

MATERI AJAR 5

Array Javascript

1.1. Materi Pembelajaran

Array Dimensi Satu dan Array Dimensi Dua

1.2. Kemampuan Akhir Pembelajaran

Sub CPMK 6 : Mahasiswa mampu menerapkan variable berindex pada web page dinamis

1.3. Materi Ajar

a. Array satu dimensi

Array adalah suatu variable yang dapat memuat beberapa nilai secara berurutan atau seri. Artinya variable yang dideklarasikan sebagai array isinya tidak satu. Berikut adalah pendeklarasian untuk array.

```
nama = new Array(3)
```

Pernyataan diatas menunjukkan bahwa variabel nama memiliki 3 elemen. Ketiga elemen tersebut akan memiliki nilai masing-masing 0, 1, 2. Nilai pertama = 0. Untuk mengisikan ketiga elemen tersebut kita dapat melakukan dengan cara :

```
nama[0] ="Dian"  
nama[1] ="Andri"  
nama[2]="Irawan"
```

Selain cara diatas kita juga bisa mendeklarasikan array sekaligus dengan mengisikan elemen-elemennya.

```
nama = new Array("Dian","Andri","Irawan")
```

Maka hasilnya seperti dibawah ini :

```
nama yang ke 1 [0] isinya adalah "Dian"  
nama yang ke 2 [1] isinya adalah "Andri"  
nama yang ke 3 [2] isinya adalah "Irawan"
```

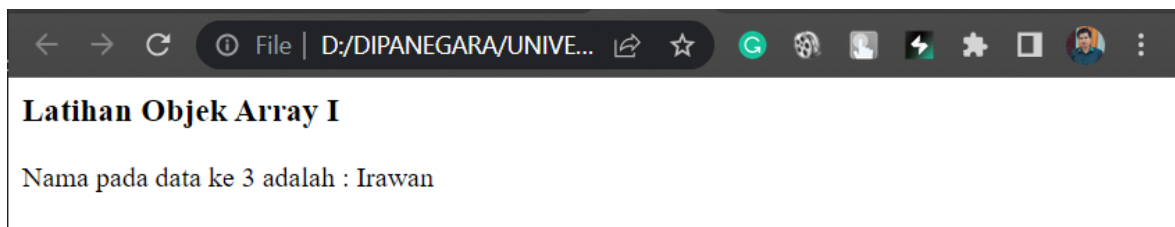
Contoh Program JavaScript

```

<HTML>
<HEAD>
<TITLE> Latihan Objek Array</TITLE>
<BODY>
<h3>Latihan Objek Array I</H3>
Nama pada data ke 3 adalah :
<SCRIPT LANGUAGE="JavaScript">
<!-- Memyembunyikan kode dari browser non-js
function cobaarray()
{
nama = new Array("Dian","Andri","Irawan");
document.write(nama[2]);
}
// akhir dari penyembunyian -->
</SCRIPT>
<SCRIPT LANGUAGE="JavaScript">
cobaarray();
</SCRIPT>
</BODY>
</HTML>

```

Hasil :

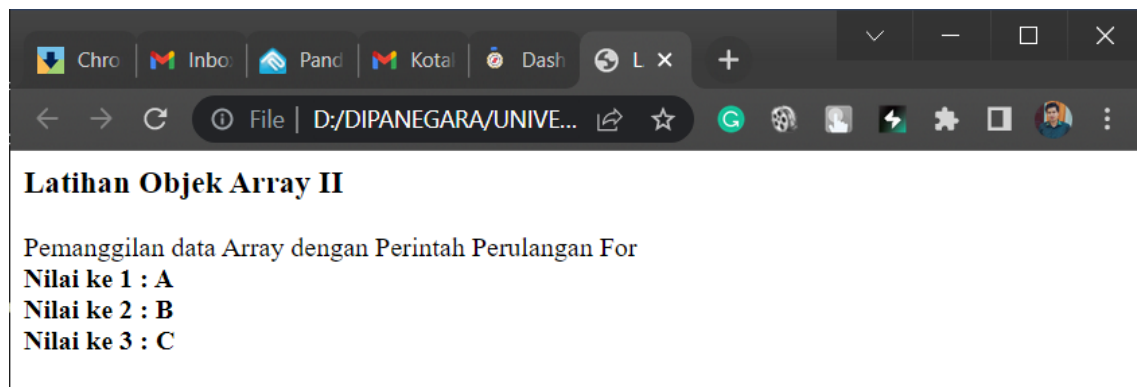
**Contoh Program JavaScript**

```

<HTML>
<HEAD>
<TITLE> Latihan Objek Array</TITLE>
<BODY>
<h3>Latihan Objek Array II</H3>
Pemanggilan data Array dengan Perintah Perulangan For<BR>
<SCRIPT LANGUAGE="JavaScript">
<!-- Memyembunyikan kode dari browser non-js
var nilai = new Array(3);
nilai[0]="A";
nilai[1]="B";
nilai[2]="C";
for (a=0;a<3;++a)
{
document.writeln("<B>Nilai ke "+ [a+1] +" : "+nilai[a]+"<BR>");
}

