

BAB 4

ARRAY DALAM JAVASCRIPT

Oleh: Taryana Suryana M.Kom
Teknik Informatika Unikom

taryanarx@email.unikom.ac.id

taryanarx@gmail.com

[Line/Telegram: 081221480577](https://line.me/tv/081221480577)

4.1. Pengertian Array

Array merupakan suatu tipe data yang berisi kumpulan nilai atau tipe data lain, misalnya seperti integer, string, number, boolean, floating dan lain-lain, selain itu Array juga digunakan untuk menampung banyak nilai dalam satu variabel

Dalam JavaScript, Array memiliki 2 cara untuk menuliskannya, cara pertama penulisan dengan menggunakan kurung siku [], yang kedua menggunakan keyword **new Array()**.

Urutan Indeks array dimulai dari nol (0) dan bukan dari 1

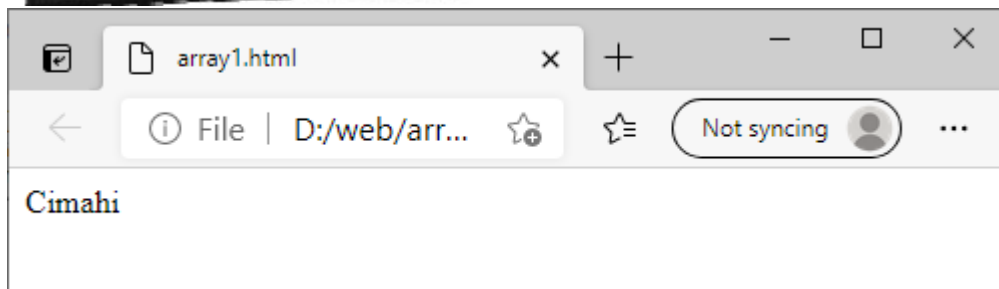
Sintak Penulisan Array

```
var array_name = [item1, item2, ...];
```

Contoh deklarasi Array dan cara menampilkannya

NamaFile:array1.html

```
<script>  
  var kota= ['Bandung', 'Ciwaruga', 'Cimahi','Garut'];  
  document.write(kota[2]);  
</script>
```



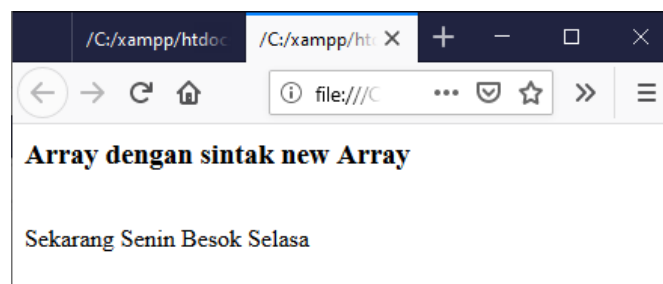
Gambar 4.1. Menampilkan Array nomor indek ke 2

Sintak Deklarasi Array Menggunakan Perintah new array

```
var array_name = new array(item1, item2, ...);
```

Contoh kode Program Menggunakan new array
NamaFile:array2.html

```
<h3>Array dengan sintak new Array</h3>
<br>
<script>
var hari=new Array('Minggu','Senin','Selasa', 'Rabu','Kamis','Jumat');
document.write("Sekarang ",hari[1]," Besok  ",hari[2],"<br>");
</script>
```



