**Praktikum Bagian 1. Belajar Javascript**

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Console Javascript dapat kita buka melalui **Inspect Element->Console**.  Console Mozilla Firefox  Di dalam console, kita bisa menulis fungsi atau kode-kode javascript dan hasilnya akan langsung ditampilkan. |
| 3 | Misalnya, mari kita coba kode berikut :    Amati apa yang muncul pada console, kemudian catat hasil pengamatanmu!. |
| 2 | Jika kamu menggunakan Nodejs, maka cara mengakses console adalah dengan mengetik perintah node pada Terminal.  Console Javascript di Nodejs |
| 3 | Amati apa yang terjadi, kemudian catat hasil pengamatanmu. Apa yang dapat disimpulkan setelah mencoba *console* Javascript? (Soal No.1) |

**Praktikum 2 : Membuat Program Javascript Pertama**

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Silahkan buka text editor, kemudian buat file baru bernama hello\_world.html |
| 2 | Ketikkan kode program dibawah ini : |
| 3 | Simpan dengan nama hello\_world.html, kemudian buka file tersebut dengan web browser. |
| 4 | Amati apa yang terjadi pada browser, kemudian catat hasil pengamatanmu (Soal No.2) |
| 5 | Sekarang coba buka console javascript, kemudian lihat pada Inspect Elemets->Console |
| 6 | Amati apa yang terjadi pada tab Console, kemudian catat hasil pengatanmu!  (Soal No. 3) |
| 7 | Tadi kita menuliskan perintah :    Menurut Anda, mengapa perintah tersebut tidak ditampilkan? (Soal No.4) |

**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 (*embed*) kode Javascript pada HTML. Tag ini dapat ditulis didalam tag <head> dan <body> |
| 2 | Ketikkan kode program dibawah ini : |
| 3 | Amati apa yang terjadi pada browser? Catat hasil pengamatanmu (Soal No.1) |
| 4 | Menurut Anda manakah yang lebih baik, dituliskan didalam tag <head> atau <body>? (Soal No. 2)  Gatau :v |

|  |  |
| --- | --- |
| 1. **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 | Ketikkan kode program dibawah ini : |
| 3 | Amati apa yang terjadi pada browser! Catat hasil pengamatanmu (Soal No.1) |
| 4 | Apa yang membedakan dari kedua kode program tersebut (Soal No. 2) |

|  |  |
| --- | --- |
| 1. **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 | Mari kita coba, buat dua file yaitu file HTML dan Javascript. |
| 3 | Isi dari file kode-program.js : |
| 4 | Isi dari file index.html : |
| 5 | Amati apa yang terjadi pada browser! Catat hasil pengamatanmu (Soal No.1) |
| 6 | Pada percobaan tersebut, kita menulis kode javascript terpisah dengan kode HTML.  Lalu pada kode HTML kita menyisipkan atribut src pada tag <script>    Maka apapun yang ada di dalam file kode-program.js akan dapat dibaca dari file index.html |
| 7 | Apa yang akan terjadi jika file javascript berada di folder yang berbeda?  Amati dan catat hasil pengamatanmu (Soal No.2) |
| 8 | Misalkan kita memiliki struktur folder seperti ini :    Maka untuk menyisipkan file kode-program.js ke dalam HTML, kita bisa menuliskan kode berikut :    Karena file kode-program.js berada didalam direktori js.  Kita juga dapat menyisipkan javascript yang ada di internet dengan memberikan alamat URL lengkap.  Contoh : |

**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 promp();

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat File baru berana alert\_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan kode program dibawah ini |
|  |  |
| 3 | Amati apa yang muncul pada bwoser |
| 4 | Catat hasil pengamatanmu (Soal No.3) |
| 5 | Buat file baru bernama confirm\_javascript.html dan simpan dalam direktori belajarjavascript |
| 6 | Ketikkan kode program dibawah ini |
| 7 | Amati apa yang muncul pada bwoser |
| 8 | Catat hasil pengamatanmu (Soal No.4) |
| 9 | Ketikkan kode program dibawah ini |
| 10 | Amati apa yang muncul pada bwoser |
| 11 | Catat hasil pengamatanmu (Soal No.5) |

**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 berana variabel\_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan kode program dibawah ini |
|  |  |
| 3 | Amati apa yang muncul pada bwoser |
| 4 | Catat hasil pengamatanmu (Soal No.6) |

