**LAPORAN PRAKTIKUM**

Laporan ini dibuat untuk memenuhi tugas mata kuliah

**“Praktikum Pemrograman Web 1”**

yang diampu oleh Ibu Yati Nurhayati, S.Kom., M.Kom.

**MODUL 4 & 5**



**NIM : 20230810088**

**NAMA : Muhammad Rizal Nurfirdaus**

**KELAS : TINFC-2023-04**

**TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS KUNINGAN**

**Praktikum**

<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript Embedded</title>

<script>

alert("Ini Latihan JavaScript Pertama Saya!");

console.log("It's Work");

document.write("Success!!");

</script>

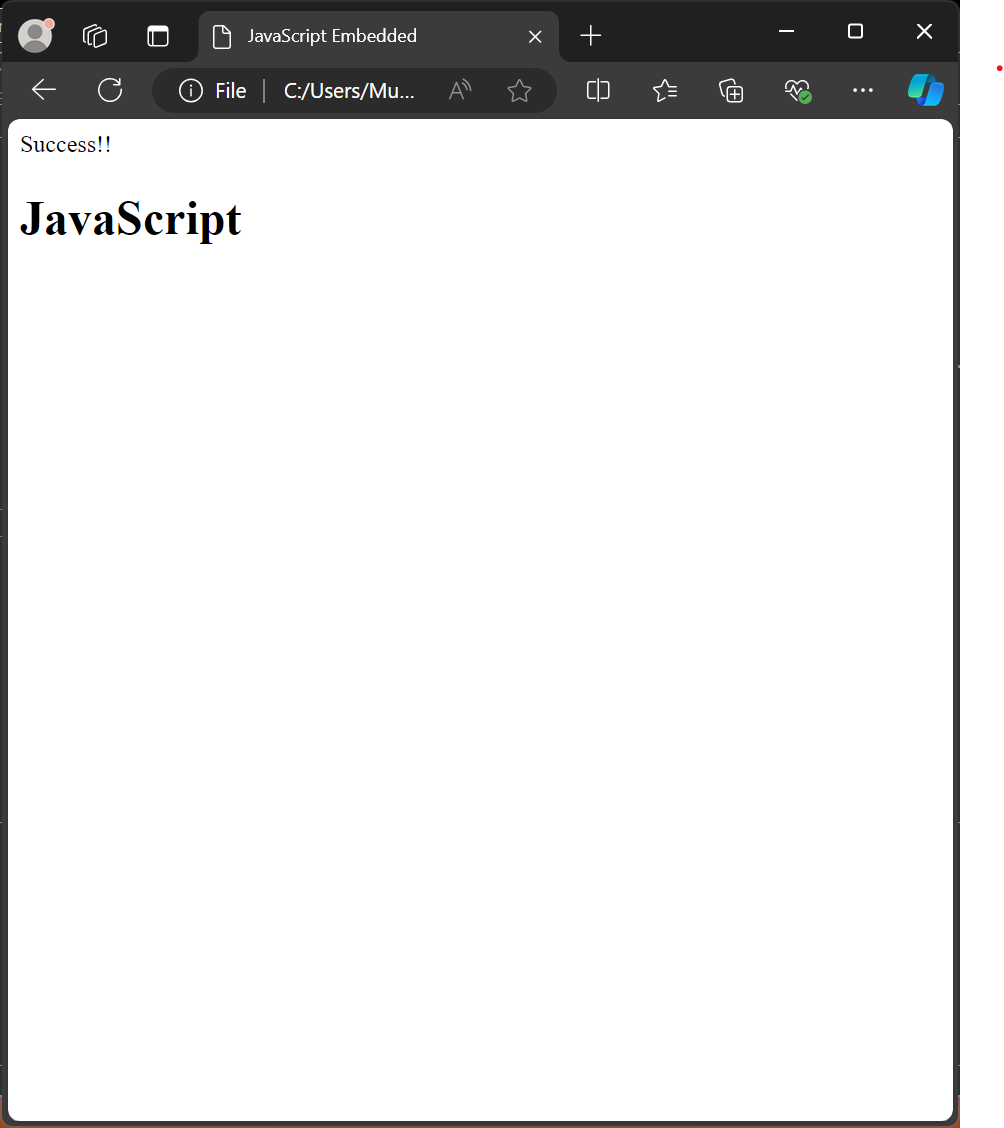
</head>

<body>

<h1>JavaScript</h1>

</body>

</html>



Analisa: Kode HTML di atas menyertakan JavaScript yang ditanamkan dalam tag <script> di bagian <head>. Saat halaman dimuat, akan muncul kotak pemberitahuan dengan pesan "Ini Latihan JavaScript Pertama Saya!", pesan "It's Work" akan tercatat di konsol browser, dan teks "Success!!" akan ditampilkan langsung di halaman. Halaman ini juga memiliki judul "JavaScript Embedded" yang muncul di tab browser dan heading utama "JavaScript". Penggunaan document.write dapat menulis konten langsung ke halaman, namun sebaiknya dihindari setelah halaman sepenuhnya dimuat karena bisa menggantikan isi halaman.

