

LAPORAN PRAKTIKUM

“JAVASCRIPT”

Dalam rangka memenuhi tugas mata kuliah Praktikum Desain Website

Dosen pengampu: Vearen Dika Sofirudin, S.Pd., M.Ed.



Disusun oleh:

Nama : Rifkhi Ardi Mustaqim

NIM : K3524033

PENDIDIKAN TEKNIK INFORMATIKA DAN KOMPUTER

FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN

UNIVERSITAS SEBELAS MARET

SURAKARTA

2025

Tujuan Praktikum

Tujuan dari praktikum ini adalah untuk:

1. Memahami struktur dasar JavaScript dan cara implementasinya dalam halaman web.
2. Mampu memanipulasi elemen HTML secara dinamis menggunakan Document Object Model (DOM).
3. Menguasai penggunaan *event handler* (seperti onclick) untuk membuat halaman web yang interaktif terhadap aksi pengguna.
4. Memahami konsep dasar penyimpanan data di sisi klien (browser) menggunakan localStorage.
5. Menerapkan logika JavaScript untuk memproses input pengguna dan melakukan operasi, seperti pada studi kasus kalkulator sederhana.

Ringkasan Materi

Praktikum ini berfokus pada JavaScript, bahasa pemrograman yang digunakan untuk membuat website menjadi interaktif dan dinamis.

1. Penulisan JavaScript: Terdapat empat cara utama untuk menyisipkan JavaScript ke dalam HTML:
 - a. Internal: Menggunakan tag `<script>...</script>` langsung di dalam file HTML.
 - b. Eksternal: Memindahkan kode ke file .js terpisah dan memanggilnya menggunakan `<script src="namafile.js"></script>`. Cara ini sangat disarankan untuk fleksibilitas.
 - c. Inline (Event Handler): Menjalankan kode saat terjadi sesuatu, misalnya `onclick="alert('Halo')"` pada sebuah tombol.
 - d. URL: Menggunakan javascript: pada atribut href sebuah link.
2. Variabel dan Tipe Data: Variabel adalah tempat menyimpan nilai dan dapat dideklarasikan menggunakan `var`, `let`, atau `const`. JavaScript memiliki tipe data implisit seperti Numerik, String, Boolean, dan Null.
3. Fungsi Output: Ada 4 cara umum untuk menampilkan output di JavaScript:
 - a. `console.log()`: Menampilkan output di console browser, berguna untuk debugging.
 - b. `alert()`: Menampilkan kotak dialog pop-up.
 - c. `document.write()`: Menulis langsung ke dokumen HTML.
 - d. `innerHTML`: Atribut untuk mengubah konten di dalam sebuah elemen HTML yang spesifik.
4. Document Object Model (DOM): DOM adalah antarmuka yang merepresentasikan dokumen HTML sebagai pohon simpul (tree of nodes). Dengan DOM, JavaScript mendapatkan kekuatan untuk mengubah semua elemen, atribut, dan gaya CSS di halaman. Ini adalah kunci untuk membuat HTML dinamis.

Langkah Kerja

Praktikum ini terdiri dari dua latihan mandiri.

2. Latihan 1 (Kalkulator Sederhana)

- Membuat file baru latihan1.HTML.
- Membuat *layout* HTML yang terdiri dari dua *input* untuk angka dan empat *button* untuk operasi (Jumlah, Kurang, Kali, Bagi).
- Setiap *button* diberi atribut onclick untuk memanggil fungsi JavaScript yang sesuai (misal, onclick="jumlah()").
- Setiap fungsi (cth: function jumlah()) mengambil nilai dari kedua *input* menggunakan document.getElementById(), mengubahnya menjadi angka (parseFloat), melakukan perhitungan, dan menampilkan hasilnya ke elemen *span* #hasil menggunakan innerHTML.

3. Latihan 2 (Highlight Teks)

- Membuat file baru latihan2.html.
- Halaman berisi paragraf teks tentang PTIK dan satu *button* "Highlight Paragraf".
- Frasa-frasa spesifik dalam paragraf yang akan di-highlight dibungkus dengan tag `` dan diberi *class* (misal, class="to-highlight").
- Sebuah *class* CSS .highlight dibuat untuk memberi style (warna merah, background kuning, tebal).