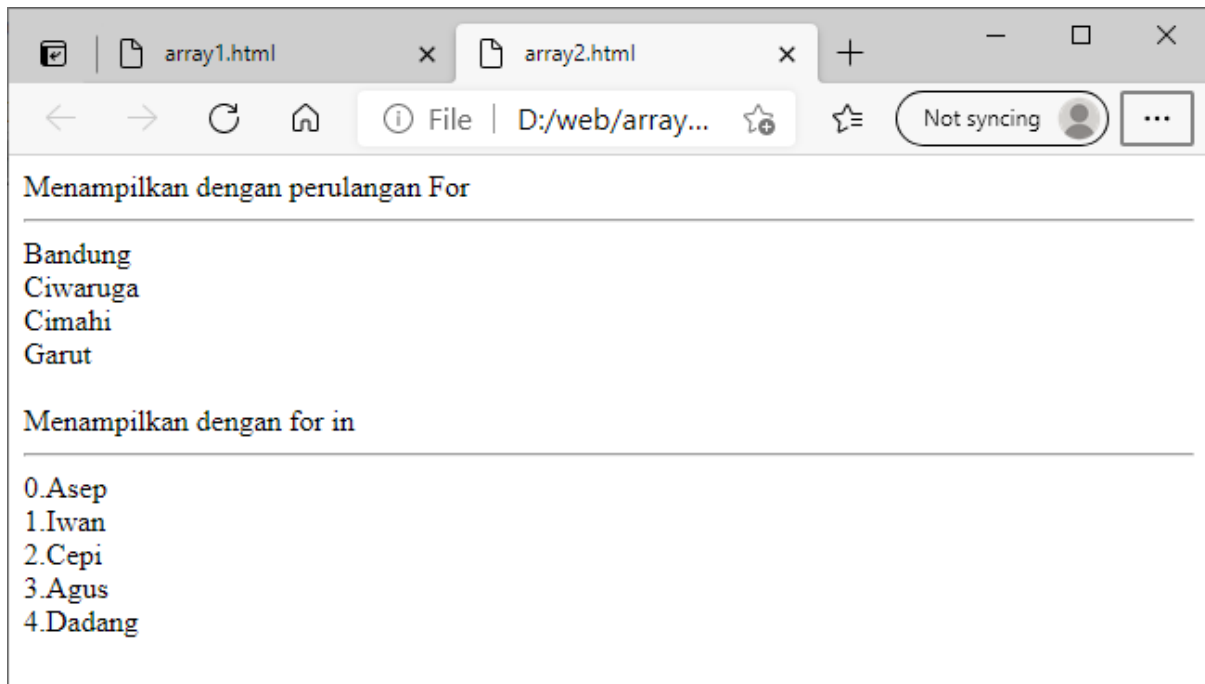
Gambar 4.2 Contoh Keluaran Array

Jika anda ingin menampilkan seluruh isi array, maka dapat dilakukan dengan menggunakan perulangan.

Namafile:array3.html

```
<script>
var kota= ['Bandung', 'Ciwaruga', 'Cimahi','Garut'];
//Menampilkan seluruh isi array
document.write("Menampilkan dengan perulangan For <hr>");
for(i=0;i<kota.length;i++)
    document.write(kota[i],"<br>");

var nama= ['Asep', 'Iwan', 'Cepi','Agus','Dadang'];
document.write("<br>Menampilkan dengan for in<hr>");
for(i in nama)
    document.write(i,".",nama[i]+"<br>");
</script>
```



Gambar 4.3.Menampilkan seluruh isi Array

4.2. Jenis-Jenis Method Array

Untuk mempermudah penggunaan array, **Java Script** menyediakan beberapa fungsi yang melekat pada setiap variabel array (atau lebih tepat disebut dengan **method**).

Berikut adalah **method objek array** di dalam **JavaScript**.

- 1) join()
- 2) reserve()
- 3) sort()
- 4) concat()
- 5) slice()
- 6) splice()
- 7) push dan pop()
- 8) shift() dan unshift()
- 9) toString() dan toLocaleString()