<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript Embed</title>

</head>

<body>

<h1>JavaScript</h1>

<button onclick = "alert('Ini Latihan JavaScript Pertama Saya!')">Pesan</button>

<button onclick = "console.log('Its Work')">Pesan di Console</button>

<button

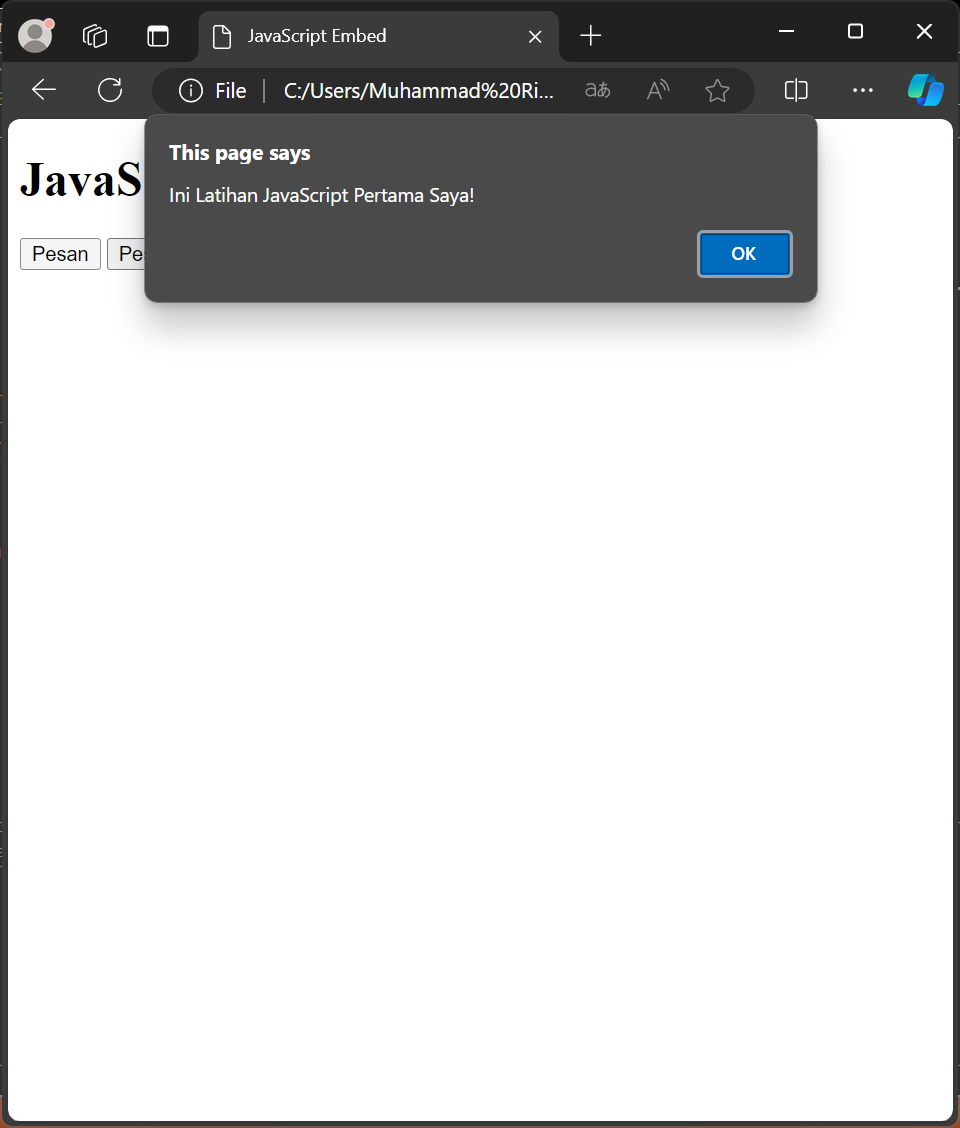
onclick = "document.write('Success <br>');

document.write('<a href = M4-2.html>Kembali</a>')">Pesan di Web

</button>

</body>

</html>



Analisa: Kode HTML di atas menampilkan tiga tombol dengan fungsi JavaScript berbeda saat diklik. Tombol pertama memunculkan kotak pesan (alert) dengan teks "Ini Latihan JavaScript Pertama Saya!". Tombol kedua mencatat pesan "Its Work" ke konsol browser, dan tombol ketiga menampilkan teks "Success" di halaman web, diikuti dengan tautan "Kembali" yang mengarah ke file M4-2.html. Penggunaan document.write pada tombol ketiga dapat mengganti seluruh konten halaman jika digunakan setelah halaman selesai dimuat, sehingga perlu diperhatikan untuk aplikasi yang lebih kompleks.

<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript External</title>

<script src = "jsM43.js"></script>

</head>

<body>

<h1>JavaScript</h1>

<button onclick = "pesan()">Pesan</button>

<button onclick = "pesanConsole()">Pesan di Console</button>

<button onclick = "pesanDok()">Pesan di Web</button>

</body>

</html>

JavaScript

function pesan(){

    alert("Ini Latihan JavaScript Pertama Saya!");

}

function pesanConsole(){

    console.log("It's Work");

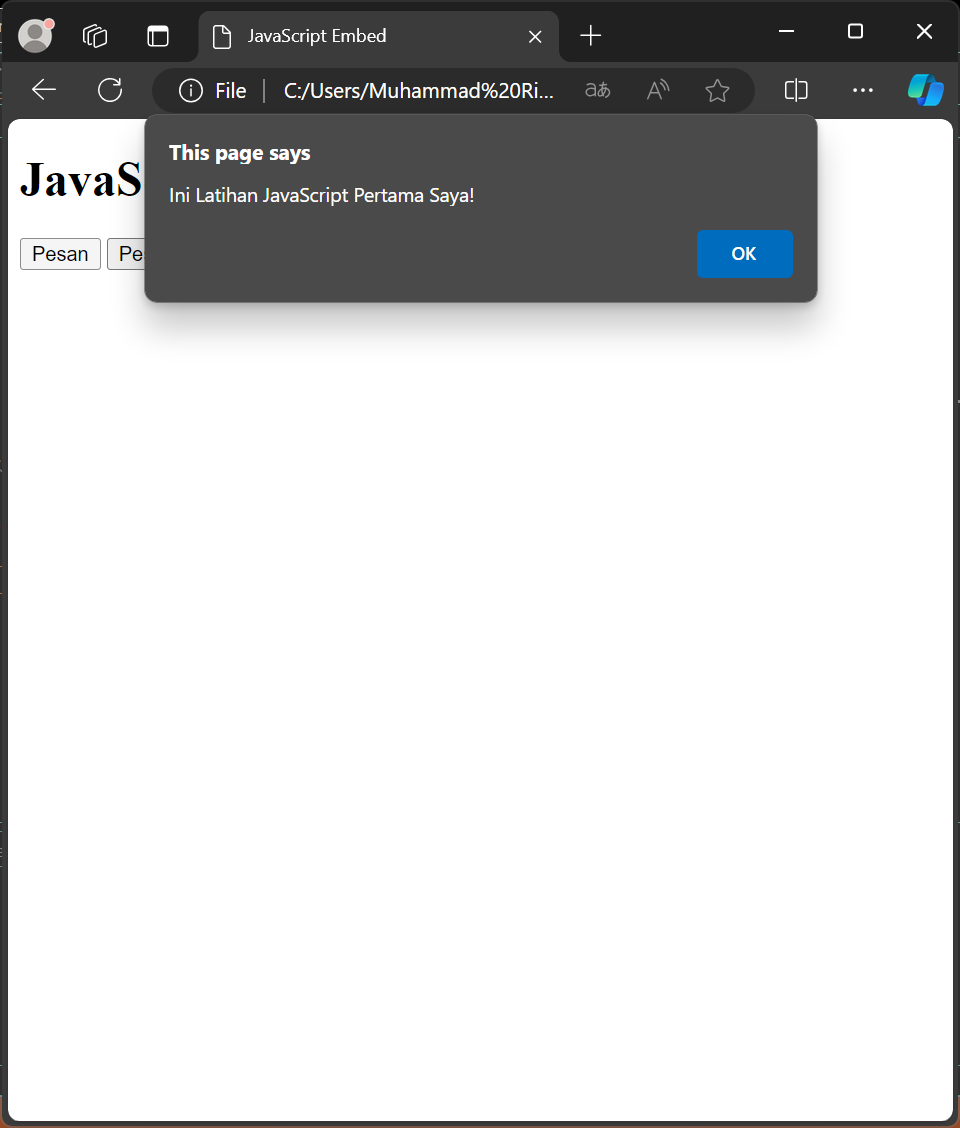
}

function pesanDok(){

    document.write("Success <br>");

    document.write("<a href = 'M4-2.html'>Kembali</a>");

}



Analisa: Kode HTML di atas menggunakan file eksternal jsM43.js untuk menjalankan fungsi JavaScript. Tiga tombol di halaman masing-masing memanggil fungsi: **pesan()** untuk menampilkan alert "Ini Latihan JavaScript Pertama Saya!", **pesanConsole()** untuk mencatat "It's Work" di konsol, dan **pesanDok()** untuk menampilkan teks "Success" beserta tautan "Kembali" ke halaman M4-2.html di halaman web. Penggunaan file JavaScript eksternal seperti ini memisahkan logika dari struktur HTML, sehingga lebih rapi dan mudah dikelola. Namun, fungsi document.write tetap perlu dihindari untuk aplikasi modern agar tidak menggantikan seluruh konten halaman.

