

Fungsi dan Logika Perulangan

Nur Muhammad Abdul Falaq

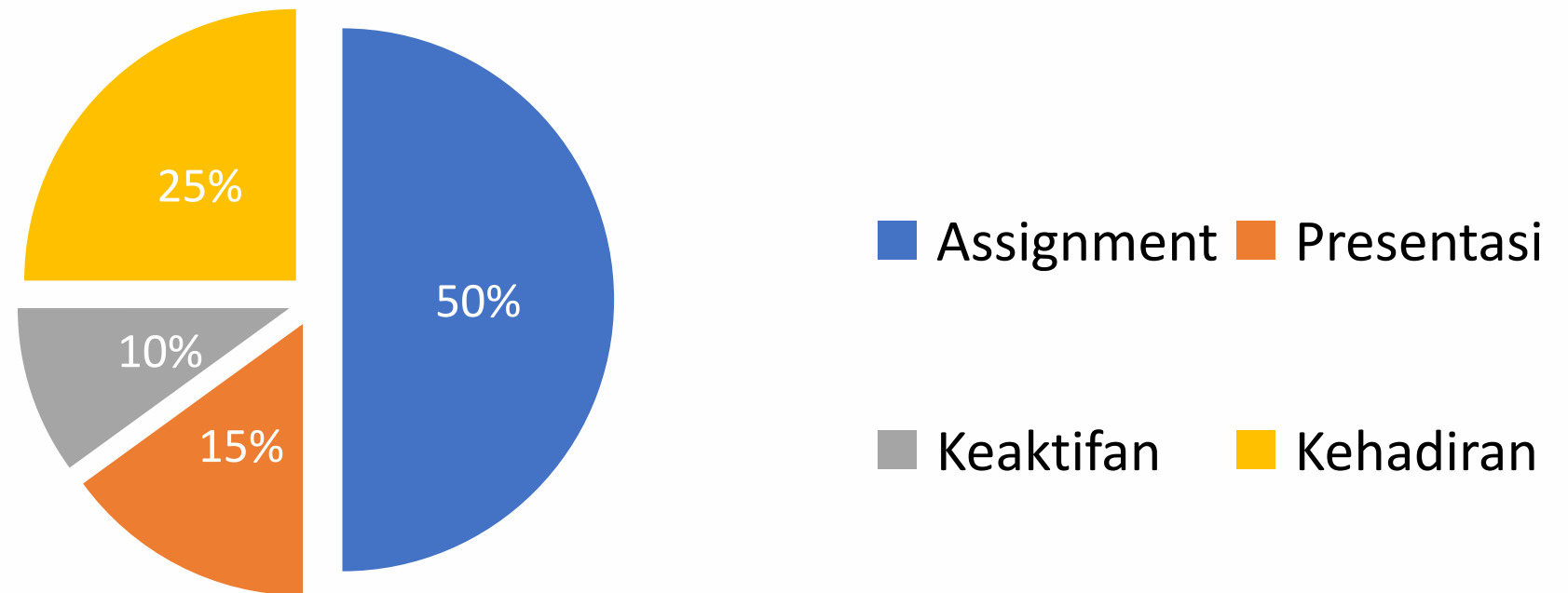


PINTAR



Penilaian

Komponen Penilaian



Scope

- Selection Javascript
- Function Javascript
- Logika Perulangan

Referensi

<https://nasrianas.website/coding/javascript/034-jenis-jenis-selector-pada-javascript>

<https://www.petanikode.com/javascript-fungsi/>

<https://www.petanikode.com/javascript-perulangan/>

Materi Sebelumnya

➤ Review JavaScript :

1. Variabel
2. Tipe Data



Menemukan Elemen HTML di JS

Seringkali, dengan JavaScript, Anda ingin memanipulasi elemen HTML.

Untuk melakukannya, Anda harus menemukan elemennya terlebih dahulu. Ada beberapa cara untuk melakukan ini:

1. Menemukan elemen HTML dengan id
2. Menemukan elemen HTML dengan nama tag
3. Menemukan elemen HTML dengan nama kelas



java script

Menemukan Elemen HTML dengan Id

Cara termudah untuk menemukan elemen HTML, adalah dengan menggunakan id elemen.