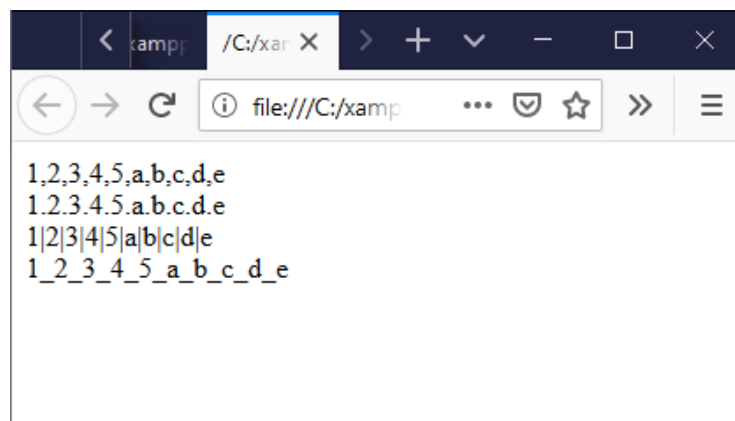
4.2.1. Method Array Join ()

Method Array .join() berfungsi untuk menggabungkan elemen array menjadi sebuah string. Sebuah argumen bisa diinputkan kedalam method ini yang akan berfungsi sebagai karakter 'pembatas' untuk menyatukan array.

Berikut adalah contoh penggunaan method join() dalam JavaScript:

NamaFile:array5.html

```
<script>
  va array1 = [1,2,3,4,5,"a","b","c","d","e"];
  document.write(array1.join(),"<br>");
  document.write(array1.join("."),"<br>");
  document.write(array1.join("|"),"<br>");
  document.write(array1.join("_"),"<br>");
</script>
```



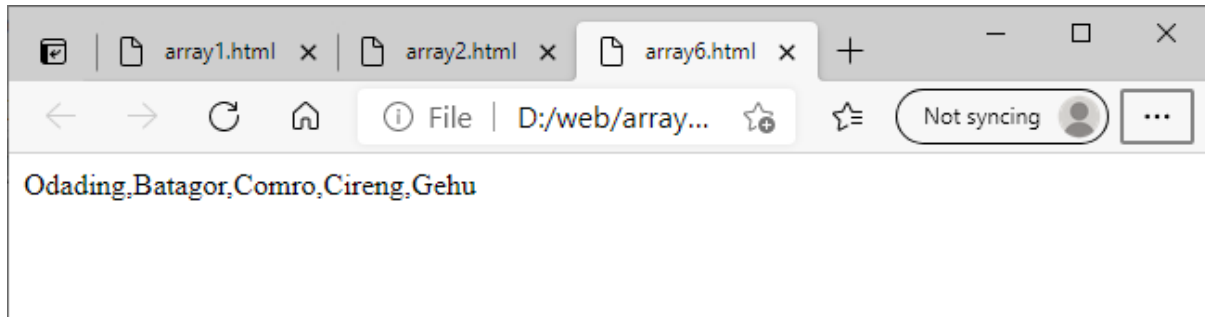
Gambar 4.4. Hasil Method Join

4.2.2. Method Array Reverse()

Method Array.reverse() digunakan untuk membalikkan urutan dari elemen di dalam array. Method ini akan mengubah array asal. Berikut adalah contoh penggunaan method reverse():

NamaFile:Array6.html

```
<script>  
  var cemilan= ['Gehu', 'Cireng', 'Comro','Batagor','Odading'];  
  cemilan.reverse();  
  document.write(cemilan.toString(),"<br>");  
</script>
```



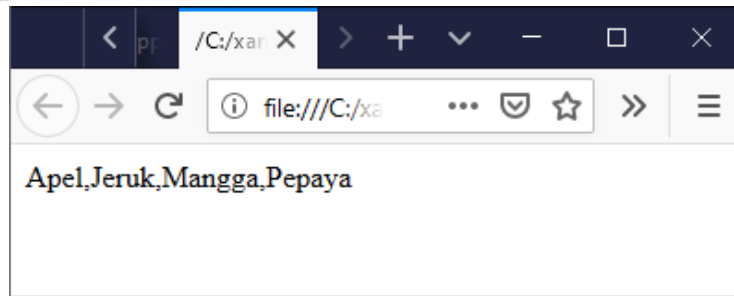
Gambar 4.5. Hasil Method Reverse()

4.2.3.Method Array sort()

Method **Array.sort()** digunakan untuk mengurutkan elemen dari array. Jika method ini dipanggil tanpa *argumen*, elemen dari array akan diurutkan berdasarkan abjad. Jika elemen di dalam array bukan huruf, maka akan dikonversi sementara menjadi **string**, baru kemudian dilakukan proses pengurutan. Jika ditemukan elemen dengan nilai **null** atau **undefined**, akan di tempatkan pada urutan akhir.

