

PraktikumJavaScript

(Dosen : Sirojul Munir S.Si,M.Kom)

Tujuan : Mempelajari dan melakukan implementasi JavaScript
Dateline : 1 pekan setelah praktikum
Gitlab Project : https://gitlab.com/NAMA_GROUP_PROJECT/praktikum04.git
Branch Repository : [PRODI_ROMBEL]_[NAMASINGKAT]_[NIM]_JS (ti01_budi_0110112001_JS)

Aturan Pengumpulan:

1. Dokumentasikan kode program anda di gitlab
2. Upload ke server kreasi.nurulfikri.ac.id
3. Tutorial yang tidak bisa dibuka / compile error / blank page dikenakan pinalti -50.

Tutorial

1. Pengantar JavaScript

JavaScript adalah bahasa skrip (scripting language) yang secara luas banyak digunakan untuk pengembangan aplikasi web di sisi client. Nama asli JavaScript awalnya adalah "ECMAScript", yang dikembangkan dan dimaintain oleh ECMA organisation. ECMA-262 adalah standard pertama dari JavaScript, dimana standarisasinya berdasarkan JavaScript (Netscape) dan JScript (Microsoft).

Bahasa JavaScript dikembangkan pertama kali oleh Brendan Eich di Netscape (Navigator 2.0), dan telah digunakan di semua browser Netscape dan Microsoft sejak tahun 1996. Berikut definisi-definisi yang menjelaskan tentang JavaScript :

- ✓ JavaScript didesain untuk membuat halaman web HTML menjadi interaktif
- ✓ JavaScript adalah scripting language yang berjalan di sisi client (browser)
- ✓ JavaScript biasa ditulis menyatu dengan Tag HTML
- ✓ JavaScript juga adalah interpreted language (artinya skrip langsung dieksekusi tanpa proses kompilasi)
- ✓ Setiap orang dapat menggunakan JavaScript tanpa membayar lisensi.

Saat ini JavaScript telah menjadi scripting language yang populer di internet dan dapat berjalan di hampir semua browser, seperti Internet Explorer, Firefox dan Opera.

JavaScript yang berjalan di sisi client aplikasi atau berjalan di browser mempunyai beberapa keunggulan, berikut hal-hal yang dapat dilakukan oleh JavaScript :

- ✓ JavaScript dapat mudah digunakan oleh desainer HTML - Penulis HTML biasanya bukan seorang programmer, JavaScript adalah bahasa skrip yang sangat simple dalam penulisan sintaksnya, setiap orang dapat membuat kode JavaScript di halaman HTML.
- ✓ JavaScript dapat meletakkan teks yang dinamis ke dokumen HTML - Sintaks JavaScript seperti ini : `document.write("<h1>" + name + "</h1>")` ; dapat menulis sebuah variable teks kedalam halaman HTML
- ✓ JavaScript dapat membaca dan menulis pada elemen-elemen HTML.
- ✓ JavaScript dapat digunakan untuk validasi data – JavaScript dapat digunakan untuk validasi form sebelum data dikirim ke server, hal ini membuat kerja program di sisi server lebih ringan.
- ✓ JavaScript dapat digunakan untuk mendeteksi pengguna browser
- ✓ JavaScript dapat digunakan untuk membuat cookies – JavaScript dapat digunakan untuk menyimpan data dan mengambil informasi dari pengguna komputer

Untuk memulai praktikum web javascript ini pastikan anda telah bekerja dalam dokumen root aplikasi web server anda, dan lakukan clone project praktikum04 dari gitlab

```
$sudo su -  
#cd /var/www/html  
#git clone NAMA_PROJECT_GITLAB/praktikum04.git
```

Buat direktory branch sesuai dengan **prodi_nama_nim** anda, misal:

```
#cd praktikum03  
#git checkout -b prodi_nama_nim
```

2. JavaScript dalam dokumen HTML

Untuk dapat menjalankan script JavaScript tentunya anda memerlukan suatu browser yang mendukung JavaScript seperti Netscape Navigator (mulai versi 2.0), FireFox, Opera, Konqueror (pada KDE 2) atau Microsoft Internet Explorer (mulai versi 3.0), ini adalah hal yang penting sebelum anda memulai menulis atau membuat script/program JavaScript. Perlu diperhatikan juga bahwa anda tentunya harus memiliki pengetahuan dan mengerti HTML.

JavaScript mengelola seluruh element pada halaman web secara hirarki. Setiap element tersebut dipandang sebagai objek yang dapat memiliki methodes dan properties tertentu. Dengan JavaScript kita dapat memanipulasi objek-objek tersebut, untuk itulah diperlukan pemahaman mengenai hirarki dari objek-objek HTML.