Tombol diberi onclick untuk memanggil fungsi toggleHighlight(). Fungsi ini menggunakan document.querySelectorAll('.to-highlight') untuk memilih semua *span* target, lalu melakukan *looping* dan menggunakan element.classList.toggle('highlight') untuk menambah atau menghapus *class* .highlight pada setiap *span*.

Dokumentasi Program dan Hasil Program

Latihan1.html Kode Program:

```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <title>Latihan 1 - Kalkulator</title>
6
7   <style>
8     body {
9       font-family: Arial, sans-serif;
10      margin: 20px;
11      background-color: #f0f0f0;
12    }
13    .container {
14      max-width: 300px;
15      margin: 0 auto;
16      padding: 20px;
17      border: 1px solid #ccc;
18      border-radius: 8px;
19      background-color: #fff;
20    }
21    input[type="number"] {
22      width: 100%;
23      padding: 8px;
24      margin-bottom: 10px;
25      box-sizing: border-box;
26    }
27    button {
28      width: 48%;
29      padding: 10px;
30      margin: 5px;
31      cursor: pointer;
32    }
33    #hasil-wrapper {
34      margin-top: 20px;
35      padding: 10px;
36      background-color: #f0f0f0;
37      border-radius: 4px;
38    }
39    #hasil {
40      font-weight: bold;
41      font-size: 1.2em;
42      color: #333;
43    }
44  </style>
45 </head>
46 <body>
47
48   <div class="container">
49     <h2>Kalkulator</h2>
50
51     <label for="angka1">Angka 1:</label>
52     <input type="number" id="angka1" placeholder="Masukkan angka pertama">
53
54     <label for="angka2">Angka 2:</label>
55     <input type="number" id="angka2" placeholder="Masukkan angka kedua">
56
57     <button onclick="jumlah()">Jumlah (+)</button>
58     <button onclick="kurang()">Kurang (-)</button>
59     <button onclick="kali()">Kali (x)</button>
60     <button onclick="bagi()">Bagi (/)</button>
61
62     <div id="hasil-wrapper">
63       <span>Hasil:</span>
64       <span id="hasil"></span>
65     </div>
66   </div>
67
68   <script>
69     function jumlah() {
70       var input1 = document.getElementById("angka1");
71       var input2 = document.getElementById("angka2");
72       var hasilElemen = document.getElementById("hasil");
73
74       var a = parseFloat(input1.value);
75       var b = parseFloat(input2.value);
76
77       if (isNaN(a)) a = 0;
78       if (isNaN(b)) b = 0;
79
80       var hasil = a + b;
81       hasilElemen.innerHTML = hasil;
82     }
83
84     function kurang() {
85       var input1 = document.getElementById("angka1");
86       var input2 = document.getElementById("angka2");
87       var hasilElemen = document.getElementById("hasil");
88
89       var a = parseFloat(input1.value);
90       var b = parseFloat(input2.value);
91
92       if (isNaN(a)) a = 0;
93       if (isNaN(b)) b = 0;
94
95       var hasil = a - b;
96       hasilElemen.innerHTML = hasil;
97     }
98
99     function kali() {
100      var input1 = document.getElementById("angka1");
101      var input2 = document.getElementById("angka2");
102      var hasilElemen = document.getElementById("hasil");
103
104      var a = parseFloat(input1.value);
105      var b = parseFloat(input2.value);
106
107      if (isNaN(a)) a = 0;
108      if (isNaN(b)) b = 0;
109
110      var hasil = a * b;
111      hasilElemen.innerHTML = hasil;
112    }
113
114    function bagi() {
115      var input1 = document.getElementById("angka1");
116      var input2 = document.getElementById("angka2");
117      var hasilElemen = document.getElementById("hasil");
118
119      var a = parseFloat(input1.value);
120      var b = parseFloat(input2.value);
121
122      if (isNaN(a)) a = 0;
123      if (isNaN(b)) b = 0;
124
125      if (b === 0) {
126        hasilElemen.innerHTML = "Error: Dibagi 0";
127      } else {
128        var hasil = a / b;
129        hasilElemen.innerHTML = hasil;
130      }
131    }
132  </script>
```

Penjelasan kode:

1. Struktur HTML

- Struktur utama dibungkus dalam `<div class="container">`.
- Terdapat dua elemen `<input type="number">` yang diberi id unik (angka1 dan angka2). ID ini krusial sebagai "pengenal" agar JavaScript dapat mengambil nilainya.
- Empat elemen `<button>` disediakan untuk operasi. Setiap tombol memiliki atribut `onclick` yang secara langsung memanggil fungsi JavaScript spesifik (misalnya, `onclick="jumlah()"`).
- Sebuah `` dengan `id="hasil"` disiapkan sebagai target untuk menampilkan output perhitungan.