NamaFile:Array7.html

```
<script>  
  var buah=['Pepaya', 'Apel', 'Jeruk','Mangga'];  
  buah.sort();  
  document.write(buah.toString(),"<br>");  
</script>
```



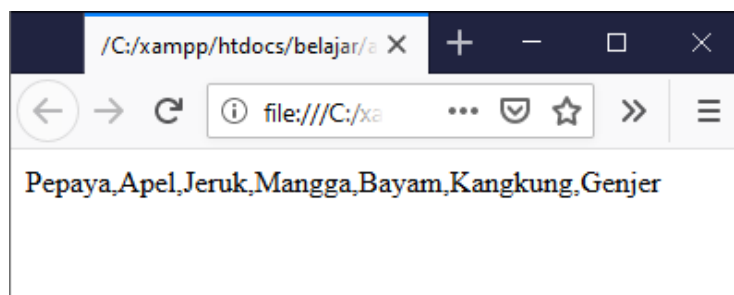
Gambar 4.6. Array yang sudah di urutkan

4.2.4. Method Array concat()

Method `array.concat()` berfungsi untuk "penyambungan" array. Hasil akhir dari pemanggilan fungsi ini adalah array baru dengan penambahan dari method `concat()`. Elemen yang ditambahkan diinput sebagai argumen dari method `concat()`. Berikut adalah contohnya:

NamaFile: Array8.html

```
<script>
  var buah=['Pepaya', 'Apel', 'Jeruk','Mangga'];
  var gabungan=buah.concat('Bayam','Kangkung','Genjer');
  document.write(gabungan.toString(),"<br>");
</script>
```



Gambar 4.7. Menggabung Array

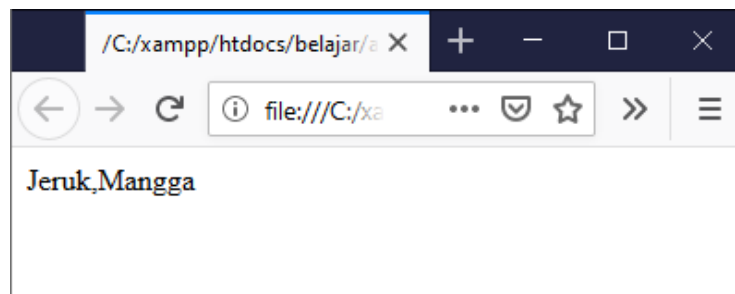
4.2.5. Method Array slice()

Method `Array.slice()` digunakan untuk 'memotong' array menjadi array baru, atau menjadi 'sub-array' dari array asal. Method `slice()` ini membutuhkan 2 buah argumen yang berisi posisi index awal dan akhir pemotongan.

Jika hanya diberikan 1 argumen, maka method ini akan mengembalikan array baru dimulai dari posisi argumen sampai dengan akhir array. Jika argumen bernilai negatif, maka perhitungan akan dimulai dari akhir array. Berikut adalah contoh penggunaan method array `slice()`:

NamaFile:Array9.html

```
<script>
  var buah=['Pepaya', 'Apel', 'Jeruk','Mangga'];
  var hasil=buah.slice(2,4);
  document.write(hasil.toString(),"<br>");
</script>
```



Gambar 4.8. Hasil Fungsi Slice

Dari contoh kode yang kita lakukan, terlihat bahwa index terakhir tidak termasuk ke dalam array baru. **`buah.slice(2,4)`** akan mengembalikan elemen array dari index ke-2, ke-3, namun tidak ke-4.

