



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-4: Javascript (Tipe Data, Fungsi dan Operator)
Mata Kuliah Desain dan Pemrograman Web
Pengampu: Tim Ajar Desain dan Pemrograman Web
September 2020

Nama : Azaria Amanda

NIM : 244107060060

Kelas : SIB 2D

Github : https://github.com/azariaamanda/04-AzariaAmanda-SIB-2D/tree/main/Praktikum_PemrogramanWeb/Jobsheet3_Javascript

Topik

- Pengenalan konsep dasar Javascript
- Tipe Data, Operator, Fungsi di dalam Javascript
- Javascript didalam HTML

Tujuan

Mahasiswa diharapkan dapat:

1. Memahami konsep Javascript
2. Memahami Tipe data, operator dan fungsi di javascript
3. Mahasiswa mampu menjalankan javascript di file HTML

Pendahuluan

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti **Google Chrome dan Mozilla Firefox**. Bahasa pemrograman *Client Side* berbeda dengan bahasa pemrograman *Server Side* seperti PHP, dimana untuk *server side* seluruh kode program dijalankan di sisi server.

Untuk menjalankan **JavaScript**, kita hanya membutuhkan aplikasi *text editor* dan *web browser*. **JavaScript** memiliki fitur: *high-level programming language, client-side, loosely typed* dan berorientasi objek. **JavaScript** pada awal perkembangannya berfungsi untuk membuat

interaksi antara user dengan situs web menjadi lebih cepat tanpa harus menunggu pemrosesan di *web server*. Sebelum *javascript*, setiap interaksi dari user harus diproses oleh *web server*.

Bayangkan ketika kita mengisi *form registrasi* untuk pendaftaran sebuah situs web, lalu men-klik tombol *submit*, menunggu sekitar 20 detik untuk website memproses isian form tersebut, dan mendapati halaman yang menyatakan bahwa terdapat kolom form yang masih belum diisi.

Untuk keperluan seperti inilah **JavaScript** dikembangkan. Pemrosesan untuk mengecek apakah seluruh form telah terisi atau tidak, bisa dipindahkan dari *web server* ke dalam *web browser*.

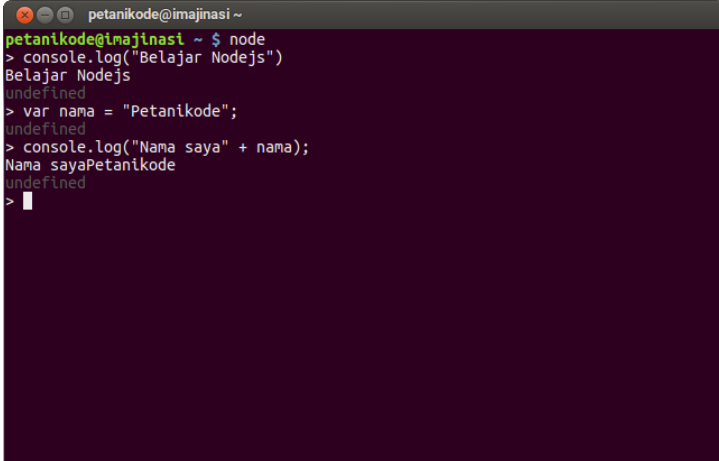
Dalam perkembangan selanjutnya, *JavaScript* tidak hanya berguna untuk *validasi form*, namun untuk berbagai keperluan yang lebih modern. Berbagai animasi untuk mempercantik halaman web, fitur chatting, efek-efek modern, games, semuanya bisa dibuat menggunakan *JavaScript*.

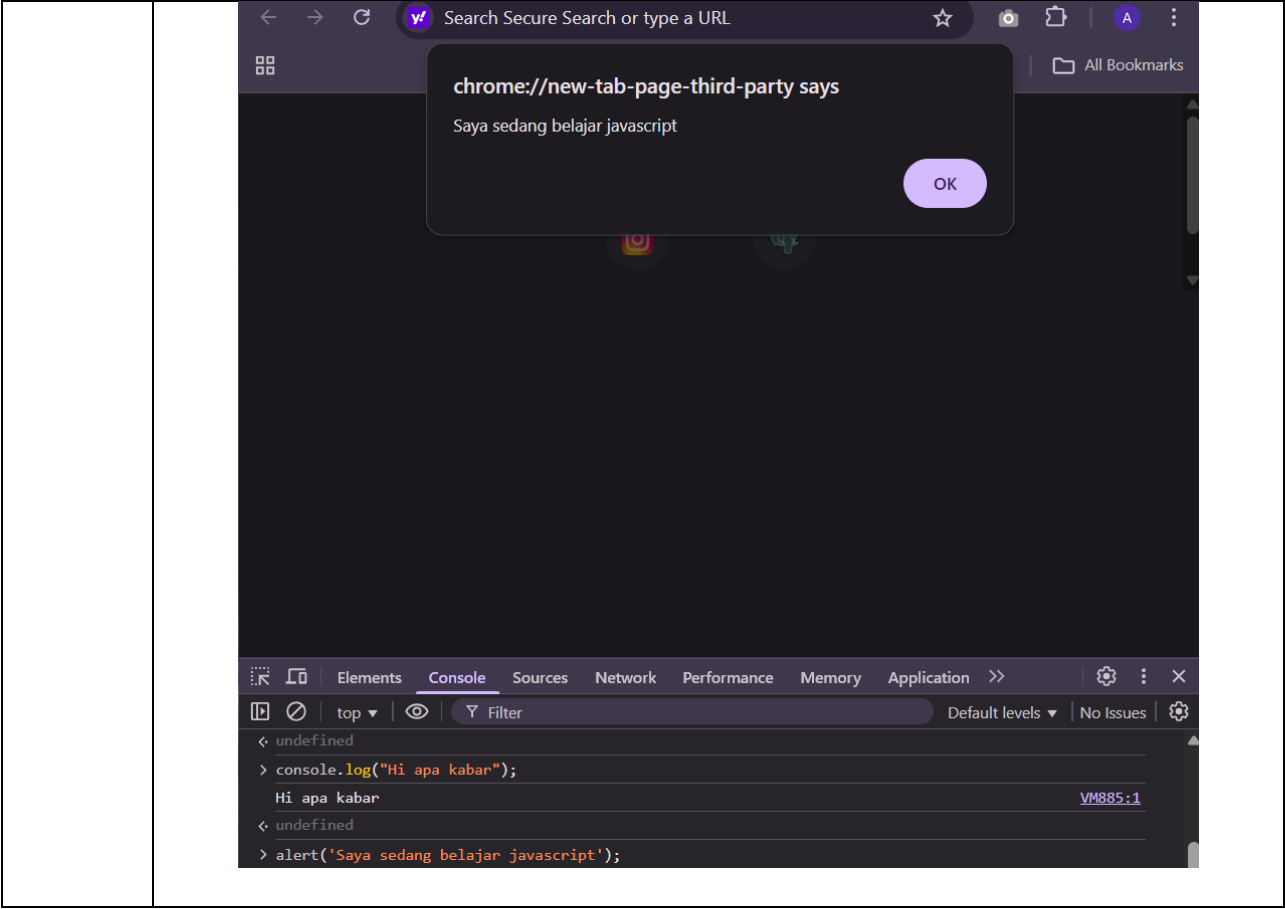
Ada 3 macam cara penulisan tag **javascript**, yaitu;

1. Menuliskan Tag dengan diawali `<script type="text/javascript">` dan diakhir dengan `</script>` atribut yang menginformasikan kepada browser bahwa program script yang ada dalam tag tersebut adalah **javascript** dalam format text.
2. Menuliskan Tag dengan diawali `<script language="javascript">` dan diakhir dengan `</script>` atribut ini digunakan dengan tujuan untuk menentukan versi dari javascript yang digunakan, sebagai contoh `<script language="javascript1.2">` menyatakan bahwa **javascript** yang kita gunakan adalah versi 2.1.
3. Menuliskan Tag dengan diawali `<script language="javascript" type="text/javascript">` dan diakhir dengan `</script>` cara campuran ini yaitu penulisan lama dan penulisan baru disatukan, dilakukan untuk mengantisipasi pengakses halaman web yang masih menggunakan browser web yang mendukung **javascript**, tetapi belum mendukung HTML

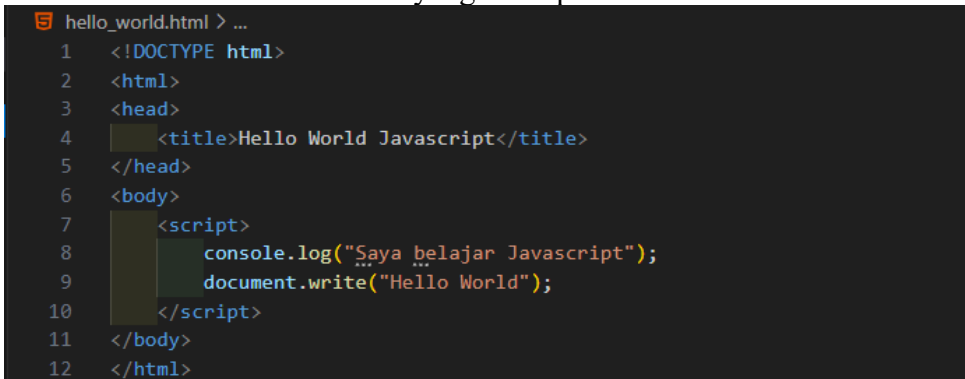
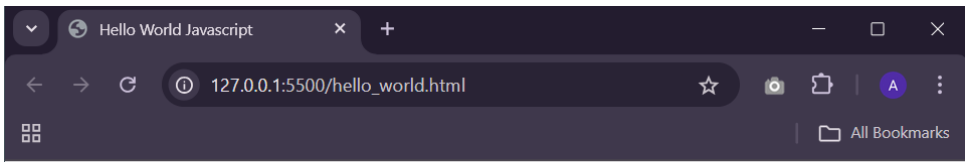
Praktikum Bagian 1. Belajar Javascript

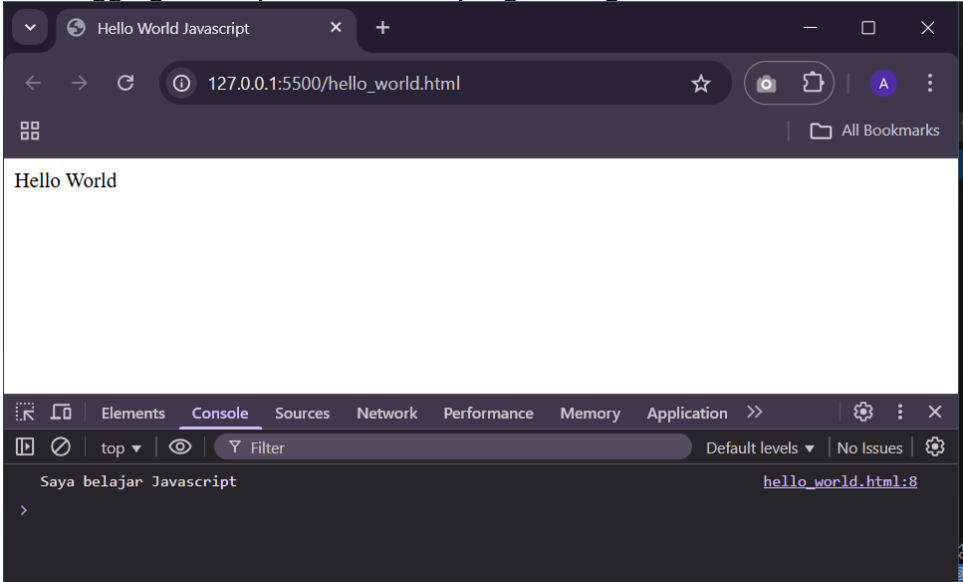
Langkah	Keterangan
1	<p>Console Javascript dapat kita buka melalui Inspect Element->Console.</p> 

	Di dalam console, kita bisa menulis fungsi atau kode-kode javascript dan hasilnya akan langsung ditampilkan.
3	<p>Misalnya, mari kita coba kode berikut :</p> <pre>console.log("Hi apa kabar!"); alert("Saya sedang belajar javascript");</pre> <p>Amati apa yang muncul pada console, kemudian catat hasil pengamatanmu!.</p>
2	<p>Jika kamu menggunakan Nodejs, maka cara mengakses <i>console</i> adalah dengan mengetik perintah node pada Terminal.</p> 
3	<p>Amati apa yang terjadi, kemudian catat hasil pengamatanmu. Apa yang dapat disimpulkan setelah mencoba <i>console</i> Javascript? (Soal No.1)</p> <ul style="list-style-type: none"> ➤ console.log() digunakan untuk mencetak informasi atau debugging langsung di panel konsol tanpa mengganggu tampilan halaman, sedangkan alert() digunakan untuk menampilkan pesan notifikasi yang interaktif kepada pengguna dalam bentuk pop-up yang memblokir.



