

Hello, World!



KAJ BIAŁAS _____
JavaScript Developer

| Ecma Script 6

CZYM JEST **Ecma Script 6**

Nowy standard języka `JavaScript` .

Nie wspierany natywnie przez wszystkie przeglądarki.

Issue #0

Uruchom przykładowy boilerplate znajdujący się w katalogu `/issue0`,
repozytorium zajęć.

TRANSPILACJA **Ecma Script 6**

W celu bezproblemowego uruchomienia kodu **ES6** za różnych przeglądarkach, używa się narzędzi "transpilujących" kod do poprzedniej wersji (ES5).

Najpopularniejszym narzędziem jest: `babel`

<https://babeljs.io>

NOWOŚCI ES6: Operator *LET*

ES 5

```
for (var i = 1; i <= 3; i++){  
  console.log(i)  
}  
  
console.log('i = ', i)  
  
// 1  
// 2  
// 3  
// 4
```

ES 6

```
for (let i = 1; i <= 3; i++){  
  console.log(i)  
}  
  
console.log('i = ', i)  
  
// 1  
// 2  
// 3  
// ReferenceError: i is not defined
```

ES6 jest kompilowane do **ES5**!

Dlaczego rezultat jest inny?

NOWOŚCI ES6: Operator *LET*

Redeklaracja w ES 5

```
if (true) {  
  var name = 'Marry'  
  var name = 'John'  
}  
  
console.log(name)  
  
// John
```

Redeklaracja w ES 6

```
if (true) {  
  let name = 'Marry'  
  let name = 'John'  
}  
  
console.log(name)  
  
// SyntaxError: Identifier 'name' has already  
// been declared
```


NOWOŚCI ES6: Operator *LET*

Hoisting w ES 5

```
if (true) {  
  var name = 'Marry'  
  var name = 'John'  
}  
  
console.log(name)  
  
// John
```

Hoisting w ES 6

```
function f() {  
  x = 2  
  let x  
  console.log(x)  
}  
  
f()  
  
// ReferenceError: x is not  
// defined
```

Issue #1

Wykorzystaj przykłady z użyciem `let` oraz `var` w swoim projekcie. Postaraj się zaobserwować różnice.

NOWOŚCI ES6: Operator *CONST*

Ma wszystkie właściwości operatora `let`

Wartości przypisane do `const` , nie mogą być nadpisane.

NOWOŚCI ES6: Operator *CONST*

Próba nadpisania wartości stałej

```
const max = 10  
  
max = 15  
  
// TypeError: Assignment to constant variable
```

Issue #2

Utwórz obiekt przypisany do stałej: `const` .

Utwórz w obiekcie pola: `name: 'Jacek'` oraz `surname: 'Kowalski'` .

W osobnej instrukcji, postaraj się nadpisać imię osoby. Jaki rezultat można zaobserwować?

NOWOŚCI ES6: **Template Strings**

Pozwalają wstawić wartość zmiennej do string'a używając specjalnej składni.

Dodatkowo możemy tworzyć wieloliniowe szablony tekstu.

NOWOŚCI ES6: Template Strings

Łączenie w ES 5

```
var name = "Kaj";  
  
var message = "Hello " + name;  
  
console.log(message);  
  
// Hello "Kaj"
```

Łączenie w ES 6

```
const name = 'Chris';  
  
const message = `Hello ${name}`;  
  
console.log(message);  
  
// Hello 'Chris'
```

NOWOŚCI ES6: Template Strings

Wiele lini w ES 5

```
var multilineMessage =  
'first line\n' +  
'second line\n' +  
'third line';  
  
console.log(multilineMessage);  
  
// first line  
// second line  
// third line
```

Wiele lini w ES 6

```
const multilineMessage =  
`first line  
second line  
third line`;  
  
console.log(multilineMessage);  
  
// first line  
// second line  
// third line
```


Issue #3

Stwórz skrypt wypisujący Twoje imię i ulubiony kolor:

```
Jestem Jacek i lubię kolor zielony!
```

Twoje imię i kolor, mają być przechowywane w stałych `const` .