

MODUL 6

JAVASCRIPT

A. TUJUAN PRAKTIKUM

- a. Mahasiswa memahami struktur javascript
- b. Mahasiswa mampu mengimplementasikan javascript dalam pembuatan web

B. ALOKASI WAKTU 1 x 50 menit

C. DASAR TEORI

1. Javascript

Pada perkembangan dunia website sekarang ini javascript sudah menjadi bahasa pemrograman yang sangat dibutuhkan. banyak website-website yang menggunakan javascript untuk membuat modal alert, pemberitahuan, form validation, efek animasi, ajax, game dan sebagainya.

Contoh salah satu website yang menggunakan javascript untuk membuat website mereka lebih interaktif atau lebih hidup adalah facebook, twittter, google dan lainnya. pada halaman facebook terdapat kotak chatting yang dapat anda buka dan tutup. form chat dapat terbuka dan tertutup jika di klik karena menggunakan bantuan javascript. kemudian lagi jika anda pernah memesan atau memeriksa ongkos tiket pesawat melalui situs-situs penyedia layanan tiket pesawat, setelah anda memilih kota sekarang dan memilih kota tujuan anda maka akan muncul maskapai pesawat yang tersedia tanpa me-reload halaman. hal ini terjadi karena menggunakan teknik ajax yang di buat dengan javascript.



JavaScript adalah bahasa pemrograman yang sangat matang dan dapat dikolaborasikan dengan dokumen HTML dan digunakan untuk membuat website yang interaktif. JavaScript diciptakan oleh Brendan Eich yang juga co-founder dari Mozilla project, Mozilla Foundation and Mozilla Corporation.

Anda dapat melakukan banyak hal dengan JavaScript. Anda akan memulai dari fitur sederhana seperti menentukan layout, membuat respon ketika mengklik button, carousels, dan gallery gambar. Namun pada akhirnya ketika anda sudah mendapat banyak pengetahuan anda juga akan dapat membuat game, animasi 2D dan 3D, aplikasi yang berhubungan dengan database, dan masih banyak lagi.

JavaScript sendiri adalah bahasa yang cukup kompleks namun sangat fleksibel, dan banyak Developer (Programmer) telah menyediakan tool yang berdiri di atas core JavaScript agar anda dapat menggunakan fungsi-fungsi ekstra, tool tersebut sebagai berikut:

- Application Programming Interfaces (APIs) dibangun pada web browser agar memungkinkan anda melakukan apapun dari dinamik dokumen HTML dan set CSS yang anda buat, untuk menangkap dan memodifikasi video dari web cam, atau membuat animasi 3D dan sampel audio.
- API pihak ketiga menyediakan akses bagi Developer untuk menghubungkan aplikasi mereka pada website atau aplikasi lain layaknya facebook dan twitter. Pernahkah anda login soundcloud dengan facebook? itu API pihak ketiga.
- Frameworks pihak ketiga dan libraries dapat digabungkan pada HTML sehingga memungkinkan Developer membangun website atau aplikasi dengan cepat

Untuk mempelajari javascript lebih lanjut silakan buka w3schools.com atau developer.mozilla.org/en-US/docs/Learn/JavaScript.

2. Penulisan Javascript

JavaScript termasuk jenis bahasa script, yang digunakan di dalam file HTML. Untuk menginput, atau memasukkan kode JavaScript ke dalam HTML, JavaScript menyediakan 4 alternatif, yaitu:

a. Menggunakan tag <script> (internal JavaScript)

Cara pertama untuk menginput kode JavaScript ke dalam halaman HTML adalah dengan menggunakan tag <script> secara internal. Internal disini berarti bahwa kode JavaScript ditulis pada halaman yang sama dengan HTML, atau di dalam satu file HTML.

