

**LAPORAN
PEMROGRAMAN WEB TI
JOBSHEET 3**

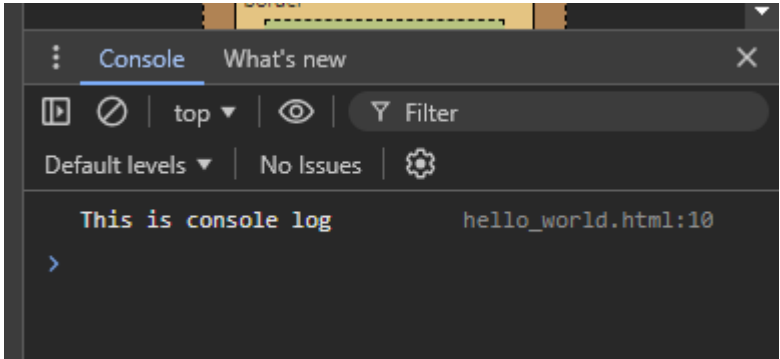
**OLEH:
DIRHAMAWAN PUTRANTO (244107023006)**



**PROGRAM STUDI
D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG**

Praktikum Bagian 1 : Membuat Program Javascript

Langkah	Keterangan
1	Buat folder baru dengan nama minggu3. Pada folder tersebut, buat file baru bernama <code>hello_world.html</code>
2	<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> <p>Ketikkan kode program di bawah ini :</p>
3	Simpan dengan nama <code>hello_world.html</code> , kemudian buka file tersebut dengan web browser.
4	<p>Amati apa yang terjadi pada browser, kemudian catat hasil pengamatanmu (Pertanyaan No. 1)</p> <p><code>console.log("Saya sinau javascript")</code> function untuk menampilkan text "saya sinau javascript" ke console browser..</p> <p><code>document.write("Hello World");</code></p>

	function untuk menampilkan text "Hello World" langsung ke HTML. Text akan muncul di halaman web.
5	Sekarang coba buka console javascript, kemudian lihat pada Inspect Elements > Console
6	<p>Amati apa yang terjadi pada tab Console, kemudian catat hasil pengamatanmu! (Pertanyaan No. 2)</p> 
7	<p>Tadi kita</p> <pre>console.log("Saya belajar Javascript");</pre> <p>menuliskan perintah :</p> <p>Menurut Anda, mengapa perintah tersebut tidak ditampilkan? (Pertanyaan No. 3)</p>

Praktikum Bagian 2 : Cara Menulis Kode Javascript di HTML

Pada Praktikum Bagian 1, kita sudah menulis kode JavaScript di dalam HTML, cara tersebut merupakan cara penulisan *embedded* (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 <code><script></code> untuk menempelkan (<i>embed</i>) kode Javascript pada HTML. Tag ini dapat ditulis didalam tag <code><head></code> dan <code><body></code>
2	<p>Buat file <code>embed.html</code>. Ketikkan kode program di bawah 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 (Pertanyaan No. 4)</p> <p>Perbedaan penulisan script console log. Pertama berada di dalam tag <code><head></code>. Kedua berada di dalam tag <code><body></code>. Keduanya memiliki fungsi yang sama, yaitu mencetak pesan ke console browser menggunakan <code>console.log()</code>.</p>
4	<p>Menurut Anda manakah yang lebih baik, dituliskan didalam tag <code><head></code> atau <code><body></code>? (Pertanyaan No. 5)</p> <p>Tergantung pada penggunaan. Jika ingin menggunakan library seperti jQuery, script bisa diletakkan di head. Untuk code yang memanipulasi elemen HTML secara langsung bisa diletakkan di body.</p>