Contoh ini menemukan elemen dengan id="intro":

Silahkan buat file html dengan nama LatihanMenemukanElementHTMLdenganID.html

```
document.getElementById("intro");
```

```
<!DOCTYPE html>  
<html>
```

```
<body>
```

```
<h2>Latihan Memanggil Element By ID</h2>
```

```
<p id="intro">Mencari Element By ID</p>
```

```
<p>Contoh Demo pemanggilan data<b>getElementById</b> method.</p>
```

```
<p id="demo"></p>
```

```
<script>
```

```
var element = document.getElementById("intro");
```

```
document.getElementById("demo").innerHTML =
```

```
"Teks dari paragraf intro adalah: " + element.innerHTML;
```

```
</script>
```

```
</body>
```

```
</html>
```

Menemukan elemen HTML dengan nama tag

contoh ini menemukan elemen dengan id="main", dan kemudian menemukan semua elemen <p> di dalam "main":

silahkan buat file html dengan nama latihanMenemukanElementByTag.html

```
document.getElementsByTagName("p");
```

```
<!DOCTYPE html>  
<html>
```

```
<body>
```

```
    <h2>Menemukan Element By Tag</h2>
```

```
    <p>Menemukan Element By Tag</p>
```

```
    <p>contoh demo <b>getElementsByTagName</b> method.</p>
```

```
    <p id="demo"></p>
```

```
    <script>
```

```
        var element = document.getElementsByTagName("p");
```

```
        document.getElementById("demo").innerHTML = 'Teks pada paragraf pertama adalah: ' +  
        element[0].innerHTML;
```

```
    </script>
```

```
</body>
```

```
</html>
```

Menemukan elemen HTML dengan nama kelas

Silahkan buat file HTML dengan nama menemukanElementByNameKelas.html

```
<!DOCTYPE html>  
<html>
```

```
<body>
```

```
<h2>Menemukan Element by nama kelas</h2>
```

```
<p>Finding HTML Elements by Class Name.</p>
```

```
<p class="intro">Hello World!</p>
```

```
<p class="intro">contoh demo <b>getElementsByClassName</b> method.</p>
```

```
<p id="demo"></p>
```

```
<script>
```

```
var x = document.getElementsByClassName("intro");
```

```
document.getElementById("demo").innerHTML =
```

```
'Paragraf pertama (indeks 0) dengan class="intro" adalah: ' + x[0].innerHTML;
```

```
</script>
```

```
</body>
```

```
</html>
```


JavaScript Function

Fungsi atau function di javascript adalah sebuah blok kode yang digunakan untuk membungkus suatu proses dengan tujuan agar penulisan kode atau proses yang sama tidak ditulis secara berulang kali



Membuat fungsi di Javascript

Ada 4 cara yang bisa kita lakukan untuk membuat fungsi di Javascript:

1. Menggunakan cara biasa;
2. Menggunakan ekspresi;
3. Menggunakan tanda panah (=>);
4. dan menggunakan Constructor.

```
function greet(name) {  
    // code  
}  
greet(name);  
// code
```

function call

Membuat Fungsi dengan Cara Biasa

```
function namaFungsi(){  
  console.log("Hello World!");  
}
```

- Cara ini paling sering digunakan, terutama buat yang baru belajar Javascript.

Membuat Fungsi dengan Ekspresi

```
var namaFungsi = function(){  
  
    console.log("Hello World!");  
  
}
```

Kita menggunakan variabel, lalu diisi dengan fungsi. Fungsi ini sebenarnya adalah fungsi anonim (anonymous function) atau fungsi tanpa nama.

Membuat Fungsi dengan Tanda Panah

Cara ini sering digunakan di kode Javascript masa kini, karena lebih sederhana. Akan tetapi sulit dipahami bagi pemula.

Sebenarnya hampir sama dengan yang menggunakan ekspresi. Bedanya, kita menggunakan tanda panah (\Rightarrow) sebagai ganti function.

Pembuatan fungsi dengan cara ini disebut arrow function.

```
var namaFungsi = () => {  
    console.log("Hello World!");  
}
```

```
// atau seperti ini (jika isi fungsi hanya satu baris):  
var namaFungsi = () => console.log("Hello World!");
```

Membuat Fungsi dengan Kostruktur

Cara ini sebenarnya tidak direkomendasikan oleh Developer Mozilla, karena terlihat kurang bagus. Soalnya body fungsinya dibuat dalam bentuk string yang dapat mempengaruhi kinerja engine javascript.

```
var namaFungsi = new Function('console.log("Hello World!");');
```

Untuk yang masih pemula, saya rekomendasikan gunakan cara yang pertama dulu. Nanti kalau sudah terbiasa baru coba gunakan cara ke-2 dan ke-3.

Cara Memanggil/Eksekusi Fungsi

Setelah mengetahui cara membuat fungsi, lalu bagaimana cara memanggилnya?

Kita bisa memanggil fungsi di dalam kode Javascript dengan menuliskan nama fungsinya seperti ini:

```
function sayHello(){  
    console.log("Hello World!");  
}
```

```
// memanggil fungsi  
sayHello() // maka akan menghasilkan -> Hello World!
```

Selain dengan cara di atas, kita juga bisa memanggil fungsi melalui atribut event pada HTML.