```

```
// akhir dari penyembunyian -->
</SCRIPT>
<SCRIPT LANGUAGE="JavaScript">
</SCRIPT>
</BODY>
</HTML>
```



Array dua dimesi

Array 2 Dimensi adalah Sebuah array yang menampung array lain sebagai data nilai dari setiap indeks array penampung. Jika kita sudah mengerti mengenai array 1 dimensi, kita bisa langsung membayangkan dan bisa langsung mengerti dari definisi array 2 dimensi diatas.

```
<html>
  <body>
    <script type="text/javascript">
      var warna = [
        ["putih","merah","hitam"],
        ["hijau","abu abu","biru"],
      ];
      document.write(warna[1][2]);
    </script>
  </body>
</html>
```

Penjelasan baris program

sebuah variable dengan nama warna lalu didalamnya saya membuat 2 kumpulan data array dimana data pertama yang berarti 0 pada urutan array adalah :

["putih","merah","hitam"]

Dan array kedua yang berarti 1 dalam urutan data array adalah :

["hijau","abu abu","biru"]

Untuk mencetak salah satu data di atas contohnya biru, dengan membuat fungsi untuk mencetak dalam javascript yaitu

document.write(warna[1][2]);

Contoh Program JavaScript

```

<!DOCTYPE html>
<html>
<head>
  <title>Tabel Nilai dengan array dua dimensi</title>
</head>
<body>
  <script>
    var nama = [
      ["Eka", "90"], ["Agustin", "70"], ["Chandra", "85"],
      ["Tika", "80"], ["Fiersa", "75"],
    ];
  </script>
  <table border="2" height="400px" width="500px" align="center">
    <tr>
      <th>NAMA</th>
      <th>NILAI</th>
    </tr>
    <tr>
      <td>
        <script>
          document.write (nama [2] [0] );
        </script>
      </td>
      <td>
        <script>
          document.write (nama [0] [1] );
        </script>
      </td>
    </tr>
    <tr>
      <td>
        <script>
          document.write (nama [4] [0] );
        </script>
      </td>
      <td>
        <script>
          document.write (nama [3] [1] );
        </script>
      </td>
    </tr>
  </table>