4.2.6. Method Objek Array: Array.splice()

Walaupun memiliki nama method yang mirip dengan slice(), method splice() sepenuhnya berbeda. Method splice() adalah method 'serba-bisa' yang bisa digunakan untuk memotong array, menambahkan elemen array, bahkan melakukan keduanya sekaligus. Tidak seperti method slice() dan concat(), pemanggilan method ini akan mengubah array asal.

Jika hanya diberikan satu buah argumen, method splice() akan berfungsi 'menghapus' array asal mulai dari index yang diberikan, dan mengembalikan nilai array yang 'dihapus'. Berikut contohnya:

NamaFile:Array10.html

```
<script>
var array1 = [1,2,3,4,"a","b","c","d"];
document.write(array1.toString(),"<br>");
// hasil: 1,2,3,4,a,b,c,d

//splice dari index ke 2
var array2 = array1.splice(2)
document.write(array1.toString(),"<br>");
// hasil: 1,2
document.write(array2.toString(),"<br>");
// hasil: 3,4,a,b,c,d
</script>
```

Jika method **splice()** memiliki 2 argumen, maka argumen kedua berfungsi untuk menentukan seberapa banyak elemen yang akan di 'hapus'. Jika elemen yang dihapus berada ditengah-tengah array asal, maka array asal akan 'tersambung'. Berikut contohnya:

NamaFile:Array11.html

```
<script>
var arr1 = [1,2,3,4,"a","b","c","d"];
document.write(arr1.toString(),"<br>"); // hasil: 1,2,3,4,a,b,c,d
//hapus 3 elemen mulai dari index ke 3
var arr2 = arr1.splice(3,3);
document.write(arr1.toString(),"<br>"); // hasil: 1,2,3,c,d
```



```
document.write(arr2.toString(),"<br>"); // hasil: 4,a,b
</script>
```

Method `splice()` mendukung argumen ke 3, 4, dan seterusnya. Jika disertakan, argumen ini akan berfungsi sebagai elemen yang akan ditambahkan ke array asal dimulai dari posisi argumen pertama. Berikut adalah contoh kode program **splice()** dengan 3 atau lebih argumen:

NamaFile:Array12.html

```
<script>
var arr1 = [1,2,3,4,"a","b","c","d"];
document.write(arr1.toString(),"<br>");
// hasil: 1,2,3,4,a,b,c,d
// masukkan elemen "x","y","z"
// mulai dari elemen ke 4
var arr2 = arr1.splice(4,0,"x","y","z")
document.write(arr1.toString(),"<br>"); //hasil:1,2,3,4,x,y,z,a,b,c,d
document.write(arr2.toString(),"<br>");
// hasil: [] (array kosong)

var arr1 = [1,2,3,4,"a","b","c","d"];
document.write(arr1.toString(),"<br>");
// hasil: 1,2,3,4,a,b,c,d

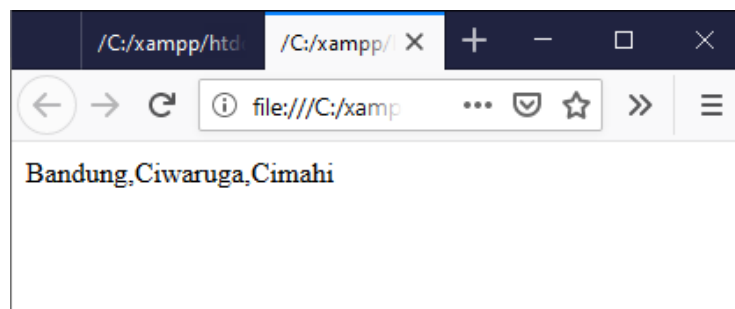
// hapus 4 elemen mulai dari elemen ke 4
// masukkan elemen "x","y","z" pada posisi penghapusan
var arr2 = arr1.splice(4,4,"x","y","z")
document.write(arr1.toString(),"<br>");
// hasil: 1,2,3,4,x,y,z
document.write(arr2.toString(),"<br>");
// hasil: a,b,c,d
</script>
```

4.2.7. Metode pop dan shift

Metode **pop** digunakan ketika kita ingin menghapus array pada elemen terakhir sedangkan metode **shift** digunakan ketika akan menghapus elemen array pada bagian pertama.