Latihan Fungsi 1

Silahkan buat satu file HTML,
dengan nama
latihanFungsiJavaScript.html
setelah itu silahkan ikuti
kodingan di samping

```
<!DOCTYPE html>
<html>

<head>
  <title>Latihan Membuat Fungsi</title>
</head>

<body>
  <!-- Memanggil fungsi saat link diklik -->
  <a href="#" onclick="sayHello()">Klik Aku!</a>

  <script>
    // membuat fungsi
    function sayHello() {
      alert("Hello World!");
    }
  </script>

</body>

</html>
```


Latihan Fungsi 2

Fungsi dengan Parameter:

Silahkan buat file html dengan nama

Latihan2FungsiJavaScript.html

silahkan buat koding seperti di samping

```
<!DOCTYPE html>
<html>

<head>
  <title>Latihan 2 Membuat Fungsi</title>
</head>

<body>
  <!-- Memanggil fungsi saat link diklik -->
  <a href="#" onclick="kali(2, 5)">Klik Aku!</a>
  <input type="text" id="hasilKali">

  <script>
    // membuat fungsi
    function kali(a, b) {
      hasilKali = a * b;
      document.getElementById("hasilKali").value = hasilKali;
    }
  </script>

</body>

</html>
```

Logika Perulangan

Perulangan akan membantu kita mengeksekusi kode yang berulang-ulang, berapapun yang kita mau.

Ada lima macam bentuk perulangan di Javascript. Secara umum, perulangan ini dibagi dua.

Yaitu: counted loop dan uncounted loop.

Perbedaanya:

Counted Loop merupakan perulangan yang jelas dan sudah tentu banyak perulangannya.

Sedangkan Uncounted Loop, merupakan perulangan yang tidak jelas berapa kali ia harus mengulan.

Perulangan yang termasuk dalam Counted Loop:

1. Perulangan For
2. Perulangan Foreach
3. Perulangan Repeat

Perulangan yang termasuk dalam Uncounted Loop:

1. Perulangan While
2. Perulangan Do/While

Perulangan For di Javascript

Perulangan For di Javascript

perulangan yang termasuk dalam couted loop, karena sudah jelas berapa kali ia akan mengulang.

silahkan buat file html dengan nama PerulanganFor.html

Yang perlu diperhatikan adalah kondisi yang ada di dalam kurung setelah kata for.

Kondisi ini akan menentukan:

Hitungan akan dimulai dari 0 ($i = 0$);

Hitungannya sampai berapa? Sampai $i < 10$;

Lalu di setiap perulangan i akan bertambah +1 ($i++$).

Variabel i pada perulangan for berfungsi untuk menyimpan nilai hitungan.

Jadi setiap perulangan dilakukan nilai i akan selalu bertambah satu. Karena kita menentukannya di bagian $i++$.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Perulangan dengan For</h2>
```

```
<script>
```

```
for (var i = 0; i < 10; i++) {
```

```
    document.write("<p>Perulangan ke-" + i + "</p>")
```

```
}
```

```
</script>
```

```
</body>
```

```
</html>
```

Perulangan While di Javascript

Perulangan while merupakan perulangan yang termasuk dalam perulangan uncounted loop.

Perulangan while juga dapat menjadi perulangan yang counted loop dengan memberikan counter di dalamnya.

silahkan buat file html dengan nama perulanganWhile.html

```
<!DOCTYPE html>
<html>
<body>
  <h2>Perulangan dengan while</h2>

  <script>
    var ulangi = confirm("Apakah anda mau mengulang?");
    var counter = 0;

    while (ulangi) {
      counter++;
      ulangi = confirm("Apakah anda mau mengulang?");
    }

    document.write("Perulangan sudah dilakuakn sebanyak " + counter + " kali");
  </script>
</body>
</html>
```

Perulangan Do/While di Javascript

Perulangan do/while sama seperti perulangan while.

Perbedaanya:

Perulangan do/while akan mengecek kondisi di belakang (sesudah mengulang), sedangkan while akan mencek kondisi di depan atau awal (sbelum mengulang).

silahkan buat file html denagn nama perulanganDoWhile.html

```
<!DOCTYPE html>
<html>
<body>
  <h2>Perulangan dengan do while</h2>

  <script>
    var ulangi = confirm("Apakah anda mau mengulang?");
    var counter = 0;

    do {
      counter++;
      ulangi = confirm("Apakah anda mau mengulang?");
    } while (ulangi)

    document.write("Perulangan sudah dilakuakn sebanyak " + counter + " kali");
  </script>
</body>
</html>
```

Perulangan Foreach di Javascript

Perulangan foreach biasanya digunakan untuk mencetak item di dalam array.

Perulangan ini termasuk dalam perulangan counted loop, karena jumlah perulangannya akan ditentukan oleh panjang dari array.

Ada dua cara menggunakan perulangan foreach di Javascript:

1. Menggunakan for dengan operator in;
2. Menggunakan method forEach().

silahkan buat file html dengan nama perulanganforeach.html