<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript Kotak Dialog</title>

<script>

function pesan(){

alert("Pemrograman Web 1");

}

function cekStatus(){

const hasil = confirm("Apakah Anda Mahasiswa Universitas Kuningan?");

if(hasil){

console.log("SIlahkan Mengakses Materi yang Telah Disediakan");

}else{

console.log("Anda Tidak Berhak Mengakses!");

}

}

function input(){

const userInput = prompt("Masukan Nama Anda :");

if(userInput !== null){

console.log("Nama Pengguna adalah : " + userInput);

}else{

console.log("Pengguna Membatalkan Input.");

}

}

</script>

</head>

<body>

<h1>JavaScript</h1>

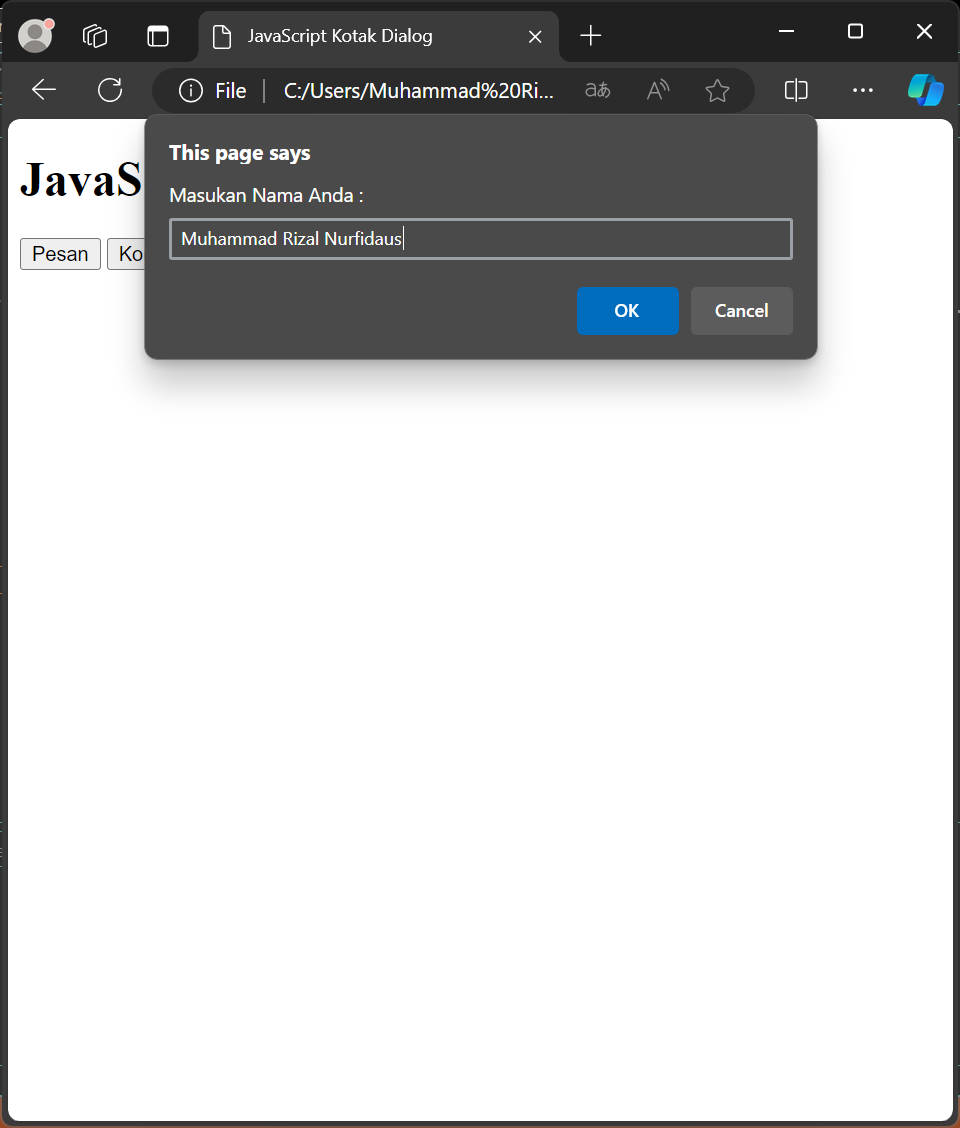
<button onclick = "pesan()">Pesan</button>

<button onclick = "cekStatus()">Konfirmasi</button>

<button onclick = "input()">Masukan</button>

</body>

</html>



Analisa: Kode HTML ini mendemonstrasikan penggunaan tiga jenis kotak dialog JavaScript: alert, confirm, dan prompt, yang masing-masing dipanggil melalui tombol. Fungsi pesan() menampilkan kotak pesan dengan teks "Pemrograman Web 1". Fungsi cekStatus() meminta konfirmasi dengan pesan "Apakah Anda Mahasiswa Universitas Kuningan?" dan memberikan respons berbeda di konsol tergantung pilihan pengguna. Fungsi input() meminta input nama pengguna, menampilkan nama tersebut di konsol jika pengguna mengisi, atau mencatat "Pengguna Membatalkan Input" jika dialog dibatalkan. Penggunaan kotak dialog ini menunjukkan cara interaktif sederhana untuk berkomunikasi dengan pengguna di JavaScript.

<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Fungsi</title>

<script>

function tampilTeks(){

document.write("Hai!! <br>");

}

function tampilNama(namaAnda){

document.write("Selamat Datang " + namaAnda);

}

tampilTeks();

nama = "Ghina";

tampilNama(nama);

