

Dosen Pengampu : Ahmad Mubarak,M.Kom
Nama Mahasiswa : Muhammad Fajar Ramadhan
Prodi : S1 Sistem Informasi
Tingkat/Semester : 1/1

Tanggal: 16 Oktober 2022

Pokok Bahasan/ Pembelajaran :

Sasaran Pembelajaran:

Di akhir modul, mahasiswa akan dapat:
Perulangan dalam JavaScript

Materi:

Alat Tulis, Modul, Laptop / Gadget

Referensi:

- Abdul Qadir, 2013, *From Zero to A Pro Javascript & JQuery*, Andi Yogyakarta
- Tutorial point, 2015, *JavaScript Language Tutorial Point*, E-book, tutorialspoint.com
- <https://www.w3bai.com/id/>

A. TINJAUAN PENDAHULUAN

Saat menulis program, Anda mungkin menghadapi situasi di mana Anda perlu melakukan tindakan berulang kali. Dalam situasi seperti itu, Anda perlu menulis pernyataan loop untuk mengurangi jumlah baris. JavaScript mendukung semua loop yang diperlukan untuk mengurangi tekanan pemrograman. Loop atau perulangan adalah suatu bentuk kegiatan mengulang suatu statement sampai batas yang diinginkan. Dalam pemrograman Loop atau perulangan sangat berguna sekali.

Alokasi waktu untuk mempelajari modul ini sekitar 2 x 50 menit.



Coba anda cari informasi tentang looping Javascript di buku, internet, atau sumber pembelajaran lainnya mengenai Operator :

Ayo kita cari tahu apa yang anda ketahui.

- 1) Apa yang dimaksud Looping dalam JavaScript
- 2) Apakah looping ada di Bahasa pemrograman lain
- 3) Bagaimana looping bekerja

Tulis jawaban Anda di bawah ini dan bagikan ke kelas Anda!

1. _Looping atau perulangan adalah suatu bentuk kegiatan mengulang suatu statement sampai batas yang diinginkan. _____

2. _Ada. Setiap bahasa pemrograman memiliki statement looping masing-masing. _____

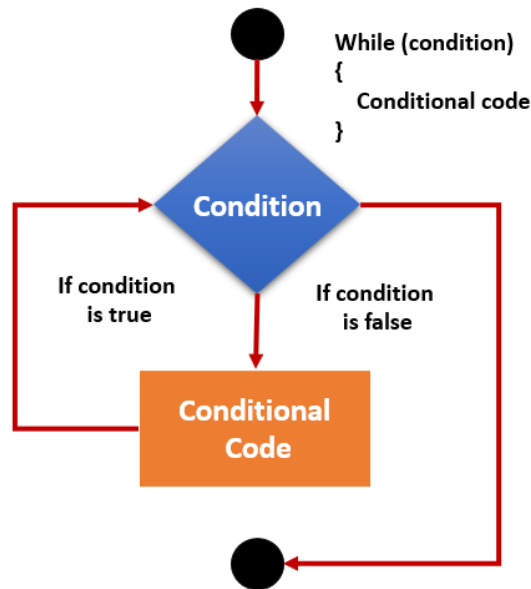
3. _Looping/Perulangan dilakukan untuk meng-eksekusi statemen-statemen dalam sebuah program secara terus-menerus sampai terdapat kondisi(True or False) untuk menghentikannya._____

B. MATERI PEMBELAJARAN

1. WHILE LOOP

Perulangan paling dasar dalam JavaScript adalah perulangan while yang akan dibahas dalam bab ini. Tujuan dari while loop adalah untuk mengeksekusi pernyataan atau blok kode berulang kali selama ekspresi benar. Setelah ekspresi menjadi salah, loop berakhir.