Praktikum 2 : Membuat Program Javascript Pertama

Langkah	Keterangan
1	Silahkan buka text editor, kemudian buat file baru bernama hello_world.html
2	<p>Ketikkan kode program dibawah ini :</p> <pre><!DOCTYPE html> <html> <head> <title>Hello World Javascript</title> </head> <body> <script> console.log("Saya belajar Javascript"); document.write("Hello World!"); </script> </body> </html></pre>
3	Simpan dengan nama hello_world.html , kemudian buka file tersebut dengan web browser.
4	<p>Amati apa yang terjadi pada browser, kemudian catat hasil pengamatanmu (Soal No.2)</p> <ul style="list-style-type: none">➤ <code>console.log("Saya belajar Javascript");</code>: Perintah ini digunakan untuk mencetak pesan ke browser. Pesan ini tidak akan muncul di halaman web yang dilihat oleh pengguna.➤ <code>document.write("Hello World");</code>: Perintah ini digunakan untuk menuliskan teks "Hello World" langsung ke dalam dokumen HTML. Hasilnya, teks ini akan muncul di halaman web yang ditampilkan   <p>The first screenshot shows a code editor with the following code:</p> <pre>hello_world.html > ... 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Hello World Javascript</title> 5 </head> 6 <body> 7 <script> 8 console.log("Saya belajar Javascript"); 9 document.write("Hello World"); 10 </script> 11 </body> 12 </html></pre> <p>The second screenshot shows a web browser window with the title 'Hello World Javascript' and the address bar showing '127.0.0.1:5500/hello_world.html'. The page content displays 'Hello World'.</p>

5	Sekarang coba buka console javascript, kemudian lihat pada Inspect Elemets->Console
6	<p>Amati apa yang terjadi pada tab Console, kemudian catat hasil pengamatanmu! (Soal No. 3)</p> <ul style="list-style-type: none"> ➤ Ketika halaman web dimuat, tab Console di dalam Developer Tools menampilkan pesan "Saya belajar Javascript". Pesan ini dihasilkan oleh perintah <code>console.log("Saya belajar Javascript");</code> yang tertulis di dalam kode JavaScript. ➤ Kesimpulannya, fungsi <code>console.log()</code> tidak menampilkan output pada halaman web yang dilihat pengguna, melainkan mencatat pesan untuk keperluan debugging dan dapat dilihat oleh pengembang di tab Console.  <p>The screenshot shows a web browser window with the title 'Hello World Javascript' and the URL '127.0.0.1:5500/hello_world.html'. The page content displays 'Hello World'. The Chrome DevTools Console is open, showing a single log entry: 'Saya belajar Javascript' from 'hello_world.html:8'.</p>
7	<p>Tadi kita menuliskan perintah :</p> <pre>console.log("Saya belajar Javascript");</pre> <p>Menurut Anda, mengapa perintah tersebut tidak ditampilkan? (Soal No.4)</p> <ul style="list-style-type: none"> ➤ karena tujuannya bukan untuk menampilkan konten bagi pengguna, melainkan sebagai alat bagi pengembang. Perintah ini dirancang khusus untuk mencetak pesan, variabel, atau hasil eksekusi kode ke Console di dalam Developer Tools.

Praktikum 3 : Cara Menulis Kode Javascript di HTML

Pada praktikum 2 kita sudah menulis kode javascript di dalam HTML, cara tersebut merupakan cara penulisan embeded (ditempel). Beberapa cara lain yang perlu kita ketahui diantaranya :

1. *Embed* (Kode Javascript ditempel langsung pada HTML)
2. *Inline* (Kode Javascript ditulis pada atribut HTML)
3. *Eksternal* (Kode Javascript ditulis dengan terpisah dengan file HTML)

1. Penulisan Kode Javascript dengan Embed	
Langkah	Keterangan
1	Pada cara ini, kita menggunakan tag <script> untuk menempelkan (<i>embed</i>) kode Javascript pada HTML. Tag ini dapat ditulis didalam tag <head> dan <body>
2	<p>Ketikkan kode program dibawah ini :</p> <pre><!DOCTYPE html> <html> <head> <title>Belajar Javascript dari Nol</title> <script> // ini adalah penulisan kode javascript // di dalam tag <head> console.log("Hello JS dari Head"); </script> </head> <body> <p>Tutorial Javascript untuk Pemula</p> <script> // ini adalah penulisan kode javascript // di dalam tag <body> console.log("Hello JS dari body"); </script> </body> </html></pre>
3	<p>Amati apa yang terjadi pada browser? Catat hasil pengamatanmu (Soal No.1)</p> <ul style="list-style-type: none">➤ Ketika halaman dimuat, output di konsol menunjukkan bahwa pesan "Hello JS dari head" dicetak terlebih dahulu, diikuti oleh "Hello JS dari body". Ini membuktikan bahwa skrip yang ada di <head> dieksekusi sebelum skrip di <body>.

The screenshot displays a web browser window with the title "Belajar Javascript dari Nol". The address bar shows the URL "127.0.0.1:5500/hello_world.html". The page content is "Tutorial Javascript untuk Pemula". The browser's developer console is open, showing two log messages: "Hello JS dari head" at line 8 of "hello_world.html" and "Hello JS dari body" at line 16 of "hello_world.html".

```
hello_world.html M X
hello_world.html > html
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Belajar Javascript dari Nol</title>
5     <script>
6       // ini adalah penulisan kode javascript
7       // di dalam tag <head>
8       console.log("Hello JS dari head");
9     </script>
10  </head>
11  <body>
12    <p>Tutorial Javascript untuk Pemula</p>
13    <script>
14      // ini adalah penulisan kode javascript
15      // di dalam tag <body>
16      console.log("Hello JS dari body");
17    </script>
18  </body>
19 </html>
```

Belajar Javascript dari Nol

127.0.0.1:5500/hello_world.html

Tutorial Javascript untuk Pemula

Elements Console Sources Network Performance Memory >>

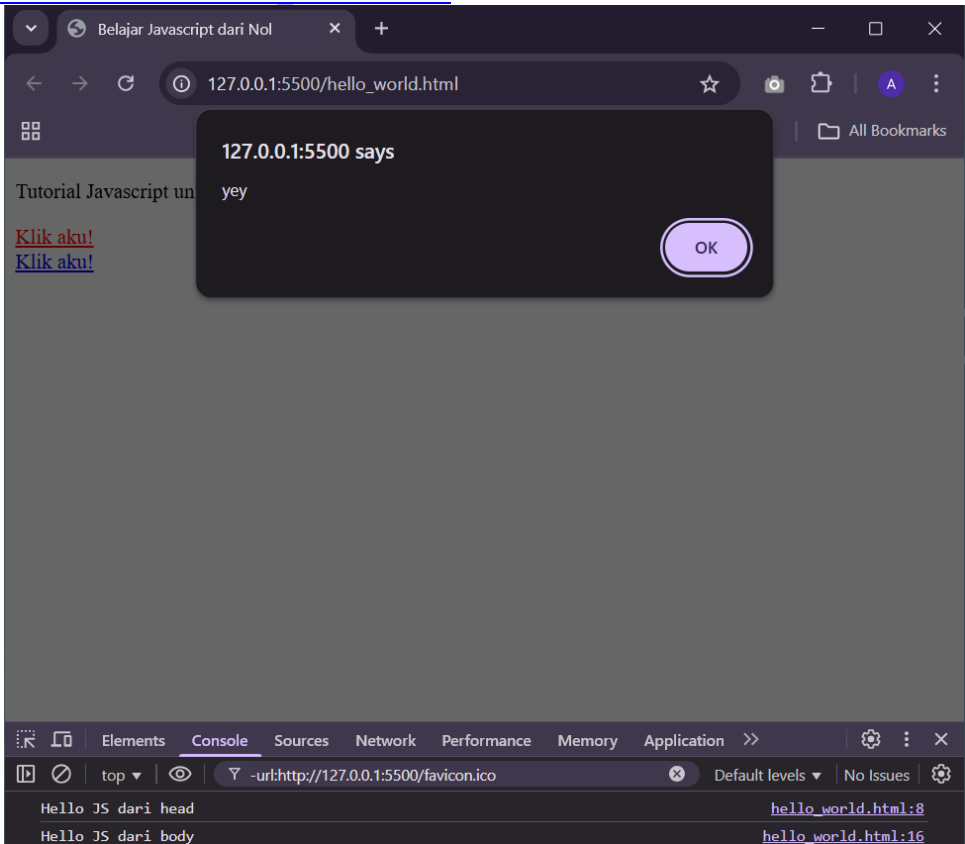
top -url:http://127.0.0.1:5500/favicon.ico Default levels No Issues

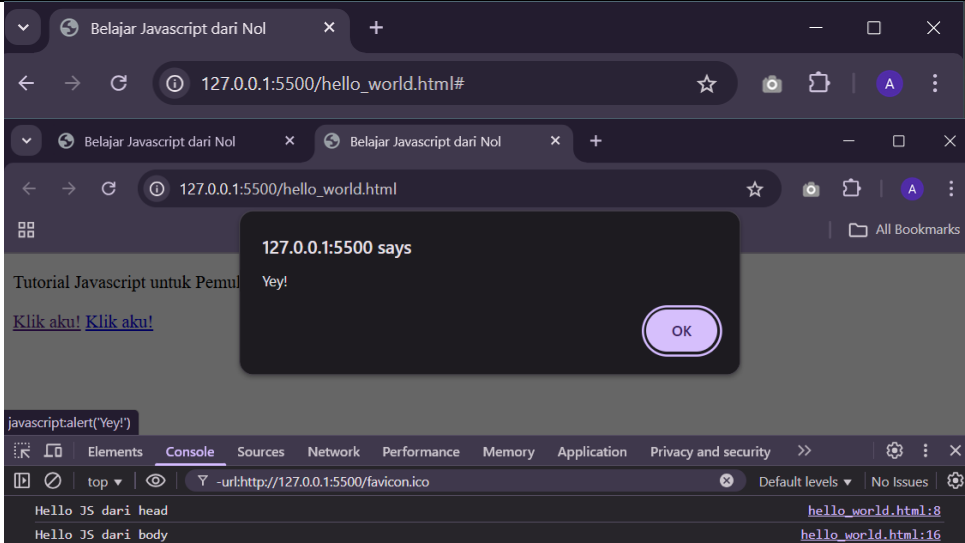
Hello JS dari head hello_world.html:8

Hello JS dari body hello_world.html:16

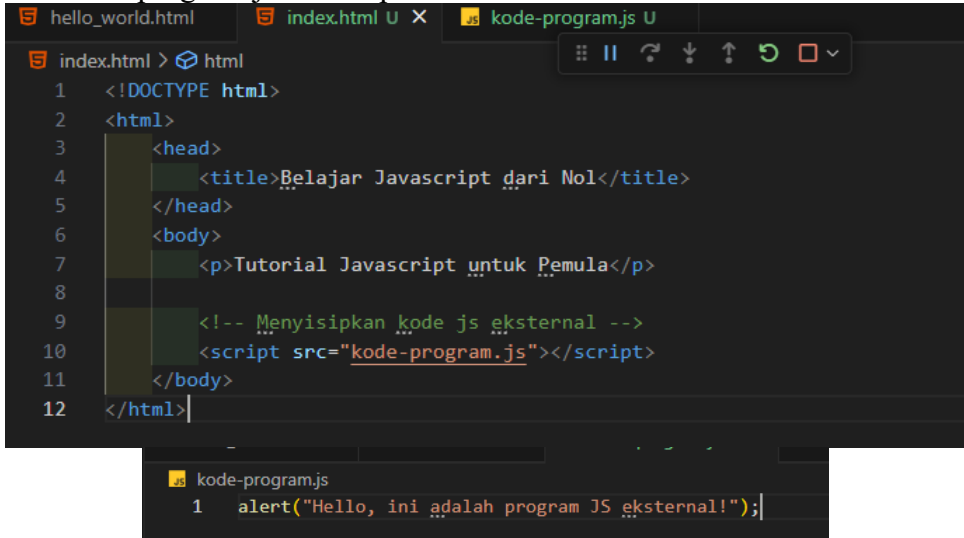
- 4 Menurut Anda manakah yang lebih baik, dituliskan didalam tag **<head>** atau **<body>**? (Soal No. 2)
- Tergantung keadaan, jika konfigurasi awal bisa menggunakan head, jika interaksi langsung dengan html bisa gunakan body.