Skrip program JavaScript ditulis dalam dokumen HTML anda dengan memasukkan tag khusus <SCRIPT> dan diakhiri dengan tag </SCRIPT> (case dalam tag HTML sifatnya case-insensitive). Seluruh blok kode JavaScript yang anda tulis harus berada dalam (diantara) tag-tag tersebut. Tag awal yang menandai sebuah blok script atau program JavaScript juga dapat ditulis lengkap pada tag awalnya dengan menambahkan atribut LANGUAGE="JavaScript", penulisannya seperti berikut:

```
1 <!DOCTYPE HTML>  
2 <html>  
3  
4 <body>  
5   <p>Before the script...</p>  
6  
7   <script>  
8     alert( 'Hello, world!' );  
9   </script>  
10  
11   <p>...After the script.</p>  
12 </body>  
13  
14 </html>
```

Atribut lain yang dapat disisipkan dalam tag <SCRIPT> adalah "SRC=url", penulisannya:

```
1 <SCRIPT LANGUAGE="JavaScript" SRC="url"></SCRIPT>.
```

Atribut "SRC" sifatnya optional, atribut ini digunakan jika anda ingin menyertakan program JavaScript anda yang terpisah/eksternal dari dokumen HTML anda kedalam dokumen tersebut. Program atau script anda harus berekstensi ".js", contoh; "basic.js". Contoh:

```
1 <SCRIPT LANGUAGE="JavaScript" SRC="/path/to/basic.js"></SCRIPT>
```

Kode program JavaScript dapat diletakan dalam direktori source (src) untuk kemudahan maintenance program

```
/praktikum04/
| --- index.html
| --- src/
|   --- css/
|       --- mystyle.css
|   --- js/
|       --- basic.js
```

Program JavaScript anda dapat diletakan diantara tag <HEAD>.....</HEAD>, namun tidak harus dalam tag tersebut anda juga dapat meletakan diantara tag <BODY>.....</BODY>. Hanya saja akan lebih baik dan tertib jika anda menaruh script anda diantara tag <HEAD>></HEAD>.

Contoh:

```
1 <HEAD><TITLE>Belajar JavaScript</TITLE>
2     <SCRIPT LANGUAGE="JavaScript" SRC="js/basic.js">
3     </SCRIPT>
4 </HEAD>
```

Masalah akan timbul jika halaman web anda yang menyertakan kode JavaScript didalamnya , diakses dengan browser lama yang belum mendukung JavaScript. Jangan khawatir akan masalah tersebut, anda cukup menyertakan tag komentar "<!--" dan ditutup "//-->" dalam kode JavaScript anda seperti berikut:

```
1 <HEAD><TITLE>Belajar JavaScript</TITLE>
2 <SCRIPT LANGUAGE="JavaScript">
3     <!-- kode ini tidak terlihat di browser lama
4     document.write("Ini blok kode JavaScript!")
5     //-->
6 </SCRIPT>
7 </HEAD>
```

Script diatas akan menghasilkan teks "Ini blok kode JavaScript!" pada browser anda dan teks "kode ini tidak terlihat dibrowser lama" tidak akan dicetak (komentar), jika browser anda adalah browser versi lama, maka tidak akan mencetak apapun pada browser anda, kecuali title "Belajar JavaScript" pada title bar.

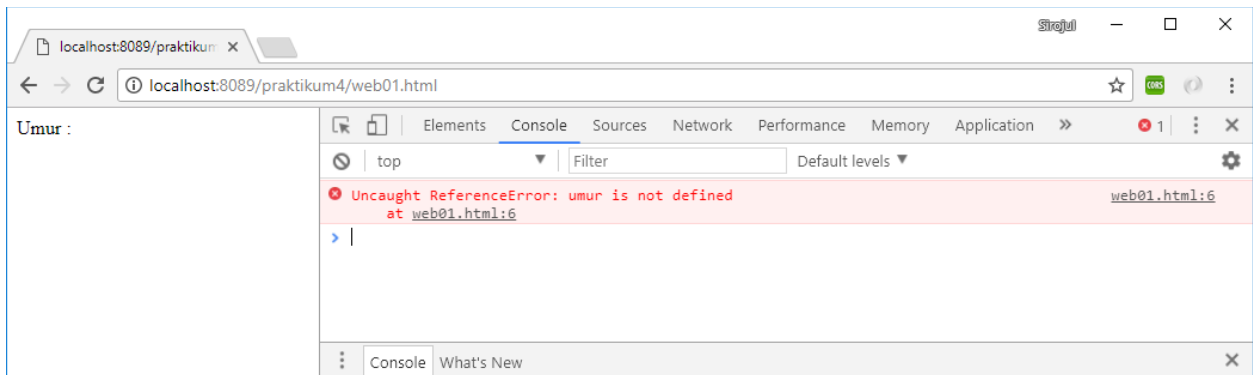
3. Developer Console

Kode program yang dibuat rentan terjadi kesalahan (error), membuat error dalam program adalah sangat manusiawi, anda bukan robot. Error bukan untuk ditakuti dan tidak untuk dihindari, namun dapat menjadi teman bagi programmer untuk mengetahui kesalahan yang terjadi pada program untuk kemudian segera diperbaiki dan disempurnakan.

Secara default browser yang digunakan tidak menampilkan error, sehingga ketika terjadi sesuatu yang tidak semestinya pada program maka kita tidak dapat melihat dan membetulkannya.

Untuk melihat error dan informasi atas yang terjadi pada kode program, browser memiliki fitur tertanam bernama “developer tools”. Disarankan bagi programmer javascript menggunakan browser chrome atau firefox karena memiliki fitur terbaik dari “developer tools”.