```

```
        </td>
    </tr>
    <tr>
        <td>
            <script>
                document.write(nama[1][0]);
            </script>
        </td>

        <td>
            <script>
                document.write(nama[2][1]);
            </script>
        </td>
    </tr>

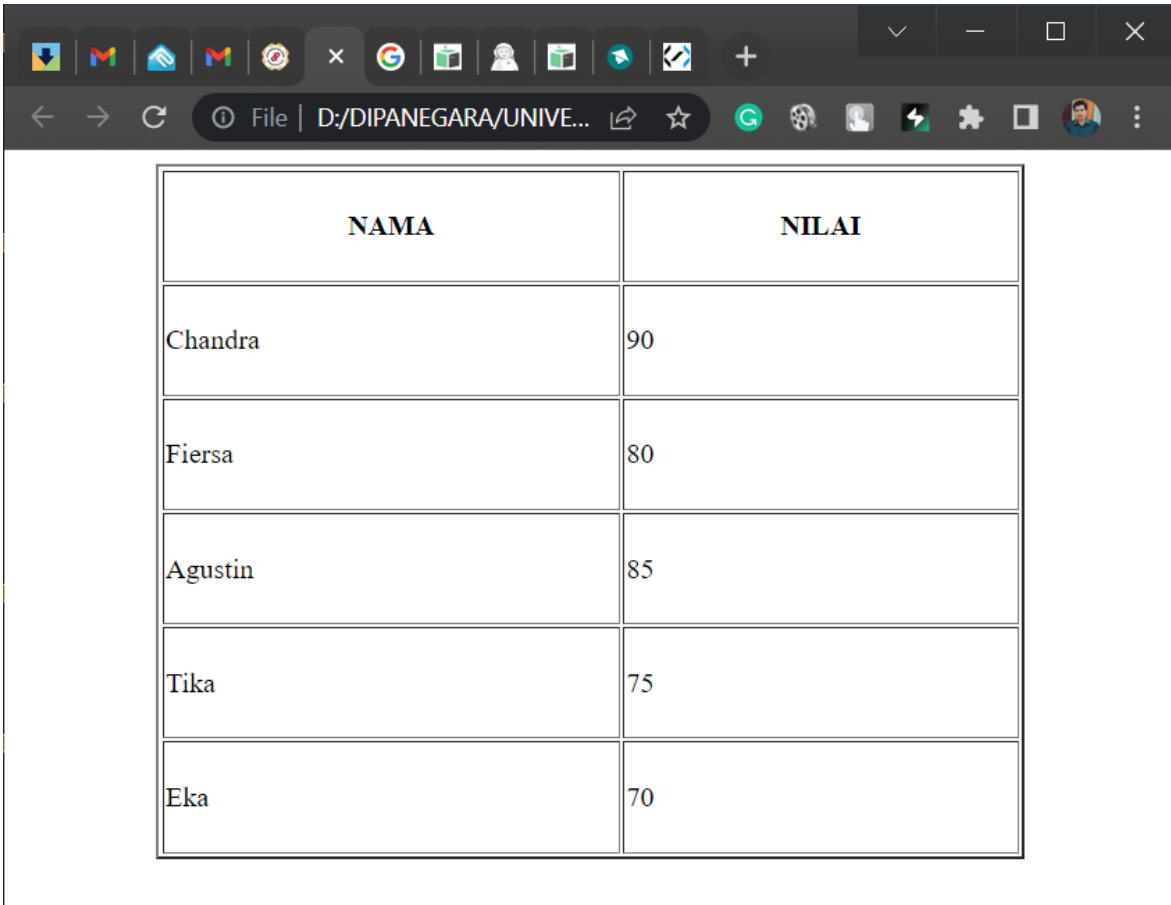
    <tr>
        <td>
            <script>
                document.write(nama[3][0]);
            </script>
        </td>

        <td>
            <script>
                document.write(nama[4][1]);
            </script>
        </td>
    </tr>

    <tr>
        <td>
            <script>
                document.write(nama[0][0]);
            </script>
        </td>

        <td>
            <script>
                document.write(nama[1][1]);
            </script>
        </td>
    </tr>

</table>
</body>
</html>
```



The image shows a screenshot of a web browser window. The address bar displays the file path "D:/DIPANEGARA/UNIVE...". The main content area contains a table with two columns: "NAMA" and "NILAI". The table lists five students with their respective scores.

NAMA	NILAI
Chandra	90
Fiersa	80
Agustin	85
Tika	75
Eka	70

b. Decision Statement

Untuk membuat suatu halaman yang dinamis dan interaktif, perancang halaman Web membutuhkan perintah-perintah yang dapat mengatur aliran dari informasi. Berdasarkan hasil komputasi yang telah dilakukan, JavaScript akan membuat keputusan jalur mana yang akan dieksekusi. Pada dasarnya dalam JavaScript terdapat dua macam pernyataan percabangan yaitu `if..else` dan `switch`

If..Else

Pernyataan ini digunakan untuk menguji sebuah kondisi dan kemudian mengeksekusi pernyataan tertentu bila kondisi tersebut terpenuhi, dan mengeksekusi pernyataan lain bila kondisi tersebut tidak terpenuhi.

```
if (kondisi) {
    //pernyataan1 dieksekusi
    //bila kondisi terpenuhi
} else {
    //pernyataan2 dieksekusi
    //bila kondisi tidak terpenuhi
}
```

kondisi adalah ekspresi JavaScript yang mana hasil evaluasinya memiliki nilai Boolean **true** atau **false**