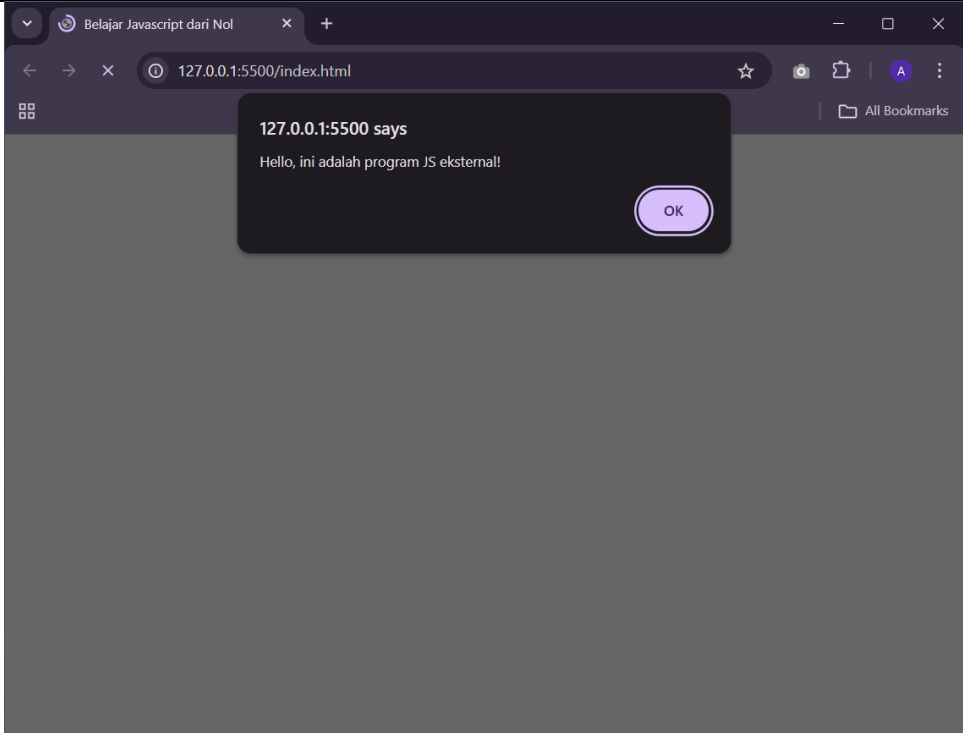
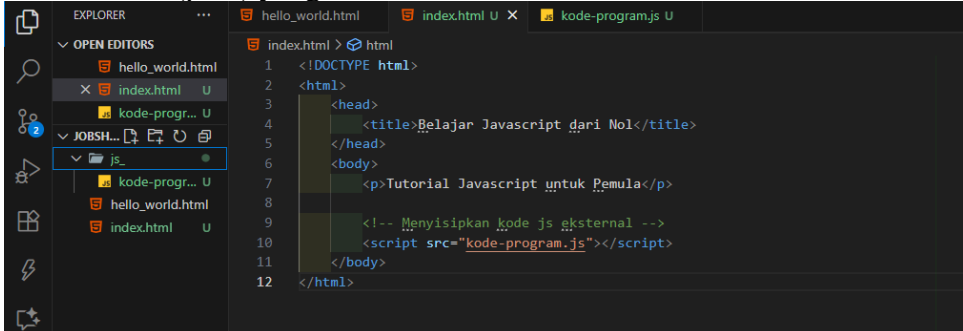
2. Penulisan Kode Javascript *Inline*

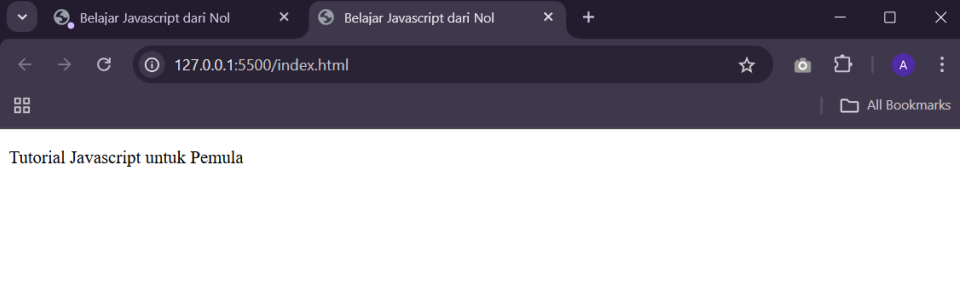
Langkah	Keterangan
1	Pada cara ini, kita akan menulis kode javascript di dalam atribut HTML. Cara ini biasanya digunakan untuk memanggil suatu fungsi pada event tertentu. Salah satu contohnya pada saat di-klik.
2	<p>Ketikkan kode program dibawah ini :</p> <pre>Klik aku!</pre> <p>Atau bisa juga seperti ini :</p> <pre>Klik aku!</pre>
3	<p>Amati apa yang terjadi pada browser! Catat hasil pengamatanmu (Soal No.1)</p> <ul style="list-style-type: none">➤ Saat link bertuliskan “Klik aku!” diklik, browser langsung menjalankan perintah <code>alert('Yey!')</code> dan menampilkan sebuah jendela pop-up (alert box) dengan pesan “Yey!” serta tombol “OK”. Setelah tombol “OK” ditekan, pop-up akan tertutup dan halaman tetap berada di posisi semula. <p>http://127.0.0.1:5500/hello_world.html#</p> 

	
4	<p>Apa yang membedakan dari kedua kode program tersebut (Soal No. 2)</p> <ul style="list-style-type: none"> ➤ Perbedaan dari kedua kode program tersebut ada pada cara eksekusinya dan efek terhadap URL di browser. Pada kode pertama, link menggunakan href="#" dan perintah JavaScript ditaruh di atribut onclick. Saat link diklik, JavaScript alert("Yey!") berjalan, tetapi setelah itu URL halaman akan berubah dengan menambahkan tanda pagar # di akhir alamat (misalnya dari hello_world.html menjadi hello_world.html#), sedangkan pada kode kedua, link langsung menggunakan href="javascript:alert('Yey!')" sehingga saat diklik, browser hanya mengeksekusi perintah JavaScript tersebut tanpa mengubah URL halaman

3. Penulisan Kode Javascript <i>Eksternal</i>	
Langkah	Keterangan
1	Pada cara ini, kita akan menulis kode javascript secara terpisah dengan file HTML. Cara ini biasanya digunakan pada proyek-proyek besar, karena diyakini dengan cara ini dapat lebih mudah mengelola kode project.
2	<p>Mari kita coba, buat dua file yaitu file HTML dan Javascript.</p> <pre> belajar-js/ ├── kode-program.js └── index.html </pre>
3	<p>Isi dari file kode-program.js :</p> <pre> alert("Hello, ini adalah program JS eksternal!"); </pre>

4	<p>Isi dari file <code>index.html</code> :</p> <pre> <!DOCTYPE html> <html> <head> <title>Belajar Javascript dari Nol</title> </head> <body> <p>Tutorial Javascript untuk Pemula</p> <!-- Menyisipkan kode js eksternal --> <script src="kode-program.js"></script> </body> </html> </pre>
5	<p>Amati apa yang terjadi pada browser! Catat hasil pengamatanmu (Soal No.1)</p> <ul style="list-style-type: none"> ➤ langsung muncul sebuah jendela pop-up (alert) yang bertuliskan “Hello, ini adalah program JS eksternal!”. Ini menunjukkan bahwa file <code>kode-program.js</code> berhasil dijalankan karena sudah dipanggil di dalam <code>index.html</code> menggunakan tag <code><script src="kode-program.js"></script></code>. 

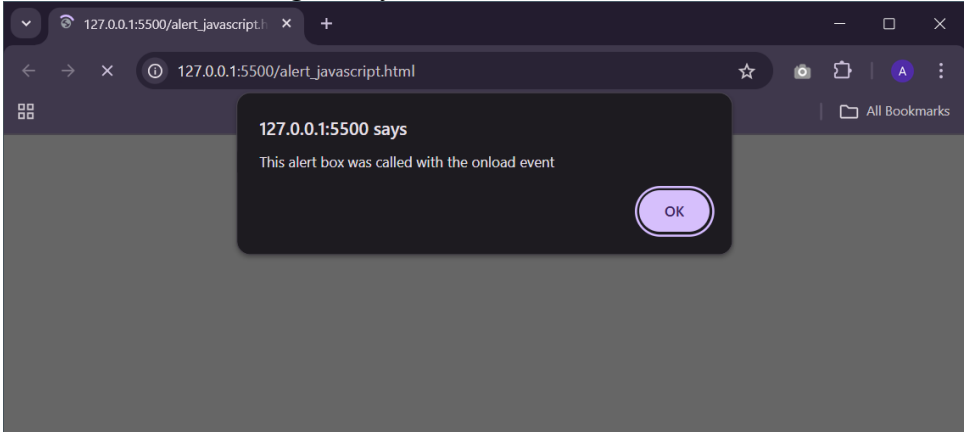
	
6	<p>Pada percobaan tersebut, kita menulis kode javascript terpisah dengan kode HTML. Lalu pada kode HTML kita menyisipkan atribut src pada tag <script></p> <pre data-bbox="355 1035 1386 1155"><!-- Menyisipkan kode js eksternal --> <script src="kode-program.js"></script></pre> <p>Maka apapun yang ada di dalam file kode-program.js akan dapat dibaca dari file index.html</p>
7	<p>Apa yang akan terjadi jika file javascript berada di folder yang berbeda? Amati dan catat hasil pengamatanmu (Soal No.2)</p> <ul style="list-style-type: none"> ➤ Jika file kode-program.js dipindahkan ke folder yang berbeda tanpa mengubah path pada atribut src, maka saat file index.html dibuka di browser, tidak muncul pop-up sama sekali. Ini karena browser tidak dapat menemukan file JavaScript tersebut akibat alamat (path) yang salah 

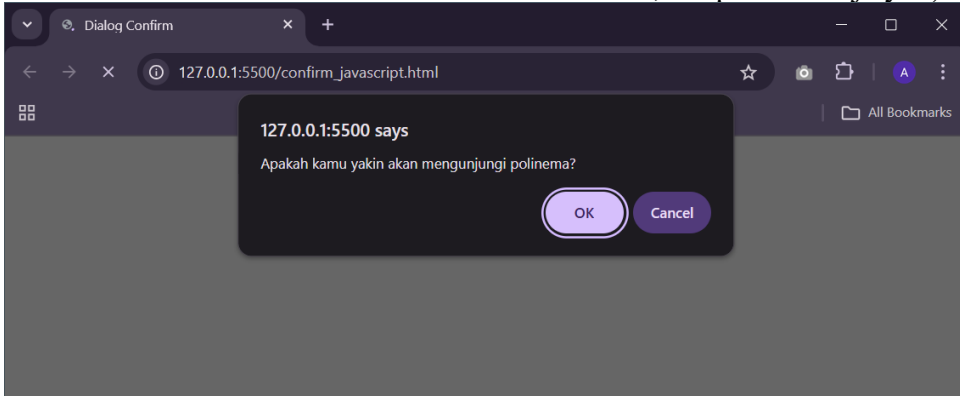
	
8	<p>Misalkan kita memiliki struktur folder seperti ini :</p> <pre> belajar-js/ ├── js/ │ └── kode-program.js └── index.html </pre> <p>Maka untuk menyisipkan file kode-program.js ke dalam HTML, kita bisa menuliskan kode berikut :</p> <pre> <script src="js/kode-program.js"></script> </pre> <p>Karena file kode-program.js berada didalam direktori js. Kita juga dapat menyisipkan javascript yang ada di internet dengan memberikan alamat URL lengkap. Contoh :</p> <pre> <script src="https://www.petanikode.com/js/kode.js"></script> </pre>

