No		
0/13	8/13	12/13	13/13	8/13	13/13	13/13	13/13	13/13	13/13	13/13	13/13	0/13	0/13	0/13	13/13	13/13	0/13	0/13	0/13	9/13	10/13	13/13	7/13	12/13	0/13	0/13	0/13	0/13	0/13	0/13	0/13		
0/24	0/24	24/24	24/24	0/24	24/24	24/24	24/24	24/24	24/24	24/24	24/24	0/24	0/24	16/24	24/24	24/24	0/24	0/24	0/24	0/24	0/24	24/24	22/24	24/24	18/24	0/24	0/24	0/24	0/24	0/24	16/24		
0/8	0/8	8/8	8/8	0/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	0/8	0/8	6/8	8/8	8/8	0/8	0/8	0/8	0/8	0/8	8/8	6/8	8/8	7/8	0/8	0/8	0/8	0/8	0/8	6/8		
0/27	0/27	27/27	27/27	20/27	25/27	25/27	25/27	25/27	25/27	27/27	27/27	0/27	0/27	0/27	27/27	27/27	0/27	0/27	0/27	20/27	20/27	27/27	16/27	27/27	0/27	0/27	0/27	0/27	16/27	0/27	0/27		
0/46	42/46	44/46	46/46	41/46	42/46	43/46	43/46	43/46	45/46	46/46	46/46	18/46	18/46	18/46	46/46	46/46	8/46	18/46	23/46	43/46	43/46	43/46	37/46	46/46	46/46	19/46	19/46	23/46	18/46	18/46	18/46		
0/19	16/19	18/19	18/19	15/19	18/19	18/19	18/19	18/19	18/19	19/19	19/19	0/19	11/19	18/19	19/19	19/19	0/19	0/19	13/19	17/19	17/19	17/19	17/19	19/19	18/19	0/19	0/19	16/19	0/19	11/19	18/19		
0/19	16/19	18/19	18/19	15/19	18/19	18/19	18/19	18/19	18/19	19/19	19/19	0/19	11/19	18/19	19/19	19/19	0/19	0/19	13/19	17/19	17/19	17/19	18/19	19/19	18/19	0/19	0/19	16/19	0/19	11/19	18/19		
0/12	11/12	11/12	11/12	8/12	10/12	11/12	11/12	11/12	11/12	12/12	12/12	0/12	7/12	12/12	12/12	12/12	0/12	0/12	7/12	11/12	11/12	11/12	9/12	11/12	11/12	0/12	6/12	9/12	0/12	7/12	12/12		
0/11	10/11	10/11	10/11	9/11	9/11	10/11	10/11	10/11	10/11	11/11	11/11	0/11	0/11	11/11	11/11	11/11	0/11	0/11	6/11	10/11	10/11	10/11	9/11	10/11	10/11	0/11	5/11	8/11	0/11	0/11	11/11		
0/34	34/34	34/34	34/34	28/34	30/34	30/34	30/34	30/34	34/34	34/34	34/34	0/34	0/34	0/34	34/34	34/34	0/34	0/34	0/34	0/34	0/34	34/34	27/34	34/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34		
0/20	13/20	20/20	20/20	0/20	20/20	19/20	19/20	19/20	19/20	20/20	20/20	0/20	0/20	0/20	20/20	20/20	0/20	0/20	0/20	0/20	0/20	20/20	14/20	16/20	0/20	0/20	0/20	0/20	0/20	0/20	0/20		
0/8	7/8	8/8	8/8	7/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8	0/8	4/8	6/8	8/8	8/8	0/8	0/8	4/8	7/8	7/8	7/8	4/8	8/8	0/8	0/8	5/8	5/8	0/8	4/8	6/8		
0/11	10/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	11/11	0/11	0/11	10/11	11/11	11/11	0/11	0/11	9/11	10/11	10/11	11/11	4/11	11/11	11/11	0/11	0/11	10/11	0/11	0/11	10/11		
0/26	3/26	9/26	9/26	1/26	8/26	9/26	9/26	9/26	21/26	26/26	26/26	0/26	0/26	3/26	25/26	25/26	0/26	0/26	2/26	3/26	3/26	14/26	23/26	25/26	22/26	0/26	0/26	3/26	0/26	0/26	3/26		
0/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	0/4	0/4	4/4	4/4	4/4	1/4	0/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	1/4	2/4	3/4	0/4	0/4	4/4		
0/17	8/17	14/17	15/17	6/17	10/17	10/17	11/17	11/17	12/17	17/17	17/17	3/17	3/17	6/17	17/17	17/17	3/17	3/17	2/17	6/17	6/17	11/17	5/17	17/17	14/17	2/17	2/17	2/17	3/17	3/17	6/17		
0/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	0/2	0/2	2/2	2/2	2/2	0/2	0/2	0/2	2/2	2/2	2/2	2/2	2/2	2/2	0/2	0/2	0/2	0/2	0/2	2/2		
0/10	9/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	9/10		
0/6	0/6	1/6	1/6	1/6	1/6	1/6	1/6	1/6	1/6	1/6	1/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	0/6	1/6		
0/11	7/11	10/11	10/11	10/11	10/11	10/11	10/11	10/11	10/11	10/11	10/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	8/11		
0/10	9/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	10/10		
0/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	7/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	7/7		
0/17	17/17	17/17	17/17	17/17	17/17	17/17	17/17	17/17	17/17	17/17	17/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	17/17		
No	No	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	Yes	Yes	No	No	No	No	No	Yes	Yes	Yes	No	No	No	No	No	No	No		

