**Loop dalam JavaScript**

Loop dalam pemrograman dibuat untuk mengulang baris kode sebanyak beberapa kali tergantung dari kondisi yang diperlukan.

**Jenis-jenis loop dalam JavaScript:**

1. **for**

Digunakan ketika kita tahu seberapa banyak pengulangan yang ingin dijalankan.

**Syntax:**

For (perintah awal; kondisi; penambahan {

// kode untuk dijalankan;

}

Contoh:

Ingin mencetak angka 1 hingga 5.

JavaScript:

for (let i = 1;i <= 5; i++) {

document.write(i + "<br>");

}

Penjelasan:

* let i = 1 🡪 inisialisasi variabel i dengan nilai 1.
* i <= 5 🡪 looping selama i sama dengan atau kurang dari 5.
* i++ 🡪 tambah i sebanyak 1 setelah setiap pengulangan.
* document.write(i + "<br>") 🡪 mencetak nilai i.

Output:

1

2

3

4

5

1. **while**

Untuk mengulang baris kode yang jumlah pengulangannya belum diketahui, tapi akan terus diulang selama masih memenuhi kondisi tertentu.

Syntax:

while (kondisi) { 🡪 kondisi: dievaluasi sebelum setiap loop.

// kode untuk dijalankan jika true, loop terus berjalan,

} jika false, loop berhenti.

Contoh:

Ingin mencetak angka 1 hingga 5.

JavaScript:

let i = 1;

while (i <= 5) {

document.write(i + "<br>");

i++; }

Penjelasan:

* let i = 1 🡪 inisialisasi variabel i dengan nilai 1.
* i <= 5 🡪 selama i sama dengan atau kurang dari 5, looping.
* document.write(i + "<br>") 🡪 mencetak nilai i.
* i++ 🡪 tambah i sebanyak 1 setelah setiap pengulangan.

Output:

1

2

3

4

5

1. **do … while**

Mirip dengan while loop. Bedanya, loop ini memastikan kode kita dijalankan paling tidak satu kali karena kondisinya baru diperiksa setelah kode kita dijalankan satu kali.

Syntax:

Do {

// kode untuk dijalankan

} while (kondisi)

Contoh:

Ingin mencetak angka 1 hingga 5.

JavaScript:

let i = 1;

do {

document.write(i + "<br>");

i++;

} while (i <= 5);

Penjelasan:

- let i = 1 🡪 inisialisasi variabel i dengan nilai 1.

- document.write(i + "<br>") 🡪 mencetak nilai i.

- i++ 🡪 penambahan nilai i sebanyak 1

- i <= 5 🡪 check apakah i sama dengan atau kurang dari 5.

Jika true, looping berjalan. Jika false, loop berhenti.

Output:

1

2

3

4

5

1. **for … in**

Digunakan untuk mengulang sejumlah elemen/property yang ada pada suatu objek.

Syntax:

For (let property in object) { 🡪 property = property name dari object

// kode untuk dijalankan }

Contoh:

Menampilkan isi data dari sebuah object.

JavaScript:

let siswa = {nama: “Irine”, usia: 17, kota: “Bandung”};

for (let property in siswa) {

document.write(property + “ : “ + siswa[property] + “<br>”); }

Output:

nama : Irine

usia : 17

kota : Bandung

1. **for … of**

Digunakan untuk mengulang sejumlah value yang ada dalam suatu objek yang iterable, contohnya arrays atau strings.

Syntax:

For (let value of array) { 🡪 key = value dari sebuah array.

// kode untuk dijalankan

}

Contoh:

Menampilkan isi data dari sebuah array.

JavaScript:

let warna = [“merah”, “kuning”, ”hijau”]; 🡪 buat variabel warna

for (let text of warna) { 🡪 perintah untuk iterasi

document.write(text + “<br>”); 🡪 cetak value

}

Output:

merah

kuning

hijau

1. **forEach()**

Digunakan untuk mengulang elemen dalam array.

Kekurangannya, tidak bisa menggunakan continue atau break untuk menghentikan pengulangan.

Syntax:

Array.forEach(function(element) {

// kode untuk dijalankan

});

Contoh:

Menampilkan isi data dari sebuah array.

JavaScript:

let warna = [“merah”, “kuning”, ”hijau”];

warna.forEach(function(text) {

document.write(text + “<br>”);

});

Output:

merah

kuning

hijau

**Statement tambahan untuk loops:**

1. **break**

Untuk memberhentikan loop segera.

Contoh:

Ingin memberhentikan loop ketika nilai mencapai 3.

Syntax:

for (let i = 1; i <= 5; i++) {

if (i === 3) {

break;

}

document.write(i + “<br>”);

}

Output:

1

2

1. **continue**

Untuk melewat pengulangan yang sedang berjalan, lalu melanjutkan pengulangan berikutnya.

Contoh:

Ingin memberhentikan loop ketika nilai mencapai 3.

Syntax:

for (let i = 1; i <= 5; i++) {

if (i === 3) {

continue;

}

document.write(i + “<br>”);

}

Output:

1

2

4

5