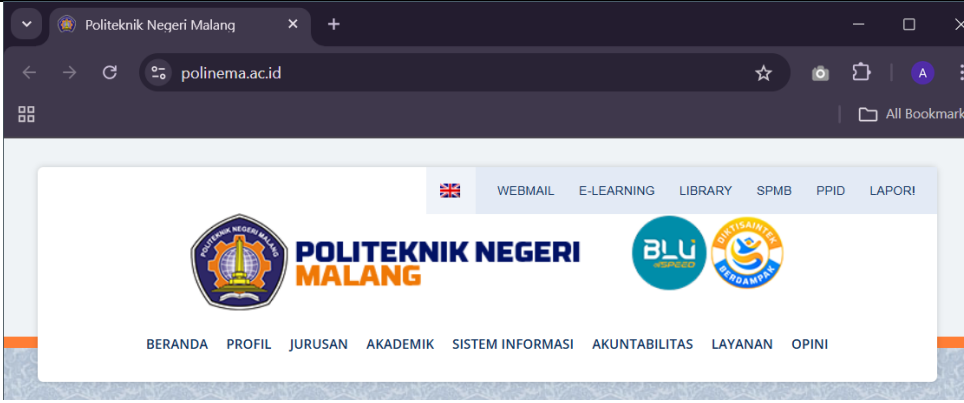

Praktikum 4: Jendela Dialog

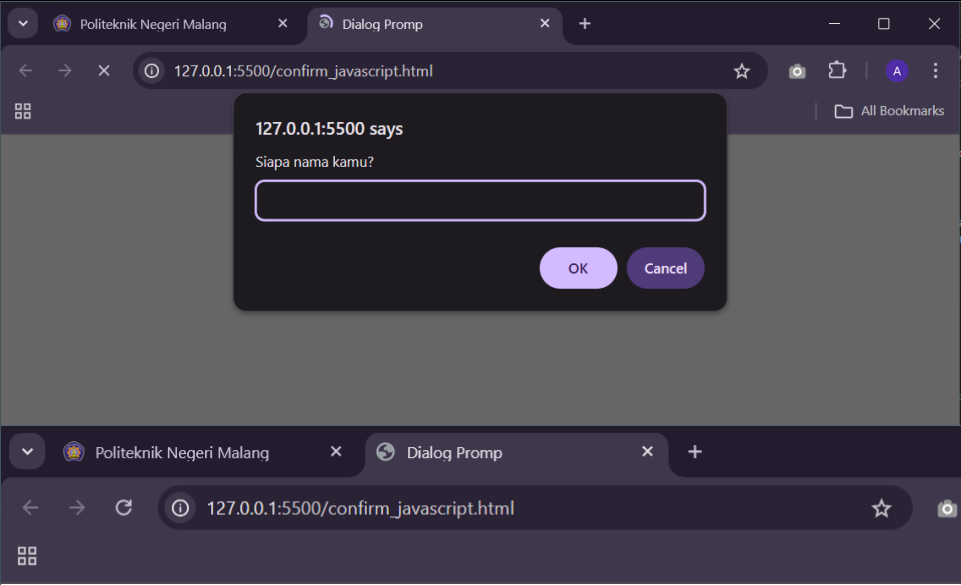
Jendela dialog merupakan jendela yang digunakan untuk berinteraksi dengan pengguna. Ada tiga macam jendela dialog pada Javascript:

1. Jendela dialog alert();
2. Jendela dialog confirm();
3. Jendela dialog prompt();

Langkah	Keterangan
1	Buat File baru bernama alert_javascript.html dan simpan dalam direktori belajarjavascript di folder localhost
2	Ketikkan kode program dibawah ini
	<pre><html> <head> <script type="text/javascript"> function message() { alert("This alert box was called with the onload event") } </script> </head> <body onload="message()"> </body> </html></pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.3)</p> <ul style="list-style-type: none">➤ angung muncul jendela pop-up (alert) bertuliskan “This alert box was called with the onload event” begitu halaman selesai dimuat. Pop-up ini muncul otomatis tanpa perlu klik apa pun karena fungsi message() dipanggil melalui atribut onload di tag <body> 

5	Buat file baru bernama confirm_javascript.html dan simpan dalam direktori belajarjavascript
6	<p>Ketikkan kode program dibawah ini</p> <pre> <!DOCTYPE html> <html> <head> <title>Dialog Confirm</title> </head> <body> <script> var yakin = confirm("Apakah kamu yakin akan mengunjungi polinema?"); if (yakin) { window.location = "https://www.polinema.ac.id"; } else { document.write("Baiklah, tetap di sini saja ya :)"); } </script> </body> </html> </pre>
7	Amati apa yang muncul pada browser
8	<p>Catat hasil pengamatanmu (Soal No.4)</p> <ul style="list-style-type: none"> ➤ Ketika program dijalankan, pada browser muncul kotak dialog konfirmasi dengan pesan “Apakah kamu yakin akan mengunjungi polinema?”. Jika pengguna menekan tombol OK, maka browser langsung diarahkan ke halaman web Polinema. Namun, jika pengguna menekan tombol Cancel, maka di halaman akan muncul tulisan “Baiklah, tetap di sini saja ya :)”. 

	
9	<p>Ketikkan kode program dibawah ini</p> <pre> <!DOCTYPE html> <html> <head> <title>Dialog Promp</title> </head> <body> <script> var nama = prompt("Siapa nama kamu?", ""); document.write("<p>Hello " + nama + "</p>"); </script> </body> </html> </pre>
10	<p>Amati apa yang muncul pada bwoser</p>
11	<p>Catat hasil pengamatanmu (Soal No.5)</p> <ul style="list-style-type: none"> ➤ Saat program dijalankan, muncul kotak dialog prompt dengan pesan “Siapa nama kamu?” dan sebuah kolom input. Setelah saya mengetikkan nama, misalnya Azaria, lalu menekan tombol OK, halaman menampilkan tulisan “Hello Azaria”. Ini menunjukkan bahwa nilai dari input prompt berhasil disimpan dan ditampilkan di browser. 

	 <p>127.0.0.1:5500 says</p> <p>Siapa nama kamu?</p> <p>OK Cancel</p> <p>Politeknik Negeri Malang Dialog Promp</p> <p>127.0.0.1:5500/confirm_javascript.html</p>
	<p>Hello Azaria</p>

Praktikum 5: Variabel

Cara membuat variabel yang umum digunakan di javascript adalah menggunakan kata kunci var lalu diikuti dengan nama variabel dan nilainya.

Contoh: var title = "Belajar Pemrograman Javascript";

Menampilkan isi Variabel

Untuk menampilkan isi variabel, kita bisa memanfaatkan fungsi-fungsi untuk menampilkan output seperti:

Fungsi console.log() menampilkan output ke console javascript;

Fungsi document.write() menampilkan output ke dokumen HTML;

dan Fungsi alert() menampilkan output ke jendela dialog.

Langkah	Keterangan
1	Buat File baru bernama variabel_javascript.html dan simpan dalam direktori belajarjavascript di folder localhost
2	Ketikkan kode program dibawah ini
	<pre><!DOCTYPE html> <html lang="en"> <head> <title>Belajar Variabel dalam Javascript</title> <script> // membuat variabel var name = "Javascript"; var visitorCount = 50322; var isActive = true; // menampilkan variabel ke jendela dialog (alert) alert("Selamat datang di " + name); // menampilkan variabel ke dalam HTML document.write("Nama Situs: " + name + "
"); document.write("Jumlah Pengunjung: " + visitorCount + "
"); document.write("Status Aktif: " + isActive + "
"); </script> </head> <body> </body> </html></pre>
3	Amati apa yang muncul pada browser
4	Catat hasil pengamatanmu (Soal No.6) <ul style="list-style-type: none">➤ Program menampilkan pesan awal berupa alert “Selamat datang di Javascript”. Setelah itu, pada halaman muncul keterangan dari variabel yang sudah dibuat, yaitu: Nama Situs: Javascript, Jumlah Pengunjung: 50322, dan Status Aktif: true. Hal ini membuktikan bahwa variabel dapat menyimpan berbagai tipe data seperti string, angka, maupun boolean, lalu ditampilkan ke browser.

	<div><div>variabel_javascript.html > html > body</div><pre>1 <!DOCTYPE html> 2 <html lang="en"> 3 <head> 4 <title>Belajar Variabel dalam Javascript</title> 5 <script> 6 // membuat variabel 7 var name = "Javascript"; 8 var visitorCount = 50322; 9 var isActive = true; 10 11 // menampilkan variabel ke jendela dialog (alert) 12 alert("Selamat datang di " + name); 13 14 // menampilkan variabel ke dalam HTML 15 document.write("Nama Situs: " + name + "
"); 16 document.write("Jumlah Pengunjung: " + visitorCount + "
"); 17 document.write("Status Aktif: " + isActive + "
"); 18 </script> 19 </head> 20 <body> 21 </body> 22 </html></pre><div>Belajar Variabel dalam Javascript</div><div>127.0.0.1:5500/variabel_javascript.html</div><div>Selamat datang di Javascript</div><div>OK</div><div>127.0.0.1:5500/variabel_javascript.html</div><div>Nama Situs: Javascript Jumlah Pengunjung: 50322 Status Aktif: true</div></div>
--	---

🚧 Menghapus Variabel

Penghapusan variabel dalam Javascript memang jarang dilakukan. Namun, untuk program yang membutuhkan ketelitian dalam alokasi memori, penghapusan variabel perlu dilakukan agar penggunaan memori lebih optimal.

Penghapusan variabel dapat dilakukan dengan katakunci delete.

Contoh:

```
bookTitle = "Belajar Pemrograman Javascript";
```

```
delete bookTitle;
```

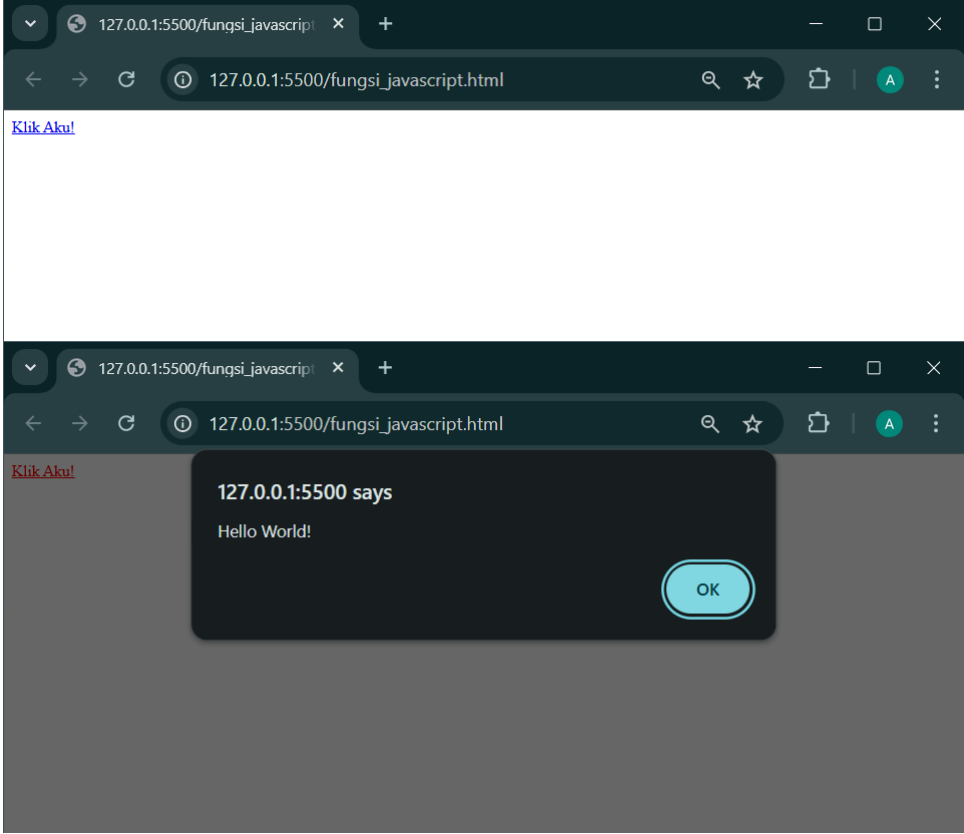
Maka variabel bookTitle akan menghilang dari memori.