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

ES2015 - narzędzia

- Kod ECMAScript 2015 możemy transpilować dla starszych przeglądarek
- Do wyboru: Traceur, TypeScript, **Babel**



- Warto:
 - szybki rozwój js, więcej możliwości już dziś, mniej boilerplate code, ...
 - „Transpilers are here to stay”

ES2015 - arrow functions

```
// Domyślnie - zwraca wyrażenie
var odds = myArr.map(v => v + 1);
var nums = myArr.map((v, i) => v + i);
var pairs = myArr.map(v => ({even: v, odd: v + 1}));

// Deklaracje umieszczamy w klamrach
nums.forEach(v => {
  if (v % 5 === 0) fives.push(v);
});

// Leksykalne this
var bob = {
  _name: "Bob",
  _friends: [],
  printFriends() {
    this._friends.forEach(f =>
      console.log(this._name + " knows " + f));
  }
}
```

ES2015 - klasy

```
class Student extends Person {  
  constructor(name, age, major) {  
    super(name, age);  
    this.major = major;  
  }  
  
  greet(greeting) {  
    console.log(greeting, this.name);  
  }  
  
  get year() {  
    return (new Date()).getFullYear() - this.age;  
  }  
  
  set age(age) {  
    this.age = parseInt(age);  
  }  
  
  static defaultAge() {  
    return 18;  
  }  
}
```

ES2015 - enhanced object literals

```
var obj = {  
  __proto__: theProtoObj,  
  
  // === 'handler: handler'  
  handler,  
  
  // === toString: function toString() {  
    toString() {  
  
      // Super calls  
      return "d " + super.toString();  
    },  
  
  // Dynamiczne nazwy własności  
  [ 'prop_' + (() => 42)() ]: 42  
};
```

ES2015 - destructuring

// list matching

```
var [a, , b] = [1,2,3];
```

// object matching

```
var { op: a, lhs: { op: b }, rhs: c }  
    = getASTNode()
```

// object matching shorthand

```
var {op, lhs, rhs} = getASTNode()
```

// Can be used in parameter position

```
function g({name: x}) {  
    console.log(x);  
}  
g({name: 5})
```

ES2015 - default, rest, spread

```
function f(x, y = 12) {  
    // domyślna wartość y (jeśli y === undefined)  
    return x + y;  
}  
f(3) === 15;
```

```
function f(x, ...y) {  
    // y jest tablicą pozostałych wartości  
    return x * y.length;  
}  
f(3, "hello", true) === 6;
```

```
function f(x, y, z) {  
    return x + y + z;  
}  
// przekazanie każdego elementu tablicy osobno  
f(...[1, 2, 3]) === 6;
```

ES2015 - let, const

```
function f() {  
  let x;  
  {  
    // let i const mają zasięg blokowy  
    const x = "sneaky";  
  
    // const nie może zmienić referencji  
    x = "foo";  
  }  
  
  // błąd, nie można ponownie zdefiniować  
  let x = "inner";  
}
```


ES2015 - moduły

```
// lib/math.js
```

```
export function sum(x, y) {  
    return x + y;  
}  
export var pi = 3.141593;  
export default var name = "Math module";
```

```
// app.js
```

```
import * as math from "lib/math";  
alert("2 $\pi$  = " + math.sum(math.pi, math.pi));
```

```
// otherApp.js
```

```
import name from "lib/math";  
import {sum, pi} from "lib/math";  
alert("2 $\pi$  = " + sum(pi, pi));
```

ES2015 - nowe metody Object i String

```
"abcde".includes("cd")    // = true  
"abc".repeat(3)           // = "abcabcabc"
```

```
// user.logged = action.logged;  
// user.updatedAt = action.updatedAt;
```

```
var action = {logged: true, updatedAt: new Date()}  
Object.assign(user, action)
```