Cara ini merupakan cara yang paling sering digunakan, jika kode JavaScript tidak begitu panjang, dan hanya digunakan di 1 halaman saja. Kode JavaScript yang akan diinput diletakkan diantara tag pembuka <script> dan tag penutup </script> seperti berikut ini:

```
<script>  
//kode javascript diletakkan disini  
</script>
```

Tag <script> akan memberitahu web browser bahwa kode diantara tag tersebut bukanlah HTML, tetapi JavaScript.

```
1  <!DOCTYPE html>
2  <html>
3  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
4  <head>
5  <title>Belajar JavaScript di DuniaIlkom</title>
6
7  <script>
8  alert("Hello World!!");
9  </script>
10 </head>
11
12 <body>
13 <h1>Belajar JavaScript</h1>
14 <p>Saya sedang belajar JavaScript di duniaIlkom.com</p>
15 <p>Belajar Web Programming di DuniaIlkom.</p>
16 </body>
17 </html>
```

b. Menggunakan tag <script scr=""> (external JavaScript)

Cara atau metode kedua untuk menginput kode JavaScript ke dalam halaman HTML adalah dengan memindahkan kode JavaScript ke dalam sebuah file terpisah, lalu ‘memanggilnya’ dari HTML. Cara ini sangat disarankan karena akan memberikan banyak keuntungan dan fleksibilitas dalam membuat program JavaScript.

Sebuah file JavaScript disimpan dalam ekstensi .js, seperti: kode.js, register.js, atau kodeku.js. Dari halaman HTML, kita memanggilnya menggunakan tag <script> dengan atribut src. Atribut src berisi alamat dari file javascript tersebut, seperti berikut ini:

```
<script src="kode_javascript.js"></script>
```

Perhatikan bahwa tag <script> tetap ditutup dengan tag penutup </script>, atau anda bisa membuatnya menjadi self closing tag seperti berikut ini:

```
<script src="kode_javascript.js" />
```

```
1 <!DOCTYPE html>
2 <html>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
4 <head>
5 <title>Belajar JavaScript di DuniaIlkom</title>
6
7 <script src="kode_javascript.js"></script>
8
9 </head>
10
11 <body>
12 <h1>Belajar JavaScript</h1>
13 <p>Saya sedang belajar JavaScript di duniaIlkom.com</p>
14 <p>Belajar Web Programming di DuniaIlkom.</p>
15 </body>
16 </html>
```

c. Menggunakan Event Handler (Inline JavaScript)

Cara ketiga untuk menjalankan JavaScript adalah dengan memanggilnya menggunakan Event Handler dari dalam tag HTML. Konsep Event Handler adalah pemanggilan kode javascript ketika ‘sesuatu’ terjadi dalam tag HTML.

Sesuatu disini maksudnya ketika sebuah element dalam HTML di klik, di klik kanan, di arahkan mouse, dan lain-lain. Event handler di dalam JavaScript ditulis dengan penambahan kata on. Sehingga jika sebuah tombol di-klik, maka disebut sebagai onclick, jika mouse berada diatas element disebut sebagai onmouseover, dan lain-lain.

```
1 <!DOCTYPE html>
2 <html>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
4 <head>
5 <title>Belajar JavaScript di DuniaIlkom</title>
6 </head>
7
8 <body>
9 <h1>Belajar JavaScript</h1>
10 <p>Saya sedang belajar JavaScript di duniaIlkom.com</p>
11 <p>Belajar Web Programming di DuniaIlkom.</p>
12
13 <button onclick="alert('Hello World!!!')>Klik Saya
14
15 </body>
16 </html>
```

d. Menggunakan URL ([href:="javascript"](#))

Cara terakhir (dan juga paling jarang digunakan saat ini) adalah dengan menyisipkan JavaScript ke dalam alamat href dari tag HTML. Cara ini disebut juga dengan protocol javascript. Disebut demikian, karena kita mengganti alamat link dari yang biasa menggunakan protocol http//: menjadi javascript:

```
1 <!DOCTYPE html>
2 <html>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
4 <head>
5 <title>Belajar JavaScript di DuniaIlkom</title>
6 </head>
7
8 <body>
9 <h1>Belajar JavaScript</h1>
10 <p>Saya sedang belajar JavaScript di duniaIlkom.com</p>
11 <p>Belajar Web Programming di DuniaIlkom.</p>
12
13 <a href="javascript:alert('Hello World!!')">Hallo Dunia...</a>
14
15 </body>
16 </html>
```

3. Variabel dan Tipe Data

Variabel adalah tempat dimana kita menyimpan nilai-nilai atau informasi-informasi pada JavaScript. Variabel yang dideklarasikan dapat diisi dengan nilai apa saja. Dalam JavaScript pendeklarasian sebuah variabel sifatnya opsional, artinya Anda boleh mendeklarasikan atau tidak hal tersebut tidak menjadi masalah. Jika Anda memberi nilai pada variabel, maka dalam JavaScript dianggap bahwa Anda telah mendeklarasikan variabel tersebut.

