

# JavaScript - operacje na strukturach danych



# Hello

Adrian Kukowski

Senior Frontend Developer at LIBRUS

# Agenda

1. Praca z tablicą
2. Praca z obiektem
3. Praca z mapą
4. Pętle
5. Destrukturyzacja
6. Chaining
7. Kompozycja funkcji

# Praca z tablicą

map - służy do modyfikacji każdego elementu tablicy z osobna

filter - służy do ograniczenia wartości w tablicy

reduce - służy do zredukowania tablicy np. Dodanie wszystkich liczb w tablicy w taki sposób aby otrzymać wynik

# Praca z obiektem

Możemy wydobywać wartość właściwości obiektów lub je modyfikować, mamy też metody jak `entries` dzięki czemu możemy “zamienić” obiekt w tablicę i użyć tych samych metod to co tablicy

# Praca z mapą

get - służy do pobierania

set - służy do ustawiania

has - służy do sprawdzania czy istnieje

delete - służy do usuwania

[Dokumentacja MDN](#)

# Pętle

- Pętla for
- Pętla for in
- Pętla for on
- Pętla while

# Destrukturyzacja

W przypadku tablicy:

```
let [nasza_nazwa, inna_nasza_nazwa] = [1,2]
```

Dzięki temu zapisowi pod zmienną “nasza\_nazwa” mamy cyfrę 1 a pod zmienną “inna\_nasza\_nazwa” mamy cyfrę 2

W przypadku obiektu:

```
let { name, surname } = { name: "Marek", surname: "Kowalski" }
```

Zwróćmy uwagę, że w przypadku obiektu nazwy z lewej strony muszą odpowiadać nazwą z prawej strony



# Chaining

```
[1,2].map(/* Jakaś funkcja do mapowania */).filter(/* Jakaś funkcja do  
filtrowania */).map(/* Inna funkcja do mapowania */)
```

Chaining polega na możliwości łączenia różnych funkcji ze sobą tutaj bardzo ważne jest słowo kluczowe **this**

# Kompozycja funkcji

```
double(add(2,5))
```

Kompozycja funkcji polega na napisaniu funkcji w taki sposób aby można było z nich komponować większą całość i wykorzystywać wielokrotnie



# Dzięki



adrian-kukowski

@ kontakt@kukowskiadrian.pl

kukowskiadrian.pl