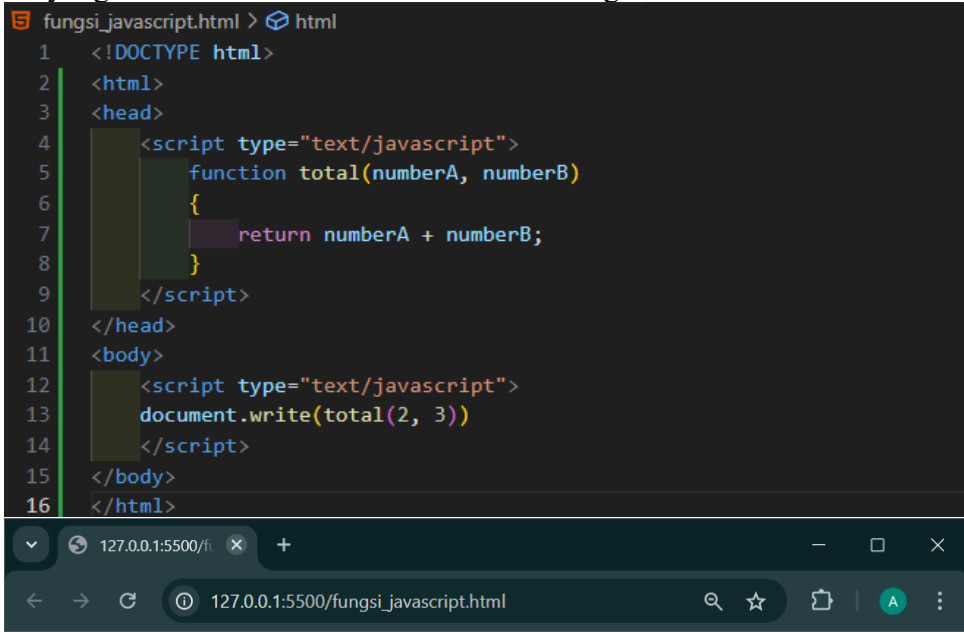
Praktikum 6 : FUNGSI

Fungsi adalah sub-program yang bisa digunakan kembali baik di dalam program itu sendiri, maupun di program yang lain.

Fungsi di dalam Javascript adalah sebuah objek. Karena memiliki properti dan juga *method*.

Langkah	Keterangan
1	Buat File baru bernama fungsi_javascript1.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	Cara memanggil fungsi didalam kode Javascript biasanya ditulis dengan : namaFungsi();
3	Ketikkan kode program berikut ini
	<pre><!DOCTYPE html> <html> <head> <script> // membuat fungsi var sayHello = () => alert("Hello World!"); </script> </head> <body> <!-- Memanggil fungsi saat link diklik --> Klik Aku! </body> </html></pre>
4	Amati apa yang muncul pada browser
5	Catat hasil pengamatanmu (Soal No.7) ➤ Saat link Klik Aku! ditekan, muncul kotak dialog alert dengan pesan “Hello World!”. Ini membuktikan bahwa fungsi dapat dipanggil ketika terjadi sebuah event, dalam hal ini event klik pada link.

	
6	<p>Parameter adalah variable yang menyimpan nilai untuk proses di dalam fungsi Cara memanggil parameter dalam javascript adalah :</p> <pre>function kali(a, b){ hasilKali = a * b; console.log("Hasil kali a*b = " + hasilKali); }</pre>
7	<p>Ketikkan kode program berikut ini</p>

	<pre> <html> <head> <script type="text/javascript"> function total(numberA,numberB) { return numberA + numberB } </script> </head> <body> <script type="text/javascript"> document.write(total(2,3)) </script> </body> </html> </pre>
8	Amati apa yang muncul pada browser
9	<p>Catat hasil pengamatanmu (Soal No.8)</p> <ul style="list-style-type: none"> ➤ Program memanggil fungsi total(2, 3) yang menjumlahkan dua angka. Hasil yang muncul di halaman browser adalah angka 5 dari numberA+numberB  <p>The screenshot shows a code editor with the following HTML and JavaScript code:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <script type="text/javascript"> 5 function total(numberA, numberB) 6 { 7 return numberA + numberB; 8 } 9 </script> 10 </head> 11 <body> 12 <script type="text/javascript"> 13 document.write(total(2, 3)) 14 </script> 15 </body> 16 </html> </pre> <p>Below the code editor, a web browser window is shown with the address bar displaying '127.0.0.1:5500/fungsi_javascript.html'. The browser's content area displays the number '5'.</p>

Praktikum 7: Tipe Data



Tipe data adalah jenis-jenis data yang bisa kita simpan di dalam variabel. Ada beberapa tipe data dalam pemrograman Javascript:

- String (teks)
- Integer atau Number (bilangan bulat)
- Float (bilangan Pecahan)
- Boolean
- Object

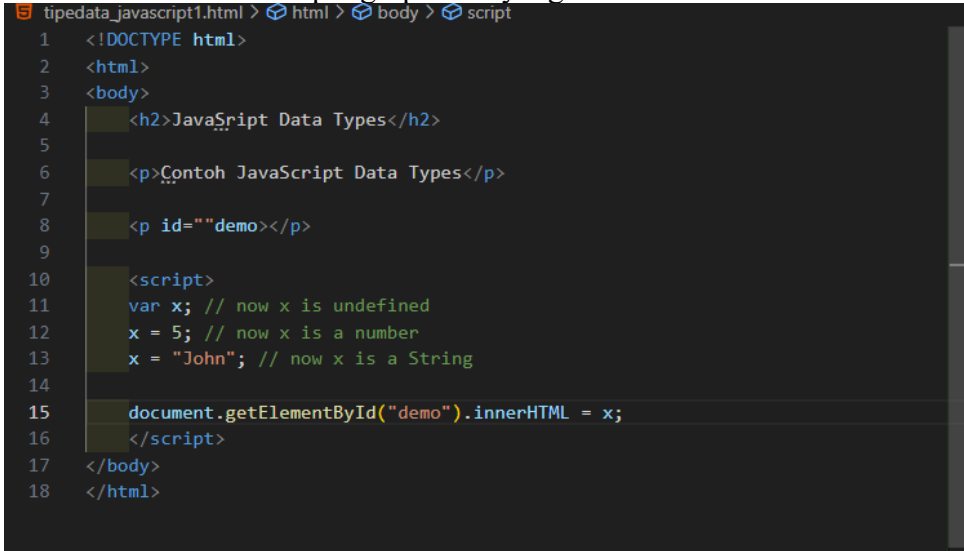
Javascript adalah bahasa yang bersifat *dynamic typing*, artinya kita tidak harus menuliskan tipe data pada saat pembuatan variabel seperti pada bahasa [C](#), [C++](#), [Java](#), dsb. yang bersifat *static typing*.

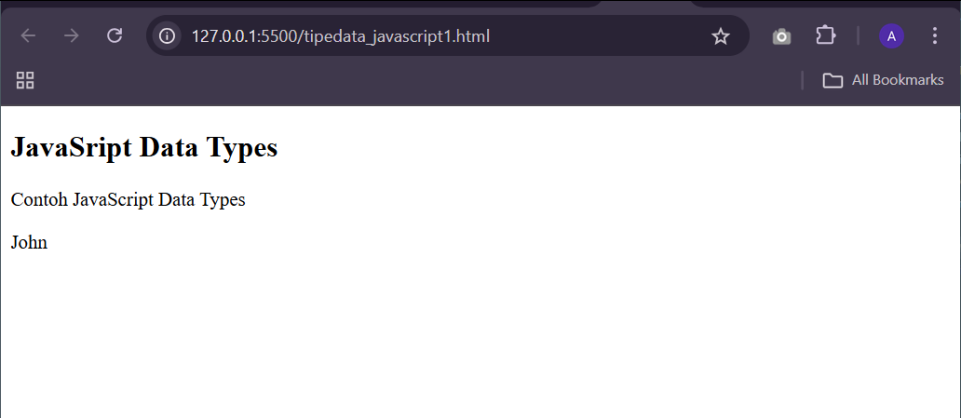
Ada beberapa aturan penulisan variabel dalam Javascript:

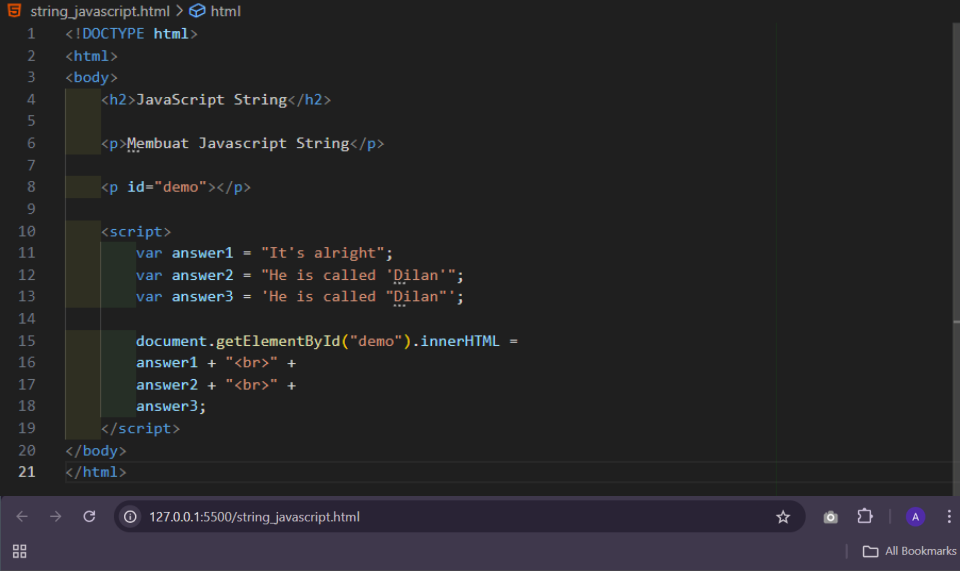
- Penamaan variabel **tidak boleh** menggunakan angka di depannya.
contoh:
`// salah`
`var 123nama = "Polinema";`

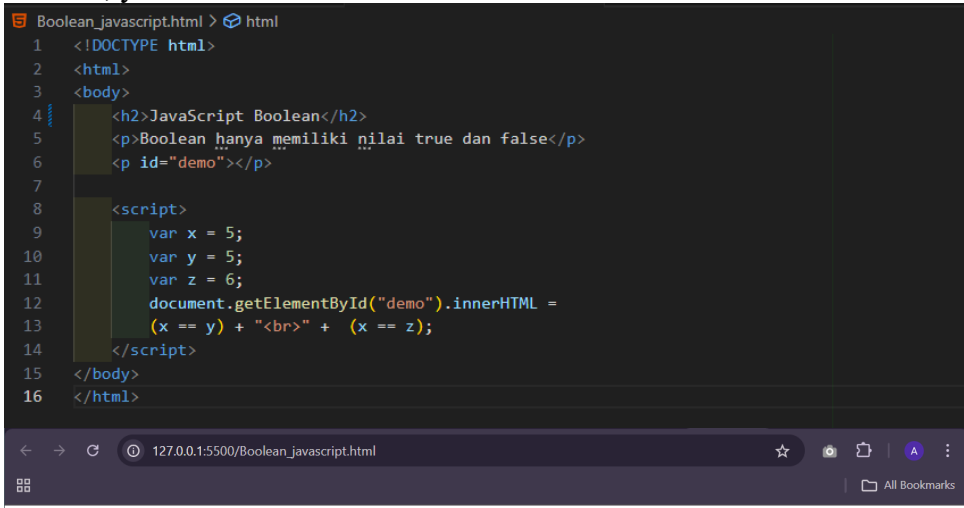
`// benar`
`var nama123 = "Polinema";`
- Penamaan variabel **boleh** menggunakan awal underscore.
contoh:
`var _nama = "Polinema";`
- Penamaan variabel **dianjurkan** menggunakan camelCase apabila terdiri dari dua suku kata.
Contoh:
`var fullName = "Polinema";`
- Penamaan variabel **dianjurkan** menggunakan bahasa inggris
Contoh:
`var postTitle = "Tutorial Javascript untuk Pemula";`

Langkah	Keterangan
1	Buat File baru bernama <code>tipedata_javascript1.html</code> dan simpan dalam direktori <code>belajarjavascript</code> .di folder <code>localhost</code>
2	Ketikkan kode program berikut ini

	<pre> <!DOCTYPE html> <html> <body> <h2>JavaScript Data Types</h2> <p>Contoh Javascript Data Types</p> <p id="demo"></p> <script> var x; // Now x is undefined x = 5; // Now x is a Number x = "John"; // Now x is a String document.getElementById("demo").innerHTML = x; </script> </body> </html> </pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.9)</p> <ul style="list-style-type: none"> ➤ Pada program tipe data, variabel x awalnya tidak memiliki nilai, lalu diisi angka, dan akhirnya diganti menjadi string. Di browser tampil teks “John”. Dari sini terlihat bahwa JavaScript bersifat dynamic typing, artinya sebuah variabel bisa menampung tipe data yang berbeda. 