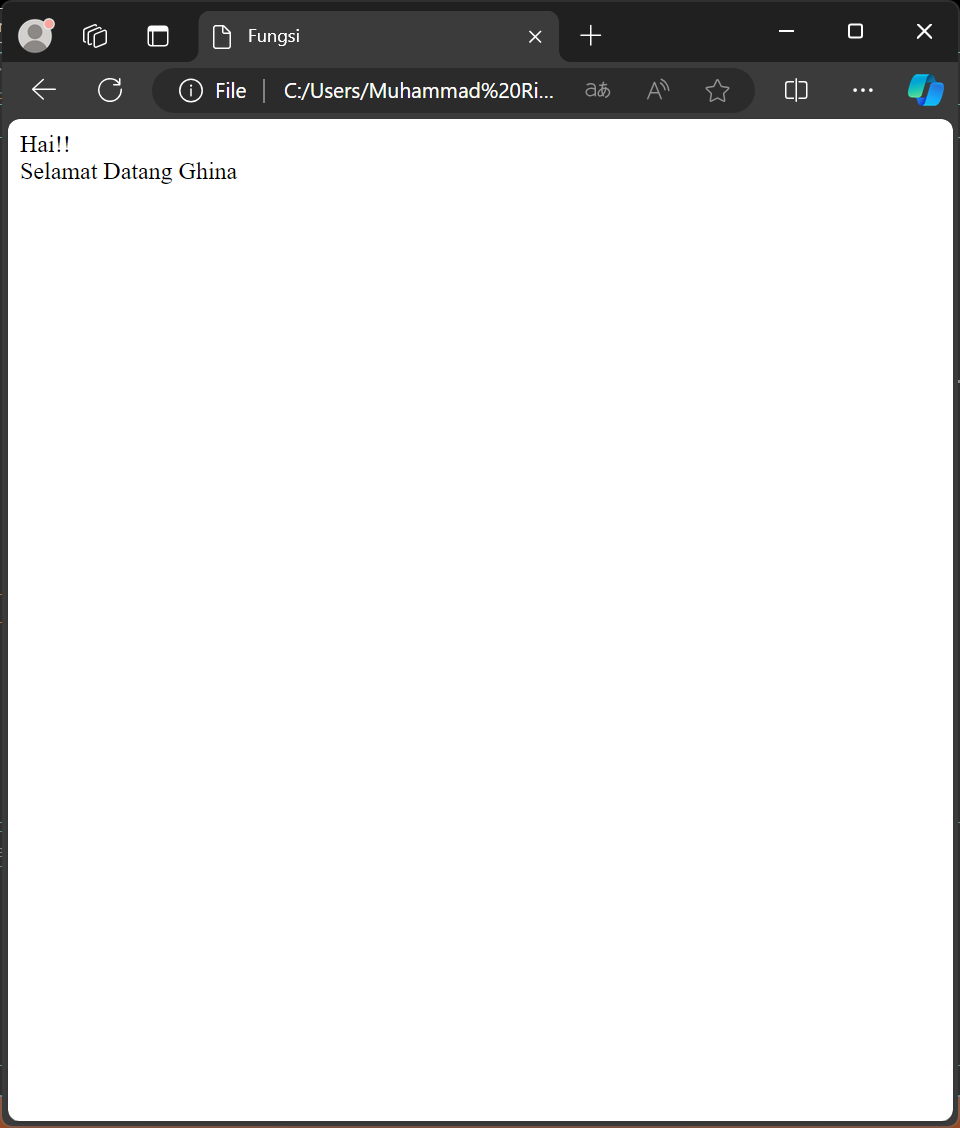
</script>

</head>

<body>

</body>

</html>



Analisis :Kode HTML ini menggunakan dua fungsi JavaScript: **tampilTeks()** untuk menampilkan teks "Hai!!" dan **tampilNama(namaAnda)** untuk menampilkan pesan "Selamat Datang" diikuti dengan nama yang diberikan. Fungsi-fungsi ini dipanggil secara otomatis saat halaman dimuat, di mana **tampilTeks()** ditampilkan terlebih dahulu, diikuti oleh pemanggilan **tampilNama("Ghina")**, sehingga halaman menampilkan "Hai!!" diikuti oleh "Selamat Datang Ghina". Namun, penggunaan **document.write** setelah halaman dimuat dapat menggantikan seluruh konten, sehingga pendekatan ini tidak disarankan untuk aplikasi modern.

6.

<!DOCTYPE html>

<html lang = "id">

    <head>

        <meta charset = "UTF-8" />

        <meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

        <title>Fungsi</title>

    </head>

    <script>

        function hitungLuasSegitiga(al, t){

            luas = (al \* t) / 2;

            document.write("Alas : " + al);

            document.write("<br>Tinggi : " + t);

            document.write("<br>Luas : " + luas);

        }

        alas = 20;

        tinggi = 40;

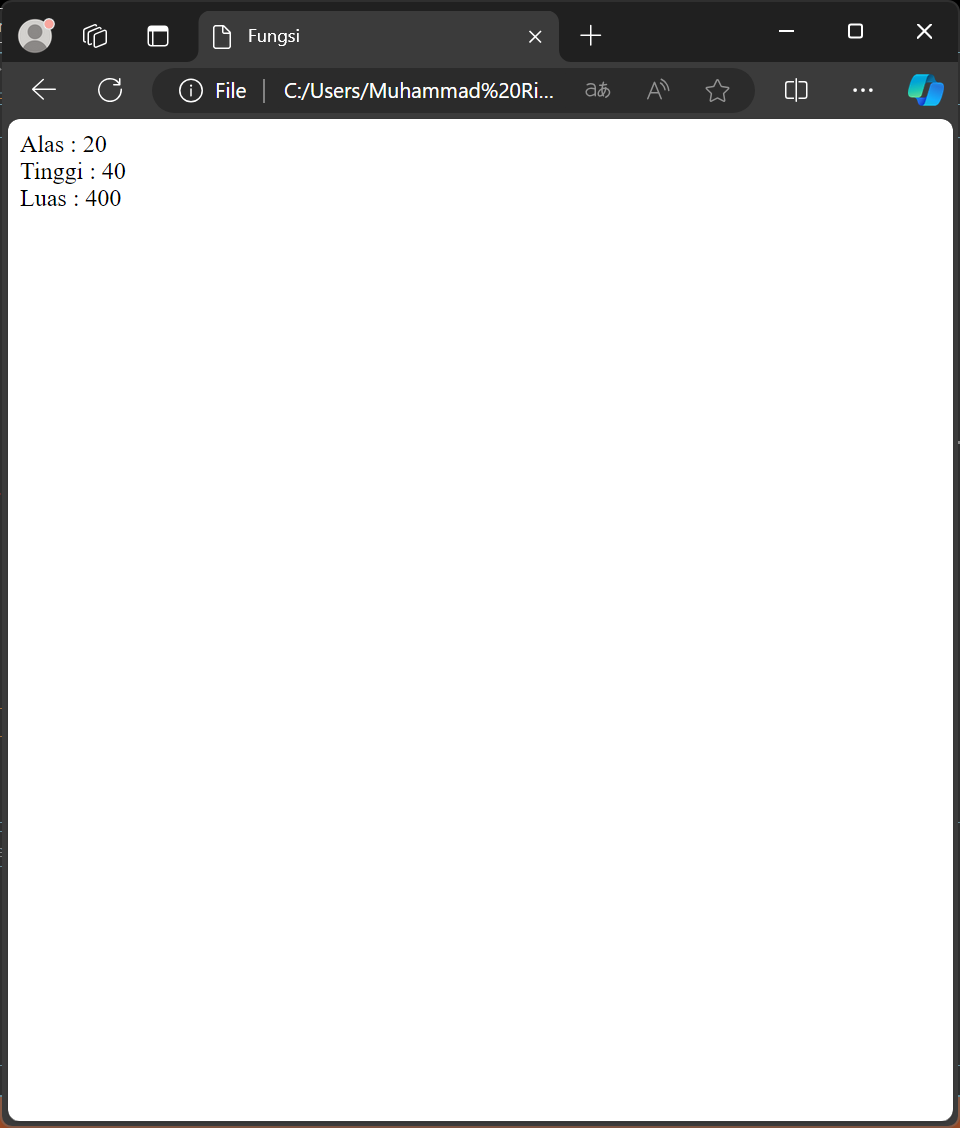
        hitungLuasSegitiga(alas, tinggi);

