



**LEMBAR ASISTENSI**  
**PRAKTIKUM PEMROGRAMAN WEB**  
**LABORATORIUM TEKNIK KOMPUER**  
**JURUSAN TEKNIK ELEKTRO FAKULTAS TEKNIK**  
**UNIVERSITAS LAMPUNG**

---

Judul Praktikum : Javascript Basic & Javascript DOM

Praktikan (NPM) : Muhammad Fitra Yudha (2115061015)

Asisten (NPM) : Asha Imalia (2015061006)

Dwindy Monica (2015061022)

No	Catatan	Tanggal	Paraf

Bandar Lampung,

2023

.....  
NPM.

## I. JUDUL

### JAVASCRIPT BASIC & JAVASCRIPT DOM

## II. TUJUAN PERCOBAAN

Adapun tujuan dari percobaan ini adalah sebagai berikut :

1. Memahami bagaimana logika komputer diimplementasikan pada dokumen web
2. Dapat memahami dan menggunakan JavaScript untuk melakukan pemrograman logika
3. Memahami DOM pada HTML
4. Dapat memanipulasi DOM pada HTML menggunakan JavaScript

## III. TEORI DASAR

### 3.1 Pengertian Javascript

JavaScript adalah bahasa pemrograman dinamis komputer yang digunakan untuk menambah interaktivitas di halaman web. Ringan dan umumnya digunakan sebagai bagian dari halaman web yang implementasinya memungkinkan skrip pada sisi klien untuk berinteraksi dengan pengguna dan membuat halaman dinamis. Merupakan bahasa pemrograman dinamis dengan kemampuan berorientasi obyek.

### 3.2 Client-Side Javascript

Client-side JavaScript adalah bentuk yang paling umum dari penggunaan JavaScript. Skrip harus disematkan atau direferensi oleh dokumen HTML agar kode dapat diinterpretasi oleh browser. Artinya, halaman web tidak harus statis, tapi dapat menyertakan program yang berinteraksi dengan pengguna, mengendalikan browser, dan dengan dinamis membuat konten HTML. Mekanisme client-side JavaScript menyediakan banyak keuntungan dibandingkan server-side script CGI

tradisional. Misalnya, JavaScript dapat digunakan untuk mengetahui apakah pengguna telah memasukkan email yang benar pada form. Kode JavaScript dieksekusi saat pengguna mengirim form, dan hanya jika semua masukan telah valid, baru kemudian dikirim ke web server. JavaScript dapat digunakan untuk menangkap event seperti tombol diklik, navigasi link dan aksi lain yang diinisiasi pengguna baik eksplisit atau pun implisit.

### 3.3 HTML DOM

Ketika sebuah halaman web dimuat, browser membuat Document Object Model (DOM) dari halaman web tersebut. Setiap halaman web berada di dalam browser window yang dapat dianggap sebagai sebuah object. Document Object merepresentasikan dokumen HTML yang ditampilkan di dalam sebuah window. Document Object memiliki berbagai properties (atribut) yang berelasi ke obyek lain untuk membolehkan akses dan modifikasi dari konten dokumen. Cara bagaimana sebuah konten dokumen diakses dan dimodifikasi disebut dengan Document Object Model atau DOM. Obyek disusun dalam sebuah hierarki. Struktur hierarki ini berlaku pada organisasi obyek dalam sebuah halaman web.

Window object – Hierarki paling atas. Merupakan elemen paling luar dari hierarki obyek.

- **Document object** – Setiap dokumen HTML yang dimuat ke browser window menjadi document object. Dokumen mengandung isi dari halaman.
- **Form object** – Semua yang diletakkan di antara tag `<form> ... </form>` membentuk form object.
- **Form control elements** – Form object mengandung semua elemen yang didefinisikan untuk obyek tersebut seperti text field, button, radio button, and checkbox.