	 <p>The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5500/tipedata_javascript1.html'. The page title is 'JavaScript Data Types'. Below the title, the text 'Contoh JavaScript Data Types' is followed by the name 'John'.</p>
5	<p>Ketikkan program dibawah ini dan simpan dengan nama string_javascript.html</p> <pre> <!DOCTYPE html> <html> <body> <h2>JavaScript Strings</h2> <p>Membuat Javascript String</p> <p id="demo"></p> <script> var answer1 = "It's alright"; var answer2 = "He is called 'Dilan'"; var answer3 = 'He is called "Dilan"'; document.getElementById("demo").innerHTML = answer1 + "
" + answer2 + "
" + answer3; </script> </body> </html> </pre>
6	Amati apa yang muncul pada browser
7	<p>Catat hasil pengamatanmu (Soal No.10)</p> <ul style="list-style-type: none"> ➤ Program string menampilkan beberapa kalimat, yaitu “It’s alright”, “He is called ‘Dilan’”, dan “He is called ‘Dilan’”. Di browser hasilnya ditampilkan per baris.

	 <p>JavaScript String</p> <p>Membuat Javascript String</p> <p>It's alright He is called 'Dilan' He is called "Dilan"</p>
8	<p>Ketikkan program dibawah ini dan simpan dengan nama Boolean_javascript.html</p> <pre> <!DOCTYPE html> <html> <body> <h2>JavaScript Booleans</h2> <p>Booleans hanya memiliki nilai true dan false</p> <p id="demo"></p> <script> var x = 5; var y = 5; var z = 6; document.getElementById("demo").innerHTML = (x == y) + "
" + (x == z); </script> </body> </html> </pre>
9	Amati apa yang muncul pada browser
10	Catat hasil pengamatanmu (Soal No.11)

	<p>➤ Pada program boolean, hasil yang muncul adalah “true” dan “false”. Nilai true muncul karena x sama dengan y, sedangkan nilai false muncul karena x tidak sama dengan z. Ini membuktikan bahwa boolean hanya memiliki dua nilai, yaitu benar dan salah.</p>  <pre> Boolean_javascript.html > html 1 <!DOCTYPE html> 2 <html> 3 <body> 4 <h2>JavaScript Boolean</h2> 5 <p>Boolean hanya memiliki nilai true dan false</p> 6 <p id="demo"></p> 7 8 <script> 9 var x = 5; 10 var y = 5; 11 var z = 6; 12 document.getElementById("demo").innerHTML = 13 (x == y) + "
" + (x == z); 14 </script> 15 </body> 16 </html> </pre> <p>JavaScript Boolean</p> <p>Boolean hanya memiliki nilai true dan false</p> <p>true false</p>
11	<p>Ketikkan program dibawah ini dan simpan dengan nama array_javascript.html</p> <pre> <!DOCTYPE html> <html> <body> <h2>JavaScript Arrays</h2> <p>Array</p> <p id="demo"></p> <script> var cars = ["Satu", "Dua", "Tiga"]; document.getElementById("demo").innerHTML = cars[0]; </script> </body> </html> </pre>
12	Amati apa yang muncul pada browser
13	<p>Catat hasil pengamatanmu (Soal No.12)</p> <p>➤ Program array menampilkan nilai pertama dari array cars, yaitu “Satu”. Dari percobaan ini terlihat bahwa array bisa menyimpan banyak data dalam satu variabel, dan kita bisa mengaksesnya berdasarkan indeks.</p>

```
array.javascript.html > html
1  <!DOCTYPE html>
2  <html>
3  <body>
4    <h2>JavaScript Array</h2>
5
6    <p>Array</p>
7
8    <p id="demo"></p>
9
10   <script>
11     var cars = ["Satu", "Dua", "Tiga"];
12     document.getElementById("demo").innerHTML = cars[0];
13   </script>
14 </body>
15 </html>
```

127.0.0.1:5500/array.javascript.html

All Bookmarks

JavaScript Array

Array

Satu

Praktikum 8: Operator

Operator adalah simbol yang digunakan untuk melakukan operasi pada suatu nilai dan variabel.

Operator dalam pemrograman terbagi dalam 6 jenis:

1. Operator aritmatika;
2. Operator Penugasan (Assignment);
3. Opeartor relasi atau perbandingan;
4. Operator Logika;
5. Operator Bitwise;
6. Operator Ternary;
7. Operator aritmatika merupakan operator untuk melakukan operasi aritmatika seperti penjumlahan, pengurangan, pembagian, perkalian, dsb.
8. Operator aritmatika terdiri dari:

Nama Operator Simbol

Penjumlahan	+
Pengurangan	-
Perkalian	*
Pemangkatan	**
Pembagian	/
Sisa Bagi	%

Langkah	Keterangan
1	Buat File baru bernama operator_javascript1.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	<p>Ketikkan program dibawah ini</p> <pre><!DOCTYPE html> <html> <body> <h2>JavaScript Operators</h2> <p>x = 5, y = 2, menghitung z = x + y, dan tampil z:</p> <p id="demo"></p> <script> var x = 5; var y = 2; var z = x + y; document.getElementById("demo").innerHTML = z; </script> </body> </html></pre>
3	Amati apa yang muncul pada browser

4

Catat hasil pengamatanmu (Soal No.13)

- Terdapat deklarasi variabel x dengan nilai 5 dan y dengan nilai 2. Sebuah variabel ketiga, z, dideklarasikan untuk menyimpan hasil dari penjumlahan x dan y. Kode JavaScript kemudian menggunakan metode `document.getElementById("demo").innerHTML` untuk menempatkan nilai z ke dalam paragraf HTML yang memiliki id "demo". Ketika halaman tersebut dijalankan di peramban (browser), bagian atas menampilkan judul "JavaScript Operator" dan deskripsi operasi, sementara bagian bawah, yang merupakan hasil eksekusi kode, menampilkan angka 7. Angka ini adalah hasil yang benar dari operasi $5 + 2$

```
operator_javascript.html > html
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <h2>JavaScript Operator</h2>
5
6   <p>x = 5, y = 2, menghitung z = x + y, dan tampil z;</p>
7
8   <p id="demo"></p>
9
10  <script>
11    var x = 5;
12    var y = 2;
13    var z = x + y;
14    document.getElementById("demo").innerHTML = z;
15  </script>
16 </body>
17 </html>
```

127.0.0.1:5500/operator_javascript.html

JavaScript Operator

x = 5, y = 2, menghitung z = x + y, dan tampil z;

7

Praktikum 9: Percabangan

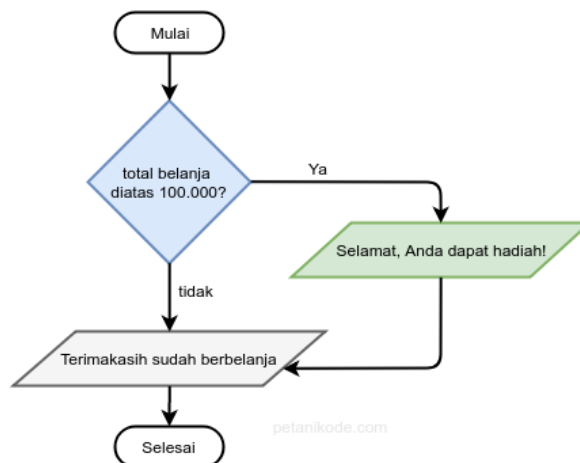
Dapat dikatakan bahwa percabangan dan putaran merupakan salah satu inti metode dalam semua bahasa pemrograman yang ada di dunia, karena dengan percabangan dan putaran akan dihasilkan sebuah program yang dinamis, dan bukan program yang linear serta bersifat statik. Karena JavaScript merupakan salah satu cara dalam melakukan pemrograman web di sisi client, maka JavaScript juga memiliki kemampuan ini.

Beberapa fungsi percabangan:

- Use if to specify a block of code to be executed, if a specified condition is true
- Use else to specify a block of code to be executed, if the same condition is false
- Use else if to specify a new condition to test, if the first condition is false
- Use switch to specify many alternative blocks of code to be executed

🚦 Percabangan if

Percabangan *if* merupakan percabangan yang hanya memiliki **satu blok pilihan** saat kondisi bernilai benar. Coba perhatikan *flowchart* berikut ini:



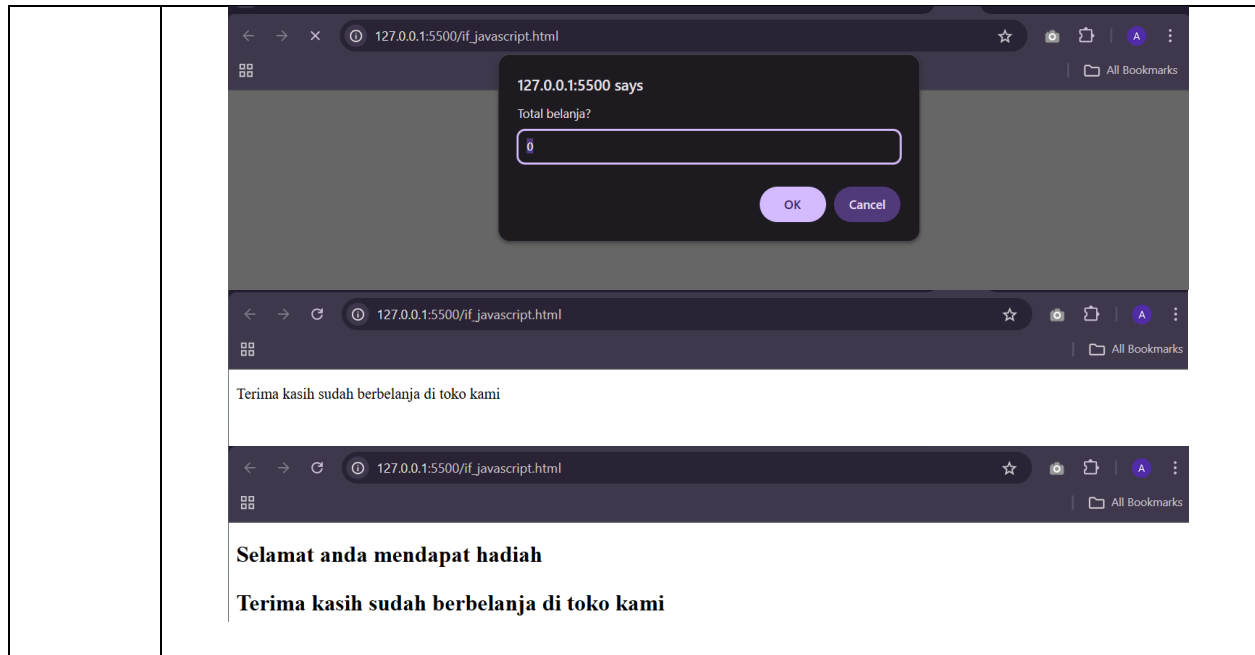
“Jika total belanja lebih besar dari Rp 100.000, Maka tampilkan pesan Selamat, Anda dapat hadiah”

Kalau dibawah Rp 100.000 bagaimana?

Ya pesannya tidak ditampilkan.