NamaFile:Arra13.html

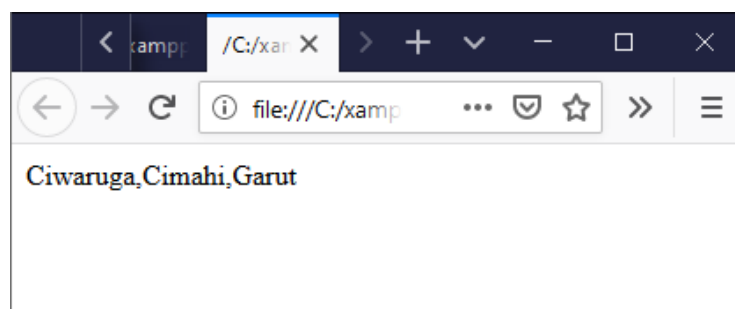
```
<script>
  var kota= ['Bandung', 'Ciwaruga', 'Cimahi','Garut'];
  kota.pop();
  document.write(kota.toString(),"<br>");
</script>
```



Gambar 4.9. Hasil Keluaran Menggunakan fungsi pop

NamaFile:Arra14.html

```
<script>
  var kota= ['Bandung', 'Ciwaruga', 'Cimahi','Garut'];
  kota.shift();
  document.write(kota.toString(),"<br>");
</script>
```



Gambar 4.10. Hasil Keluaran Penggunaan shift

4.2.8. Metode push dan unshift

Berikut ini akan dibahas mengenai penggunaan **metode push dan unshift**, Untuk **menambah data pada variabel array**

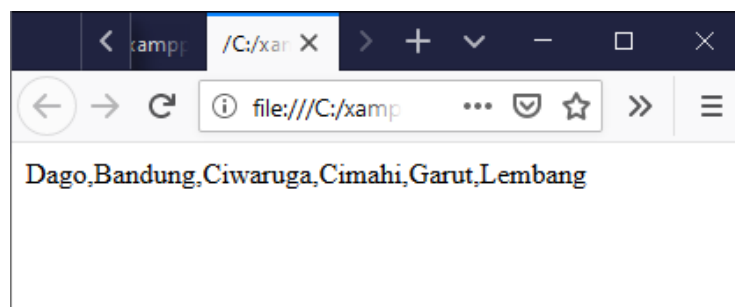
1. **push (some_value)**: merupakan fungsi yang digunakan untuk memasukkan value ke sebuah array dan disimpan di paling akhir / paling kanan.
2. **unshift (some_value)**: fungsi yang digunakan untuk memasukkan value ke sebuah array tetapi disimpan di paling awal / ujung kiri.

Contoh Metode push()

Misal akan ditambahkan Kota “**Lembang**” kedalam variabel array kota. Maka kode proramnya adalah sebagai berikut

NamaFile:Arra15.html

```
<script>
  var kota= ['Bandung', 'Ciwaruga', 'Cimahi','Garut'];
  kota.push("Lembang");
  document.write(kota.toString(),"<br>");
</script>
```



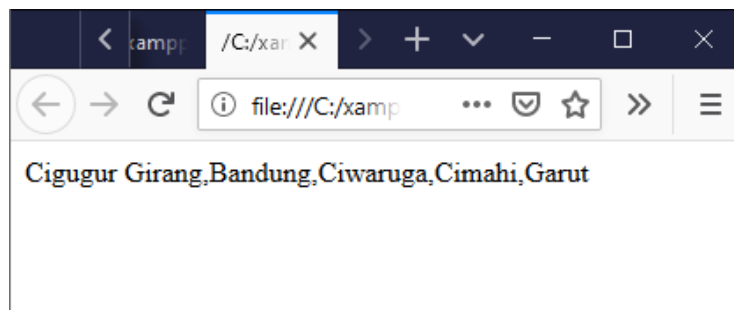
Gambar 4.11. Hasil setelah dilakukan Metode push