Untuk kasus yang melibatkan lebih banyak kondisi, maka kita dapat meletakkan pernyataan `if` lain setelah `else`

```
if (kondisi1){
    //pernyataan1 dieksekusi
    //bila kondisi1 terpenuhi
} else if (kondisi2) {
    //pernyataan2 dieksekusi
    //bila kondisi1 tidak terpenuhi
} else {
    //pernyataan3 dieksekusi
    //bila kondisi2 tidak terpenuhi
}
```

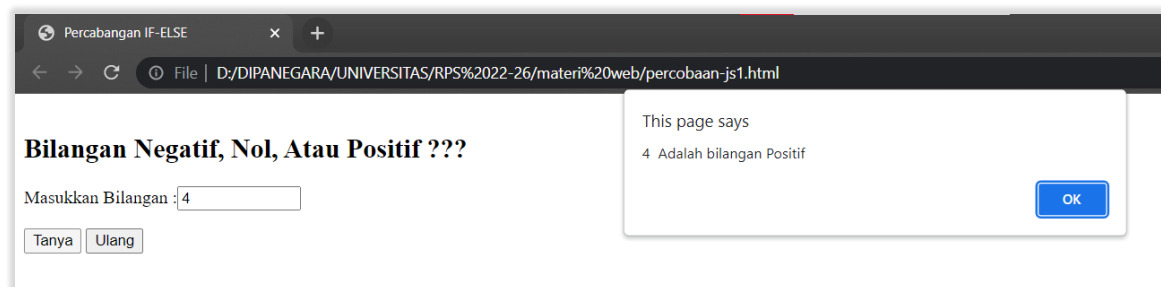
Contoh Program JavaScript

```

<HTML>
<HEAD><TITLE>Percabangan IF-ELSE</TITLE></HEAD>
<BODY>
<SCRIPT language="JavaScript">
<!--
function tanyabilangan(){
  var bil = parseFloat(document.fform.bilangan.value);
  var jenis = " ";
  if(isNaN(bil)) {
    alert("Anda Belum memasukkan Bilangan");
  } else {
    if (bil > 0) {
      jenis = " Adalah bilangan Positif";
    } else if (bil < 0) {
      jenis = " Adalah bilangan Negatif";
    } else {
      jenis = " Adalah Nol";
    }
    alert (bil+" "+jenis);
  }
}
//--></SCRIPT>
<FORM NAME ="fform">
<H2><BR>Bilangan Negatif, Nol, Atau Positif ???</H2>
Masukkan Bilangan :<input type="text" size="11" name="bilangan">
<P>
<INPUT TYPE="button" value="Tanya" onclick="tanyabilangan()">
<INPUT TYPE="reset" value="Ulang"> </p>
</FORM>
</BODY>
</HTML>

```

Hasil :



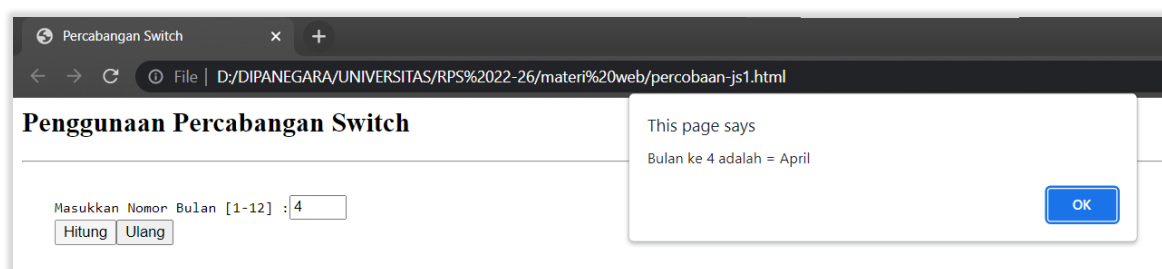
Switch

Selain menggunakan if..else, percabangan juga dapat ditangani dengan perintah switch. Dengan kata lain pernyataan switch digunakan untuk menyederhanakan pernyataan if..else yang terlalu banyak.