Diagram alir dari while loop terlihat sebagai berikut:



Sintaks while loop dalam JavaScript adalah sebagai berikut:

```
while (expression)
{
    Statement(s) to be executed if expression is true
}
```

Contoh 1

```
<html>
<body>
    <script type="text/javascript">
        var count = 0;
        document.write("Starting Loop ");
        while (count < 10)
        {
            document.write("Current Count : " + count + "<br />");
            count++;
        }
    </script>
</body>
```

```

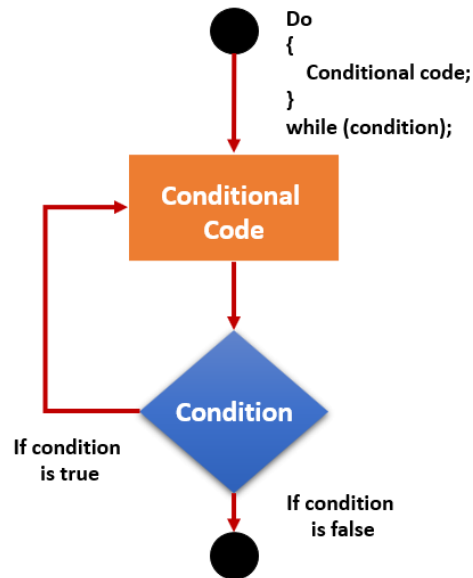
document.write("Loop stopped!");
</script>
<p>Set the variable to different value and then try...</p>
</body>
</html>

```

2. DO ... WHILE LOOP

Perulangan **do ... while** mirip dengan perulangan **while** kecuali bahwa pemeriksaan kondisi terjadi di akhir perulangan. Ini berarti bahwa loop akan selalu dieksekusi setidaknya sekali, bahkan jika kondisinya salah.

Bagan alur dari perulangan do-while adalah sebagai berikut:



Bagan alur dari perulangan do-while adalah sebagai berikut:

```

Do
{
    Statement(s) to be executed;
}
while (expression);

```

Note : Jangan lewatkan titik koma yang digunakan di akhir do ... while loop.

Contoh 2

```

<html>
<body>
<script type="text/javascript">
    var count = 0;
    document.write("Starting Loop" + "<br />");
    do{
        document.write("Current Count : " + count + "<br />");
        count++;
    }while (count < 5);

```

```
document.write ("Loop stopped!");  
</script>  
<p>Set the variable to different value and then try...</p>  
</body>  
</html>
```

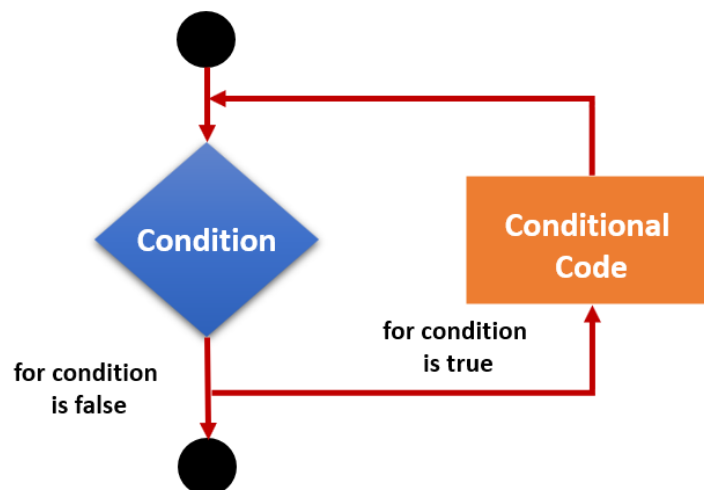
3. FOR LOOP

Perulangan 'for' adalah bentuk perulangan yang paling ringkas. Ini mencakup tiga bagian penting berikut:

- Inisialisasi loop tempat untuk menginisialisasi penghitung kami ke awal nilai.
- Pernyataan inisialisasi dijalankan sebelum loop dimulai.
- Pernyataan pengujian yang akan menguji apakah suatu kondisi yang diberikan benar atau tidak. Jika kondisinya benar, maka kode yang diberikan di dalam loop akan dieksekusi, jika tidak, kontrol akan keluar dari loop.

Pernyataan iterasi di mana anda dapat menambah atau mengurangi.