Short cut untuk menampilkan ke window developer tools pada browser chrome dengan tekan Ctrl-Shift-I



Gambar 1.1 Developer Console

Pada gambar 1.1 terjadi kesalahan kode program dimana variable umur belum didefinisikan pada baris ke 6 file web01.html.

Perintah `console.log()` (“cetak ke console”);

dapat digunakan untuk cetak ke console sebagai log debugging dari program.

Pada window developer console anda juga bisa dapat menguji pernyataan program javascript. Detail penggunaan developer console dapat diakses halaman web ini: <https://javascript.info/debugging-chrome>

4. Variable dan Tipe Data

3.1 Values

Pendefinisian nilai string menggunakan single-quote ‘pemrograman web’ atau bisa juga menggunakan double-quote “pemrograman web”. Bilangan didefinisikan dengan atau tanpa desimal, seperti 3.14.

3.2 Variable

Variable didefinisikan dengan keyword `var` atau `let`, modern javascript menggunakan `let`, berikut contohnya:

```

1      ...
2      var tinggi = 161; // deklarasi bilangan bulat
3      let berat = 60.20; // deklarasi bilangan desimal
4      let lulus = TRUE; // deklarasi variable boolean
5      ...

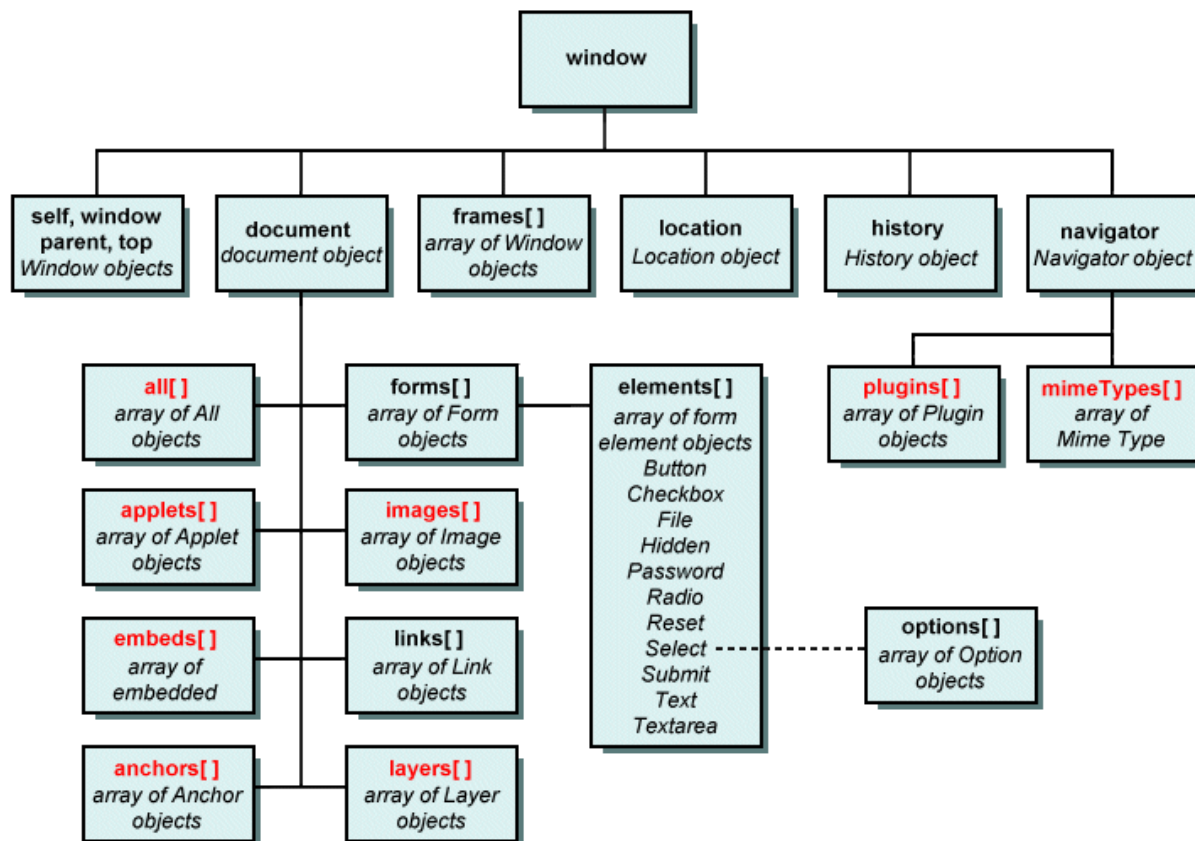
```

Pada pendefinisian variable JavaScript mempunyai ciri **loosely typed** (dynamic language) yaitu variable dideklarasikan tidak perlu ditentukan jenis tipe datanya (head variable), JavaScript akan secara otomatis membaca jenis data berdasarkan standard yang ada.

5. Document Object Model

Halaman dokumen HTML adalah sebuah object document yang merepresentasikan dokumen HTML yang di-load pada saat itu. Didalam object document terdapat object-object lainnya seperti images, forms, links, frame dan area-area lainnya.