### 3.4 Javascript & DOM

Menggunakan DOM, Javascript mendapatkan semua kemungkinan untuk

membuat halaman HTML yang dinamis:

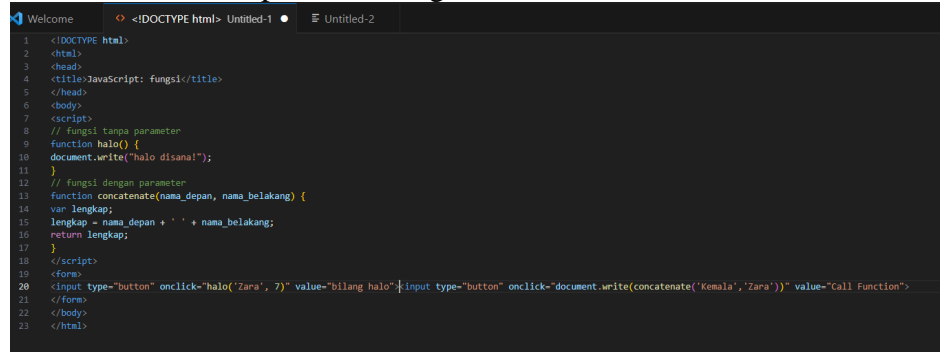
- Kita dapat menggunakan JavaScript untuk menambahkan elemen HTML ke halaman web, seperti menambahkan elemen baru, tombol, gambar, atau teks ke halaman. Kita juga dapat menghapus elemen yang ada jika diperlukan
- JavaScript memungkinkan kita untuk mengubah teks, gambar, atau konten lain dari elemen HTML secara dinamis. Ini memungkinkan pembuatan efek seperti pembaruan teks secara real-time atau animasi yang merespons input pengguna.
- Kita dapat mengubah atribut-atribut elemen HTML, seperti mengubah nilai atribut `src` untuk gambar, `href` untuk tautan, dan lainnya. Ini berguna untuk membuat galery gambar, mengubah tautan, dan lain sebagainya.
- Kita dapat mengubah gaya elemen HTML menggunakan JavaScript, seperti mengubah warna latar belakang, ukuran teks, posisi elemen, dan lainnya. Ini memungkinkan pembuatan animasi dan efek visual yang dinamis.
- Kita dapat menggunakan JavaScript untuk memvalidasi input dalam formulir sebelum mengirimkannya ke server. Ini membantu memastikan bahwa data yang dimasukkan oleh pengguna sesuai dengan aturan yang ditetapkan.
- JavaScript memungkinkan kita untuk mengontrol navigasi pengguna di halaman web. Kita dapat mengarahkan pengguna ke halaman lain, mengganti isi halaman tanpa perlu memuat ulang seluruh halaman, dan sebagainya.
- JavaScript memungkinkan kita untuk mengambil data dari server dan menampilkannya di halaman tanpa memuat ulang halaman. Ini memungkinkan pembuatan aplikasi web yang responsif dan interaktif.
- Kita dapat menggunakan JavaScript untuk membuat animasi dan transisi yang halus. Misalnya, Kita dapat membuat slide show, menu tarik-turun, atau efek hover dengan bantuan JavaScript.
- Kita dapat mengubah URL di peramban pengguna menggunakan JavaScript. Ini berguna untuk mengimplementasikan tautan dalam satu halaman (single-page application) atau mengatur tautan terkait dengan navigasi dinamis.

- Kita dapat menangani peristiwa seperti klik tombol, input pengguna, dan peristiwa lainnya dengan JavaScript. Ini memungkinkan reaksi yang dinamis terhadap interaksi pengguna

## IV. PROSEDUR PERCOBAAN

Adapun prosedur percobaan sebagai berikut :

### 4.1 Percobaan 1 : Javascript Basic, Fungsi



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>JavaScript: fungsi</title>
5 </head>
6 <body>
7 <script>
8 // fungsi tanpa parameter
9 function halo() {
10 document.write("halo disana!");
11 }
12 // fungsi dengan parameter
13 function concatenate(nama_depan, nama_belakang) {
14 var lengkap;
15 lengkap = nama_depan + ' ' + nama_belakang;
16 return lengkap;
17 }
18 </script>
19 <form>
20 <input type="button" onclick="halo('Zara', 7)" value="bilang halo"><input type="button" onclick="document.write(concatenate('Kemala','Zara'))" value="Call Function">
21 </form>
22 </body>
23 </html>
```

Gambar 4.1 Percobaan 1 : Javascript Basic, Fungsi

### 4.2 Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>JavaScript: redirect, timeout, & print</title>
5 </head>
6 <body>
7 <script>
8 function Redirect() {
9 window.location = "http://meizano.github.io";
10 }
11 alert("Anda akan diarahkan ke halaman lain dalam 10 detik.");setTimeout('Redirect()', 10000);
12 </script>
13 <form>
14 <input type="button" value="Print" onclick="window.print()" />
15 </form>
16 </body>
17 </html>
```