2. Tampilan CSS

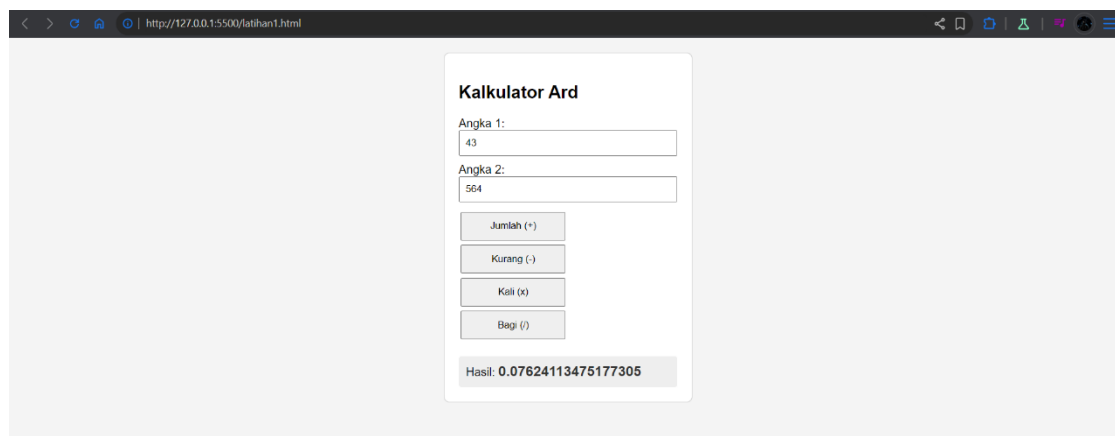
- CSS di dalam tag `<style>` digunakan untuk presentasi visual.
- Selektor `.container` dipakai untuk mengatur lebar (`max-width: 300px`) dan memposisikan kalkulator di tengah layar (`margin: 0 auto`).

- Lebar tombol (width: 48%) diatur agar pas tampil dua kolom dalam *container*.
- Area #hasil-wrapper diberi *styling* latar belakang untuk membedakannya sebagai area output.

3. Fungsionalitas JavaScript

- Logika inti dipecah menjadi empat fungsi terpisah: jumlah(), kurang(), kali(), dan bagi().
- Di dalam setiap fungsi, alurnya adalah sebagai berikut:
 1. **Seleksi Elemen:** document.getElementById() digunakan untuk mengambil elemen input dan elemen hasil berdasarkan id mereka.
 2. **Pengambilan Nilai:** Properti .value diambil dari elemen input.
 3. **Konversi Tipe:** parseFloat() digunakan untuk mengonversi nilai (yang aslinya *string*) menjadi *number*, agar operasi matematis dapat dilakukan.
 4. **Validasi:** isNaN() (Is Not a Number) mengecek jika input kosong. Jika true, nilai *default* 0 diberikan untuk menghindari *error* kalkulasi.
 5. **Perhitungan & Output:** Hasil kalkulasi disimpan dalam variabel hasil, yang kemudian disuntikkan ke target menggunakan hasilElemen.innerHTML.
- **Logika Khusus:** Fungsi bagi() memiliki satu pemeriksaan kondisional if (b === 0) untuk menangani kasus pembagian dengan nol, yang akan menampilkan pesan error alih-alih hasil "Infinity".