Contoh Program JavaScript

```
<HTML>
<HEAD><TITLE>Percabangan Switch</TITLE></HEAD>
<BODY>
<SCRIPT language="JavaScript">
<!--
function tanyabulan(){
    var bulan = parseFloat(document.fform.ibulan.value);
    var namabulan=" ";
    switch (bulan){
        case 1 : namabulan="Bulan ke 1 adalah = Januari";break;
        case 2 : namabulan="Bulan ke 2 adalah = Februari";break;
        case 3 : namabulan="Bulan ke 3 adalah = Maret";break;
        case 4 : namabulan="Bulan ke 4 adalah = April";break;
        case 5 : namabulan="Bulan ke 5 adalah = Mei";break;
        case 6 : namabulan="Bulan ke 6 adalah = Juni";break;
        case 7 : namabulan="Bulan ke 7 adalah = Juli";break;
        case 8 : namabulan="Bulan ke 8 adalah = Agustus";break;
        case 9 : namabulan="Bulan ke 9 adalah = September";break;
        case 10 : namabulan="Bulan ke 10 adalah = Oktober";break;
        case 11 : namabulan="Bulan ke 11 adalah = November";break;
        case 12 : namabulan="Bulan ke 12 adalah = Desember";break;
        default : namabulan="Anda salah mengisi";
    }
    alert(namabulan);
}
//--></SCRIPT>
<FORM NAME ="fform">
<H2>Penggunaan Percabangan Switch</H2><HR>
<PRE>
Masukkan Nomor Bulan [1-12] :<input type="text" size="2"
name="ibulan">
<INPUT TYPE="button" value="Hitung" onclick="tanyabulan()"><INPUT
TYPE="reset" value="Ulang">
</PRE>
</FORM>
</BODY>
</HTML>
```

Hasil



c. Loop Control

Untuk mengulang kejadian beberapa kali maka kita membutuhkan proses perulangan. Pada JavaScript dikenal beberapa metode/cara perulangan.

Perulangan For

Digunakan untuk mengeksekusi pernyataan-pernyataan beberapa kali. Perulangan For paling sering dipakai, jika anda sudah tahu akhir dari perulangan tersebut. . Perintah **for** mengulang suatu loop sampai kondisi menghasilkan evaluasi true atau loop keluar dengan perintah **break** .

Contoh :

```
for (nilai awal;kondisi;penambahan){
    ulang pernyataan ini;
}
```

Contoh dalam program :

```
For(x=1;x<=10;x++){
    document.writeln("Belajar JavaScript Yuuuuu..");
}
```

Contoh Program JavaScript

```
<HTML>
<HEAD><TITLE>Latihan Perulangan I</TITLE></HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!-- menyembunyikan script dan browser non-JS
document.writeln("<H2>Latihan Perulangan For</H2>");
document.writeln("-----");
for (i=1; i<=10; i++) {
    sq=i*i;
    document.writeln("<PRE>");
    document.write("Angka:" + i + " -----> Kuadrat: " + sq + "<BR>");
    document.writeln("</PRE>");
}
document.writeln("-----");
// akhir dari penyembunyian -->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Hasil

```

Latihan Perulangan For
-----
Angka:1 -----> Kuadrat: 1
Angka:2 -----> Kuadrat: 4
Angka:3 -----> Kuadrat: 9
Angka:4 -----> Kuadrat: 16
Angka:5 -----> Kuadrat: 25
Angka:6 -----> Kuadrat: 36
Angka:7 -----> Kuadrat: 49
Angka:8 -----> Kuadrat: 64
Angka:9 -----> Kuadrat: 81
Angka:10 -----> Kuadrat: 100
-----

```

Perulangan While

Perulangan lain yang dapat digunakan adalah dengan menggunakan perintah While. Perintah while digunakan untuk perulangan yang tidak diketahui berapa kali proses perulangannya. Perintah while terus mengulangi loop selama kondisi memiliki nilai true. Syntax untuk perintah while adalah sebagai berikut :