Berikut ini adalah gambar yang menjelaskan hirarki Object pada dokumen HTML:



Gambar 1.2 Document Object Model (DOM)

3.3 Memberi ID pada object

Object pada dokumen HTML dapat diberi inisialisasi dengan nama identifikasi atau ID tertentu, misalnya suatu area diberi inisialisasi ID dengan nama “nurulfikri” dalam area tag atau tag <div>.

```

1 <span id="nurulfikri" style="font-family:Arial;color:blue">
2   NurulFikri Computer
3 </span>

```

Atau sebuah kumpulan elemen-elemen pada object form diberi inisilasi dengan nama ID "frm_cari", seperti kode berikut :

```

1 <form id="frm_cari" method="GET">
2   Cari Siswa:
3     <input type="text" name="nama_cari" value="" size="30"/>
4     <input type="button" value="Go" name="proses_cari"/>
5 </form>
6 Nama Yang Anda Cari : <span id="katacari"></span>

```

3.4 Mendefinisikan variable object

Dari area atau object yang telah diberi inisialasi ID dapat dibuat variable yang mewakili object tersebut dengan menggunakan method `document.getElementById()`.

Berikut contoh potongan kode JavaScript yang mendefinisikan variabel object pada dokumen HTML berdasarkan nama identifikasi (ID) sesuai dengan kode HTML sebelumnya.

```

1 <script language="JavaScript">
2 ----
3 let obj1 = document.getElementById("nurulfikri");
4
5 let frm = document.getElementById("frm_cari");
6 alert("Kata Yang Anda Cari : " + frm.nama_cari.value;
7
8 ----
9 </script>

```

Pada kode diatas obj1 adalah variable yang mewakili object dari area `` dan object frm yang mewakili koleksi object dari form.

Dalam object form terdapat elemen-elemen didalamnya seperti element input text, button, select yang memiliki name masing-masing. Pada contoh diatas untuk mengambil nilai text pada elemen form text : `<input type="text" name="nama_cari" value="" />` dituliskan dengan kode berikut: `frm.nama_cari.value;`

6. JavaScript Function

Fungsi dalam javascript didefinisikan dengan sintaks berikut ini:

```

1 function nama_fungsi(var1,var2,...,varX)
2 {
3   Statement program
4 }

```

Berikut contoh mendefinisikan fungsi dalam javascript:

```
1 ...
2 <script type="text/javascript">
3   function salam()
4   {
5       alert("Assalamualaikum Saudaraku !");
6   }
7   function salamKenal(teman){
8       alert("Apa kabar " + teman);
9   }
10  function jumlah(a,b){
11      let c = a + b;
12      Return c;
13  }
14 </script>
15 ...
```

7. Event Handler

Event handlers suatu karakteristik dari OOP yang dimiliki JavaScript yang sangat berguna dalam menciptakan web lebih interaktif. Event Handlers sebenarnya adalah suatu fungsi built-in yang akan menjalankan kode program tertentu atau fungsi tertentu (di definisikan oleh programmer) yang diakibatkan suatu aksi (event) dari seorang user terhadap object halaman web anda.

Jika user atau anda menekan (meng-klick) suatu tombol (button) maka terjadilah suatu Click-event. Jika pointer mouse bergerak melewati atau mengarah pada posisi suatu link teks maka akan terjadi MouseOver-event. Event Handlers yang menangani event tersebut dalam JavaScript adalah onClick dan onMouseOver.

Berikut ini beberapa daftar event handlers built-in dalam JavaScript:

Event Handlers	Deskripsi	Objek
onClick	Reaksi terhadap aksi meng-click-mouse pada suatu objek	checkboxes, links, radio buttons, reset buttons, dan submit buttons
onMouseOver	Reaksi jika suatu cursor atau mousepointer yang menunjuk ke suatu objek	Links
onMouseOut	Reaksi jika suatu cursor atau mousepointer yang meninggalkan suatu objek	Links
onLoad	Reaksi jika suatu objek selesai di-load.	images, windows
onUnload	Reaksi jika suatu dokumen ditutup/diakhiri.	windows
onAbort	Reaksi jika suatu objek diberhentikan dari proses loading	images
onChange	Reaksi jika suatu nilai(value) dari objek dimodifikasi	file uploads, select objects, text boxes, textarea
onSelect	Reaksi jika suatu teks dipilih dari objek text box atau textarea	text boxes, text areas
onError	Reaksi jika terjadi error JavaScript	images, windows
onReset	Reaksi jika suatu tombol reset dalam form ditekan	forms
onSubmit	Reaksi jika suatu tombol button dalam form ditekan	forms

Jika anda ingin program JavaScript anda bereaksi/bekerja terhadap suatu aksi (event) tertentu. Hal ini dapat dilakukan dengan bantuan dari event-handlers. Sebuah tombol(button) bisa menciptakan sebuah popup window jika ditekan (di-click).

Hal ini berarti akan menyebabkan sebuah pop up window muncul sebagai reaksi terhadap Click-event. Event-handler yang diperlukab untuk maksud tersebut adalah onClick. onClick ini akan memberitahukan komputer untuk melakukan sesuatu jika event tersebut terjadi.