Gambar 4.2 Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print

#### 4.3 Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

```
1 <!doctype html>
2 <html>
3 <head>
4   <title>Mengakses Elemen DOM</title>
5 </head>
6 <body>
7   <h1>Mengakses Elemen</h1>
8   <p>Elemen yang diakses adalah yang memiliki kata pada script.</p>
9   <p>Modul ini akan dibuat menjadi sebuah buku.</p>
10  <script>
11    var elemen = "belum ketemu";
12    function mencariString(node, string) {
13      if (node.nodeType == Node.ELEMENT_NODE) {
14        for (let i = 0; i < node.childNodes.length; i++) {
15          if (mencariString(node.childNodes[i], string)) {
16            elemen = node.childNodes[i];
17            return true;
18          }
19        }
20        return false;
21      } else if (node.nodeType == Node.TEXT_NODE) {
22        return node.nodeValue.indexOf(string) > -1;
23      }
24    }
25    console.log(elemen);
26    console.log(mencariString(document.body, "buku"));
27    console.log(elemen);
28    // → true
29  </script>
30 </body>
31 </html>
```

Gambar 4.3 Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

#### 4.4 Percobaan 4 : Javascript DOM, Membuat Node

```
1 <!doctype html>
2 <html>
3 <head>
4   <title>Membuat Node</title>
5 </head>
6 <body>
7   <p>Cari sendiri gambar kupu-kupu dan bunganya.</p>
8   <p> di dalam
9   .</p>
10  <p><button onclick="replaceImages()">Ganti</button></p>
11  <script>
12    function replaceImages() {
13      let images = document.body.getElementsByTagName("img");
14      for (let i = images.length - 1; i >= 0; i--) {
15        let image = images[i];
16        if (image.alt) {
17          let text = document.createTextNode(image.alt);
18          image.parentNode.replaceChild(text, image);
19        }
20      }
21    }
22  </script>
23 </body>
24 </html>
```

Gambar 4.4 Percobaan 4 : Javascript DOM, Membuat Node

#### 4.5 Percobaan 5 : Javascript DOM, Animasi DOM

```
ed-3  <!doctype html> Untitled-4  <!doctype html> Untitled-
1  <!doctype html>
2  <html>
3  <head>
4  <title>Animasi DOM</title>
5  </head>
6  <body>
7  <p>Cari sendiri gambar kupu-kupunya.</p>
8  <p style="text-align: center">
9  
10 </p>
11 <script>
12 let kupu = document.querySelector("img");
13 let angle = Math.PI / 2;
14 function animate(time, lastTime) {
15   if (lastTime != null) {
16     angle += (time - lastTime) * 0.001;
17   }
18   kupu.style.top = (Math.sin(angle) * 20) + "px";
19   kupu.style.left = (Math.cos(angle) * 200) + "px";
20   requestAnimationFrame(newTime => animate(newTime, time));
21 }
22 requestAnimationFrame(animate);
23 </script>
24 </body>
25 </html>
```

Gambar 4.5 Percobaan 5 : Javascript DOM, Animasi DOM

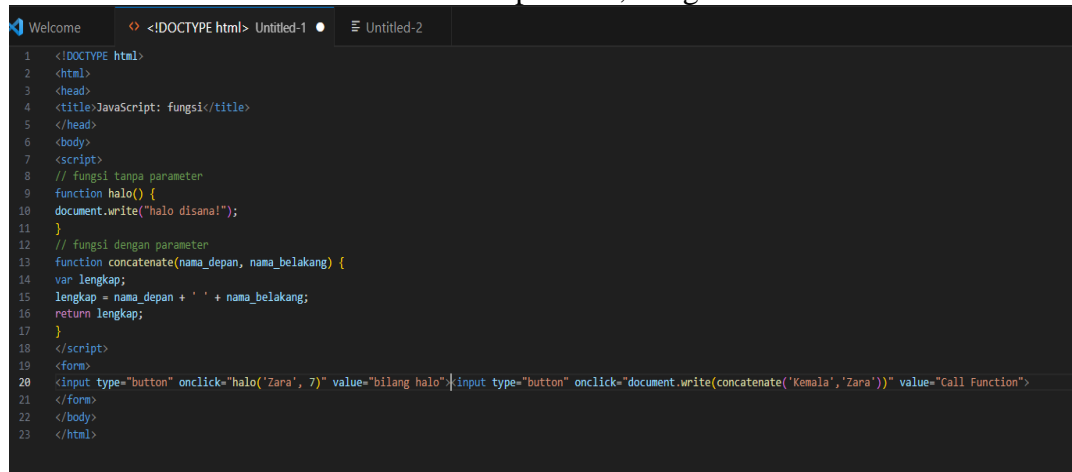


## V. PEMBAHASAN

Adapun pembahasan sebagai berikut :