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>Buatlah file <code>inline.html</code>.</p> <pre>Klik aku!</pre> <p>Ketikkan kode program di bawah ini :</p> <pre>Klik aku!</pre> <p>Atau bisa juga seperti ini :</p>
3	<p>Amati apa yang terjadi pada browser! Catat hasil pengamatanmu (Pertanyaan No. 6)</p> <p>Keduanya sama sama memunculkan alert pada browser.</p>
4	<p>Apa yang membedakan dari kedua kode program tersebut (Pertanyaan No. 7)</p> <p>Perbedaan nya ada pada cara memunculkan alert nya.</p> <p>Pada kode pertama menggunakan atribut onclick dan kode kedua menampilkan alert dari href. Namun penggunaan onclick lebih baik karena memisahkan struktur HTML dari javascript.</p>

3. Penulisan Kode Javascript Eksternal

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	<div>  </div> <p>dua file yaitu file HTML dan Javascript.</p> <p>Mari kita coba, buat</p>

3	<p>Isi dari file <code>kode-program.js</code>:</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 (Pertanyaan No. 8)</p> <p>script mengambil file kode-program.js untuk menjalankan kode ada di dalam file tersebut.</p>
6	<p>Pada percobaan tersebut, kita menulis kode javascript terpisah dengan kode HTML. Lalu pada kode HTML kita menyisipkan atribut <code>src</code> pada tag <code><script></code></p> <pre><!-- Menyisipkan kode js eksternal --> <script src="kode-program.js"></script></pre> <p>Maka apapun yang ada di dalam file <code>kode-program.js</code> akan dapat dibaca dari file <code>index.html</code></p>
7	<p>Pindahkan file <code>kode-program.js</code> pada folder lain, apa yang akan terjadi jika file javascript berada di folder yang berbeda?</p> <p>Amati dan catat hasil pengamatanmu (Pertanyaan No. 9)</p> <p>Jika file js berada di folder yang berbeda, kita tinggal mengatur path nya pada tag script di atribut src.</p>

8

```

├── js/
│   └── kode-program.js
└── index.html

```

Misalkan kita memiliki struktur folder seperti ini:

Maka untuk menyisipkan file `kode-program.js` ke dalam HTML, kita bisa

```
<script src="js/kode-program.js"></script>
```

menuliskan kode berikut :

Praktikum Bagian 3: 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 <code>alert_javascript.html</code>	
2	Ketikkan kode program di bawah 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 (Pertanyaan No. 10)</p> <p>function message() adalah fungsi bernama message. Fungsi ini akan menjalankan kode di dalamnya. Contoh di atas function menjalankan alert.</p> <p>Atribut onload pada tag <body> digunakan untuk menentukan fungsi yang akan dijalankan ketika halaman web selesai dimuat. Pada contoh ini, fungsi message akan dipanggil.</p>
5	Buat file baru bernama <code>confirm_javascript.html</code>
6	<p>Ketikkan kode program di bawah 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 (Pertanyaan No. 11)</p> <p>Terdapat variable yakin yang dideklarasikan dengan “var” Fungsi confirm menampilkan pop up konfirmasi. Di fungsi confirm terdapat pengembalian nilai true yang dimana jika user klik Ok Akan di arahkan ke link yang ditentukan dengan window.location. Dan nilai false akan menjalankan document.write.</p>
9	Ketikkan kode program di bawah ini

	<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	Amati apa yang muncul pada browser	
11	Catat hasil pengamatanmu (Pertanyaan No. 12) Saat website dibuka akan ada alert dengan text input di dalamnya, fungsi ini dari prompt. Prompt terdapat 2 argumen. Argumen pertama adalah teks yang akan ditampilkan. Argumen kedua adalah nilai default yang akan di inputkan user.	

Praktikum Bagian 4: 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()` mengembalikan output ke konsol JavaScript; –

Fungsi `document.write()` mengembalikan output ke dokumen HTML; – dan

`alert()` mengembalikan output ke jendela dialog.