ES2015 - nowe metody Array

// zwraca prawdziwą tablicę

```
Array.from(document.querySelectorAll('*'))
```

// podobne do *new Array()*, inne zachowanie dla 1 parametru

```
Array.of(1, 2, 3)
```

```
[0, 0, 0].fill(7, 1) // [0, 7, 7]
```

```
[1, 2, 3].find(x => x == 3) // 3
```

```
[1, 2, 3].findIndex(x => x == 2) // 1
```

```
[1, 2, 3, 4, 5].copyWithin(3, 0) // [1, 2, 3, 1, 2]
```

```
["a", "b", "c"].entries() // iterator [0,"a"],[1,"b"],[2,"c"]
```

```
["a", "b", "c"].keys() // iterator 0, 1, 2
```

```
["a", "b", "c"].values() // iterator "a", "b", "c"
```

ES2015 - obiekt Promise

```
function timeout(duration = 0) {  
    return new Promise((resolve, reject) => {  
        setTimeout(resolve, duration);  
    })  
}
```

```
var p = timeout(1000).then(( ) => {  
    return timeout(2000);  
}).then(( ) => {  
    throw new Error("hmm");  
}).catch(err => {  
    return Promise.all([timeout(100), timeout(200)]);  
})
```



TypeScript

TypeScript

- **Superset** ECMAScript
- ... oraz **z silnym typowaniem**
- Konieczna jest transpilacja (do ES2015, ES5 lub ES3)
- **Błędy w typach** wyłapywane na etapie kompilacji i przez IDE
- **Instalacja:** `npm i -g typescript`
- **Kompilacja:** `tsc my-app.ts`
- **Linting:** `npm i -D tslint`

TypeScript: konfiguracja

```
// file: tsconfig.json
{
  "compilerOptions": {
    "module": "commonjs",           // jaki typ modułów?
    "target": "es5",               // docelowa platforma?
    "emitDecoratorMetadata": true,
    "experimentalDecorators": true,
    "noImplicitAny": false,        // "any" jako domyślny typ?
    "removeComments": false,
    "sourceMap": true              // czy generować sourcemaps?
  },
  "exclude": [
    "node_modules",
    "dist/"
  ]
}
```

TypeScript: użycie z bibliotekami js

- Możemy korzystać z zewnętrznych bibliotek bez potrzeby ich modyfikacji
- Potrzebujemy definicji ich interfejsów
- Przechowujemy je w plikach `*.d.ts`
- Większość bibliotek JavaScript ma już takie definicje dostępne
- Instalujemy je za pomocą npm:
`npm i -D @types/jquery`

TypeScript: typy

typ:	przykładowe wartości:
boolean	true, false
number	liczby, Infinity, NaN
string	znaki, ciągi znaków
[]	tablice innych typów, np. number[], boolean[]
{ }	obiekt
undefined	nie zdefiniowany

TypeScript: typy

typ:	przykładowe wartości:
enum	wylistowanie, np. { Admin, Publisher, User }
any	dowolny typ
void	nic
-	w przypadku braku określenia typu, TypeScript może się go "domyślić" (<i>type inference</i>)

TypeScript: klasy

```
class Person {  
  
    dateOfBirth: number;           // publiczna własność  
    private verified: boolean;     // prywatna własność  
  
    constructor(  
        name: string,              // definicja parametru  
        public city: string,       // i publicznej własności  
        age?: number               // parametr opcjonalny  
    ) {  
        /* ... */  
    }  
}  
  
var p1 = new Person('Tomek', 'Gdynia', 33);
```

p1.

- city (property) Person.city: string
- dateOfBirth

TypeScript: interfejsy

- Kiedy definiowanie klasy to za dużo
- Interfejsów używamy tylko na potrzeby sprawdzania typów (nie pozostawiają śladu w kodzie wynikowym)
- Mamy możliwość przeciążania interfejsów

```
interface Announcer {  
    (message: string, announceDate: number): void;  
    (message: string[], announceDate: Date): void;  
}
```

Linki

Przydatne zasoby

- **Dokumentacja**

- <https://developer.mozilla.org/en-US/>
- <https://zealdocs.org/>

- **Więści ze świata JavaScript, tips**

- <https://devchat.tv/>
- <https://twitter.com/sulco>
- <https://github.com/gothinkster/realworld>

- **Backend**

- <https://github.com/typicode/json-server>
- <https://feathersjs.com>
- <https://nestjs.com>

- **Online IDE for frontend:**

- <https://stackblitz.com>