```

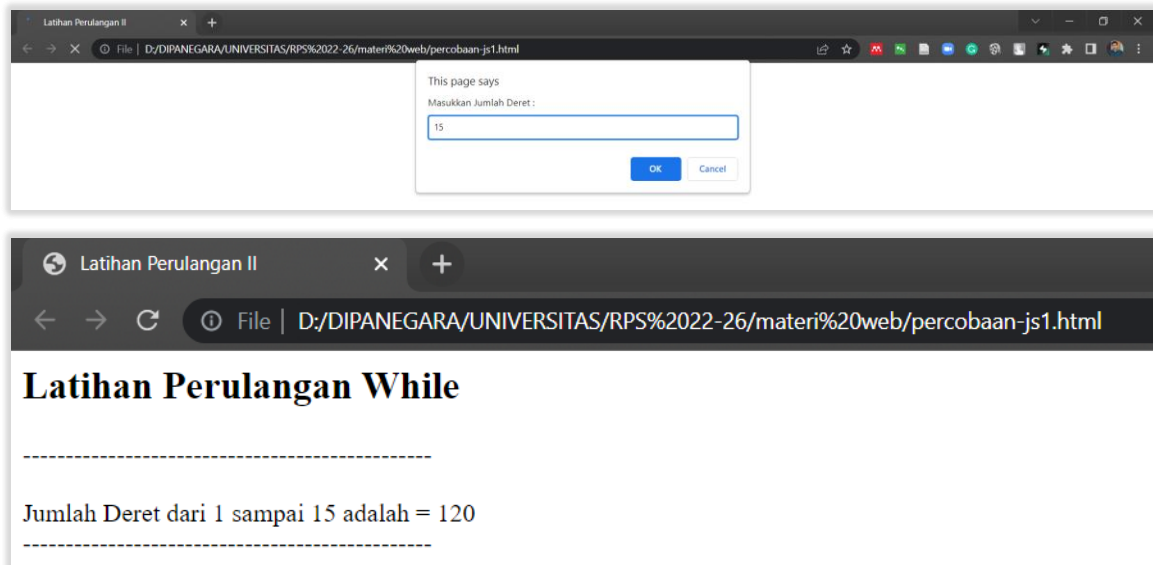
while (kondisi){
    ulang pernyataan ini;
}

```

Contoh Program JavaScript

```
<HTML>
<HEAD><TITLE>Latihan Perulangan II</TITLE></HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!-- menyembunyikan script dan browser non-JS
var deret = prompt('Masukkan Jumlah Deret :','');
document.writeln("<H2>Latihan Perulangan While</H2>");
document.writeln("-----");
document.writeln("<BR>");
var jml = 0.0;
var angka = 1;
while (angka <= deret){
    jml= jml+angka;
    angka++
}
document.writeln("<BR>");
document.writeln("Jumlah Deret dari 1 sampai "+deret+" adalah =
"+jml);
document.writeln("<BR>");
document.writeln("-----");
// akhir dari penyembunyian -->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Hasil



Perulangan Do While

Perulangan ini hampir sama seperti while, digunakan apabila kita belum tahu berapa kali perulangan harus dilakukan. Bedanya pernyataan do..while pengujiannya dilakukan di akhir pernyataan.

```
Do
{
    //pernyataan1 dieksekusi
}
while (kondisi);
```

Contoh Program JavaScript

```

<HTML>
<HEAD><TITLE>Latihan Perulangan III (Do-While)</TITLE></HEAD>
<BODY>
<SCRIPT LANGUAGE="JavaScript">
<!-- menyembunyikan script dan browser non-JS
function pass(){
    var coba = 1;
    do{
        p = prompt("Tuliskan password dengan benar","");
        if (p=="UNDIPA"){
            alert("Selamat Datang di UNDIPA MAKASSAR");
            window.open("welcome.html");
            break;
        }else {
            alert("Password Salah !!! Ulangi lagi.");
        }
        if (coba==3){
            alert("maaf, kesempatan anda hanya 3 kali");
            history.go(-1);
        }
        coba=coba+1;
    } while (coba<=3)
}
// akhir dari penyembunyian -->
</SCRIPT>
<FORM METHOD="post">
<INPUT TYPE="button" VALUE="Password" ONCLICK="pass()">
</FORM>
</BODY>
</HTML>

```

HASIL