Bagan alur dari for loop dalam JavaScript adalah sebagai berikut:



Sintaks dari for loop dalam JavaScript adalah sebagai berikut:

```
for (initialization; test condition; iteration statement)  
{  
    Statement(s) to be executed if test condition is true  
}
```

Contoh penggunaan dalam HTML

Contoh 3

```
<html>  
<body>  
<script type="text/javascript">  
    var count;  
    document.write("Starting Loop" + "<br />");  
    for(count = 0; count < 10; count++){  
        document.write("Current Count : " + count );  
    }  
</script>  
</body>  
</html>
```

```
document.write("<br />");
}
document.write("Loop stopped!");

</script>
<p>Set the variable to different value and then try...</p>
</body>
</html>
```

4. FOR IN LOOP

For ... in loop digunakan untuk mengulang properti objek. Karena kita belum membahas Objects, Anda mungkin merasa kurang faham dengan loop ini. Tapi begitu Anda memahami bagaimana objek berperilaku dalam JavaScript, Anda akan menemukan loop ini sangat berguna.

```
for (variablename in object)
{
    statement or block to execute
}
```

Dalam setiap iterasi, satu properti dari objek ditetapkan ke nama variabel dan loop ini berlanjut hingga semua properti objek habis.

Contoh 4

```
<html>
<body>
<script type="text/javascript">
    var aProperty;
    document.write("Navigator Object Properties<br /> ");
    for (aProperty in navigator)
    {
        document.write(aProperty);
        document.write("<br />");
    }
    document.write ("Exiting from the loop!");
</script>
<p>Set the variable to different object and then try...</p>
</body>
</html>
```

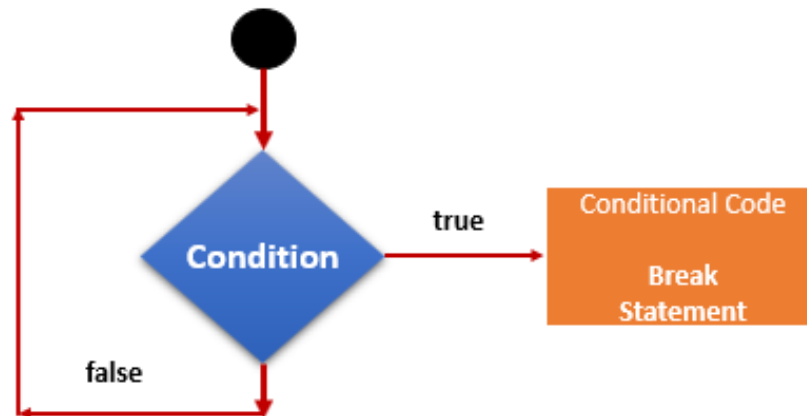
5. LOOP CONTROL

JavaScript menyediakan kontrol penuh untuk menangani loop dan beralih pernyataan. Mungkin ada situasi ketika anda harus keluar dari lingkaran tanpa mencapai dasarnya. Mungkin juga ada situasi ketika anda ingin melewati bagian dari blok kode dan memulai iterasi tampilan berikutnya. Untuk menangani semua situasi seperti itu, JavaScript menyediakan pernyataan *break* dan *continue*. Pernyataan-pernyataan ini digunakan untuk keluar dari setiap loop atau untuk memulai iterasi berikutnya dari setiap loop masing-masing.

a. Statement Break

Pernyataan break, yang diperkenalkan secara singkat dengan pernyataan switch, digunakan untuk

keluar dari loop lebih awal, keluar dari kurung kurawal yang menyertainya.



Contoh berikut mengilustrasikan penggunaan pernyataan break dengan perulangan while. Perhatikan bagaimana loop pecah lebih awal setelah x mencapai 5 dan mencapai pernyataan document.write (..) tepat di bawah ke kurung kurawal penutup:

Contoh 5

```
<html>
<body>
  <script type="text/javascript">

    var x = 1;
    document.write("Entering the loop<br /> ");
    while (x < 20)
    {
      if (x == 5)
      {
        break; // breaks out of loop completely
      }
      x = x + 1;
      document.write( x + "<br />");
    }
    document.write("Exiting the loop!<br /> ");

  </script>
<p>Set the variable to different value and then try...</p>
</body>
</html>
```

b. Statement Continue

Pernyataan continue memberitahu interpreter untuk segera memulai iterasi berikutnya dari loop dan melewati blok kode yang tersisa. Kapan pernyataan lanjutan ditemukan, aliran program segera pindah

ke ekspresi pemeriksaan loop dan jika kondisinya tetap benar, maka ia memulai iterasi berikutnya, jika tidak, kontrol keluar dari loop.