### 5.1 Percobaan 1 : Javascript Basic, Fungsi

#### 5.1.a Source Code Percobaan 1 : Javascript Basic, Fungsi

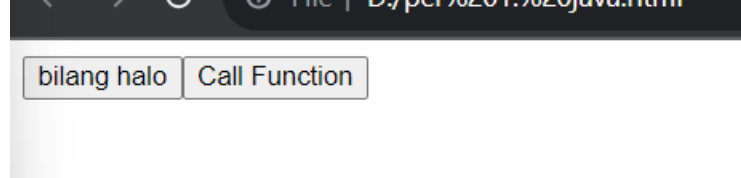


```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>JavaScript: fungsi</title>
5 </head>
6 <body>
7 <script>
8 // fungsi tanpa parameter
9 function halo() {
10 document.write("halo disana!");
11 }
12 // fungsi dengan parameter
13 function concatenate(nama_depan, nama_belakang) {
14 var lengkap;
15 lengkap = nama_depan + ' ' + nama_belakang;
16 return lengkap;
17 }
18 </script>
19 <form>
20 <input type="button" onclick="halo('Zara', 7)" value="bilang halo"><input type="button" onclick="document.write(concatenate('Kemala','Zara'))" value="Call Function">
21 </form>
22 </body>
23 </html>
```

Gambar 5.1.a Source Code Percobaan 1 : Javascript Basic, Fungsi

Berdasarkan Gambar 5.1.a Source code, fungsi pada baris pertama memiliki `<!DOCTYPE html>` untuk mengeksekusi kode program sebagai HTML dan memiliki header yang akan menampilkan judul. Pada baris ke 9 terdapat fungsition berisi `halo` yang akan mencetak `halo disana`, dengan fungsi parameter yang mempunyai fungsi `join` berisi nama depan dan belakang, dengan variabel `lengkap` dan variabel tersebut akan berjalan jika lengkapnya sama dengan namanya ditambah spasi dikembalikan secara utuh dan diakhiri dengan simbol `script`, dimana pada baris ke 21 dan 22 terdapat semacam tombol, dengan elemen di klik `Halo Zahra` dengan nilai `halo` dan di bawahnya terdapat konflik antara penulisan dokumen dan panggilan ke nilai fungsi.

### 5.1.b Output Percobaan 1 : Javascript Basic, Fungsi

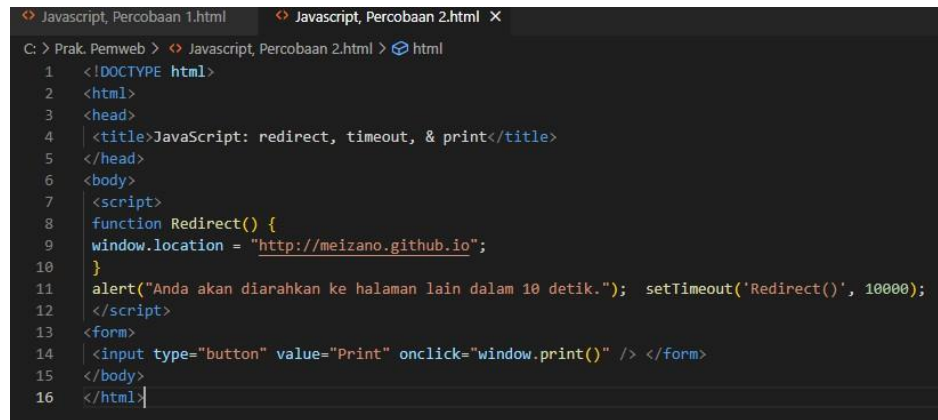


Gambar 5.1.b Output Percobaan 1 : Javascript Basic, Fungsi

Berdasarkan Gambar 5.1.b output yang diberikan dari source code di atas, yang memiliki teks "bilang halo" dan "call function" yang dapat diklik. Saat kita mengklik opsi "say hello", outputnya adalah "halo disana!" dan ketika saya klik pada fungsi panggilan, hasilnya adalah "Kemala Zara". Hal ini terjadi karena menjalankan baris 20 dari source code

