

JS Fundamentals

Isumi
Batam, Jan 2020

GOALS

1. Conditional
2. Loop
3. Math
4. Array

1. Conditional: if

Conditional paling sederhana di programming menggunakan statement if.

```
var lapar = prompt('apakah kamu lapar?');
```

```
if (lapar == 'iya') {  
  console.log('saya butuh makan');  
}
```

Operators

Conditional if menggunakan operator perbandingan dan logika.

Contoh:

```
5 >= 5
```

```
10 <= 100
```

```
5 == "5"
```

```
5 === 5
```

```
5 === "5"
```

```
grade == "A" && attitude == "good"
```

Operator	Description
&&	logical and
	logical or
!	logical not

Operator	Description
==	equal to
===	equal value and equal type
!=	not equal
!==	not equal value or not equal type
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
?	ternary operator

Logical Operator : OR (||)

```
if ( ownKTP || ownSIM ) {  
    console.log('bisa beli tiket');  
}
```

Logical Operator : AND (&&)

```
if ( myAge >= 17 && myHeight > 150 ) {  
    console.log('Bisa memiliki SIM');  
}
```

Conditional : if else

```
var tinggi = prompt('berapa tinggi kamu?');  
  
if (tinggi > '160') {  
    console.log('hore, bisa naik roller coaster');  
} else {  
    console.log('hmm, naik komedi putar saja deh');  
};
```

Conditional : else if

```
var nilai = prompt('berapa nilai kamu?');
```

```
if (nilai > '90') {  
    console.log('Luar biasa!!');  
} else if (nilai > '70') {  
    console.log('Bagus');  
} else {  
    console.log('Masih kurang');  
};
```

Notice:

Remember how to convert
string -> Int

`parseInt()`

Exercise Conditional

Minta user memasukkan nilai mereka 1 - 100

- Jika **nilai ≥ 80 && nilai ≤ 100** , alert “Dapat nilai A”
- Jika **nilai < 80 && nilai ≥ 60** , alert “Dapat nilai B”
- Jika **nilai < 60 && nilai ≥ 40** , alert “Dapat nilai C”
- Jika **nilai < 40** , alert “Dapat nilai D”
- **else** Jika nilainya bukan 1 - 100, maka alert “Nilai tidak valid”

Conditional: switch

Conditional dengan banyak kondisi.

```
switch(expression) {  
    case x:  
        // code block  
        break;  
    case y:  
        // code block  
        break;  
    default:  
        // code block  
}
```

2. Loop

Loop atau perulangan adalah tindakan mengulang / merepetisi sebuah proses.

Untuk melakukan looping/iteration, JavaScript menyediakan beberapa jenis iteration, yaitu:

- For Loop
- While Loop
- do-While Loop

Increment & Decrement

Increment penambahan satu angka

```
i = 1;
```

```
i++;
```

Decrement pengurangan satu angka

```
i = 5;
```

```
i--;
```

For Loop

Pada for-loop, statement for akan menampung tiga parameter, yaitu sebut saja **inisialisasi**, **kondisi**, dan **incremental/decremental**. Ketiga parameter ini akan menjadi kontrol kapan loop ini harus berhenti.

```
for(var angka = 1; angka < 10; angka++) {  
    console.log('Iterasi ke-' + angka);  
}
```

While Loop

Apabila statement **while** mendapatkan **nilai true**, maka proses yang berada di dalam curly bracket ({ }) akan dijalankan.

```
var flag = 1;  
while(flag < 10) {  
    console.log('Iterasi ke-' + flag);  
    flag++;  
}
```

Do/While Loop

Do / while loop adalah varian dari while loop. Loop ini akan **mengeksekusi blok kode sekali**, sebelum memeriksa kondisinya, lalu akan mengulangi loop selama kondisi **true**.

```
var flag = 1;
```

```
do {
```

```
    console.log('Iterasi ke-' + flag);
```

```
    flag++;
```

```
} while(flag < 1)
```

3. JavaScript Math

```
Math.round(4.7);
```

```
Math.pow(8, 2);
```

```
Math.sqrt(64);
```

```
Math.abs(-4.7);
```

```
Math.ceil(4.4);
```

```
Math.floor(4.7);
```

```
Math.min(0, 150, 30, 20, -8, -200);
```

```
Math.max(0, 150, 30, 20, -8, -200);
```

```
Math.random();
```

4. Array

Array adalah kumpulan dari berbagai data.

