



Spreading Operator and Rest Parameter

Potongan kode pada materi ini:

- https://repl.it/@dicodingacademy/163-02-rest-parameter?lite=true

Masih membahas mengenai array, ES6 memiliki dua fitur menarik untuk membantu pengelo parameter fungsi dan arrays menjadi mudah, yakni spreading operator dan rest parameter.

Spreading Operator

Spreading operator dituliskan dengan three consecutive dots (_), Sesuai namanya "Spread", fitur baru ES6 ini digunakan untuk membentangkan nilai array atau lebih tepatnya iterable object menjadi beberapa elements. Mari kita lihat contoh kode berikut:

```
1. const favorites = ["Seafood", "Salad", "Nugget", "Soup"];
5. /* output:
6. [ 'Seafood', 'Salad', 'Nugget', 'Soup' ]
7. */
```

Pada kode tersebut hasil yang dicetak adalah sebuah array (ditunjukkan dengan tanda []), karena memang kita mencetak nilai favorites itu sendiri. Nah, de dapat membentangkan nilai – nilai dalam array tersebut.

```
    const favorites = ["Seafood", "Salad", "Nugget", "Soup"];
    console.log(...favorites);
```

nilai array menjadi beberapa elemen sesuai dengan panjang nilai *array-*nya, sehingga jika kita menuliskan kode seperti ini:

```
    console.log(...favorites);
```

Sama seperti kita menuliskan kode seperti ini:

```
    console.log(favorites[0], favorites[1], favorites[2], favorites[3]);
```

console.log() atau Math.max().

```
    /* Math.max() -> Mencari nilai terbesar */
    const numbers = [12, 32, 90, 12, 32];

      // Sama seperti kita menuliskan
// console.log(Math.max(numbers[0], numbers[1], numbers[2], numbers[3])
```

Spread operator dapat digunakan untuk menggabungkan dua buah array dalam objek array baru. Jika tidak menggunakan spread operator ini maka hasilnya seperti ini:

```
    const favorites = ["Seafood", "Salad", "Nugget", "Soup"];
    const others = ["Cake", "Pie", "Donut"];

7.

8. /* output:

9. [[ 'Seafood', 'Salad', 'Nugget', 'Soup' ], [ 'Cake', 'Pie', 'Donut' ]]

10. */
```

allFavorite menjadi objek array baru yang menampung dua array di dalamnya. Nah lantas bagaimana jika kita mencoba menggunakan spread operator?

```
    const favorites = ["Seafood", "Salad", "Nugget", "Soup"];
    const others = ["Cake", "Pie", "Donut"];

     const allFavorites = [...favorites, ...others]
s. /* output:

9. [[ 'Sasfood', 'Salad', 'Nugget', 'Soup', 'Cake', 'Pie', 'Donut' ]

10. */
```

Yups, dengan menggunakan spread operator nilai dari dua array tersebut berhasil tergabung.

Rest parameter

Sebelumnya kita sudah tahu bahwa *variadic function* dapat menerima banyak paran apakah kita tahu bagaimana caranya agar function dapat menerima parameter? Jika spread operator adalah pelebur array menjadi beberapa elemen yang berbeda, rest parameter ini adalah kebalikan dari operator tersebut. Penasaran?

kita dapat menggabungkan beberapa elemen menjadi satu array. Tentu teknik ini sangat bermanfaat ketika kita hendak membuat sebuah *variadic function*.

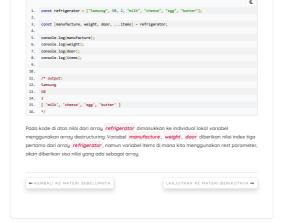
Sebagai contoh penggunaanya, mari kita buat sebuah *variadic function* yang berfungsi untuk menjumlahkan seluruh nilai argument fungsi yang diberikan.

```
    function sum(...numbers) {
    var result = 0;
    for(let number of numbers) {
    result += number

6. return result;
7. }

    console.log(sum(1,2,3,4,5));
```

mengelompokkan nilai-nilai array yang terdestruksi pada variabel dalam bentuk array yang lain Sedikit bingung? Mari lihat contoh kode berikut ini:







DIBANTU

Copyright © 2020 - Dicoding Indonesia. All rights reserve

Terms Privac