## 5.2 Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print

### 5.2.a Source Code Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print

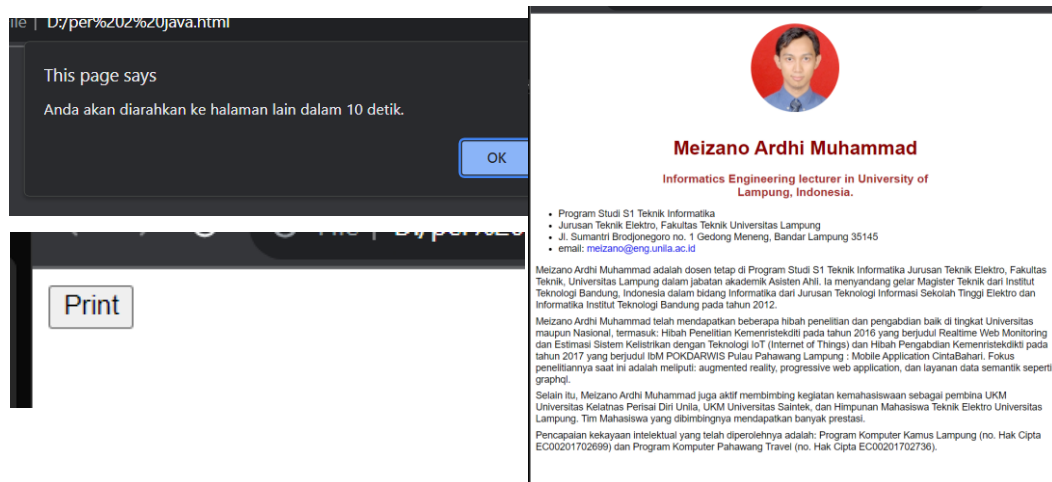


```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>JavaScript: redirect, timeout, & print</title>
5 </head>
6 <body>
7 <script>
8 function Redirect() {
9   window.location = "http://meizano.github.io";
10 }
11 alert("Anda akan diarahkan ke halaman lain dalam 10 detik."); setTimeout('Redirect()', 10000);
12 </script>
13 <form>
14 <input type="button" value="Print" onclick="window.print()" /> </form>
15 </body>
16 </html>
```

Gambar Source Code 5.2.a Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print

Berdasarkan Gambar Source Code 5.2.a Pada baris 1 dan 4 berfungsi untuk menjalankan kode program yaitu html dan memiliki header yang akan menampilkan judul di browser. Pada baris ke 8 terdapat fungsi redirect, dimana windows.location akan mengarahkan ke link website Pak Meizano, dengan peringatan yang berarti Anda akan diarahkan ke halaman lain setelah 10 detik dengan pengaturan waktu yang ditentukan, khususnya pengalihan dengan waktu 10000, dan pada baris ke 14 terdapat entry berupa tombol type yang nilainya tercetak atau ditampilkan di atas Windows

## 5.2.b Output Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print



Gambar 5.2.b Output Percobaan 2 : Javascript Basic, Redirect, Timeout, & Print Berdasarkan Gambar 5.2.b Output. Saat Anda menjalankannya, kita akan mendapatkan pesan "Anda akan diarahkan ke halaman lain dalam 10 detik", kemudian jika Anda klik OK Anda akan langsung mengakses menu cetak yang berasal dari baris ke-14 sc. Setelah 10 detik website akan dialihkan ke meizano.github.io yang berasal dari source baris 8 sampai 11

### 5.3 Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

#### 5.3.a Source Code Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

```
1  <!doctype html>
2  <html>
3  <head>
4  |   <title>Mengakses Elemen DOM</title>
5  </head>
6  <body>
7  <h1>Mengakses Elemen</h1>
8  <p>Elemen yang diakses adalah yang memiliki kata pada script.</p>
9  <p>Modul ini akan dibuat menjadi sebuah buku.</p>
10 <script>
11 var elemen = "belum ketemu";
12 function mencariString(node, string) {
13   if (node.nodeType == Node.ELEMENT_NODE) {
14     for (let i = 0; i < node.childNodes.length; i++) {
15       if (mencariString(node.childNodes[i], string)) {
16         elemen = node.childNodes[i];
17         return true;
18       }
19     }
20     return false;
21   } else if (node.nodeType == Node.TEXT_NODE) {
22     return node.nodeValue.indexOf(string) > -1;
23   }
24 }
25 console.log(elemen);
26 console.log(mencariString(document.body, "buku"));
27 console.log(elemen);
28 // → true
29 </script>
30 </body>
31 </html>
```

Gambar 5.3.a Source Code Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

Berdasarkan Gambar 5.3.a Source Code. Pada Baris pertama berisi `html>` yang digunakan untuk memberitahu browser bahwa dokumen ini adalah dokumen HTML.