Metode unshift digunakan apabila ingin menambahkan data baru pada bagian awal array

Contoh metode unshift kode programnya adalah sebagai berikut:

NamaFile:Arra16.html

```
<script>
  var kota= ['Bandung', 'Ciwaruga', 'Cimahi', 'Garut'];
  kota.unshift("Cigugur Girang");
  document.write(kota.toString(),"<br>");
</script>
```



Gambar 4.12. Menggunakan perintah unshift

TUGAS

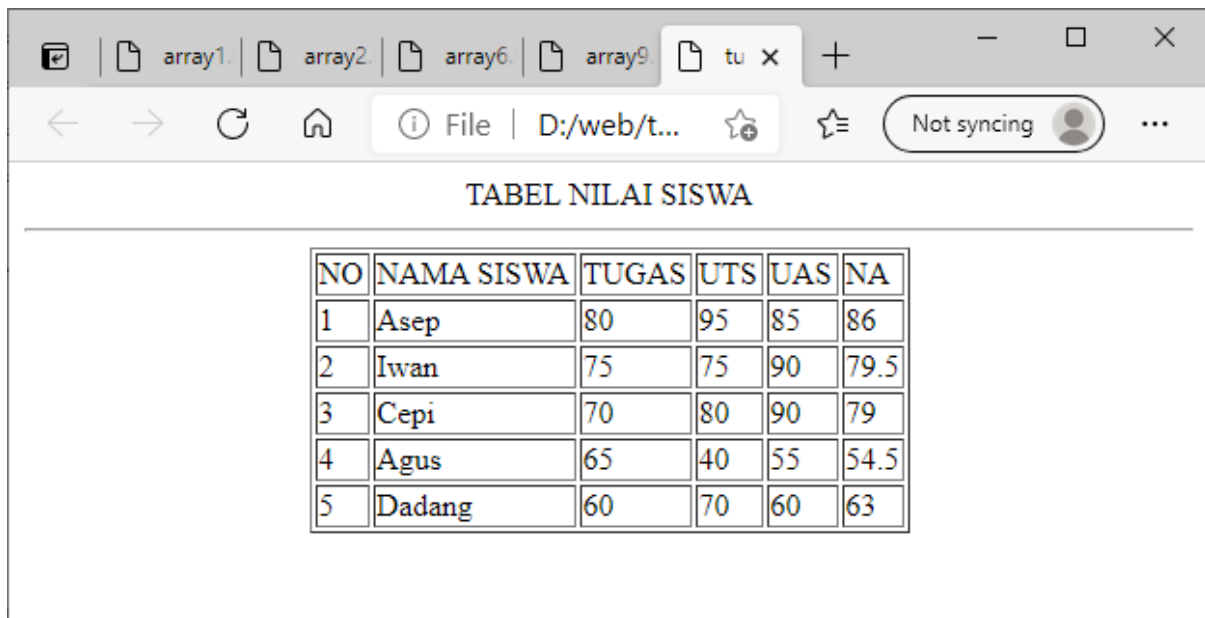
Tugas4.1.