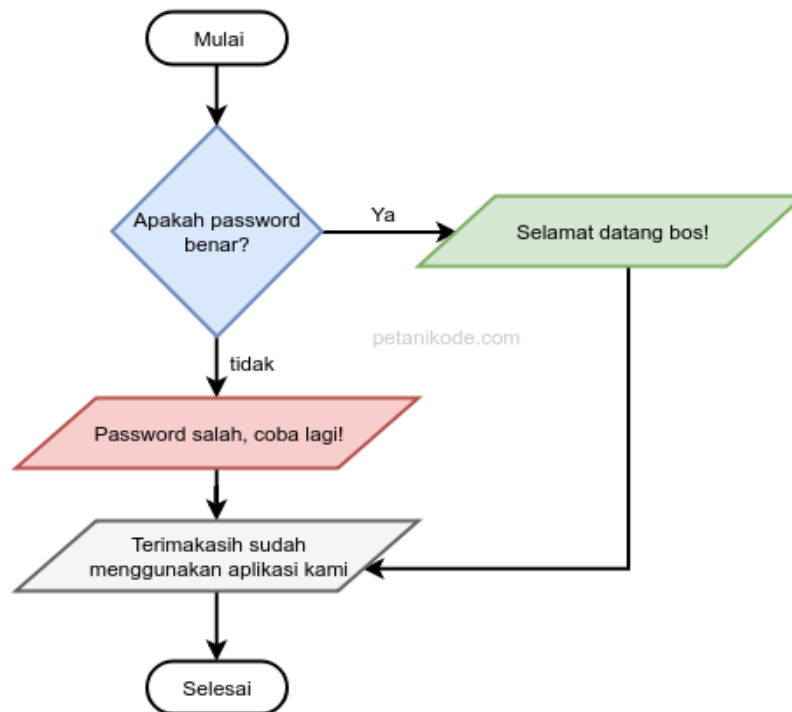
Langkah	Keterangan
1	Buat File baru bernama if_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	Ketikkan program dibawah ini

	<pre> <!DOCTYPE html> <html lang="en"> <head> <title>Percabangan if</title> </head> <body> <script> var totalBelanja = prompt("Total belanja?", 0); if(totalBelanja > 30000){ document.write("<h2>Selamat Anda dapat hadiah</h2>"); } document.write("<p>Terimakasih sudah berbelanja di toko kami</p>"); </script> </body> </html> </pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.14)</p> <ul style="list-style-type: none"> ➤ Program meminta pengguna memasukkan "Total belanja?" menggunakan fungsi prompt(). Jika nilai yang dimasukkan oleh pengguna lebih besar dari 30000, maka peramban akan menampilkan teks "Selamat anda mendapat hadiah". Terlepas dari kondisi tersebut terpenuhi atau tidak, program akan selalu menampilkan teks "Terima kasih sudah berbelanja di toko kami". ➤ Skenario pertama, yang terletak di bagian bawah, menampilkan teks "Terima kasih sudah berbelanja di toko kami" saja. Ini terjadi ketika pengguna memasukkan nilai total belanja yang tidak lebih dari 30000, sehingga kondisi if tidak terpenuhi dan teks hadiah tidak muncul. Skenario kedua, yang berada di bagian paling bawah, menunjukkan kedua teks: "Selamat anda mendapat hadiah" dan "Terima kasih sudah berbelanja di toko kami". Hal ini terjadi ketika pengguna memasukkan nilai yang lebih besar dari 30000, <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Percabangan if</title> 5 </head> 6 <body> 7 <script> 8 var totalBelanja = prompt("Total belanja?", 0); 9 10 if(totalBelanja > 30000){ 11 document.write("<h2>Selamat anda mendapat hadiah</2>"); 12 } 13 14 document.write("<p>Terima kasih sudah berbelanja di toko kami</p>"); 15 </script> 16 </body> 17 </html> </pre>



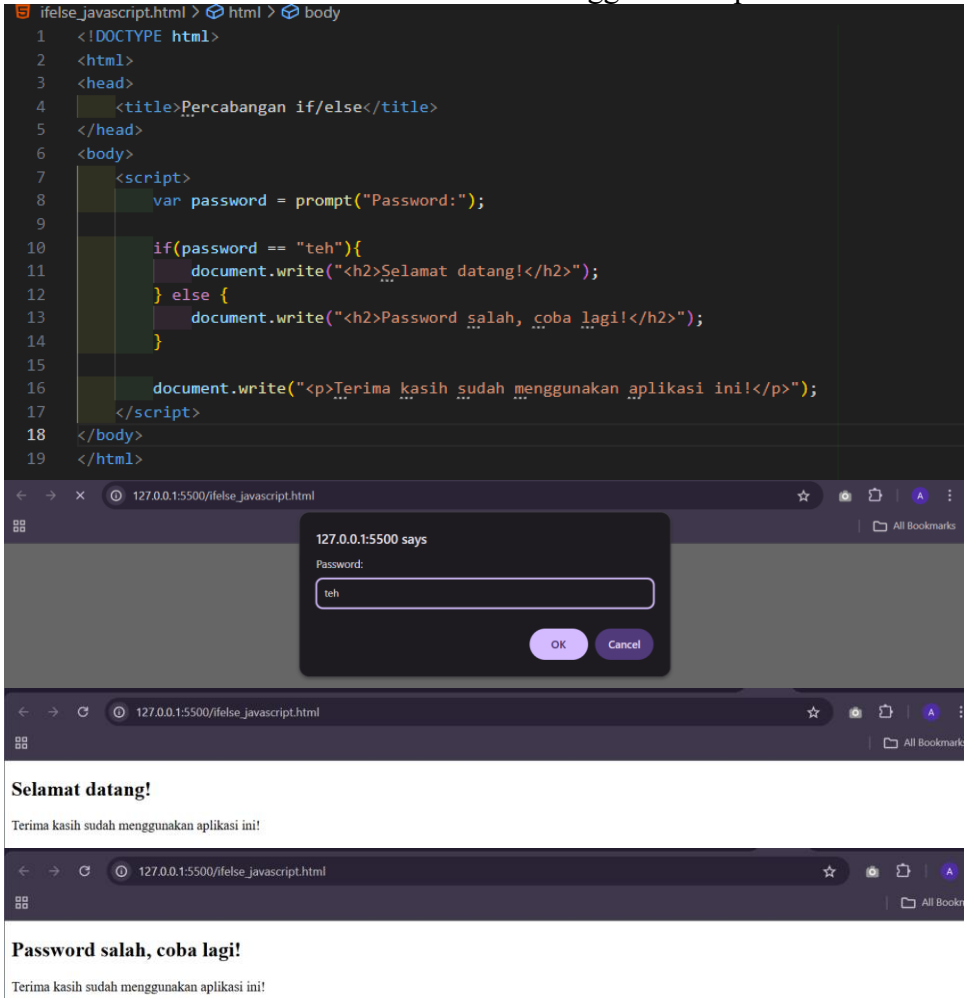
🚦 Percabangan *if/else*

Percabangan *if/else* merupakan percabangan yang memiliki **dua blok pilihan**. Pilihan pertama untuk kondisi **benar**, dan pilihan kedua untuk kondisi **salah** (*else*). Coba perhatikan flowchart ini:



Ini adalah flowchart untuk mengecek password. Apabila password benar, pesan yang ada pada blok hijau akan ditampilkan: **“Selamat datang bos!”** Tapi kalau salah, maka pesan yang ada di blok merah yang akan ditampilkan: **“Password salah, coba lagi!”**

Langkah	Keterangan
1	Buat File baru bernama ifelse_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	Ketikkan program dibawah ini <pre> <!DOCTYPE html> <html> <head> <title>Percabangan if/else</title> </head> <body> <script> var password = prompt("Password:"); if(password == "teh"){ document.write("<h2>Selamat datang !</h2>"); } else { document.write("<p>Password salah, coba lagi!</p>"); } document.write("<p>Terima kasih sudah menggunakan aplikasi ini!</p>"); </script> </body> </html> </pre>

3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.15)</p> <p>➤ Program meminta pengguna untuk memasukkan "Password" menggunakan fungsi prompt(). Kondisi if memeriksa apakah password yang dimasukkan sama dengan "teh". Jika benar, halaman akan menampilkan "Selamat datang!". Sebaliknya, jika password salah, program akan menampilkan "Password salah, coba lagi!". Kedua skenario tersebut sama-sama akan diikuti oleh teks "Terima kasih sudah menggunakan aplikasi ini!".</p>  <p>The screenshot shows a code editor with the following JavaScript code:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Percabangan if/else</title> 5 </head> 6 <body> 7 <script> 8 var password = prompt("Password:"); 9 10 if(password == "teh"){ 11 document.write("<h2>Selamat datang!</h2>"); 12 } else { 13 document.write("<h2>Password salah, coba lagi!</h2>"); 14 } 15 16 document.write("<p>Terima kasih sudah menggunakan aplikasi ini!</p>"); 17 </script> 18 </body> 19 </html> </pre> <p>Below the code, three browser screenshots are shown:</p> <ul style="list-style-type: none"> The first screenshot shows a prompt box asking for a password. The user has entered "teh". The second screenshot shows the browser displaying "Selamat datang!" followed by "Terima kasih sudah menggunakan aplikasi ini!". The third screenshot shows the browser displaying "Password salah, coba lagi!" followed by "Terima kasih sudah menggunakan aplikasi ini!".

🚦 Percabangan *switch/case*

Percabangan *switch/case* adalah bentuk lain dari percabangan *if/else/if*.

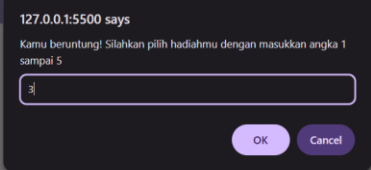
Strukturnya seperti ini:

```
switch(variabel){
    case <value>:
        // blok kode
        break;
    case <value>:
        // blok kode
        break;
    default:
        // blok kode
}
```

Langkah	Keterangan
1	Buat File baru bernama switchcase_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	<div>Ketikkan program dibawah ini</div> <pre><!DOCTYPE html> <html> <head> <title>Percabangan switch/case</title> </head> <body> <script> var jawab = prompt("Kamu beruntung! Silahakn pilih hadiahmu dengan memasukan angka 1 sampai 5"); var hadiah = ""; switch(jawab){ case "1": hadiah = "Tisu"; break; case "2": hadiah = "1 Kotak Kopi"; break; case "3": hadiah = "Sticker"; break; case "4": hadiah = "Minyak Goreng"; break; case "5": hadiah = "Uang Rp 50.000"; break; default: document.write("<p>Oops! anda salah pilih</p>"); } if(hadiah == ""){ document.write("<p>Kamu gagal mendapat hadiah</p>"); } else { document.write("<h2>Selamat kamu mendapatkan " + hadiah + "</h2>"); } </script> </body> </html></pre>
3	Amati apa yang muncul pada browser
4	<div>Catat hasil pengamatanmu (Soal No.16)</div> <div>➤ Program meminta pengguna memasukkan angka dari 1 hingga 5. Jawaban tersebut kemudian dievaluasi dalam switch statement. Jika pengguna memasukkan "3", misalnya, program akan menjalankan kode di case "3", yang menetapkan hadiah sebagai "Sticker". Jika tidak ada kasus yang cocok, default</div>

akan dijalankan. Di luar percabangan switch, terdapat percabangan if/else yang memeriksa apakah variabel hadiah kosong atau tidak

```
switchcase_javascript.html > html > body > script
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Percabangan switch/case</title>
5  </head>
6  <body>
7    <script>
8      var jawab = prompt ("Kamu beruntung! Silahkan pilih hadiahmu dengan masukkan angka 1 sampai 5");
9      var hadiah = "";
10     switch(jawab){
11       case "1":
12         hadiah = "Tisu";
13         break;
14       case "2":
15         hadiah = "1 Kotak Kopi";
16         break;
17       case "3":
18         hadiah = "Sticker";
19         break;
20       case "4":
21         hadiah = "Minyak Goreng";
22         break;
23       case "5":
24         hadiah = "Uang Rp 50.000";
25         break;
26       default:
27         document.write("<p>Opps! anda salah pilih</p>");
28     }
29     if(hadiah === ""){
30       document.write("<p>kamu gagal mendapat hadiah</p>");
31     } else {
32       document.write("<h2>Selamat! Kamu mendapat " + hadiah + "</h2>");
33     }
34   </script>
35 </body>
36 </html>
```



Selamat! Kamu mendapat Sticker

Percabangan bersarang

Kita juga dapat membuat blok percabangan di dalam percabangan. Ini disebut percabangan bersarang atau *nested if*.