Langkah	Keterangan
1	Buat file baru bernama <code>variable_javascript.html</code>
2	Ketikkan kode program di bawah 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

4	<p>(Pertanyaan No.13)</p> <p>Mendeklarasikan tiga variabel nama, visitorCount, dan isActive.</p> <p>Menampilkan alert jika variabel name sudah dideklarasikan.</p> <p>Menuliskan informasi tentang nama situs, jumlah pengunjung, dan status aktif ke halaman web.</p>
---	---

Deleting Variables

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 kata kunci

delete. Contoh:

```
var bookTitle = "Learn Javascript Programming"; 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 <code>function_javascript.html</code>
2	Cara memanggil fungsi didalam kode Javascript biasanya ditulis dengan : <code>namaFungsi () ;</code>
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	<p>Catat hasil pengamatanmu (Pertanyaan No.14)</p> <p><code>() =></code> ini sintaks untuk membuat fungsi arrow function. Fungsi ini tidak memiliki nama parameter, jadi tidak perlu mendefinisikan argumen saat memanggilnya.</p> <p><code>onclick="sayHello()"</code> Atribut onclick pada tag <code><a></code> ini akan menjalankan fungsi <code>sayHello()</code> ketika link tersebut diklik.</p>

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

9	<p>(Pertanyaan No.15)</p> <p>function total(numberA, numberB) Fungsi ini menerima dua parameter numberA dan numberB.</p>
---	--

	<p>return numberA + numberB Pernyataan ini mengembalikan hasil penjumlahan dari numberA dan numberB.</p> <p>total(2, 3) adalah panggilan fungsi total dengan argumen 2 dan 3. Fungsi ini akan mengembalikan hasil penjumlahan 2 dan 3, yaitu 5.</p>
--	---

Praktikum Bagian 6: 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 123name =
"Polinema";
// benar
var name123 = "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 = "Javascript Tutorials";
```

Langkah	Keterangan
1	Buat File baru bernama tipedata_javascript.html

2	<p>Ketikkan kode program berikut ini</p> <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 (Pertanyaan No. 16)</p> <p>x = 5 memberikan nilai angka 5 kepada variabel x. Tipe data ini adalah number.</p> <p>x = "John" memberikan string "John" kepada variabel x. Tipe data ini adalah String.</p> <p>Ini menunjukkan fleksibilitas JavaScript dalam mengubah tipe data variabel</p>
5	<p>Ketikkan program di bawah 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 (Pertanyaan No. 17)</p> <p>Kita bebas memilih antara tanda kutip tunggal atau ganda untuk membungkus string.</p>	
8	<p>Ketikkan program di bawah ini dan simpan dengan nama <u>boolean_javascript.html</u></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	<p>Catat hasil pengamatanmu (Pertanyaan No. 18)</p> <p>Konversi ke Boolean $(x == y)$ melakukan perbandingan antara nilai x dan y. Karena kedua nilai sama-sama 5, maka hasilnya adalah true.</p> <p>$(x == z)$ melakukan perbandingan antara nilai x dan z. Karena nilai x adalah 5 dan nilai z adalah 6, maka hasilnya adalah false.</p> <p>Hasil perbandingan secara otomatis dikonversi menjadi string ketika ditambahkan ke konten HTML. true akan dikonversi menjadi "true" dan false akan dikonversi menjadi "false".</p>
11	<p>Ketikkan program di bawah 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 (Pertanyaan No. 19)</p> <p><code>cars[0]</code> mengakses elemen pertama dari array cars. Indeks array dimulai dari 0, sehingga elemen pertama memunculkan “Satu”.</p>

Praktikum Bagian 7: 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. Opearator relasi atau perbandingan;
4. Operator Logika;
5. Operator Bitwise;
6. Operator Ternary;

Operator aritmatika merupakan operator untuk melakukan operasi aritmatika seperti penjumlahan, pengurangan, pembagian, perkalian, dsb. Operator aritmatika terdiri dari:

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

Langkah	Keterangan
1	Buat File baru bernama <code>operator_javascript.html</code>
2	<p>Ketikkan program di bawah 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	<p>Catat hasil pengamatanmu (Pertanyaan No.20)</p> <p>var z = x + y melakukan operasi penjumlahan antara nilai x dan y, kemudian menyimpan hasilnya dalam variabel z.</p>