Buat Program JavaScript Untuk Menampilkan, tabel berikut:

Diketahui array nama, tugas, uts,dan uas,

Cari Nilai Akhir (NA)

Rumus $na = 40\% \text{ tugas} + 30\% \text{ uts} + 30\% \text{ uas}$;



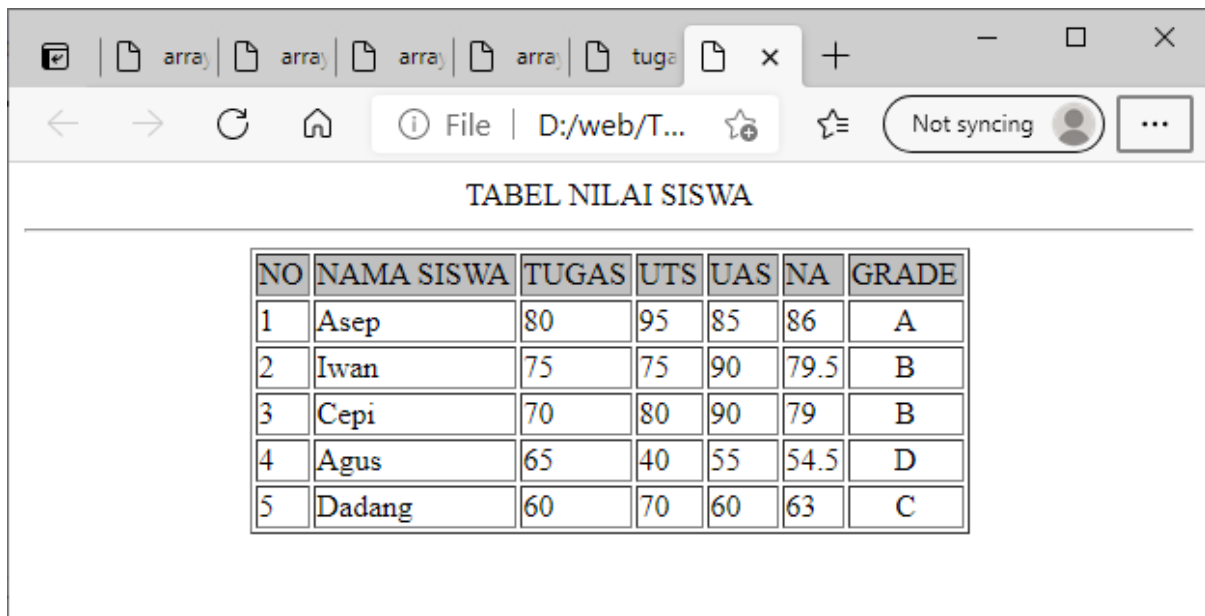
The screenshot shows a web browser window with several tabs open. The active tab is titled 'tu x'. The address bar shows the file path 'D:/web/t...'. The page content displays a table titled 'TABEL NILAI SISWA' with the following data:

NO	NAMA SISWA	TUGAS	UTS	UAS	NA
1	Asep	80	95	85	86
2	Iwan	75	75	90	79.5
3	Cepi	70	80	90	79
4	Agus	65	40	55	54.5
5	Dadang	60	70	60	63

Tugas4.2.

Gunakan Fungsi Logika IF, untuk menampilkan Grade dari Nilai Akhir, dengan ketentuan,

NILAI AKHIR	GRADE
0..49	E
50-59	D
60..69	C
70-79	B
80-100	A



The screenshot shows a web browser window with the title "TABEL NILAI SISWA". The browser's address bar shows "File | D:/web/T...". The table displayed in the browser is as follows:

NO	NAMA SISWA	TUGAS	UTS	UAS	NA	GRADE
1	Asep	80	95	85	86	A
2	Iwan	75	75	90	79.5	B
3	Cepi	70	80	90	79	B
4	Agus	65	40	55	54.5	D
5	Dadang	60	70	60	63	C

setelah semua program selesai dibuat dan hasilnya sesuai :

Dengan menggunakan Microsoft Word Copy Paste Kode program beserta tangkapan layar hasil keluarannya (screenshot), kemudian kirim ke Modul Tugas yang ada di kuliahonline.unikom.ac.id, **Nama Filenya: Tugas-bab4-nama-nim-kelas.pdf**

Dikumpulkan Paling Lambat Setiap Hari Minggu Jam 18:00

[Ok...selamat mencoba](#)