Langkah	Keterangan
1	Buat File baru bernama nestedif_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	<p>Ketikkan program dibawah ini</p> <pre><!DOCTYPE html> <html> <head> <title>Percabangan Ternary</title> </head> <body> <script> var username = prompt("Username:"); var password = prompt("Password:"); if(username == "mahasiswa"){ if(password == "kopi"){ document.write("<h2>Selamat datang </h2>"); } else { document.write("<p>Password salah, coba lagi!</p>"); } } else { document.write("<p>Anda tidak terdaftar!</p>"); } </script> </body> </html></pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.17)</p> <ul style="list-style-type: none">➤ Program meminta pengguna untuk memasukkan "Username" dan "Password" secara berurutan. Kondisi if pertama memeriksa apakah username sama dengan "mahasiswa". Jika benar, program akan melanjutkan ke if kedua yang memeriksa apakah password sama dengan "kopi. Hasil di peramban memperlihatkan dua kemungkinan: "Selamat datang!" jika kedua kredensial benar, dan "Anda tidak terdaftar!" jika username atau password salah.

nestedif javascript.html > html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Percabangan Ternary</title>
5 </head>
6 <body>
7   <script>
8     var username = prompt("Username:");
9     var password = prompt("Password:");
10
11     if (username == "mahasiswa") {
12       if(password == "kopi") {
13         document.write("<h2>Selamat datang!</h2>");
14       } else {
15         document.write("<h2>Password salah, coba lagi!</h2>");
16       }
17     } else {
18       document.write("<h2>Anda tidak terdaftar!</h2>");
19     }
20   </script>
21 </body>
22 </html>
```

127.0.0.1:5500/nestedif_javascript.html

127.0.0.1:5500 says

Username:

OK Cancel

127.0.0.1:5500/nestedif_javascript.html

127.0.0.1:5500 says

Password:

OK Cancel

127.0.0.1:5500/nestedif_javascript.html

Selamat datang!

127.0.0.1:5500/nestedif_javascript.html

Anda tidak terdaftar!

Praktikum 10: 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 mengulang.

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

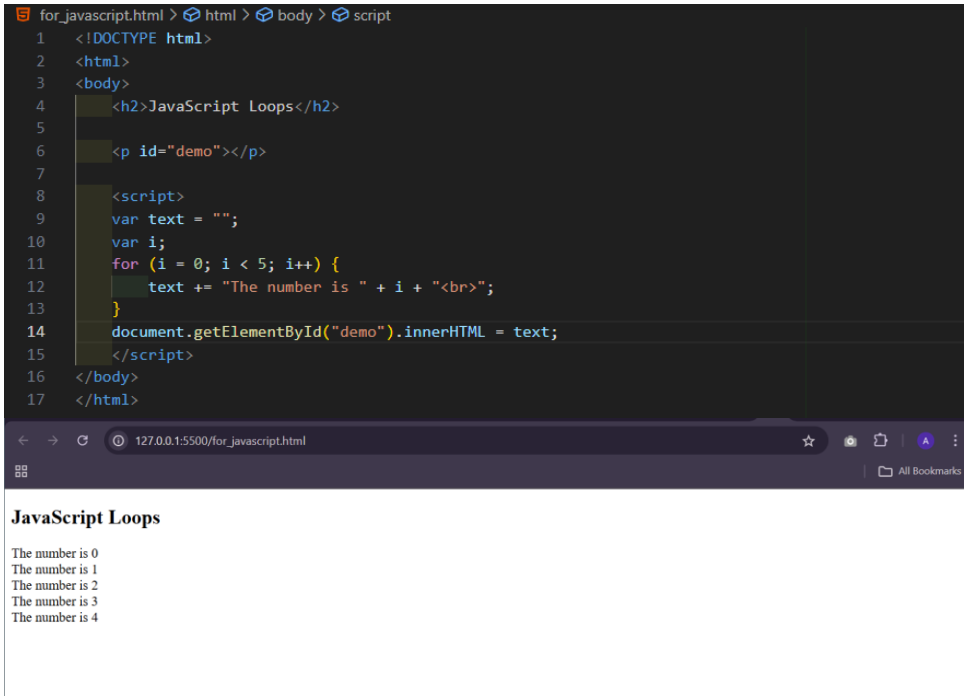
Perulangan For di Javascript

Perulangan for merupakan perulangan yang termasuk dalam *counted loop*, karena sudah jelas berapa kali ia akan mengulang.

Bentuknya seperti ini:

```
for(let i = 0; i < 10; i++){  
    document.write("<p>Perulangan ke-" + i + "</p>")  
}
```

Langkah	Keterangan
1	Buat File baru bernama for_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	Ketikkan program dibawah ini

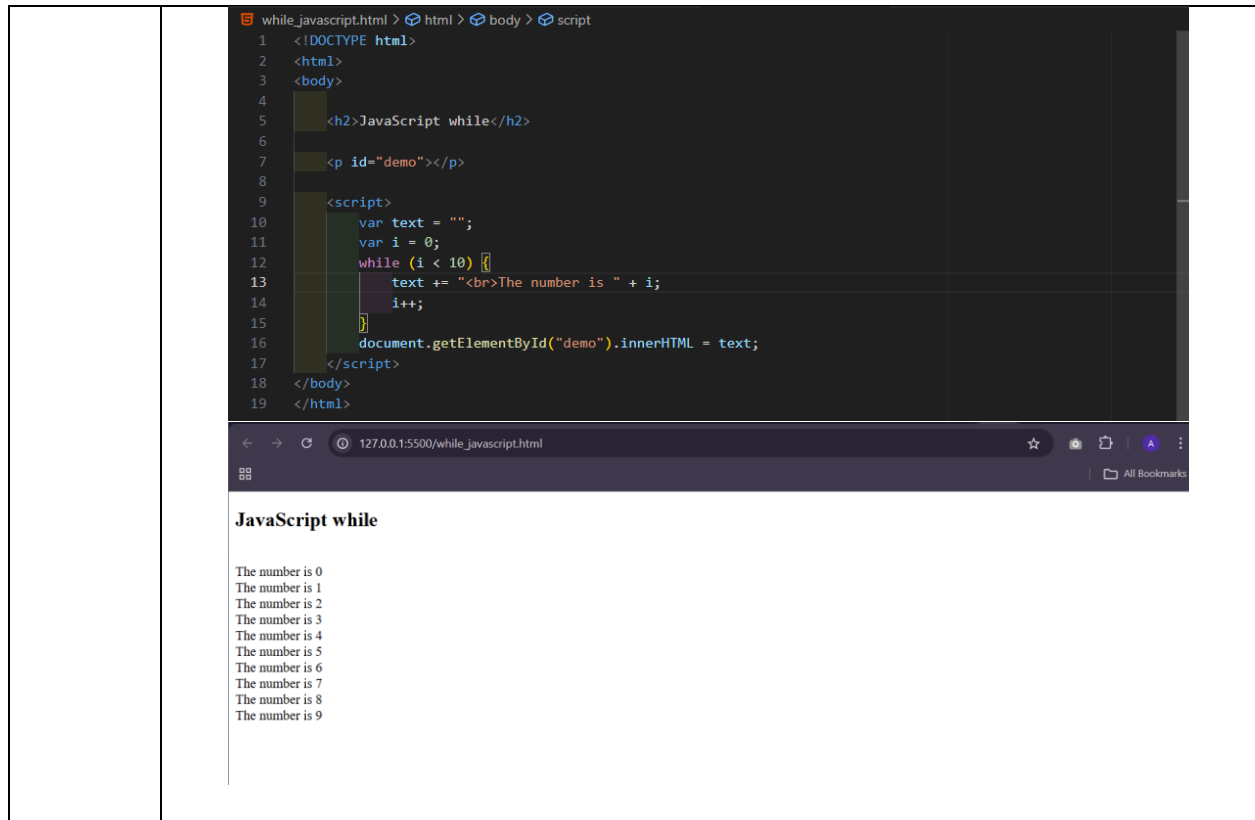
	<pre> <!DOCTYPE html> <html> <body> <h2>JavaScript Loops</h2> <p id="demo"></p> <script> var text = ""; var i; for (i = 0; i < 5; i++) { text += "The number is " + i + "
"; } document.getElementById("demo").innerHTML = text; </script> </body> </html> </pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.18)</p> <ul style="list-style-type: none"> ➤ Program mendeklarasikan sebuah variabel text yang awalnya kosong. Perulangan for dimulai dari i = 0 dan akan terus berjalan selama i kurang dari 5 (i < 5), dengan nilai i bertambah satu setiap kali perulangan selesai (i++). Di setiap iterasi, perulangan menambahkan string "The number is" diikuti oleh nilai i saat itu ke dalam variabel text. Hasil akhirnya, yang ditampilkan di elemen HTML dengan id "demo", adalah daftar angka dari 0 hingga 4 yang masing-masing berada di baris baru karena penggunaan tag
. 

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.


Langkah	Keterangan
1	Buat File baru bernama while_javascript.html dan simpan dalam direktori belajarjavascript di folder localhost
2	<p>Ketikkan program dibawah ini</p> <pre><!DOCTYPE html> <html> <body> <h2>JavaScript while</h2> <p id="demo"></p> <script> var text = ""; var i = 0; while (i < 10) { text += "
The number is " + i; i++; } document.getElementById("demo").innerHTML = text; </script> </body> </html></pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.19)</p> <ul style="list-style-type: none">➤ Variabel i diinisialisasi dengan nilai 0. Perulangan while akan terus berjalan selama kondisi i < 10 terpenuhi. Di setiap putaran, teks "The number is" ditambah dengan nilai i saat itu dan ditambahkan ke variabel text. Setelah itu, nilai i dinaikkan satu (i++). Hasil akhir yang ditampilkan di halaman web adalah daftar angka dari 0 sampai 9.

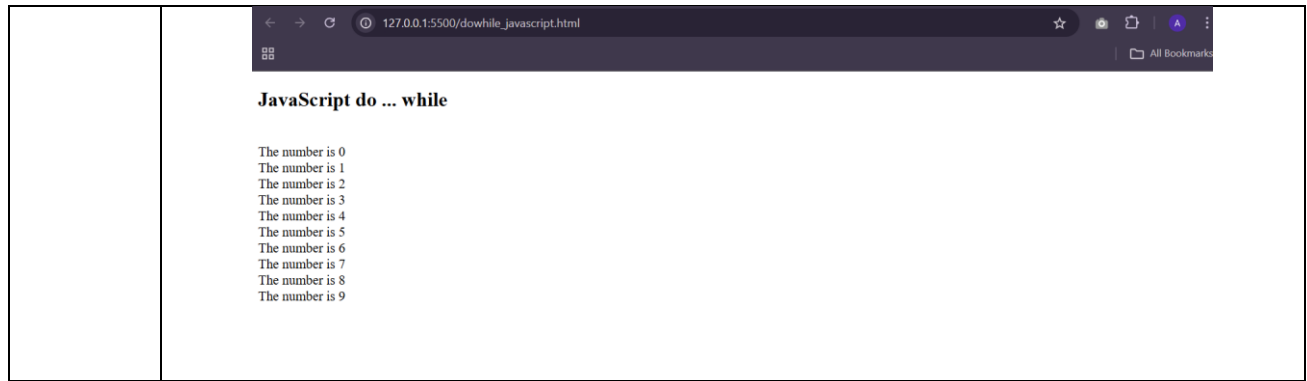


- ✚ Perulangan Do/While di Javascript
 - Perulangan do/while sama seperti perulangan while.
 - Perbedaanya:
 - Perulangan do/while akan melakukan perulangan sebanyak 1 kali terlebih dahulu, lalu mengecek kondisi yang ada di dalam kurung while
 - Bentuknya seperti ini:

```
do {
  // blok kode yang akan diulang
} while (<kondisi>);
```

Langkah	Keterangan
1	Buat File baru bernama dowhile_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost
2	Ketikkan program dibawah ini

	<pre> <!DOCTYPE html> <html> <body> <h2>JavaScript do ... while</h2> <p id="demo"></p> <script> var text = "" var i = 0; do { text += "
The number is " + i; i++; } while (i < 10); document.getElementById("demo").innerHTML = text; </script> </body> </html> </pre>
3	Amati apa yang muncul pada browser
4	<p>Catat hasil pengamatanmu (Soal No.20)</p> <ul style="list-style-type: none"> ➤ Gambar ini menampilkan kode perulangan, namun kali ini menggunakan struktur do...while. Pada perulangan ini, blok kode di dalam do akan dieksekusi setidaknya satu kali, sebelum kondisi while ($i < 10$) dievaluasi. Variabel i diinisialisasi ke 0, dan di dalam blok do, teks "The number is" ditambahkan ke variabel text, dan i dinaikkan. Perulangan ini akan terus berjalan sampai i mencapai 10. Hasil akhirnya di halaman web menunjukkan daftar angka dari 0 hingga 9, 



Referensi :

- 1) Jason Beaird, The principles of Beautiful Web Design
- 2) Rian Ariona, Belajar HTML dan CSS (Tutorial Fundamental dalam mempelajari HTML dan CSS)
- 3) Adi Hadisaputra, HTML dan CSS Fundamental dari Akar menuju Daun John Duckett, HTML dan CSS design and build websites