Praktikum Bagian 8: Percabangan

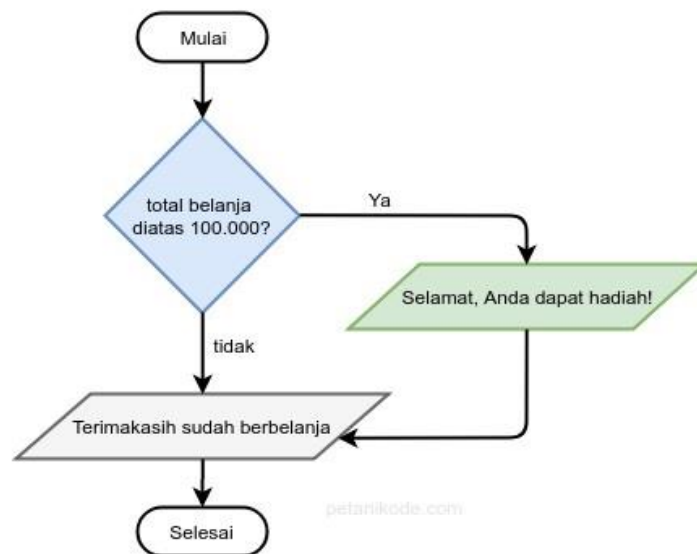
Dapat dikatakan bahwa percabangan dan *looping* merupakan salah satu inti metode dalam semua bahasa pemrograman yang ada di dunia, karena dengan percabangan dan *looping* akan dihasilkan sebuah program yang dinamis, dan bukan program yang linear serta bersifat statik. Karena JavaScript merupakan salah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

Beberapa fungsi percabangan:

- Gunakan **if** untuk menentukan blok kode yang akan dieksekusi, jika kondisi yang ditentukan adalah true
- Gunakan **else** untuk menentukan blok kode yang akan dieksekusi, jika kondisi yang sama adalah false
- Gunakan **else if** untuk menentukan kondisi baru yang akan diuji, jika kondisi pertama adalah false
- Gunakan **switch** untuk menentukan banyak blok kode alternatif yang akan dieksekusi

Percabangan IF

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



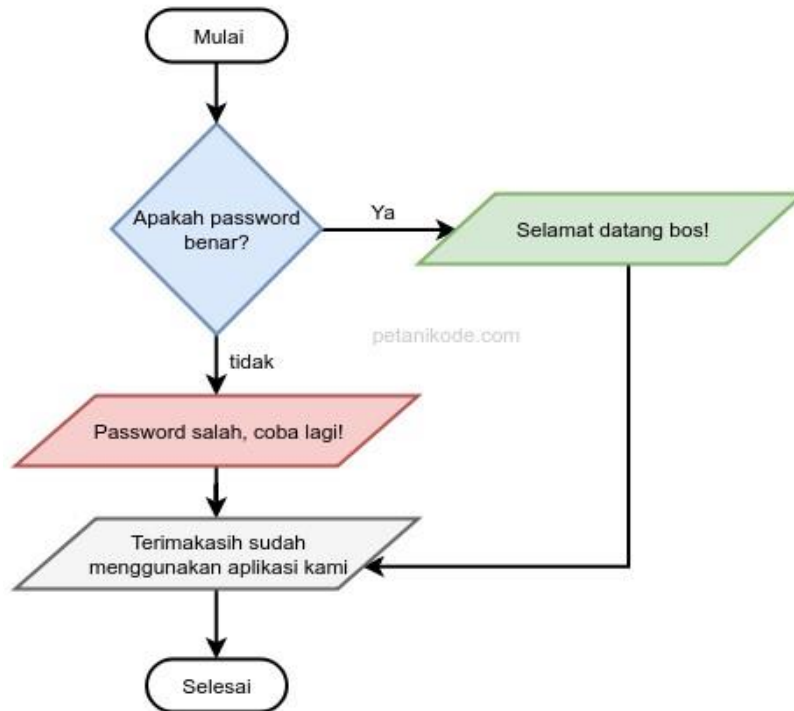
- Jika total belanja lebih besar dari Rp 100.000, maka tampilkan pesan “Selamat, Anda dapat hadiah!”.
- Jika di bawah Rp 100.000, maka pesan “Selamat, Anda dapat hadiah!” tidak ditampilkan.