Penulisan variabel di dalam JavaScript dapat dilakukan dalam beberapa cara:

```
var nama = "Vearen Dika Sofirudin";
let nilai = 85.5;
const phi = 3.14;
Alamat = "Jl. Raya Pemogan"
```

Tidak seperti bahasa pemrograman lainnya, JavaScript tidak memiliki tipe data secara explisit. Hal ini dapat dilihat dari beberapa contoh variabel diatas. Anda mendeklarasikan variabel tapi tidak menentukan tipenya.

Meskipun JavaScript tidak memiliki tipe data secara explisit. JavaScript mempunyai tipe data implisit. Terdapat empat macam tipe data implisit yang dimiliki oleh JavaScript yaitu :

- Numerik, seperti : 0222532531, 1000, 45, 3.146789 dsb
- String, seperti : “Hallo”, “April”, “Jl. Setiabudi No 17A”, “Cece Kirani” dsb
- Boolean, bernilai true atau false
- Null, variabel yang tidak diinisialisasi
- Object

4. Fungsi Output

Output adalah sebuah tampilan program yang biasanya digunakan untuk memperlihatkan hasil akhir. Pada Bahasa pemrograman pada umumnya output ditampilkan dalam bentuk teks dengan fungsi print().

Ada 4 cara menampilkan output pada Javascript:

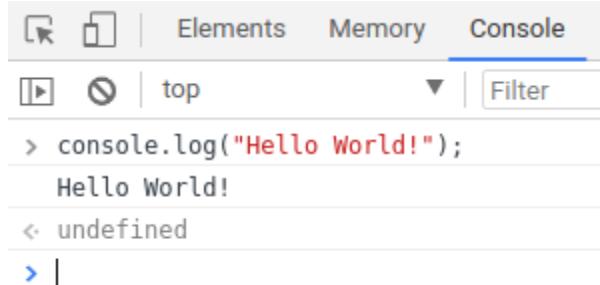
a. Menggunakan Fungsi console.log();

Fungsi console.log() adalah fungsi untuk menampilkan teks ke console Javascript. Fungsi console.log() biasanya digunakan untuk *debugging*. Karena setiap pesan error di Javascript selalu ditampilkan di dalam Console.

Contoh penggunaan:

```
console.log("Hello World!");
```

Akan menghasilkan



b. Menggunakan Fungsi alert();

Fungsi alert() adalah fungsi untuk menampilkan jendela dialog. Fungsi sebenarnya berada pada objek window. Fungsi alert(), hanya bisa digunakan di dalam browser saja.

Secara lengkap bisa ditulis seperti ini:

```
window.alert("Hello World!");
```

Bisa juga ditulis alert() saja seperti ini:

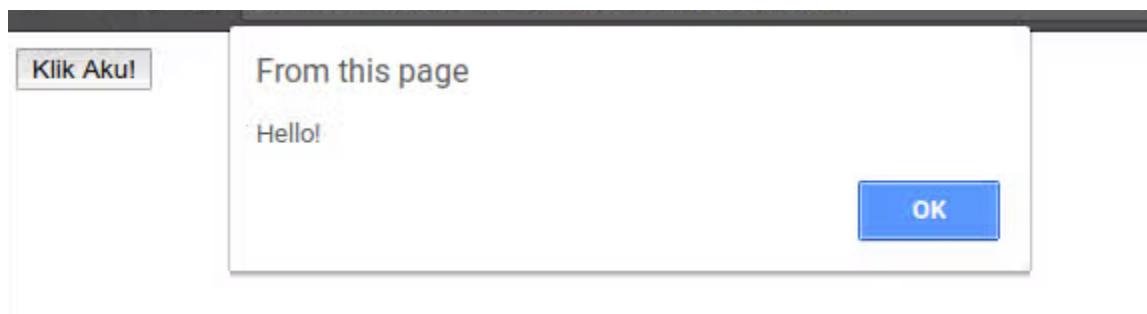
```
alert("Hello World!");
```

Contoh penggunaan fungsi ‘alert()’ :