Berikut ini ditunjukkan kode program contoh suatu event-handler onClick, dalam hal ini event-handlers akan didefinisikan dalam suatu tag HTML :

```

1 <form>
2 <input type="button" value="Click Aku" onClick="alert('Halo Kawan !')">
3 <input type="button" value="Go" onClick="salam()">
4 </form>

```

8. Skup Variable

8.1 Variable lokal

Variable yang didefinisikan dalam fungsi dan hanya dapat diakses oleh kode didalam fungsi tersebut, variable bersifat lokal.

```

function fungsiSatu(){
    let namasiswa = "Ahmad Rizky";
}

```

```
// hanya kode dalam fungsi ini yang dapat akses variable namasiswa  
}
```

8.2 Variable global

Variable yang didefinisikan DI LUAR fungsi bersifat global dan dapat diakses oleh kode lain dalam file JavaScript tersebut.

```
let namasiswa = "Ahmad Rizky";  
function fungsiSatu(){  
    // kode di dalam fungsi ini dapat meng-akses variable namasiswa  
}
```

8.3 Variable global automatically

Value yang di-assign pada variable yang belum dideklarasikan otomatis menjadi global variable walaupun variable tersebut berada di dalam suatu fungsi.

```
functisatu();// fungsi dipanggil terlebih dahulu  
console.log(namasiswa); // cetak variable ke log  
function functisatu(){  
    namasiswa = "Ahmad Zaki"; //  
}
```

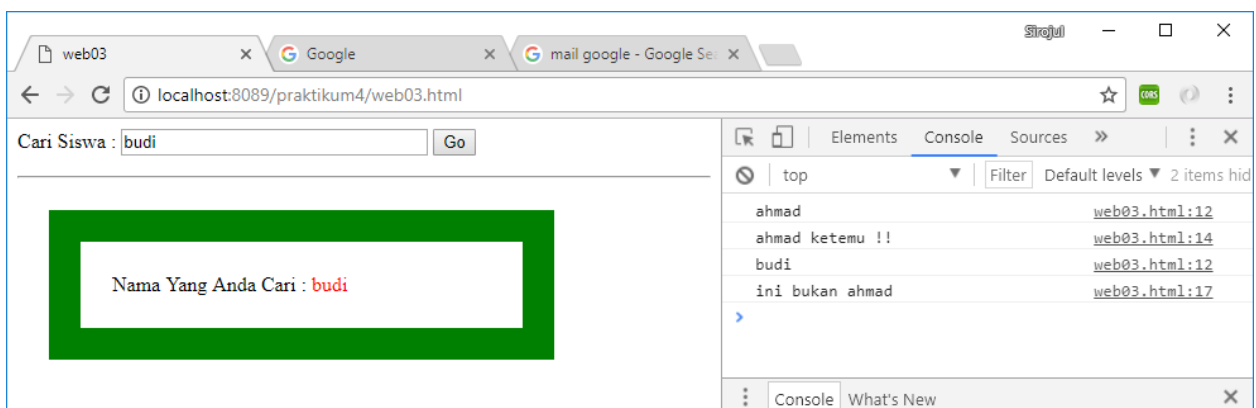
Latihan Praktikum

1. Program Cari Teman (cariteman.html)

```

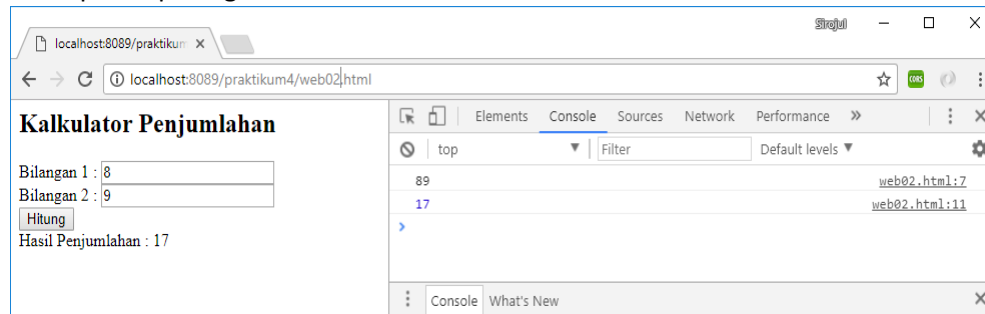
1  <html>
2  <head>
3      <title>web03</title>
4      <script language="JavaScript">
5          function cariTeman(){
6              let frm = document.getElementById("frm_cari");
7              let str = frm.nama_cari.value;
8              alert("Kata Cari : " + frm.nama_cari.value);
9              document.getElementById("katacari").innerHTML= str;
10             console.log(str);
11             if(str == "ahmad"){
12                 console.log("ahmad ketemu !!");
13                 document.getElementById("katacari").style="color:blue";
14             }else{
15                 console.log("ini bukan ahmad ");
16                 document.getElementById("katacari").style="color:red";
17             }
18         }
19     </script>
20 </head>
21 <body>
22     <form id="frm_cari">
23         Cari Siswa :
24         <input type="text" name="nama_cari" value="" size="30"/>
25         <input type="button" value="Go" name="proses_cari" onClick="cariTeman()"/>
26     </form>
27     <hr/>
28     <div style=" width: 300px;border: 25px solid green;padding: 25px;margin: 25px;">
29         Nama Yang Anda Cari : <span id="katacari"></span>
30     </div>
31 </body>
32 </html>

```



2. Program Kalkulator penjumlahan sederhana (kalkulator1.html)

Berikut ini contoh operasi aritmatika program kalkulator penjumlahan sederhana, seperti yang ditampilkan pada gambar 1.2.