Contoh ini mengilustrasikan penggunaan pernyataan continue dengan perulangan while. Perhatikan bagaimana pernyataan continue digunakan untuk melewati pencetakan ketika indeks yang disimpan dalam variabel x mencapai 5.

Contoh 6

```
<html>
<body>
  <script type="text/javascript">
    var x = 1;
    document.write("Entering the loop<br /> ");
    while (x < 10)
    {
      x = x + 1;
      if (x == 5)
      {
        continue; // skip rest of the loop body
      }
      document.write( x + "<br />");
    }
    document.write("Exiting the loop!<br /> ");
  </script>
<p>Set the variable to different value and then try...</p>
</body>
</html>
```

Jika ada materi yang belum anda pahami atau ada hal – hal yang mau anda ditanyakan, Maka janganlah segan – segan untuk menghubungi dosen anda.

Untuk lebih memahami mengenai Perulangan JavaScript, silakan anda baca dan pelajari materi mengenai Perulangan JavaScript di buku-buku sumber lain. Atau dapat juga melihat pada channel youtube.

C. MENGECEK PEMAHAMAN



Ayo saatnya kita berlatih !

Jawablah latihan ini secara singkat dan jelas yang telah dilampirkan pada kuliah hari ini sesuai dengan disiplin ilmu kita. Jika anda memiliki pertanyaan atau komentar tentang latihan ini, anda dapat membagikannya di bawah ini. Periksa jawaban Anda menggunakan kunci jawaban.

1. Apa yang anda ketahui tentang *Perulangan*
2. Buatlah Perulangan untuk menampilkan data dari 1 sampai 20 dengan menggunakan perulangan for, dan while menggunakan bahasa pemrograman JavaScript :

