

Funkcja strzałkowa - arrow function

📅 Created	
👤 Created by	
🏷️ Tags	

Wstęp

Funkcje strzałkowe charakteryzują się krótszym zapisem w stosunku do normalnie deklarowanych funkcji oraz nie posiadają własnego bindowania ze słowem **this**. Co to oznacza postaramy się wyjaśnić w praktyce.



Zacznijmy od składni arrow function zwanej też **fat arrow** (ang. gruba strzała), która przedstawia się następująco:

argumenty przekazywane do funkcji; jeżeli jest ich kilka, to są oddzielane od siebie przecinkami	znak szczególny dla funkcji strzałkowej oznaczający zwrócenie wartości	ciało funkcji zawierające instrukcje do wykonania
↓	↓	↓
(argumenty)	=>	{ instrukcje do wykonania }

Jak zapewne zauważyłeś funkcja strzałkowa domyślnie jest funkcją bez nazwy. Aby móc ją wykorzystać w kilku miejscach lub wywołać, powinno się ją przypisać do jakiejś

zmiennej, np.

```
const myFunction = (params) => {  
  // ciało funkcji  
}
```

Rozpatrzmy teraz kilka możliwości na skrócenie kodu :

- kiedy funkcja nie posiada argumentów należy użyć pustych nawiasów okrągłych:

```
const showMyNumber = () => {  
  console.log('123456789');  
}
```

- kiedy funkcja ma dokładnie **jeden** argument nawiasy okrągłe () są opcjonalne:

```
const addFour = b => { return b + 4; }
```

- jeżeli funkcja posiada **kilka argumentów**, to należy je umieścić w nawiasach okrągłych i oddzielić przecinkiem:

```
const sum = (a, b) => a + b;
```

- jeżeli funkcja ma zwracać tylko jedno wyrażenie bądź wartość, to możesz pominąć 'wąsy' - { } oraz słówko **return**:

```
const addFour = b => b + 4;
```

- jeżeli w ciele funkcji jest więcej niż jedno wyrażenie, to musisz dołączyć wąsy i słówko **return**:

```
const addFour = b => {  
  const newValue = b + 4;  
  return newValue;  
}
```

A teraz przejdźmy do mniej przyjemnej części. Wspomnieliśmy we wstępie, że funkcja strzałkowa nie posiada własnego bindowania z **this**. Jako przykład rozpatrzmy obiekt, którego właściwości będą odnosić się do przypisanych metod (funkcji) :

this przy użyciu zwykłej funkcji:

```
const object = {  
  showThisRegularFunction: function() {  
    return console.log(this);  
  }  
}  
  
object.showThisRegularFunction();  
  
// zostanie zwrócone:  
  
{  
  showThisRegularFunction: function() {  
    return console.log(this);  
  }  
}
```

W przypadku wywołania zwykłej funkcji jako metody w obiekcie **this** zawsze będzie się odwoływać do obiektu, w którym została wywołana metoda.

this przy użyciu funkcji strzałkowej:

```
const object = {  
  arrowInMethod: function() {  
    const res = () => this;  
    return res();  
  },  
  arrow: () => this  
}  
  
console.log(object.arrowInMethod());  
console.log(object.arrow() === window);  
  
// zostanie zwrócone:  
  
{  
  arrow: () => this,  
  arrowInMethod: function() {  
    const res = () => this;  
    return res();  
  }  
}  
  
true
```

Zauważ, że pierwszy `console.log` po wywołaniu metody **arrowInMethod()** zwróci cały obiekt, w którym jest zawarty. Jednakże w przypadku zastosowaniu funkcji strzałkowej jest to możliwe tylko

wtedy, kiedy arrow function jest wewnątrz funkcji - tworzy się w ten sposób referencję do rodzica; zawęża się kontekst z globalnego do tej funkcji.

Jak widać w przypadku wywołania metody **arrow()** - **this** będzie się odwoływać do obiektu globalnego window (przy użyciu Node.js będzie to global).