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Belajar Javascript</title>
    <script>
        alert("Selamat datang di tutorial belajar Javascript");

        function sayHello(){
            alert("Hello!");
        }
    </script>
</head>
<body>
    <button onclick="sayHello()">Klik Aku!</button>
</body>
</html>
```

Hasilnya



c. Menggunakan Fungsi document.write();

Objek document adalah objek yang mewakili dokumen HTML di dalam Javascript. Dalam objek document, terdapat fungsi write() untuk menulis sesuatu ke dokumen HTML. Selain fungsi write(), objek document juga menyediakan berbagai macam fungsi untuk manipulasi dokumen HTML.

Contoh

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Belajar Javascript</title>
    <script>
        document.write("<h1>Hello World!</h1>");
        document.write("<hr>");
        document.write("<p>Saya sedang belajar Javascript</p>");
        document.write("di <b>PTIK UNS</b>")
    </script>
</head>
<body>

</body>
</html>
```

Hasilnya

Hello World!

Saya sedang belajar Javascript

di PTIK UNS

d. Menggunakan innerHTML.

innerHTML adalah sebuah atribut di dalam (objek) elemen HTML yang berisi string HTML. Dengan innerHTML, kita dapat menampilkan output ke elemen yang lebih spesifik.

Contoh:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Belajar Javascript</title>
</head>
<body>

    <h1>Tutorial Javascript untuk Pemula</h1>
    <div id="hasil-output"></div>

    <script>
        // membuat objek elemen
        var hasil = document.getElementById("hasil-output");

        // menampilkan output ke elemen hasil
        hasil.innerHTML = "<p>Aku suka Javascript, mungkin....</p>";
    </script>

</body>
</html>
```

Hasilnya:

Tutorial Javascript untuk Pemula

Aku suka Javascript, mungkin....

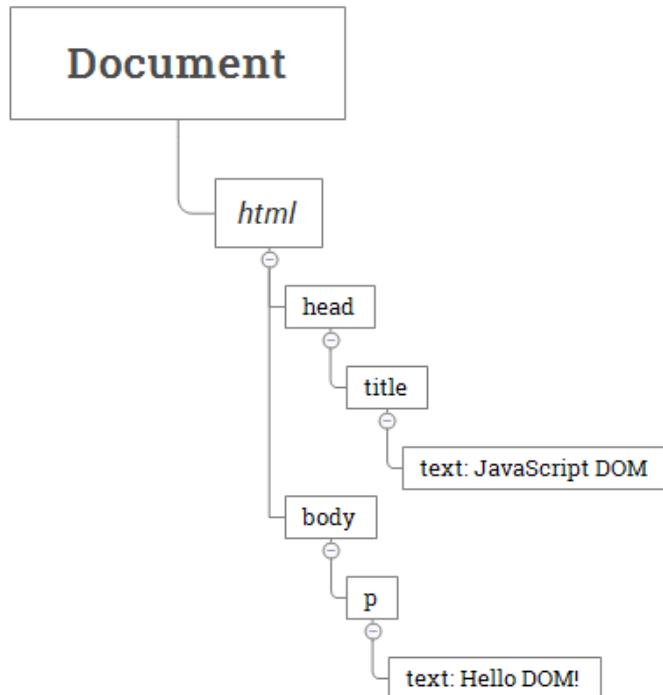
5. Document Object Model

Document Object Model (DOM) adalah antarmuka pemrograman aplikasi (API) untuk memanipulasi dokumen HTML. DOM mewakili dokumen HTML sebagai pohon simpul. DOM menyediakan fungsi yang memungkinkan Anda untuk menambah, menghapus, dan memodifikasi bagian dokumen secara efektif. Perhatikan bahwa DOM adalah cara lintas platform dan bahasa-independen untuk memanipulasi dokumen HTML dan XML.

DOM mewakili dokumen HTML sebagai hierarki node. Perhatikan dokumen HTML berikut:

```
<!DOCTYPE html>
<html>
  <head>
    <title>JavaScript DOM</title>
  </head>
  <body>
    <p>Hello DOM!</p>
  </body>
</html>
```

Pohon berikut mewakili dokumen HTML di atas:



Di pohon DOM ini, dokumen disebut sebagai root node. Root node memiliki satu node anak yaitu adalah elemen <html>. Elemen <html> disebut root element. Root element ini kemudian memiliki element <head> dan element <body> dimana masing-masing node ini akan memiliki lebih banyak element lainnya di dalamnya.

Dengan model objek, JavaScript mendapatkan semua kekuatan yang dibutuhkan untuk membuat HTML dinamis:

- JavaScript dapat mengubah semua elemen HTML di halaman
- JavaScript dapat mengubah semua atribut HTML di halaman
- JavaScript dapat mengubah semua gaya CSS di halaman
- JavaScript dapat menghapus elemen dan atribut HTML yang ada
- JavaScript dapat menambahkan elemen dan atribut HTML baru
- JavaScript dapat bereaksi terhadap semua event HTML yang ada di halaman
- JavaScript dapat membuat event HTML baru di halaman

Berikut contoh penulisan DOM untuk membuat elemen baru (node):