Screenshot hasil:



Latihan2.html

Kode Program:

```
1 <DOCTYPE html>
2 <html>
3 <head>
4 <title>Latihan 2 - Highlight Teks</title>
5 <style>
6 body { font-family: sans-serif; padding: 20px; line-height: 1.6; }
7 #ptik-text {
8   border: 1px solid #ccc;
9   padding: 15px;
10  max-width: 800px;
11 }
12 button {
13   margin-top: 15px;
14   padding: 10px 20px;
15   font-size: 1em;
16 }
17 .highlight {
18   color: red;
19   background-color: yellow;
20   font-weight: bold;
21 }
22 </style>
23 </head>
24 <body>
25
26 <div id="ptik-text">
27   <h3>PTIK FKIP UNS</h3>
28
29   <p>The <span class="to-highlight">Faculty of Teacher Training and Education</span>, <span class="to-highlight">Sebelas Maret University</span>, Surakarta is an <span class="to-highlight">Educational Person</span>
30
31   <p>The <span class="to-highlight">Informatics and Computer Engineering Education (PTIK)</span> study program is planned to take shelter in the management of the <span class="to-highlight">Engineering and M</span>
32 </div>
33
34 <button onclick="toggleHighlight()">Highlight Paragraf</button>
35
36 <script>
37   function toggleHighlight() {
38     var elements = document.querySelectorAll('.to-highlight');
39
40     elements.forEach(function(element) {
41       element.classList.toggle('highlight');
42     });
43   }
44 </script>
45
46 </body>
47 </html>
```

Penjelasan Kode:

1. Struktur HTML

- Teks paragraf ditempatkan di dalam `<div id="ptik-text">`.
- Kunci Persiapan:** Frasa-frasa spesifik yang akan di-highlight **dibungkus satu per satu** menggunakan tag ``.
- Setiap `` diberi **class="to-highlight"**. Class ini berfungsi sebagai "penanda" atau *selector* agar JavaScript dapat menemukan semua elemen tersebut.
- Sebuah `<button>` disiapkan dengan atribut **onclick="toggleHighlight()"**, yang bertindak sebagai pemicu (trigger) untuk menjalankan fungsi JavaScript.

2. Tampilan CSS

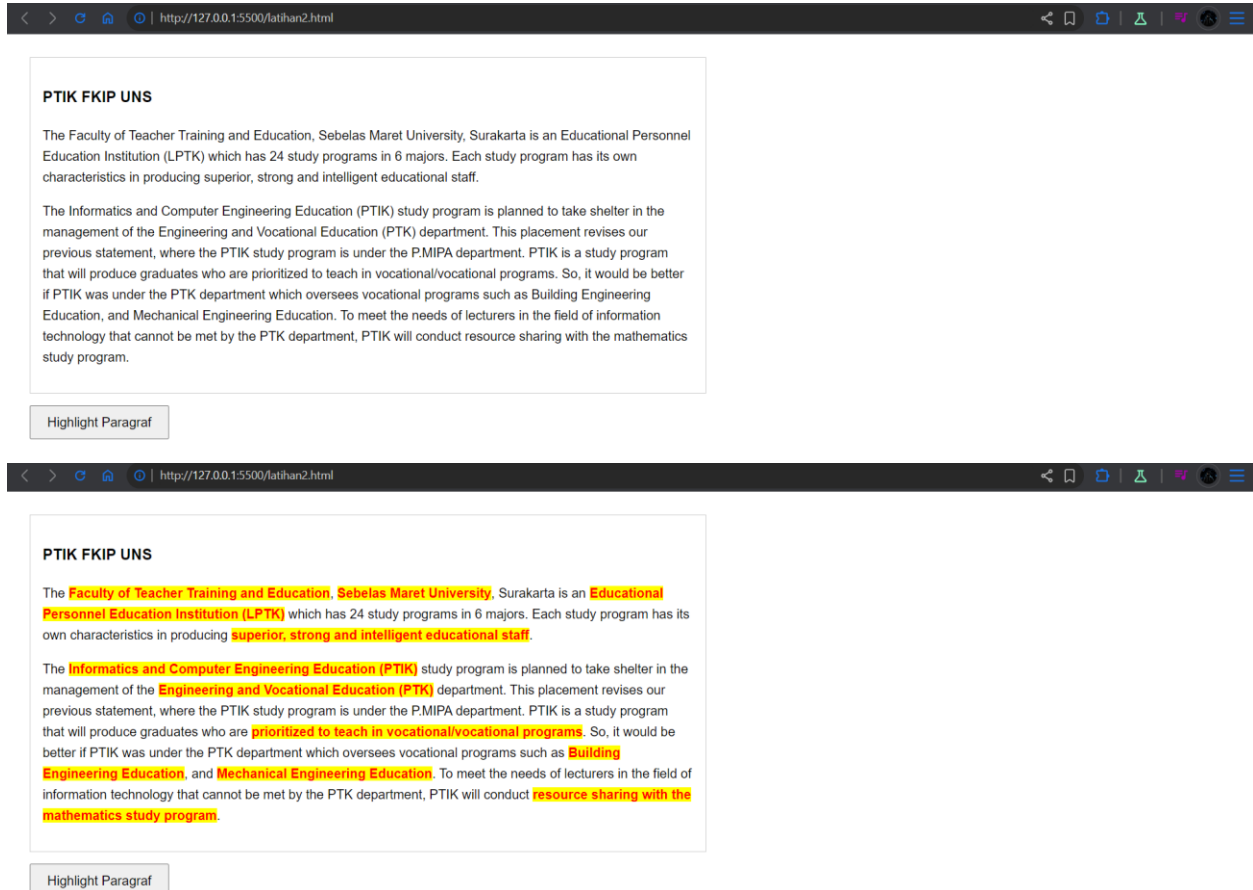
- CSS di dalam tag `<style>` mendefinisikan tampilan akhir dari *highlight*.
- Dibuat sebuah *class* bernama **.highlight**. Class ini berisi properti visual yang diinginkan (teks merah, latar kuning, dan tebal).
- Penting dicatat bahwa *class .highlight* ini **tidak diterapkan** pada elemen mana pun saat halaman pertama kali dimuat. Class ini hanya akan "dipakai" ketika JavaScript menambahkannya.