WAŻNA INFORMACJA DO ZADAŃ!!!

!! Do każdego artykułu utwórzcie sobie na komputerze folder i w nim pliki z rozwiązaniami zadań. Zawsze, nawet jeśli pytanie jest otwarte i czysto teoretyczne to warto napisać na nie odpowiedź by nasz mózg lepiej zapamiętał to co piszemy.

Legenda:

bez gwiazdek - zadania podstawowe z artykułów oraz filmu,

* - na podstawie artykułu, ale nie znajdziesz tego na filmie,

** - nie ma tego w artykule ani w filmiku ale dostajesz link gdzie znajduje się rozwiązanie,

*** - brak informacji w artykule, filmie oraz brak linku do rozwiązania. Są to najbardziej rozwijające zadania, które również uczą samodzielności :)

1. Kiedy w funkcji strzałkowej można opuścić nawiasy okrągłe, w których zwykle znajdują się argumenty funkcji?
 - a) kiedy funkcja posiada wiele argumentów,
 - b) kiedy funkcja posiada tylko jeden argument,
 - c) nie jest możliwe opuszczenie nawiasów okrągłych,

2. W jakiej sytuacji można pominąć stosowanie instrukcji **return** używając funkcji strzałkowej?
 - a) nie można pominąć,
 - b) można pominąć, kiedy zwracane jest jedno wyrażenie lub wartość,
 - c) można pominąć, kiedy ciało funkcji zawiera kilka instrukcji,
3. Napisz funkcję strzałkową, która będzie zwracać połowę wartości wprowadzonej jako argument. Np. argument = 4, funkcja zwraca 2.
4. Napisz funkcję strzałkową, która przyjmować będzie jeden argument (liczbę). Następnie sprawdź czy liczba jest dodatnia, ujemna, czy równa 0 i zwróć odpowiednią wiadomość do użytkownika w konsoli.
5. Napisz funkcję strzałkową, która przyjmować będzie trzy argumenty liczbowe (całkowite). Sprawdź czy w każdej z tych liczb ostatnia cyfra jest taka sama, np. 7,17,77 i zwróć odpowiednią informację. Jeżeli w przynajmniej jednym przypadku cyfra będzie się różnić, to warunek nie zostaje spełniony - również zwracamy informację.
6. ** Napisz funkcję strzałkową, która jako argument przyjmie tekst "Czarna krowa w kropki bordo". Następnie z wykorzystaniem JavaScriptowych metod wyświetl w konsoli słowa tego tekstu w kolejności alfabetycznej ("w" również będzie traktowane jako słowo).
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/split
7. Utwórz funkcję strzałkową, która jako argument przyjmie tablicę z kilkoma obiektami. Niech każdy z obiektów opisuje Twojego znajomego : jak ma na imię, jaki jest jego wiek, jaki zawód wykonuje. Następnie za pomocą pętli wywołanej w funkcji postaraj się wyświetlić informacje o każdym z przyjaciół, np. "Mój przyjaciel (imię przyjaciela) to z zawodu (zawód przyjaciela) i ma (wiek przyjaciela) lat".
8. Napisz funkcję strzałkową, która będzie usuwać wybrany element z tablicy. Załóżmy, że przekazujemy do funkcji tablicę [1,2,3] oraz jako drugi argument liczbę 2. W rezultacie powinniśmy otrzymać tablicę [1,3].
9. Napisz funkcję strzałkową, która będzie dodawać element do tablicy w zadeklarowanym miejscu. Na przykład, mając tablicę z wartościami [3,4,5,6,7] chcesz dodać liczbę 1 na miejsce o indeksie 2.

10. Napisz funkcję strzałkową, która będzie zwracać jedynie string "Hello". Następnie utwórz kolejną funkcję, która przyjmie tę pierwszą jako argument a dodatkowo kolejnym jej argumentem będzie imię osoby. Postaraj się wywołać funkcję w funkcji, aby otrzymać na koniec w konsoli wiadomość "Hello (imię osoby)".