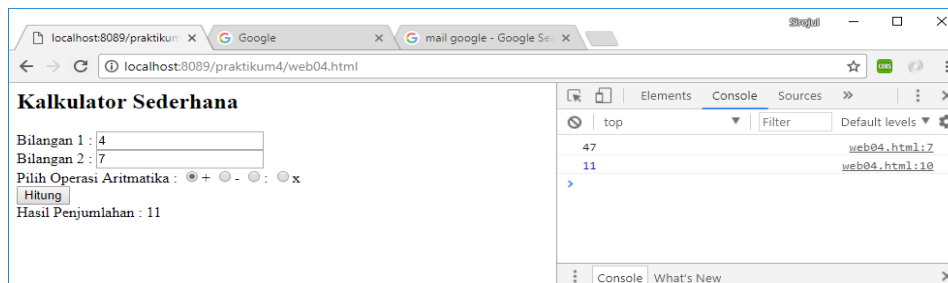
Gambar 1.2 Kalkulator Penjumlahan

```

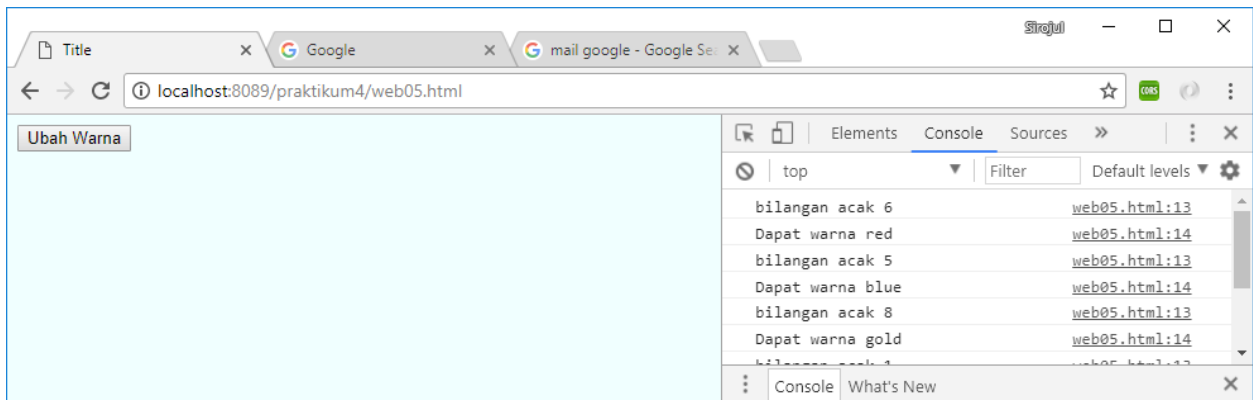
1  <html>
2  <head>
3      <script language="JavaScript">
4          function jumlahkan() {
5              let strbill = document.getElementById("bill").value; // ambil nilai id=bill
6              let strbil2 = document.getElementById("bil2").value; // ambil nilai id=bil2
7              console.log(strbill + strbil2); // cetak ke log console hasil strbill + strbil2
8              let bill = parseInt(strbill); // casting string ke integer simpan ke bill
9              let bil2 = parseInt(strbil2); // casting string ke integer simpan ke bil2
10             console.log(bill + bil2); // cetak ke log console hasil bill + bil2
11             let bilhasil = bill + bil2; // jumlahkan bill + bil2 dan simpan ke bilhasil
12
13             document.getElementById("hasil").innerHTML = bilhasil; // cetak bilhasil di area id=hasil
14         }
15     </script>
16 </head>
17 <body>
18     <h2>Kalkulator Penjumlahan</h2>
19     <form>
20         Bilangan 1 : <input type="text" id="bill" />
21         <br/> Bilangan 2 : <input type="text" id="bil2" />
22         <br/>
23         <input type="button" value="Hitung" onclick="jumlahkan();" />
24         <br/> Hasil Penjumlahan : <span id="hasil"></span>
25     </form>
26 </body>
27 </html>

```

Modifikasi program diatas sehingga kalkulator bisa mensupport operasi dasar aritmatika, beri nama file kalkulator2.html, penampakan tampilan program kalkulator versi 2 seperti gambar dibawah ini:



3. Program Ubah Warna



```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6      <script language="JavaScript">
7          function ubahWarna() {
8              let acak = Math.round(Math.random() * 10);
9              let warnas = ["beige", "cyan", "yellow", "pink", "green", "blue",
10                 "red", "azure", "gold", "gray", "purple"];
11              console.log("bilangan acak " + acak);
12              console.log("Dapat warna " + warnas[acak]);
13
14              document.getElementById("id1").style="background-color:" + warnas[acak];
15          }
16      </script>
17  </head>
18  <body id="id1">
19      <form>
20          <input type="button" value="Ubah Warna" onclick="ubahWarna()" />
21      </form>
22  </body>
23  </html>

```

Praktikum Mandiri JavaScript

DATELINE: 31 Desember 2021, 23:55

1. Login Authentication (10%) : login.html

Pada praktikum ini anda diminta membuat sistem login menggunakan JavaScript, user aplikasi akan memasukkan username dan password yang sudah anda tentukan dalam form login misal :

username=ahmad2017 , password=integrity

Terlebih dahulu anda diminta membuat form login, setelah tombol login di click maka kode javascript anda akan melakukan pengecekan apakah sesuai dengan username dan password yang telah ditentukan, munculkan javascript alert jika login sukses maupun gagal. Jika login sukses arahkan ke halaman baru dan munculkan text login berhasil <h1>Login Sukses</h1>. Jika gagal login maka web tidak pindah halaman.

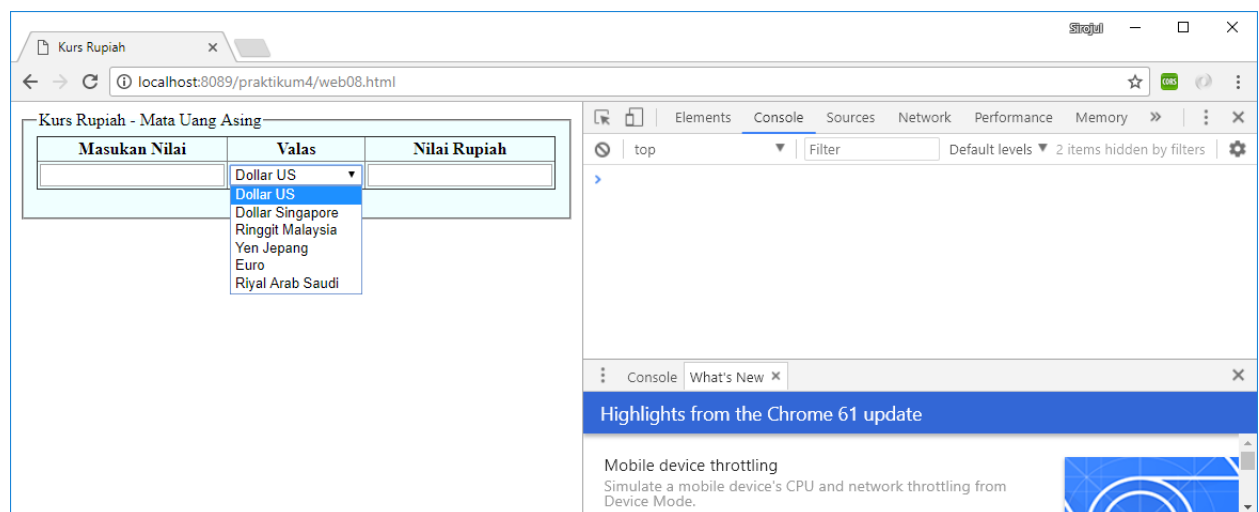
Kode JavaScript disimpan dalam file **otentikasi.js** dan digunakan di file **login.html**.

2. Kurs Rupiah (15%)

Pada praktikum kurs rupiah ini anda diminta untuk membuat konversi matauang dari matauang asing ke rupiah, user jika ingin menghitung konversi harus memasukkan dahulu nilai matauang asing untuk kemudian dikonversi sesuai dengan pilihan valas, jika dropdown telah dipilih otomatis nilai rupiah akan muncul di kotak nilai rupiah.

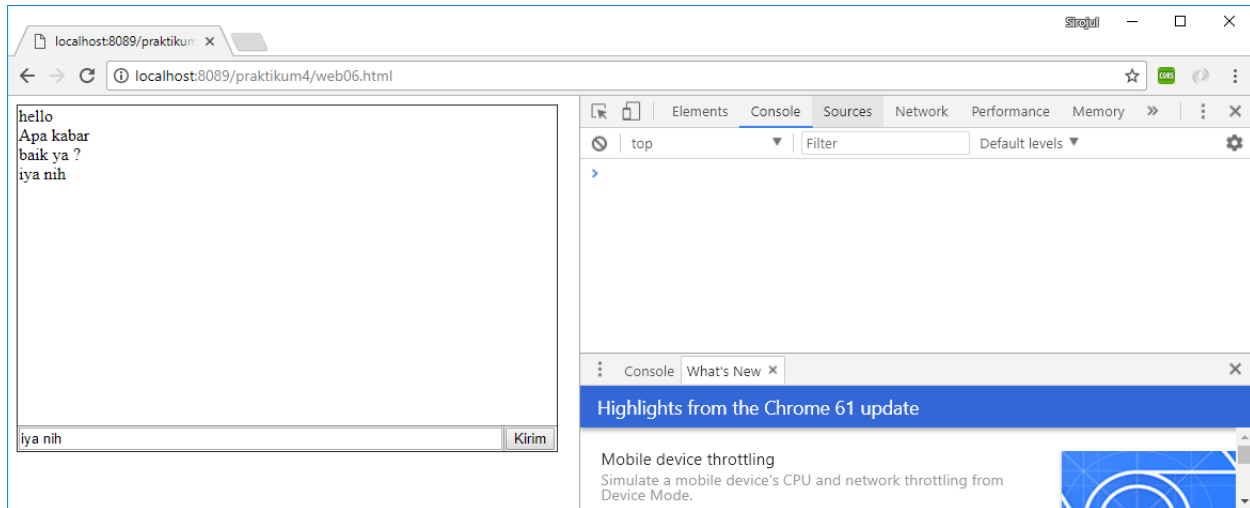
Untuk kesamaan data valas berikut nilai valas matauang asing terhadap rupiah