    </script>

    <body>

    </body>

</html>



Analisis : Kode HTML di atas menggunakan fungsi JavaScript hitungLuasSegitiga(al, t) untuk menghitung luas segitiga berdasarkan parameter alas (al) dan tinggi (t). Fungsi ini mengalikan alas dan tinggi, membaginya dengan 2, lalu menampilkan nilai alas, tinggi, dan luas menggunakan document.write. Pada saat halaman dimuat, variabel alas diisi dengan nilai 20 dan tinggi dengan nilai 40, sehingga fungsi secara otomatis menghitung dan menampilkan hasilnya di halaman web.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript</title>

<script>

function showData(){

nm = document.getElementById("nama").value;

jk = document.getElementById("jenKel").value;

document.getElementById("hasil").innerHTML = "<strong>Nama : </strong > "

+ nm + "<br><strong>Jenis Kelamin : </strong> " + jk;

}

</script>

</head>

<body>

<h1>Form Input</h1>

<form nama = "form1">

<table border = "1">

<tr>

<td><label>Nama</label></td>

<td><input type = "text" name = "nama" id = "nama" /></td>

</tr>

<tr>

<td><label>Jenis Kelamin</label></td>

<td>

<input type = "radio" name = "jenKel" id = "jenKel" value = "L" /> Laki - Laki

<input type = "radio" name = "jenKel" id = "jenKel" value = "P" /> Perempuan

</td>

</tr>

<tr>

<td colspan = "2">

<button type = "button" name = "tampil" value = "tampil"

onclick = "showData()">Tampilkan</button>

<button type = "reset" name = "batal" value = "batal">Batal</button>

</td>

</tr>

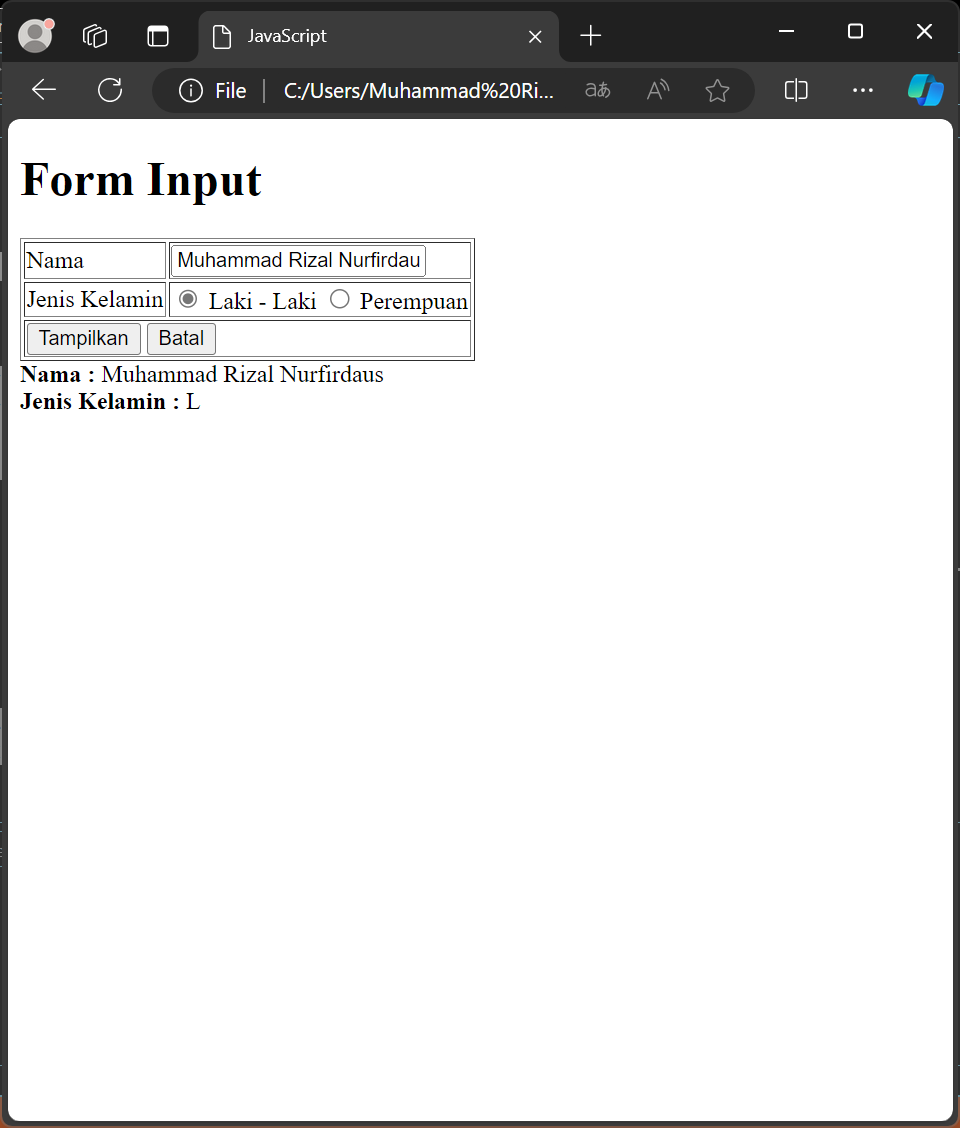
</table>

</form>

<div id = "hasil"></div>

</body>

</html>



Analisis: Kode HTML ini membuat formulir input sederhana dengan dua field: nama (input teks) dan jenis kelamin (radio button). Fungsi JavaScript showData() membaca nilai dari field nama dan jenis kelamin menggunakan document.getElementById, lalu menampilkan hasilnya dalam elemen <div> dengan ID "hasil" menggunakan innerHTML. Tombol "Tampilkan" memanggil fungsi ini untuk menampilkan data, sementara tombol "Batal" mereset formulir. Kode ini memungkinkan pengguna memasukkan data dan melihat hasilnya secara langsung di halaman tanpa harus memuat ulang.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript</title>

<script>

function showData(){

pr = document.getElementById("prodi").value;

document.getElementById("programStudi").value = pr;