Langkah	Keterangan
1	Buat File baru bernama <code>if-javascript.html</code>
2	Ketikkan program di bawah 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	Catat hasil pengamatanmu (Pertanyaan No. 21)

4	(Pertanyaan No.21) Muncul insert untuk user memasukkan total belanja. Nilai yang dimasukkan user disimpan dalam variabel totalBelanja. Jika nilai totalBelanja lebih dari 30000, maka pesan "Selamat anda dapat hadiah" akan ditampilkan. Terlepas dari hasil kondisi, pesan "Thanks." akan selalu ditampilkan.
---	---

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 memeriksa password.

- Apabila password benar, pesan yang ada pada blok hijau akan ditampilkan: **“Selamat datang bos!”**
- Jika salah, maka pesan yang ada di blok merah yang akan ditampilkan: **“Password salah, coba lagi!”**

Langkah	Keterangan
1	Buat File baru bernama <code>ifelse-javascript.html</code>
2	Ketikkan program di bawah ini <pre> <!DOCTYPE html> <html> <head> <title>Percabangan if/else</title> </head> <body> <script> var password = prompt("Password:"); </pre>

	<pre> 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	Catat hasil pengamatanmu (Pertanyaan No. 22)
4	<p>(Pertanyaan No.22)</p> <p>Nilai yang dimasukkan oleh user disimpan dalam variabel password. Jika password sama dengan "teh", pesan "Selamat datang" akan ditampilkan. Jika password tidak sama dengan "teh", pesan "Password salah" akan ditampilkan.</p>

Percabangan SWITCH CASE

switch/case adalah bentuk alternatif dari struktur percabangan **if/else/if**. Dalam pernyataan **switch/case**, alih-alih mengevaluasi beberapa kondisi **if**, program mengevaluasi nilai variabel atau ekspresi dan membandingkannya dengan beberapa kemungkinan kasus. Setiap kasus mewakili nilai potensial, dan ketika kecocokan ditemukan, blok kode yang sesuai dieksekusi. Jika tidak ada kasus yang cocok, kasus **default** dieksekusi (jika disediakan), mirip dengan blok **else** dalam pernyataan **if/else**.

Struktur **switch/case** dapat membuat kode lebih mudah dibaca dan terorganisir, terutama saat berhadapan dengan beberapa kondisi berdasarkan satu variabel. Strukturnya terlihat seperti ini:

```

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

```

Langkah	Keterangan
1	Buat File baru bernama <code>switchcase_javascript.html</code>
2	Ketikkan program di bawah ini
	<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>Opps! 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	Catat hasil pengamatanmu (Pertanyaan No. 23)

4	<p>(Pertanyaan No.23)</p> <p>Switch digunakan untuk mengevaluasi banyak kemungkinan nilai dari sebuah variabel.</p> <p>case "1" sampai case "5" masing-masing kasus memeriksa apakah nilai jawab sama dengan angka 1 sampai 5.</p> <p>break menghentikan eksekusi switch setelah menemukan kasus yang cocok.</p>
---	--

Percabangan Bersarang

Percabangan Bersarang mengacu pada kondisi di mana satu pernyataan percabangan (seperti **jika**, **else**, **switch**, dll.) ditempatkan di dalam pernyataan percabangan lainnya. Hal ini memungkinkan proses pengambilan keputusan yang lebih kompleks di mana beberapa kondisi perlu dievaluasi pada tingkat yang berbeda. Dalam percabangan bersarang, hasil dari satu kondisi dapat bergantung pada hasil kondisi lainnya, memberikan kontrol yang lebih halus atas alur program.