- **One-step array allocation**

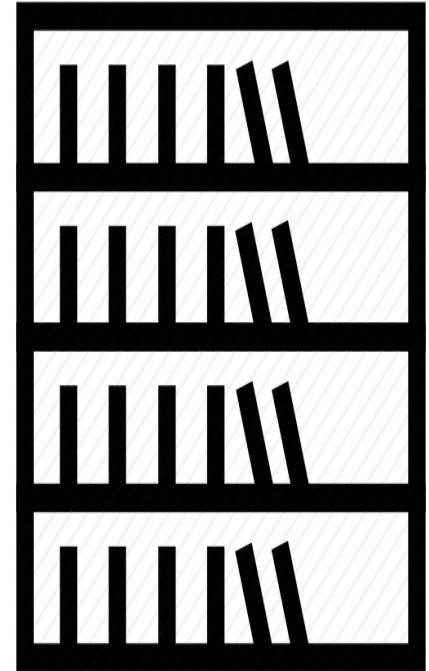
```
var primes = [2, 3, 5, 7, 11, 13];  
var names = ["Joe", "Jane", "John", "Juan"];  
• No trailing comma after last element
```

- **Accessing array elements**

```
primes[0] → 2  
primes[5] → 13  
names[0] → "Joe";  
names[3] → "Juan";  
names[3] = "Jill"; // Replace entry at index 3  
names[3] → "Jill";
```

- **The length property**

```
var names = ["Joe", "Jane", "John", "Juan"];  
names.length → 4  
• Notice that if the length is 4, the index of the last entry is 3  
• This is true in general: array indexes run from 0 to length-1
```



Built-in Function Array

- `push`: menambah 1 nilai ke `array` ke index paling belakang
- `pop`: menghapus 1 nilai dari `array` index paling belakang
- `unshift`: menambah 1 nilai ke `array` index paling depan (index 0)
- `shift`: menghapus 1 nilai dari `array` index paling depan (index 0)
- `join`: menggabungkan seluruh elemen `array` menjadi sebuah string dan mengambil parameter sebagai simbol penyambung antar elemen
- `sort`: mengurutkan elemen di dalam `array` sesuai alphabet
- `slice`: mengambil beberapa lapis data
- `splice`: mengubah nilai `array` dengan menghapus dan/atau menambah nilai baru ke `array`
- `split`: memecah string dan mengembalikan `array` sesuai dengan separator / pemisah yang didefinisikan

<https://jsbin.com/qehefet/edit?js.console>

Array Popular Methods

- `forEach`: melakukan pengulangan untuk setiap elemen array.
- `map`: membuat sebuah array baru melalui penerapan sebuah fungsi terhadap setiap elemen array.
- `filter`: membuat sebuah *array* baru dengan memperhatikan kondisi tertentu pada setiap elemen dari *array* yang sudah ada.
- `reduce`: *meredu*ce elemen *array* menjadi *single value* dengan menjumlah setiap elemen (dari kiri ke kanan).
- `includes`: mengecek apakah pada elemen array memenuhi suatu kondisi atau tidak.
- `some`: mengecek *minimal* salah satu elemen *array* memenuhi kondisi.
- `every`: mengecek apakah *setiap* elemen dalam *array* memenuhi kondisi.
- `concat`: menggabungkan 2 *array* menjadi 1 *array*.

<https://jsbin.com/wukamoy/edit?js.console>

Final Exercise

1. Buatlah website sederhana yang berfungsi sebagai **kalkulator**.
 - (Prompt) Minta user memasukkan **angka pertama** dan angka **kedua**
 - (Prompt) Minta user **memilih operasi matematika** yg mau dilakukan:
Tambah, Kurang, Kali, Bagi, Pangkat, Min, Max
 - (Alert) Munculkan hasilnya

Notes:

- Kalian dapat menggunakan function `parseInt` untuk mengubah string menjadi integer.

Final Exercise

2. Filtering

- Buka link swapi.co/api/planets
- Buat tabel yg memunculkan data planet (**property results**)
- Berikan filtering menggunakan **nama planet**

Cari nama planet...

Search

127.0.0.1:5500 says

Kata pencarian tidak boleh kosong!

OK

No.	Name	Diameter	Climate	Population
1.	Alderaan	12500	temperate	2000000000
2.	Yavin IV	10200	temperate, tropical	1000
3.	Hoth	7200	frozen	unknown
4.	Dagobah	8900	murky	unknown
5.	Bespin	118000	temperate	6000000
6.	Endor	4900	temperate	30000000
7.	Naboo	12120	temperate	4500000000
8.	Coruscant	12240	temperate	1000000000000
9.	Kamino	19720	temperate	1000000000
10.	Geonosis	11370	temperate, arid	100000000000