- 1 Dollar US ---- 9.915 rupiah
- 1 Dollar Singapore ---- 13.472 rupiah
- 1 Ringgit Malaysia ---- 874 rupiah
- 1 Yen Jepang ---- 120 rupiah
- 1 Euro ---- 15.888 rupiah
- 1 Riyal Arab Saudi ---- 3.592 rupiah



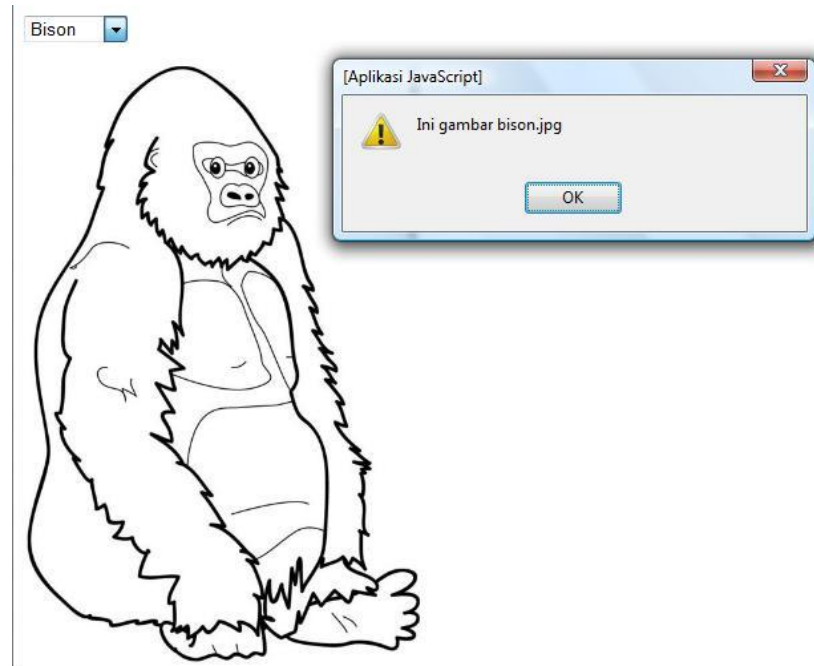
3. Chat Message (25%)

Pada praktikum chat message ini anda diminta membuat aplikasi chatbox sederhana. Chatbox diletakan di pojok kanan bawah halaman web anda walaupun halaman di-scroll. Chatbox seperti pada gambar dibawah ini terdiri atas kotak yang menampilkan pesan dan kotak untuk menuliskan pesan, tombol kirim digunakan untuk mengirimkan pesan untuk ditampilkan dalam kotak pesan. Chatbox anda tidak perlu menyimpan data hasil chat (jika browser di close data hilang).



4. Image Picker (15%)

Pada praktikum ini anda diminta untuk menampilkan gambar sesuai dengan data yang ada pada dropdown, jika drop down dipilih maka akan menampilkan gambar pada area dibawah dropdown dan juga kotak alert untuk informasi nama file gambar. Jumlah gambar harus lebih dari 5 gambar.



5. Form Validation (35%)

Pada praktikum ini anda diminta membuat form validasi dengan menggunakan JavaScript. Berikut data pada komponen form yang anda kirim untuk di validasi.

- Nama Pelanggan : (required validation, maksimum 30 karakter)
- Email: (required validation, email format)
- Jam Keberangkatan : (required validation, jam dengan format jam antara 00.00 – 23.59)
- Tujuan Keberangkatan : (required validation)
- Jumlah Tiket : (required validation, bilangan bulat antara 1 – 10)

Ketika tombol submit di tekan, JavaScript akan melakukan validasi, jika terjadi kesalahan inputan yang tidak sesuai dengan format data yang ditentukan maka halaman web akan menampilkan pesan kesalahan (error) disamping field berwarna merah bahwa data tidak valid, jika data form valid tampilkan keseluruhan data nama pelanggan, email, jam keberangkatan, tujuan dan jumlah tiket dibawah form.

Aturan Pengerjaan

- a. Hasil pekerjaan di **commit ke server gitlab** anda sesuai dengan project praktikum 2 ini
- b. Dilarang keras melakukan plagiarisme