Lalu pada baris ke 2 ada `<html>` yang digunakan untuk memulai dokumen HTML. Kemudian pada baris ke 3 terdapat `<head>` yang digunakan untuk mengawali head dari dokumen HTML. Kemudian pada baris ke 4 terdapat `<title>Mengakses Elemen DOM</title>` digunakan untuk menyatakan judul halaman web. Judul ini akan ditampilkan di tab browser. Lalu pada baris ke 5 terdapat `</head>` yang menutup judul dokumen HTML. Kemudian pada baris 6 terdapat `<body>` yang membuka badan dokumen HTML yang berisi konten halaman web. Kemudian pada baris 7 terdapat `<h1>Mengakses Elemen</h1>` yang mengidentifikasi elemen h1 dengan teks "Mengakses Elemen". Kemudian, pada baris 8, kita mengakses elemen yang berisi `script.</p>` digunakan untuk mengidentifikasi elemen p dengan teks "Elemen yang diakses adalah yang memiliki kata pada script.". Kemudian pada baris ke-9 terdapat Modul ini akan dibuat menjadi sebuah buku. yang berfungsi untuk mendefinisikan

elemen p dengan teks "Modul ini akan dibuat menjadi sebuah buku.".Kemudian pada baris ke 10 terdapat [removed] yang berfungsi untuk menandai awal dari kode JavaScript.Kemudian pada baris ke 11 terdapat var elemen = "belum ketemu"; yang berfungsi untuk mendeklarasikan variabel elemen dengan nilai awal "belum ketemu". Kemudian pada baris ke 12 terdapat function mencariString(node, string) { yang berfungsi untuk mendefinisikan fungsi JavaScript bernama mencariString().Fungsi ini menerima dua parameter: node dan string. Kemudian pada baris ke-13 terdapat if (node.nodeType == Node.ELEMENT\_NODE) { untuk memeriksa apakah node tersebut merupakan elemen atau bukan. Jika demikian, fungsi tersebut akan melakukan iterasi ke semua anak node.Kemudian pada baris ke-14 ada for (let i = 0; i < node.xss=removed> -1; yang mengembalikan nilai true jika string ada di teks node. Kemudian pada baris ke-25 ada console.log (element ); memungkinkan kita mencetak nilai variabel elemen ke konsol.Kemudian pada baris ke-26 ada console.log(mencariString (document.body, "buku")); digunakan untuk memanggil fungsi pencarian String() dengan parameter document.body dan "buku".Parameter document.body adalah elemen konten dokumen HTML.Parameter "buku" adalah kata yang akan dicari.Kemudian pada baris 27 ada console.log(element); digunakan untuk mencetak nilai variabel elemen ke konsol.Kemudian pada baris 30 terdapat </body> yang digunakan untuk menandai akhir badan dokumen HTML. Kemudian pada baris ke-31 terdapat </html> yang digunakan untuk menandai akhir dari dokumen HTML.

### 5.3.b Output Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

## Mengakses Elemen

Elemen yang diakses adalah yang memiliki kata pada script.

Modul ini akan dibuat menjadi sebuah buku.

Gambar 5.3.b Output Percobaan 3 : Javascript DOM, Mengakses Elemen DOM

Berdasarkan Gambar 5.3.b Output Terdapat judul yang ada pada bar website yaitu “Menjalankan Elemen DOM” yang merupakan hasil dari eksekusi kode baris ke 4. Lalu terdapat teks “Mengakses Elemen” pada header, lalu terdapat paragraf teks yang merupakan hasil dari eksekusi kode program baris ke 7 sampai baris ke 9. Jadi pada source code tersebut jika ada elemen yang mengandung kata "buku" dalam teksnya, maka output kedua akan menjadi true, dan output ketiga akan mencetak elemen DOM yang mengandung kata "buku." Jika tidak ada elemen yang mengandung kata "buku," maka kedua dan ketiga akan tetap "belum ketemu."