}

</script>

</head>

<body>

<h1>Form Pilih Program Studi</h1>

<form nama = "form1">

<table border = "0">

<tr>

<td><label>Program Studi</label></td>

<td>

<select name = "prodi" id = "prodi" onchange = "showData()">

<option value = "Belum ada yang dipilih!">-- Pilih --</option>

<option value = "ti">Teknik Informatika</option>

<option value = "si">Sistem Informasi</option>

<option value = "dkv">Desain Komunikasi Visual</option>

<option value = "ts">Teknik Sipil</option>

<option value = "mi">Manajemen Informatika</option>

</select>

</td>

</tr>

<tr>

<td><label>Data</label></td>

<td>

<input type = "text" name = "programStudi" id = "programStudi" readonly />

</td>

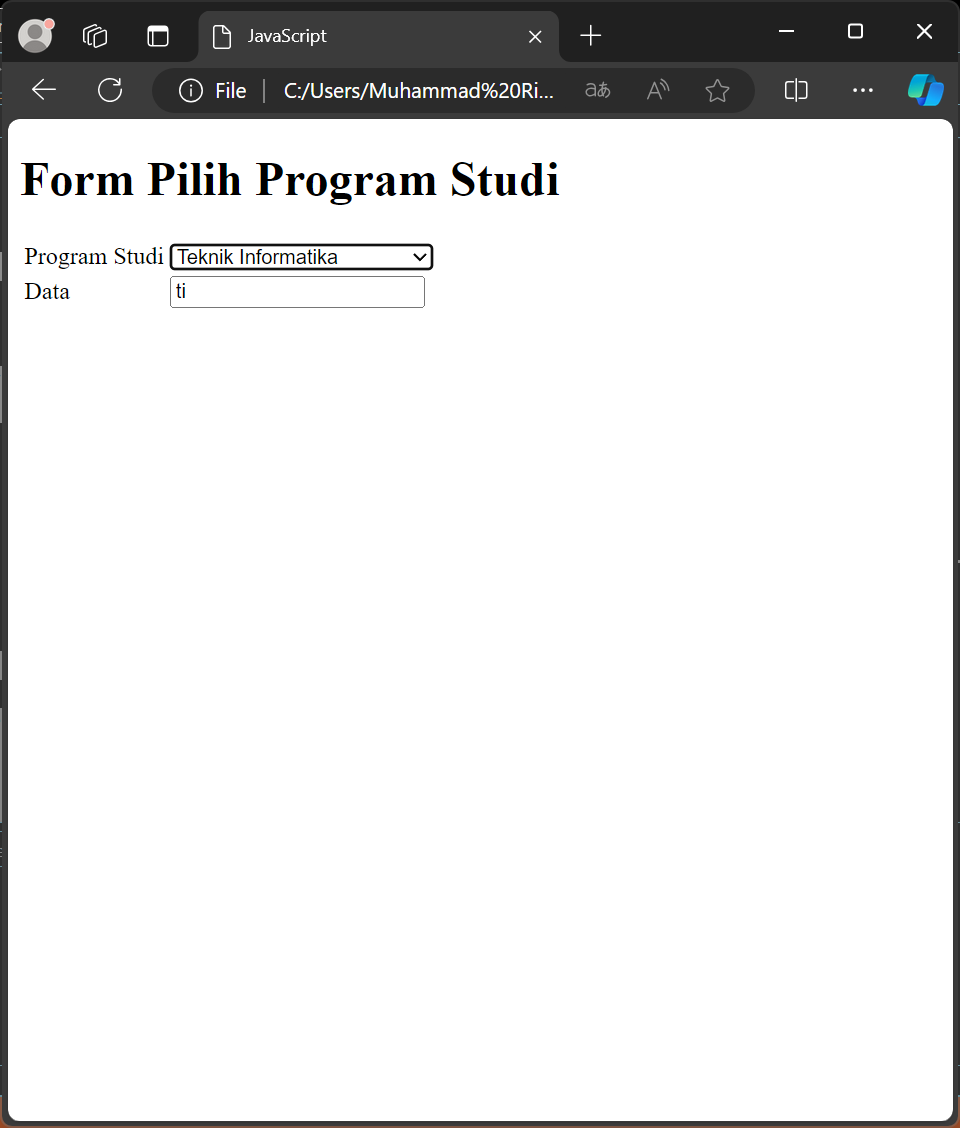
</tr>

</table>

</form>

</body>

</html>



Analisis : Kode HTML di atas membuat formulir untuk memilih program studi menggunakan dropdown menu. Fungsi JavaScript **showData()** mengambil nilai yang dipilih dari dropdown dengan ID "prodi" dan menampilkannya di input teks dengan ID "programStudi". Dropdown memiliki beberapa pilihan program studi, termasuk opsi default "-- Pilih --". Saat pengguna memilih program studi, nilai yang dipilih langsung diperbarui ke input teks di bawahnya. Fitur ini memungkinkan tampilan data secara dinamis, namun nilai yang dikirimkan masih berupa kode singkatan seperti "ti" atau "si", sehingga bisa ditambahkan deskripsi yang lebih jelas untuk pengalaman pengguna yang lebih baik.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>JavaScript</title>

<script>

function penjumlahan(){

bilangan1 = parseFloat(document.getElementById("angka1").value);

bilangan2 = parseFloat(document.getElementById("angka2").value);

total = bilangan1 + bilangan2;

document.getElementById("hasil").value = total;

}

</script>

</head>

<body>

<h1>Penjumlahan</h1>

<form nama = "form1">

<table border = "0">

<tr>

<td><label>Angka 1</label></td>

<td><input type = "number" name = "angka1" id = "angka1" /></td>

</tr>

<tr>

<td colspan =" 2">+</td>

</tr>

<tr>

<td><label>Angka 2</label></td>

<td><input type = "number" name = "angka2" id = "angka2" /></td>

</tr>

<tr>

<td colspan =" 2">

<button type = "button" name = "tampil" value = "tampil"