## Menghapus Variabel

Penghapusan variabel dalam Javscript 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 |
|  |  |
| 4 | Amati apa yang muncul pada browser |
| 5 | Catat hasil pengamatanmu (Soal No.7) |
| 6 | Parameter adalah variable yang menyimpan nilai untuk proses di dalam fungsi  Cara memanggil parameter dalam javascript adalah : |
| 7 | Ketikkan kode program berikut ini |
| 8 | Amati apa yang muncul pada browser |
| 9 | Catat hasil pengamatanmu (Soal No.8) |

**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](https://www.petanikode.com/topik/c), [C++](https://www.petanikode.com/topik/c++), [Java](https://www.petanikode.com/topik/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 tediri 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 tipedata\_javascript1.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan kode program berikut ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.9) |
| 5 | Ketikkan program dibawah ini dan simpan dengan nama string\_javascript.html |
| 6 | Amati apa yang muncul pada browser |
| 7 | Catat hasil pengamatanmu (Soal No.10) |
| 8 | Ketikkan program dibawah ini dan simpan dengan nama Boolean\_javascript.html |
| 9 | Amati apa yang muncul pada browser |
| 10 | Catat hasil pengamatanmu (Soal No.11) |
| 11 | Ketikkan program dibawah ini dan simpan dengan nama array\_javascript.html |
| 12 | Amati apa yang muncul pada browser |
| 13 | Catat hasil pengamatanmu (Soal No.12) |

**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 | Ketikkan program dibawah ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.13) |

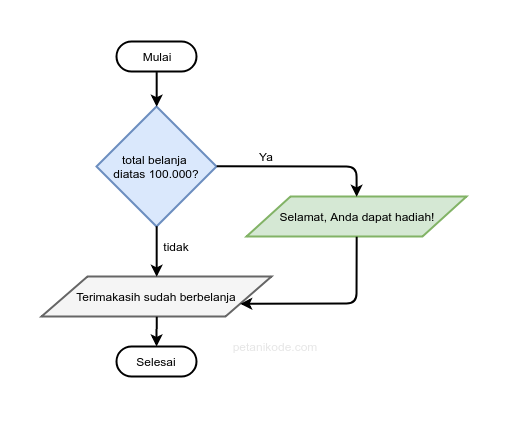
**Praktikum 9: Percabangan**

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“

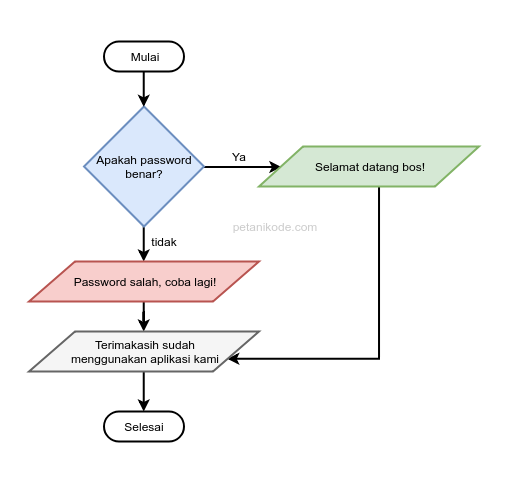
Kalau dibawah Rp 100.000 bagaimana?

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 |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.14) |

## 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 |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.15) |

## Percabangan switch/case

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

Strukturnya seperti ini:



|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat File baru bernama switchcase\_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan program dibawah ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.16) |

## Percabangan bersarang

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

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat File baru bernama nestedif\_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan program dibawah ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.17) |

**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 mengulan.

Perulangan yang termasuk dalam *Counted Loop*:

1. Perulangan For
2. Perulangan Foreach
3. Perulangan Repeat

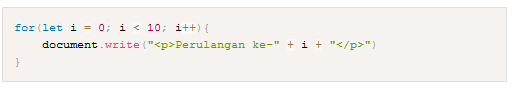
Perulangan yang termasuk dalam *Uncounted Loop*:

1. Perulangan While
2. Perulangan Do/While

## Perulangan For di Javascript

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

Bentuknya seperti ini:



|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat File baru bernama for\_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan program dibawah ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.18) |

* 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 | Ketikkan program dibawah ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.19) |

* 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:



|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat File baru bernama dowhile\_javascript.html dan simpan dalam direktori belajarjavascript.di folder localhost |
| 2 | Ketikkan program dibawah ini |
| 3 | Amati apa yang muncul pada browser |
| 4 | Catat hasil pengamatanmu (Soal No.20) |

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