3. Fungsionalitas JavaScript

- Logika inti berada di dalam fungsi `toggleHighlight()`.
- document.querySelectorAll('.to-highlight')**:
 - Ini adalah perintah untuk mencari di seluruh dokumen HTML dan **memilih semua elemen** yang memiliki *class* `.to-highlight`.
 - Hasilnya adalah `NodeList` (kumpulan elemen), yang disimpan dalam variabel `elements`.

- c. **elements.forEach(function(element) { ... });**:
- 1) Kode ini melakukan *looping* (perulangan) pada setiap elemen tunggal (element) yang ditemukan di dalam NodeList tadi.
- d. **element.classList.toggle('highlight');**:
- 1) Ini adalah **mekanisme utamanya**. Untuk setiap yang sedang di-loop:
 - 2) `.classList.toggle()` akan memeriksa apakah elemen tersebut *sudah memiliki class .highlight*.
 - 3) **Jika belum ada**: `Class .highlight` akan ditambahkan, sehingga CSS membuatnya ter-highlight.
 - 4) **Jika sudah ada**: `Class .highlight` akan dihapus, sehingga elemen kembali ke tampilan normal.
- e. Hasilnya, satu tombol dapat mengaktifkan dan menonaktifkan *highlight* pada banyak target sekaligus.

Screenshot hasil:



Kesimpulan

Berdasarkan praktikum yang telah dilakukan, dapat ditarik kesimpulan:

1. **JavaScript** terbukti esensial untuk mengubah HTML statis menjadi halaman

yang dinamis dan interaktif.

2. **Document Object Model (DOM)** adalah jembatan yang memungkinkan JavaScript untuk "menemukan" dan "memanipulasi" elemen HTML.
3. Keberhasilan manipulasi bergantung pada **selektor** (seperti `getElementById` dan `querySelectorAll`) untuk memilih target dan **event handler** (seperti `onclick`) untuk memicu aksi.
4. Latihan ini menerapkan dua metode manipulasi yang berbeda: `.innerHTML` untuk mengubah konten (di kalkulator) dan `.classList.toggle()` untuk mengubah style CSS (di fitur highlight).
5. JavaScript berhasil digunakan untuk mengeksekusi **logika** pemrograman, seperti validasi input (`isNaN`), penanganan kondisi (`if...else`), dan perulangan (`forEach`) untuk memproses data secara efektif.