## 5.4 Percobaan 4 : Javascript, Membuat Node

### 5.4.a Source Code Percobaan 4 : Javascript, Membuat Node

```
D: > <> per 4 js.html > html > body > p > img
1  <!doctype html>
2  <html>
3  <head>
4  <title>Membuat Node</title>
5  </head>
6  <body>
7  <p>Cari sendiri gambar kupu-kupu dan bunganya.</p>
8  <p> di dalam
9  .</p>
10 <p><button onclick="replaceImages()">Ganti</button></p>
11 <script>
12 function replaceImages() {
13   let images = document.body.getElementsByTagName("img");
14   for (let i = images.length - 1; i >= 0; i--) {
15     let image = images[i];
16     if (image.alt) {
17       let text = document.createTextNode(image.alt);
18       image.parentNode.replaceChild(text, image);
19     }
20   }
21 }
22 </script>
23 </body>
24 </html>
```

Gambar 5.4.a Source Code Percobaan 4 : Javascript, Membuat Node

Berdasarkan Gambar 5.4.a Source Code pada baris pertama terdapat untuk menjalankan bahwa kode program tersebut merupakan html, dan terdapat title yang akan menampilkan sebuah judul Pada baris ke-7 terdapat paragraf yang di mana akan menampilkan cari sendiri gambar kupu-kupu dan bunga dan pada baris ke-8 terdapat paragraf yang di mana image image-nya yaitu sebuah gambar kupu-kupu dengan alt-nya kupu dan pada baris ke-9 dia mencari image-nya yaitu bunga dengan alt bunga dan pada baris ke-10 baris ke-10 terdapat paragraf button yang di mana on click replaceimage dia akan mengganti dengan fungsi replaceimage dan pada baris ke-14 terdapat let image di mana dia menampilkan gambar yaitu gambar kupu-kupu dan bunga.



#### 5.4.b Output Percobaan 4 : Javascript DOM, Membuat Node

---

Cari sendiri gambar kupu-kupu dan bunganya.



Cari sendiri gambar kupu-kupu dan bunganya.

kupu di dalam bunga.

Ganti

Gambar 5.4.b Output Percobaan 4 : Javascript DOM, Membuat Node

Berdasarkan Gambar 5.4 Output menampilkan gambar bunga yang berasal dari source code pada baris ke 8 dan 9 dengan posisi yang telah di atur. Lalu terdapat tombol ganti yang dapat diklik merupakan hasil dari source code baris ke 10. ketika kita mengklik tulisan ganti maka tampilannya akan berubah yang merupakan hasil dari eksekusi code pada baris ke 12 sampai baris ke 13.

## 5.5 Percobaan 5 : Javascript DOM, Animasi DOM

### 5.5.a Source Code Percobaan 5 : Javascript DOM, Animasi DOM

```
<> me 5.html > html > body > p > img
1  <!doctype html>
2  <html>
3  <head>
4  <title>Animasi DOM</title>
5  </head>
6  <body>
7  <p>Cari sendiri gambar kupu-kupunya.</p>
8  <p style="text-align: center">
9  
10 </p>
11 <script>
12 let kupu = document.querySelector("img");
13 let angle = Math.PI / 2;
14 function animate(time, lastTime) {
15   if (lastTime != null) {
16     angle += (time - lastTime) * 0.001;
17   }
18   kupu.style.top = (Math.sin(angle) * 20) + "px";
19   kupu.style.left = (Math.cos(angle) * 200) + "px";
20   requestAnimationFrame(newTime => animate(newTime, time));
21 }
22 requestAnimationFrame(animate);
23 </script>
24 </body>
25 </html>
```

Gambar 5.5.a Source Code Percobaan 5 : Javascript DOM, Animasi DOM

Berdasarkan Gambar 5.5.a Source Code pada baris pertama terdapat untuk menjalankan bahwa kode program tersebut merupakan html, dan terdapat title yang akan menampilkan sebuah judul, baris ke 7 terdapat paragraf yaitu cari sendiri gambar kupu-kupu dan pada baris ke 8 terdapat paragraf yang di mana style yaitu center atau berada di tengah dan terdapat image dia akan mencari gambar yang telah dimasukkan dan stylenya yaitu position dan pada baris ke 12 terdapat let kupu-kupu yang di mana terdapat dokumen dengan image dan pada baris ke 13 sampai baris ke 22 terdapat let Angle, terdapat function animate yaitu waktu dan last time dan di bawahnya terdapat if itu lifetime! = null, Dan terdapat style yang di mana dikali 20 +px dan di bawahnya yaitu dikali 200 + px, dan diakhiri dengan requestAnimationFrame(animate);

### 5.5.b Output Percobaan 5 : CSS Text Slide

---

Cari sendiri gambar kupu-kupunya.