```
<!DOCTYPE html>
<html>
<body>
<h2>Perulangan dengan foreach</h2>
<script>
  ///Berikut ini bentuk perulangan "foreach" tanpa menggunakan operator in:
  var makanan = ["nasi", "daging", "sayur", "buah-buahan"];
  for (i = 0; i < makanan.length; i++) {
    document.write(i + ". " + makanan[i] + "<br/>");
  }

  //Perulangan ini dapat dibuat lebih sederhana lagi dengan menggunakan operator in seperti ini:
  var alatMakan = ["piring", "gelas", "sendok", "garpu"];
  for (i in alatMakan) {
    document.write(i + ". " + alatMakan[i] + "<br/>");
  }
</script>
</body>
</html>
```

Perulangan dengan Method repeat()

Perulangan dengan method atau fungsi repeat() termasuk dalam perulangan counted loop.

Fungsi ini khusus digunakan untuk mengulang sebuah teks (string).

Bisa dibilang:

Ini merupakan singkat dari perulangan for.

Contoh:

Apabila kita menggunakan perulangan for:

```
for( let i = 0; i < 100; i++){  
    document.write("Ulangi kalimat ini!");  
}
```

Apabila kita menggunakan fungsi repeat():

```
document.write("Ulangi kalimat ini! ".repeat(100));
```

silahkan buat file html dengan nama perulanganRepeat.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Perulangan dengan repeat</h2>
```

```
<script>
```

```
    document.write("Ulangi kalimat ini! ".repeat(100));
```

```
</script>
```

```
</body>
```

```
</html>
```

Bonus Latihan CRUD di JS

Setelah kita paham dasar-dasar pembuatan fungsi dan jenis-jenisnya, sekarang mari kita coba membuat program sederhana.

Program ini berisi CRUD (Create, Read, Update, Delete) data barang yang tersimpan dalam sebuah array.

Silahkan Buat satu file html dengan nama bonus.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Belajar Fungsi di Javascript</title>
</head>
<body>
  <fieldset>
    <legend>Input Form</legend>
    <input type="text" name="barang" placeholder="input nama barang..." />
    <input type="button" onclick="addBarang()" value="Tambah" />
  </fieldset>
  <div>
    <ul id="list-barang">
    </ul>
  </div>
  <script src="fungsi.js"></script>
</body>
</html>
```


Bonus Latihan CRUD di JS 2

Silahkan Buat file javascript dengan nama fungsi.js

```
var dataBarang = [
  "Buku Tulis",
  "Pensil",
  "Spidol"
];
function showBarang(){
  var listBarang = document.getElementById("list-barang");
  // clear list barang
  listBarang.innerHTML = "";

  // cetak semua barang
  for(let i = 0; i < dataBarang.length; i++){
    var btnEdit = "<a href='#' onclick='editBarang(\"++\")>Edit</a>";
    var btnHapus = "<a href='#' onclick='deleteBarang(\"++\")>Hapus</a>";

    listBarang.innerHTML += "<li>" + dataBarang[i] + " [" + btnEdit + " | " + btnHapus + "]</li>";
  }
}
function addBarang(){
  var input = document.querySelector("input[name=barang]");
  dataBarang.push(input.value);
  showBarang();
}
function editBarang(id){
  var newBarang = prompt("Nama baru", dataBarang[id]);
  dataBarang[id] = newBarang;
  showBarang();
}
function deleteBarang(id){
  dataBarang.splice(id, 1);
  showBarang();
}

showBarang();
```

Tugas Latihan...

Silahkan Buat Program Perulangan, minimal seperti dibawah ini tampilannya.

point nilai :

1. ada inputan data dan ada inputan jumlah perulangan
2. ada proses perulangan untuk data yg sudah di inputkan
3. tampilan bebas sesai kreatifitas
4. proses bebas sesuai kreatifitas contoh proses mengulangi kata seperti contoh di bawah ini, atau proses menjumlahkan data atau lainnya, intinya memanfaatkan perulangan di javascript

Perulangan Kata

Silahkan masukan kata yang mau di ulangi

Silahkan maukan jumlah perulangan

Mulai

Perulangan Kata

Silahkan masukan kata yang mau di ulangi

pintar

Silahkan maukan jumlah perulangan

6

Mulai

tulisan pintar Perulangan ke-0

tulisan pintar Perulangan ke-1

tulisan pintar Perulangan ke-2

tulisan pintar Perulangan ke-3

tulisan pintar Perulangan ke-4

tulisan pintar Perulangan ke-5