```
<!DOCTYPE html>
<html>
  <head>
    <title>JavaScript HTML DOM</title>
  </head>
  <body>
    <div id="div1">
      <p id="p1">Ini paragraf 1.</p>
      <p id="p2">Ini paragraf 2.</p>
    </div>

    <script>
      const para = document.createElement("p");
      const node = document.createTextNode("Ini paragraf baru.");
      para.appendChild(node);
      const element = document.getElementById("div1");
      element.appendChild(para);
    </script>
  </body>
</html>
```

Hasilnya :

Ini paragraf 1.

Ini paragraf 2.

Ini paragraf baru.

D. PRAKTIKUM

a. Persiapan File

1. Salin terlebih dahulu file .zip yang sudah diberikan pada praktikum ini
2. Ekstrak file ke dalam folder dengan nama javascript1

Name	Date modified	Type	Size
images	04-Apr-19 21:30	File folder	
styles	04-Apr-19 21:30	File folder	
CODE_OF_CONDUCT.md	04-Apr-19 21:30	MD File	1 KB
index.html	04-Apr-19 21:30	Firefox HTML Doc...	2 KB
LICENSE	04-Apr-19 21:30	File	7 KB
README.md	04-Apr-19 21:30	MD File	1 KB

b. Mengubah teks

1. Pertama-tama buatlah sebuah file baru bernama **main.js**. Simpan di dalam folder **scripts**.
2. Selanjutnya, buka file index.html Anda, dan masukkan elemen berikut pada sebuah baris **sebelum tag penutup </body>**:

```
1 | <script src="scripts/main.js"></script>
```

Ini sama halnya dengan cara kerja elemen <link> untuk CSS — ini menempatkan JavaScript ke halaman, sehingga dapat memberikan pengaruh pada HTML, CSS, dan lainnya pada halaman.

3. Tambahkan kode berikut pada file **main.js**:

```
1 | var myHeading = document.querySelector('h1');
2 | myHeading.innerHTML = 'Hello world!';
```

4. Sekarang pastikan file HTML dan JavaScript disimpan, dan muat index.html di browser. Anda seharusnya mendapatkan hasil seperti berikut:



c. Menambahkan pengubah gambar

Pada bagian ini kita akan menambahkan gambar lain pada situs kita, dan tambahkan beberapa Javascript sederhana untuk mengubah antara keduanya ketika gambar di-klik.

1. Pertama, cari gambar lain yang ingin anda gunakan di situs anda. Pastikan ukurannya sama dengan gambar pertama, atau kurang lebih sama.
2. Simpan gambar pada folder images anda.
3. Pada file main.js , tuliskan kode JavaScript berikut (Jika kode Javascript hello world masih disana, hapus saja):

```
1 | var myImage = document.querySelector('img');
2 |
3 | myImage.onclick = function() {
4 |   var mySrc = myImage.getAttribute('src');
5 |   if(mySrc === 'images/firefox-icon.png') {
6 |     myImage.setAttribute ('src','images/firefox2.png');
7 |   } else {
8 |     myImage.setAttribute ('src','images/firefox-icon.png');
9 |   }
10 | }
```

4. Simpan semua file dan muat index.html di browser. Sekarang kita klik gambarnya, seharusnya berubah tampilannya menjadi gambar yang satunya!

Kita menyimpan referensi ke elemen gambar Kita dalam variabel **myImage**. Selanjutnya, Kita membuat fungsi tanpa nama (fungsi "anomim") pada property variabel pengendali event onclick. Sekarang, setiap kali elemen gambar ini diklik:

1. Kita mendapatkan nilai atribut src dari gambar.
2. Kita menggunakan kondisional untuk memeriksa apakah nilai src sama dengan path gambar asli:
 - a. Jika ya, kita mengubah nilai src ke path gambar kedua, memaksa gambar lain dimuat dalam elemen <image>.
 - b. Jika tidak (artinya harus sudah mengalami perubahan), kita mengubah nilai src kembali ke jalur gambar asli, untuk membalikkannya kembali ke semula.

Jadi, intinya, apabila dimuat gambar1, jika gambar1 diklik maka akan berubah menjadi gambar2, begitu juga sebaliknya.

d. Menambahkan pesan selamat datang yang dinamis

Selanjutnya, Kita akan menambahkan sedikit kode lagi untuk mengubah judul halaman untuk menyertakan pesan pembuka yang dinamis saat pengguna pertama kali masuk ke situs. Pesan pembuka ini akan bertahan saat pengguna meninggalkan situs dan kemudian kembali lagi. Kita juga akan menyertakan opsi untuk mengubah pengguna dan oleh karena itu pesan selamat datang diperlukan setiap waktu.

1. Pada **index.html**, Tambahkan baris berikut **sebelum** elemen <script>:

```
1 | <button>Change user</button>
```

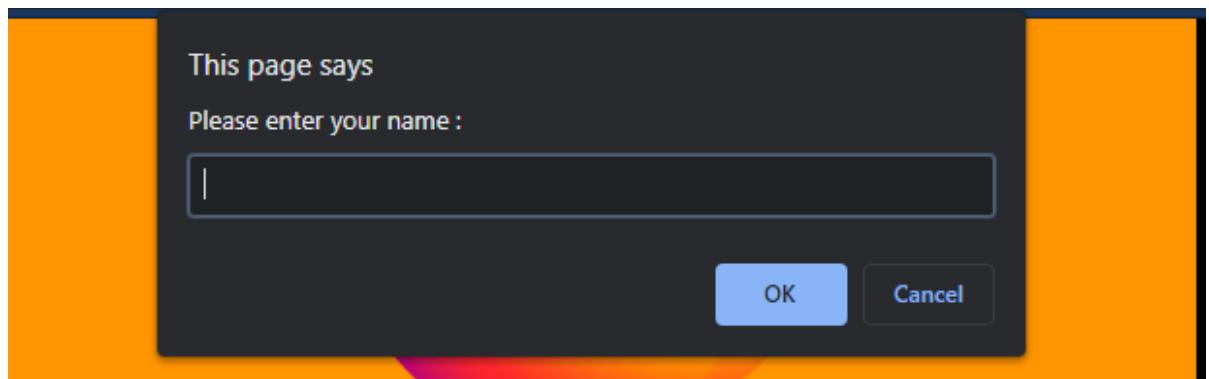
2. Pada **main.js**, tambahkan kode berikut di bagian bawah file, persis seperti yang tertulis - ini mengacu pada tombol baru yang akan kita tambahkan, judulnya, dan simpan pada variabel:

```
1 | var myButton = document.querySelector('button');
2 | var myHeading = document.querySelector('h1');
```

3. Sekarang tambahkan fungsi berikut untuk mengatur ucapan yang dinamis - ini belum berinteraksi tapi nanti akan kita gunakan:

```
1 | function setUsername() {  
2 |     var myName = prompt('Please enter your name.');//  
3 |     localStorage.setItem('name', myName);  
4 |     myHeading.innerHTML = 'Mozilla is cool, ' + myName;  
5 | }
```

Fungsi ini berisi fungsi **prompt()** yang menampilkan **kotak dialog** seperti **alert()**. Perbedaannya adalah **prompt()** meminta pengguna untuk memasukkan data, dan menyimpan data tersebut dalam variabel saat tombol dialog OK ditekan.



Dalam kasus ini, kita meminta pengguna memasukkan namanya. Selanjutnya, kita memanggil **API** yang disebut **localStorage**, yang memungkinkan kita menyimpan data di browser, dan mengambilnya nanti. Kita menggunakan fungsi `setItem` `localStorage()` untuk membuat dan menyimpan item data yang disebut 'name', dan menetapkan nilainya ke variabel `myName` yang berisi nama pengguna yang dimasukkan. Akhirnya, kita mengatur `innerHTML` dari judul ke string, ditambah nama penggunanya.

4. Selanjutnya, tambahkan blok if ... else — kita bisa memanggil kode inisialisasi, sebagai pengatur saat aplikasi pertama kali dimuat:

```
1 | if(!localStorage.getItem('name')) {  
2 |     setUsername();  
3 | } else {  
4 |     var storedName = localStorage.getItem('name');  
5 |     myHeading.innerHTML = 'Mozilla is cool, ' + storedName;  
6 | }
```

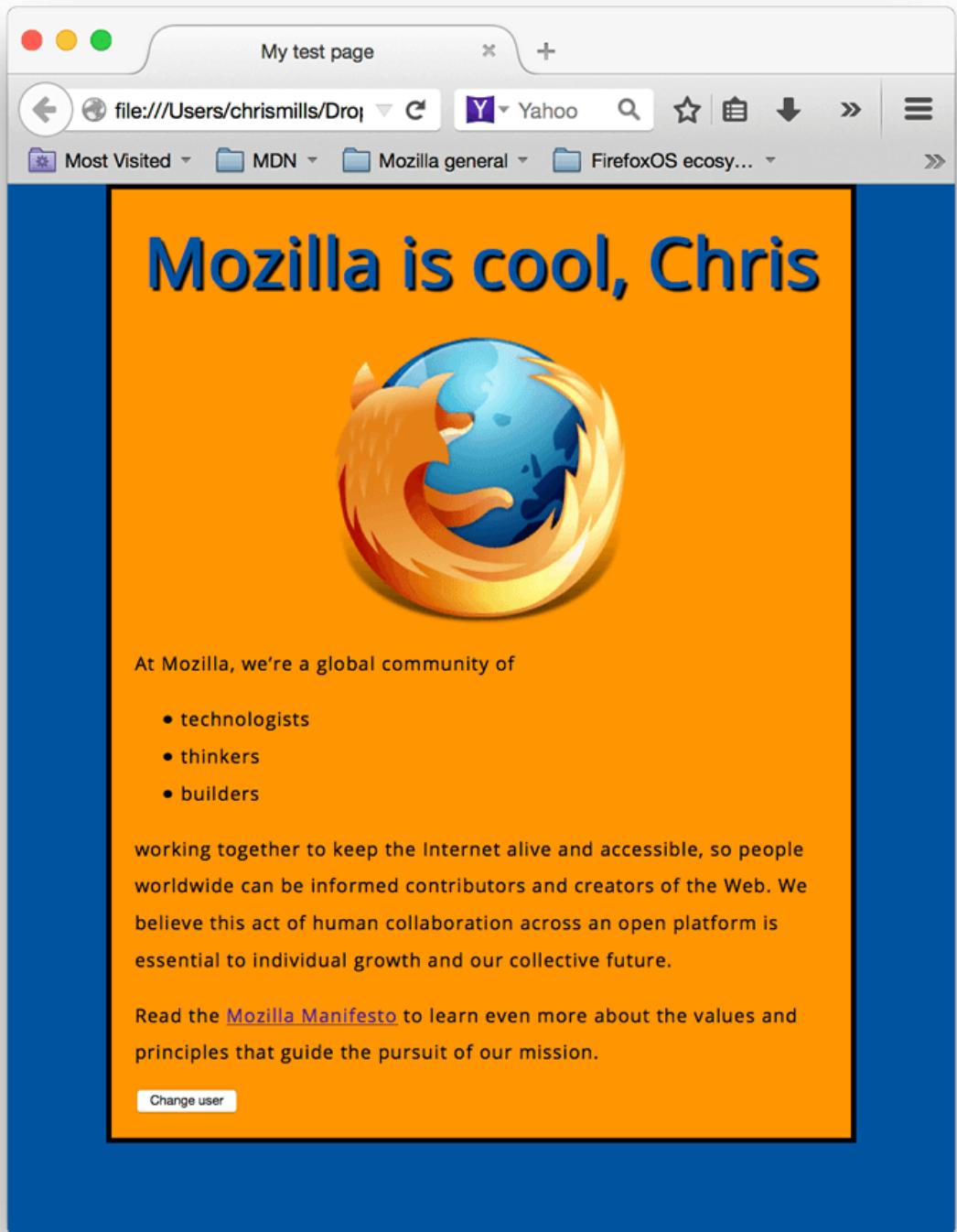
Blok kode pertama kali menggunakan operator negasi (logical NOT) untuk memeriksa apakah item data ada - jika tidak ada, fungsi setUsername() dijalankan untuk membuatnya. Jika sudah ada (misalnya pengguna mengaturnya saat mereka mengunjungi situs), kita mengambil nama yang tersimpan dengan menggunakan getItem() dan menetapkan innerHTML dari judul ke string, ditambah nama pengguna, sama seperti yang kita lakukan di dalam setUsername().

5. Terakhir, letakkan pengendali event di bawah onclick pada tombolnya, sehingga saat diklik fungsi setUsername() dijalankan. Hal ini memungkinkan pengguna untuk mengatur nama baru kapan pun mereka mau dengan menekan tombol:

```
1 | myButton.onclick = function() {  
2 |     setUsername();  
3 | }
```

*) myButton adalah variable dari elemen **button** yang telah didefinisikan sebelumnya

6. Sekarang saat pertama kali mengunjungi situs, akan meminta nama pengguna kemudian memberi pesan yang dinamis kepada anda. Kemudian Anda bisa mengganti nama kapan saja dengan menekan tombol. Sebagai bonus tambahan, karena namanya tersimpan di dalam localStorage, tetap ada setelah situs ditutup, jadi pesan yang dinamis akan tetap ada saat Anda membuka situs ini lagi!



Simpan 1 file tersebut dengan nama **javascript.html**.

E. LATIHAN

1. Buatlah sebuah **kalkulator sederhana** yang dapat melakukan perhitungan dua buah variabel.
 - Jumlah
 - Kurang
 - Kali
 - Bagi

Simpan dengan nama latihan1.HTML

2. Diketahui sebuah halaman web profil PTIK mengandung tulisan berikut

PTIK FKIP UNS

The Faculty of Teacher Training and Education, Sebelas Maret University, Surakarta is an Educational Personnel Education Institution (LPTK) which has 24 study programs in 6 majors. Each study program has its own characteristics in producing superior, strong and intelligent educational staff.

The Informatics and Computer Technology Education (PTIK) study program is planned to take shelter in the management of the Engineering and Vocational Education (PTK) department. This placement revises our previous statement, where the PTIK study program is under the P.MIPA department. PTIK is a study program that will produce graduates who are prioritized to teach in vocational/vocational programs. So, it would be better if PTIK was under the PTK department which oversees vocational programs such as Building Engineering Education, and Mechanical Engineering Education. To meet the needs of lecturers in the field of information technology that cannot be met by the PTK department, PTIK will conduct resource sharing with the mathematics study program.

Buatlah halaman yang dapat **menampilkan paragraph dan sebuah tombol**. Apabila tombol tersebut ditekan maka dapat **melakukan highlight tulisan tertentu** di dalam paragraph sehingga tampilan yang awalnya seperti ini:

The Faculty of Teacher Training and Education, Sebelas Maret University, Surakarta is an Educational Personnel Education Institution (LPTK) which has 24 study programs in 6 majors. Each study program has its own characteristics in producing superior, strong and intelligent educational staff.

The Informatics and Computer Technology Education (PTIK) study program is planned to take shelter in the management of the Engineering and Vocational Education (PTK) department. This placement revises our previous statement, where the PTIK study program is under the PMIPA department. PTIK is a study program that will produce graduates who are prioritized to teach in vocational/vocational programs. So, it would be better if PTIK was under the PTK department which oversees vocational programs such as Building Engineering Education, and Mechanical Engineering Education. To meet the needs of lecturers in the field of information technology that cannot be met by the PTK department, PTIK will conduct resource sharing with the mathematics study program.

Berubah menjadi :

The **Faculty of Teacher Training and Education**, **Sebelas Maret University**, Surakarta is an **Educational Personnel Education Institution (LPTK)** which has 24 study programs in 6 majors. Each study program has its own characteristics in producing **superior, strong and intelligent educational staff**.

The **Informatics and Computer Technology Education (PTIK)** study program is planned to take shelter in the management of the **Engineering and Vocational Education (PTK)** department. This placement revises our previous statement, where the PTIK study program is under the PMIPA department. PTIK is a study program that will produce graduates who are **prioritized to teach in vocational/vocational programs**. So, it would be better if PTIK was under the PTK department which oversees vocational programs such as **Building Engineering Education**, and **Mechanical Engineering Education**. To meet the needs of lecturers in the field of information technology that cannot be met by the PTK department, PTIK will conduct **resource sharing with the mathematics study program**.

Warna berubah merah, background kuning dan teks tebal. Jika ditekan kembali, maka highlight akan menghilang.

Simpan halaman web dengan nama **latihan2.html**