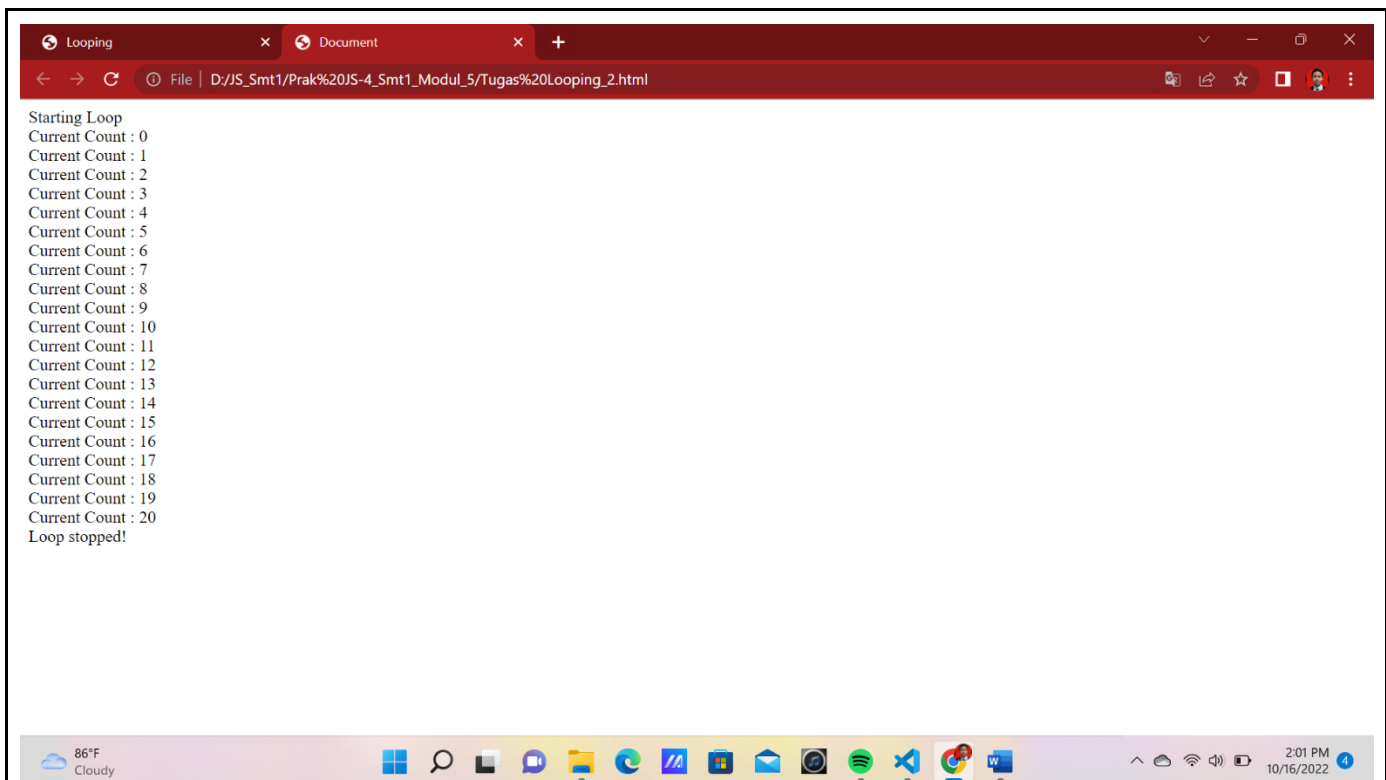
Jawaban

1. Looping atau perulangan adalah suatu bentuk kegiatan mengulang suatu statement sampai batas yang diinginkan. Looping terbagi kedalam 5 jenis, yaitu : While Loop, Do-While Loop, For Loop, For In Loop, dan Loop Control.