onclick = "penjumlahan()">=</button>

<button type = "reset" name = "batal" value = "batal">C</button>

</td>

</tr>

<tr>

<td><label>Hasil</label></td>

<td><input type = "number" name = "hasil" id = "hasil" readonly /></td>

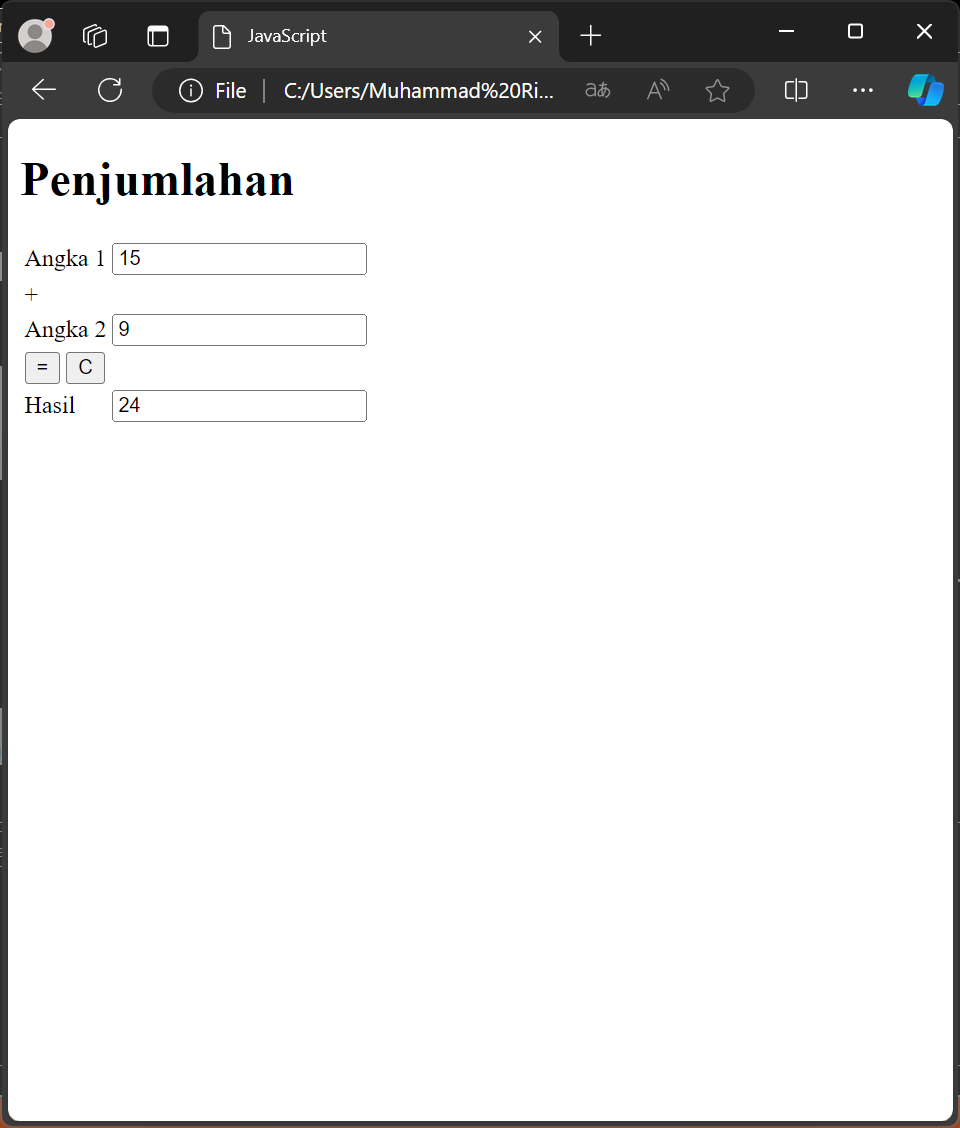
</tr>

</table>

</form>

</body>

</html>



Analisis : Kode HTML di atas membuat aplikasi sederhana untuk melakukan penjumlahan dua angka menggunakan JavaScript. Fungsi **penjumlahan()** membaca nilai dari input dengan ID "angka1" dan "angka2", mengubahnya menjadi tipe angka menggunakan **parseFloat**, lalu menjumlahkan keduanya. Hasil penjumlahan ditampilkan pada input teks dengan ID "hasil". Tombol "=" memanggil fungsi ini untuk melakukan perhitungan, sedangkan tombol "C" mereset semua input.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Pemilihan</title>

</head>

<body>

<script>

nilai = 80;

if (nilai >= 81 && nilai <= 100) nh = "A";

else if (nilai >= 75) nh = "AB";

else if (nilai >= 69) nh = "B";

else if (nilai >= 63) nh = "BC";

else if (nilai >= 57) nh = "C";

else if (nilai >= 51) nh = "CD";

else if (nilai >= 45) nh = "D";

else nh = "E";

document.write("Nilai Anda = " +nh);

</script>

</body>

</html>



Analisis : Kode HTML di atas menggunakan JavaScript untuk menentukan nilai huruf (grade) berdasarkan nilai angka yang telah ditentukan, yaitu 80. Logika dalam skrip menggunakan struktur **if-else** untuk memeriksa rentang nilai dan memberikan hasil sesuai dengan kategori: A, AB, B, BC, C, CD, D, atau E. Dengan nilai 80, skrip menentukan bahwa nilai huruf adalah "AB", yang kemudian ditampilkan langsung di halaman menggunakan **document.write**. Pendekatan ini sederhana namun efektif untuk menunjukkan konsep pemilihan bersyarat dalam JavaScript.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Pemilihan</title>

</head>

<body>

<script>

nilai = 80;

let nh;

switch(true){

case nilai >= 81 && nilai <= 100 :

nh = "A";

break;

case nilai >= 75 :

nh = "AB";

break;

case nilai >= 69 :

nh = "B";

break;

case nilai >= 63 :

nh = "BC";

break;

case nilai >= 57 :

nh = "C";

break;

case nilai >= 51 :

nh = "CD";

break;

case nilai >= 45 :

nh = "D";

break;

default :

nh = "E"

break;

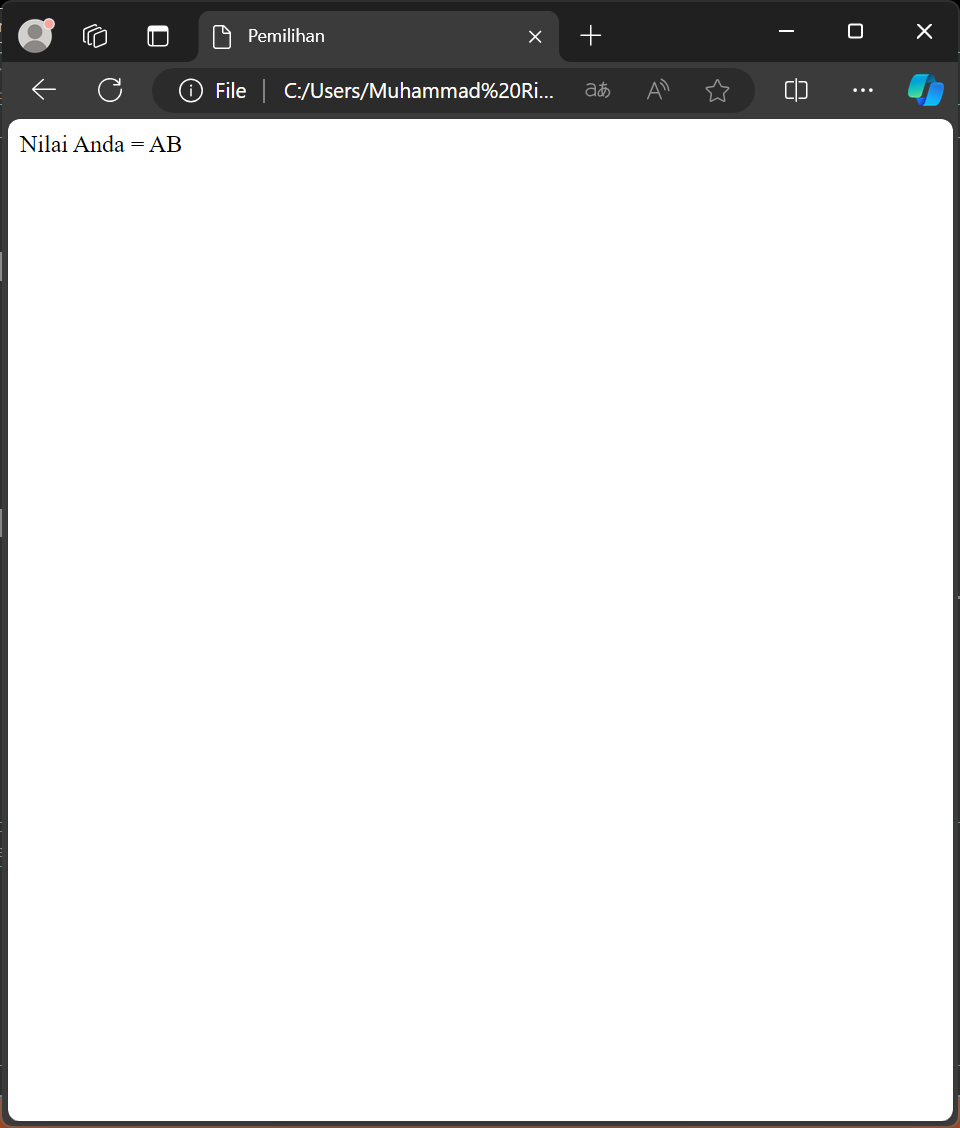
}

document.write("Nilai Anda = " +nh);