Gambar 5.5.b Output Percobaan 5 : CSS Text Slide

Berdasarkan Gambar 5.5.b Output terdapat gambar kupu kupu yang merupakan hasil dari baris ke 9 yang memanggil gambar yang sesuai dengan format kode yang telah di atur. Gambar kupu kupu tersebut bergerak ke arah dan posisi yang telah di tentukan pada baris ke 15 sampai baris ke 2

## VI. KESIMPULAN

Adapun kesimpulan sebagai berikut :

1. Berdasarkan percobaan 5.1, 5.2, 5.3, 5.4, dan 5.5 Kode JavaScript ditulis dalam elemen `<script>`, yang mengizinkan penulisan dan eksekusi kode JavaScript dalam halaman web.
2. Berdasarkan percobaan 5.2, JavaScript dapat digunakan untuk mengarahkan pengguna ke halaman web lain dengan jeda waktu yang telah ditentukan oleh user dengan menggunakan fungsi `redirect`
3. Berdasarkan percobaan 5.3 pada baris ke 12 terdapat fungsi `mencariString` yang akan berjalan secara rekursif melalui elemen dalam dokumen dan mencoba mencocokkan teks yang diberikan dengan elemen-elemen dalam dokumen.
4. Berdasarkan percobaan 5.5 terdapat Fungsi `animate` pada source code baris ke 14 digunakan untuk menganimasikan agar gambar kupu-kupu bisa ke kiri kekanan . Fungsi ini dijalankan menggunakan `requestAnimationFrame` pada source code baris ke 22 untuk menciptakan animasi yang mulus.
5. Berdasarkan percobaan 5.5 Fungsi `animate` menerima dua parameter, `time` (waktu saat ini) dan `lastTime` (waktu sebelumnya). Fungsi ini menghitung perubahan sudut berdasarkan perbedaan waktu antara saat ini dan waktu sebelumnya.

## DAFTAR PUSTAKA

Nugroho, Bunafit. 2004. Aplikasi Pemrograman Web Dinamis dengan PHP dan MySQL. Yogyakarta : Penerbit Gavamedia

Gunawan, Wahyu. 2010. Kebut Sehari jadi Master PHP. Yogyakarta: Genius

Publisher Hakim, Lukmanul. 2008. Membongkar Trik Bahasa Para Master PHP.

Yogyakarta: Penerbit Lokomedia

## TUGAS AKHIR

SOAL :

Buat halaman yang menampilkan nilai ganjil dari 1 s.d. 10000 jika tombol Ganjil ditekan dan nilai genap dari 10001 s.d. 20000 jika tombol Genap ditekan. Letakkan kode pada berkas JS eksternal!

JAWAB :

### 1. Source Code : HTML Tugas akhir

```
C: > Prak. Pemweb > <> TA, Laprak 3.html > html > body > script
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Tampilan Bilangan Ganjil dan Genap</title>
5  </head>
6  <body>
7  |   <button id="btnGanjil">Ganjil</button>
8  |   <button id="btnGenap">Genap</button>
9  |   <div id="result"></div>
10 |   <script src="TA, Laprak 3.js"></script>
11 </body>
12 </html>
13
```

Gambar 1 Source Code : HTML Tugas akhir

### 2. Source Code : Java Script Tugas Akhir

```
1  document.getElementById("btnGanjil").addEventListener("click", function() {
2      let result = "";
3      for (let i = 1; i <= 10000; i++) {
4          if (i % 2 !== 0) {
5              result += i + " ";
6          }
7      }
8      document.getElementById("result").textContent = result;
9  });
10
11 document.getElementById("btnGenap").addEventListener("click", function() {
12     let result = "";
13     for (let i = 10001; i <= 20000; i++) {
14         if (i % 2 === 0) {
15             result += i + " ";
16         }
17     }
18     document.getElementById("result").textContent = result;
19 });
20
```

Gambar 2 Source Code Java Script

### 3. Output Tugas Akhir