Misalnya, Anda dapat menumpuk pernyataan **if** di dalam pernyataan **if** lain untuk terlebih dahulu memeriksa satu kondisi dan kemudian, berdasarkan itu, memeriksa kondisi kedua dalam blok pertama.

Langkah	Keterangan
1	Buat file baru bernama nestedif_javascript.html
2	Ketikkan program di bawah ini

	<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	Catat hasil pengamatanmu (Pertanyaan No.24)
4	<p>(Pertanyaan No.24)</p> <p>Kondisi if (password == "kopi") adalah kondisi dalam (nested). Kondisi ini hanya akan diperiksa jika kondisi luar benar (username adalah "mahasiswa").</p>

Praktikum Bagian 9: 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*. Perbedaanannya adalah:

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

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 <code>for_javascript.html</code>
2	<p>Ketikkan program di bawah ini</p> <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 (Pertanyaan No. 25)</p> <p>Loop for dimulai dengan i bernilai 0. Selama i kurang dari 5, loop akan berulang. i++ setelah setiap loop, nilai i akan ditambah 1.</p> <p>Dalam setiap loop, nilai i ditambahkan ke string text bersama dengan teks "The number is " dan tag
.</p> <p>Loop terus berjalan hingga i mencapai 5.</p>

Perulangan WHILE

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 <code>while_javascript.html</code>
2	Ketikkan program di bawah ini
	<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 (Pertanyaan No. 26)</p> <p>Loop while akan terus berjalan selama kondisi $i < 10$ benar. Selama i kurang dari 10, loop akan berulang.</p> <p><code>text +=</code> digunakan untuk menambahkan nilai ke variabel <code>text</code>. <code>
The number is " + 1</code> menggabungkan string <code>
The number is</code> dengan nilai 1. Hasilnya akan ditambahkan ke variabel <code>text</code>.</p> <p><code>i++</code> setelah setiap loop, nilai i akan ditambah 1.</p>

Perulangan DO WHILE

Langkah	Keterangan
1	Buat file baru bernama <code>dowhile_javascript.html</code>

2	Ketikkan program di bawah ini
---	-------------------------------

Perulangan do/while adalah variasi dari perulangan while di JavaScript. Perbedaan utama di antara keduanya adalah bahwa perulangan do/while akan selalu mengeksekusi kode di dalam perulangan **setidaknya sekali**, terlepas dari apakah kondisinya benar atau salah. Ini karena kondisi dievaluasi **setelah** blok kode dieksekusi, bukan sebelumnya, seperti dalam perulangan while standar:

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

Karakteristik utama:

- Kode di dalam blok do berjalan terlebih dahulu, lalu kondisinya diperiksa.
- Jika kondisinya benar, perulangan berulang; Jika false, loop berhenti.

Jenis perulangan ini memastikan bahwa kode di dalam perulangan dieksekusi setidaknya sekali, bahkan jika kondisinya salah sejak awal.

	<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	Catat hasil pengamatanmu (Pertanyaan No.27) Perbedaan antara while dan dowhile While Kondisi diperiksa sebelum blok kode dijalankan. Jika kondisi tidak terpenuhi, blok kode tidak akan dijalankan sama sekali.	

	<p>Dowhile Kode dijalankan setidaknya sekali, kemudian kondisi diperiksa. Jika kondisi masih benar, blok kode akan dijalankan kembali.</p> <p>Dalam kode ini, penggunaan dowhile tidak memberikan perbedaan yang signifikan karena kondisi $i < 10$ selalu benar pada loop pertama.</p>
--	---

Referensi:

- 1) Jason Beard, 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
- 4) John Duckett, HTML and CSS design and build websites

Link github : <https://github.com/EEuRekaa/WebPemrogramanTI.git>