

Więcej o funkcjach
i zmiennych w JS

typeof: "function"

Jeśli tworzymy funkcję za pomocą słowa **function** lub tworzymy zmienną za pomocą słowa **var**

- występuje tzw. *hoisting*
 - funkcja lub zmienna jest dostępna w całej funkcji niezależnie od tego, w której linii została zadeklarowana
- jeśli deklarujemy poza jakąś funkcją, tworzymy zmienną lub funkcję globalną
- taka funkcja lub zmienna globalna będzie dostępna jako właściwość obiektu `window`

Zmienna `let` lub stała `const` poza
blokiem lub funkcją
stanie się globalna

ale nie będzie właściwością obiektu `window`

Ponowna deklaracja **var**
nie jest błędem

let i **const** - tak

var i let mogę zadeklarować,
a później przypisać wartość

nie mogę tak zrobić z **const**

Można zadeklarować kilka zmiennych po przecinku

```
...  
1 let result, amount = 10;  
2  
3 const  
4   number = 5,  
5   string = "tekst";  
6  
7 var myVariable;
```

Funkcja może posiadać parametry domyślne



```
1 const power = (a, b = 1) => a ** b;
```


Funkcja może wywoływać samą siebie

```
1  const factorial = (number) => {  
2      if (!number) {  
3          return 1;  
4      }  
5  
6      return factorial(number - 1) *  
    number;  
7  };
```

nazywa się to rekurencją (*recursion*)