2. Menampilkan data 1 sampai 20

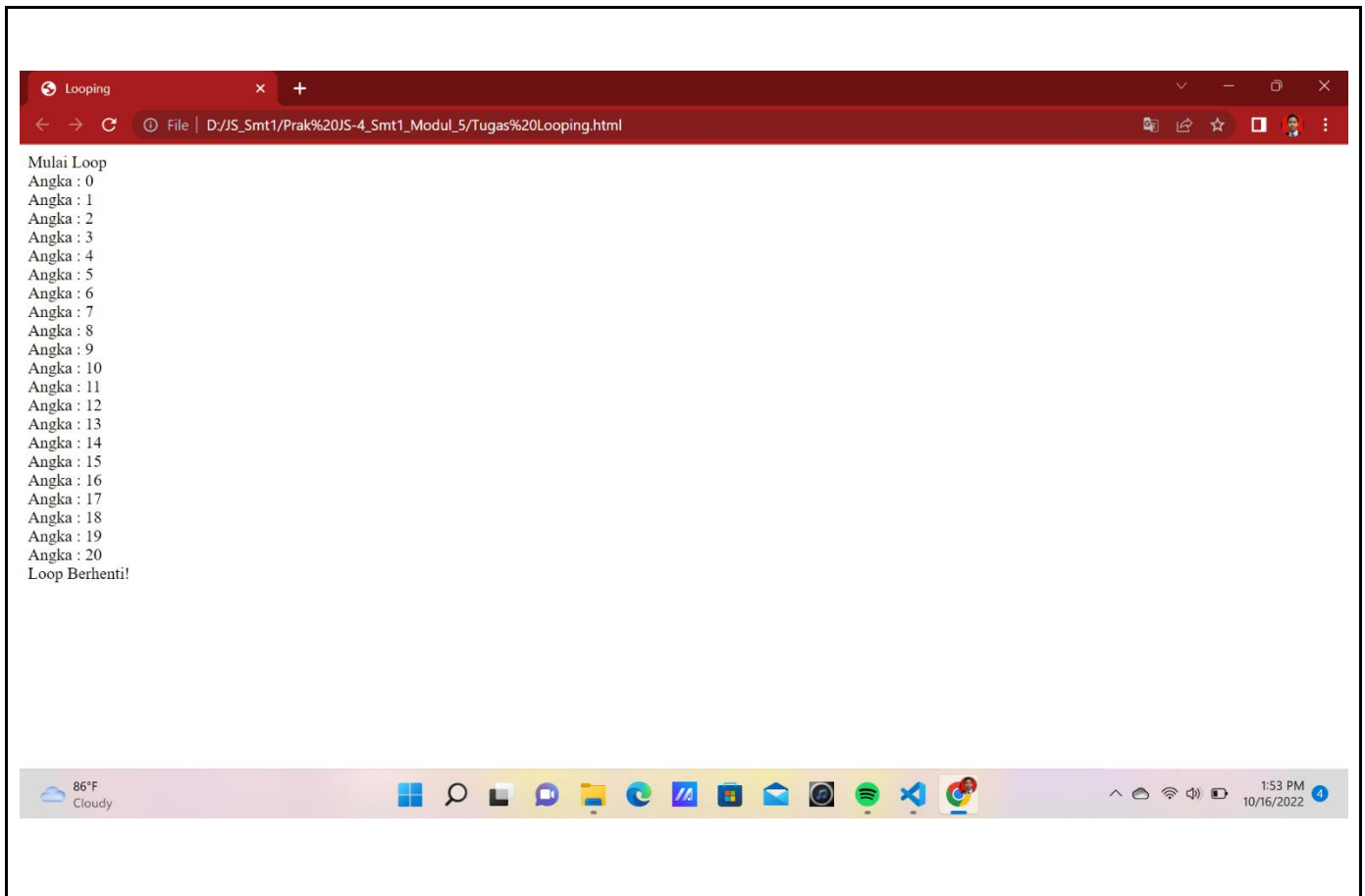
- **Perulangan For**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <script type="text/javascript">
    var count;
    document.write("Starting Loop" + "<br>");
    for(count = 0; count <= 20; count++)
    {
      document.write("Current Count : "+ count + "<br>");
    }
    document.write("Loop stopped!");
  </script>
</body>
</html>
```

- **Perulangan While**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Looping</title>
</head>
<body>
  <script type="text/javascript">
    var count = 0;
    document.write("Mulai Loop <br>");
    while (count <= 20)
    {
      document.write("Angka : " + count + "<br>");
      count++;
    }
    document.write("Loop Berhenti!");
  </script>
</body>
</html>
```

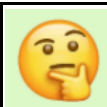


The screenshot shows a web browser window with a red title bar labeled 'Looping'. The address bar shows the file path 'D:/JS_Smt1/Prak%20JS-4_Smt1_Modul_5/Tugas%20Looping.html'. The main content area displays the output of a JavaScript loop: 'Mulai Loop', followed by 'Angka : 0' through 'Angka : 20', and finally 'Loop Berhenti!'. The Windows taskbar at the bottom shows the date as 10/16/2022 and the time as 1:53 PM.

```
Mulai Loop
Angka : 0
Angka : 1
Angka : 2
Angka : 3
Angka : 4
Angka : 5
Angka : 6
Angka : 7
Angka : 8
Angka : 9
Angka : 10
Angka : 11
Angka : 12
Angka : 13
Angka : 14
Angka : 15
Angka : 16
Angka : 17
Angka : 18
Angka : 19
Angka : 20
Loop Berhenti!
```

D. PENUTUP PEMBELAJARAN

Anda telah mencapai bagian terakhir dari kuliah hari ini. Sekedar ulasan, baca ringkasan singkat tentang JavaScript di bawah ini.



IDE KUNCI

Berpikir tentang Pembelajaran



Silakan periksa jawaban yang anda kerjakan pada soal latihan MODUL 5 dengan kunci jawaban di bawah ini!

Kunci Jawaban

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

<p>Click the button to loop through a block of code five times.</p>

<button onclick="myFunction()">Try it</button>

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

<script>
function myFunction() {
  var text = "";
  var i;
  for (i = 1; i < 21; i++) {
    text += "The number is " + i + "<br>";
  }
  document.getElementById("demo").innerHTML = text;
}
</script>
</body>
</html>
```