</script>

</body>

</html>



Analisis : Kode HTML di atas menggunakan JavaScript dengan struktur switch untuk menentukan nilai huruf (grade) berdasarkan nilai angka yang telah ditentukan, yaitu 80. Logika switch(true) memeriksa kondisi rentang nilai dari yang tertinggi ke terendah, menetapkan grade mulai dari "A" hingga "E". Untuk nilai 80, grade yang ditampilkan adalah "AB". Hasil grade ditampilkan langsung di halaman menggunakan document.write. Pendekatan ini memperlihatkan penggunaan switch sebagai alternatif dari if-else untuk menangani banyak kondisi secara terstruktur.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Perulangan</title>

</head>

<body>

<h1>Perulangan dengan For</h1>

<div id = "hasilFor"></div>

<script>

let hasil = "";

for (let i = 1; i <= 10; i++){

hasil += "Angka : " + i + "<br>";

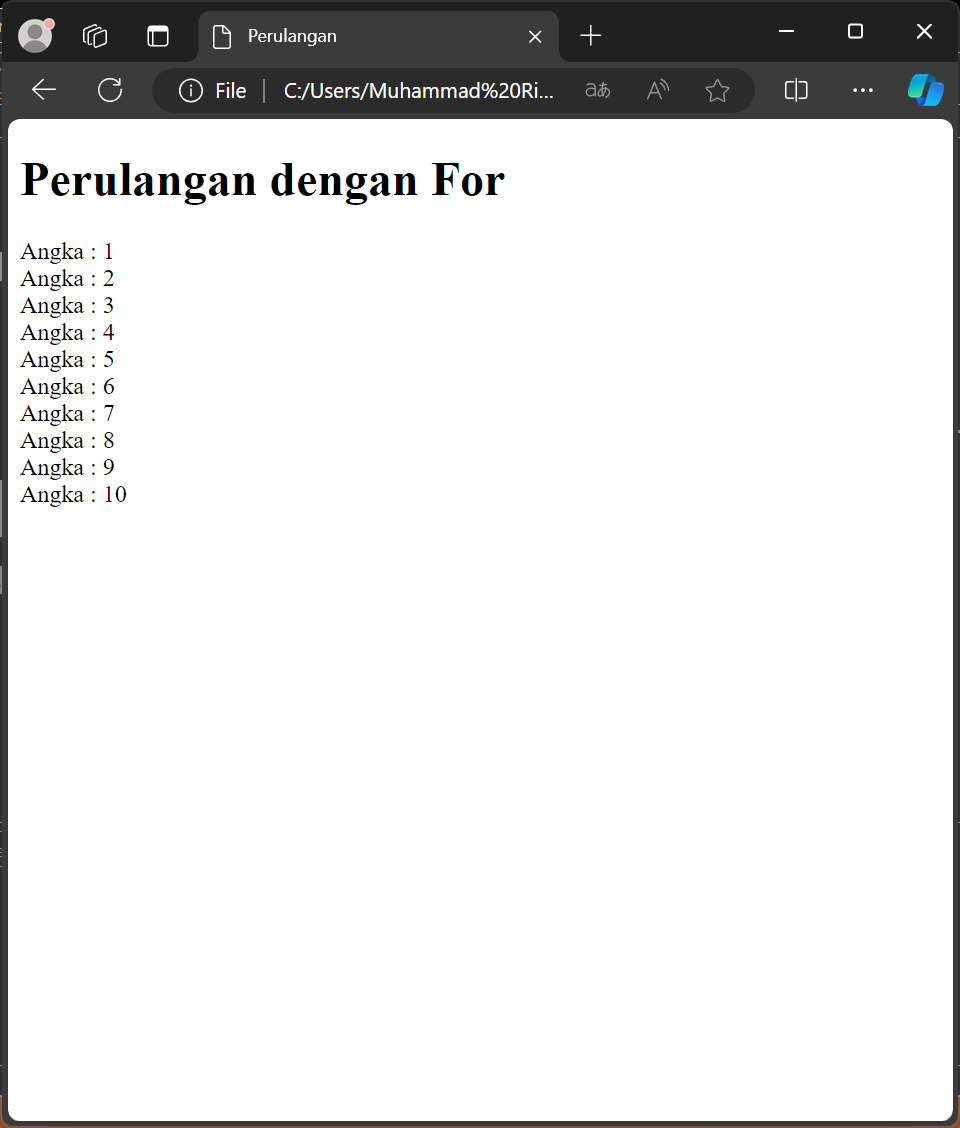
}

document.getElementById("hasilFor").innerHTML = hasil;

</script>

</body>

</html>



Analisis : Kode HTML di atas membuat halaman dengan judul "Perulangan" yang menggunakan elemen <h1> untuk menampilkan teks "Perulangan dengan For" dan sebuah <div> dengan id hasilFor untuk menampilkan hasil perulangan. Di dalam tag <script>, terdapat sebuah perulangan for yang dimulai dari angka 1 hingga 10. Pada setiap iterasi, angka tersebut ditambahkan ke dalam string hasil dengan format "Angka : x" diikuti dengan tag <br> untuk pemisahan baris. Setelah perulangan selesai, hasilnya ditampilkan di dalam elemen dengan id hasilFor menggunakan innerHTML.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Array</title>

</head>

<script>

let buah = ["Anggur", "Apel", "Jeruk", "Mangga"];

buah[4] = "Semangka";

document.write("Buah-Buahan : <br>");

document.write("0 : " + buah[0] + "<br>");

document.write("1 : " + buah[1] + "<br>");

document.write("2 : " + buah[2] + "<br>");

document.write("3 : " + buah[3] + "<br>");

document.write("4 : " + buah[4] + "<br>");

document.write("<br><br>Buah-Buahan dengan for : <br>");

for (let i = 0; i < buah.length; i++){

document.write(i + " : " + buah[i] + "<br>");

}

buah[2] = "Kedondong";

buah[3] = "";

document.write("<br><br>Sekarang Buah-Buahan : <br>");

for (let i = 0; i < buah.length; i++){

document.write(i + " : " + buah[i] + "<br>");

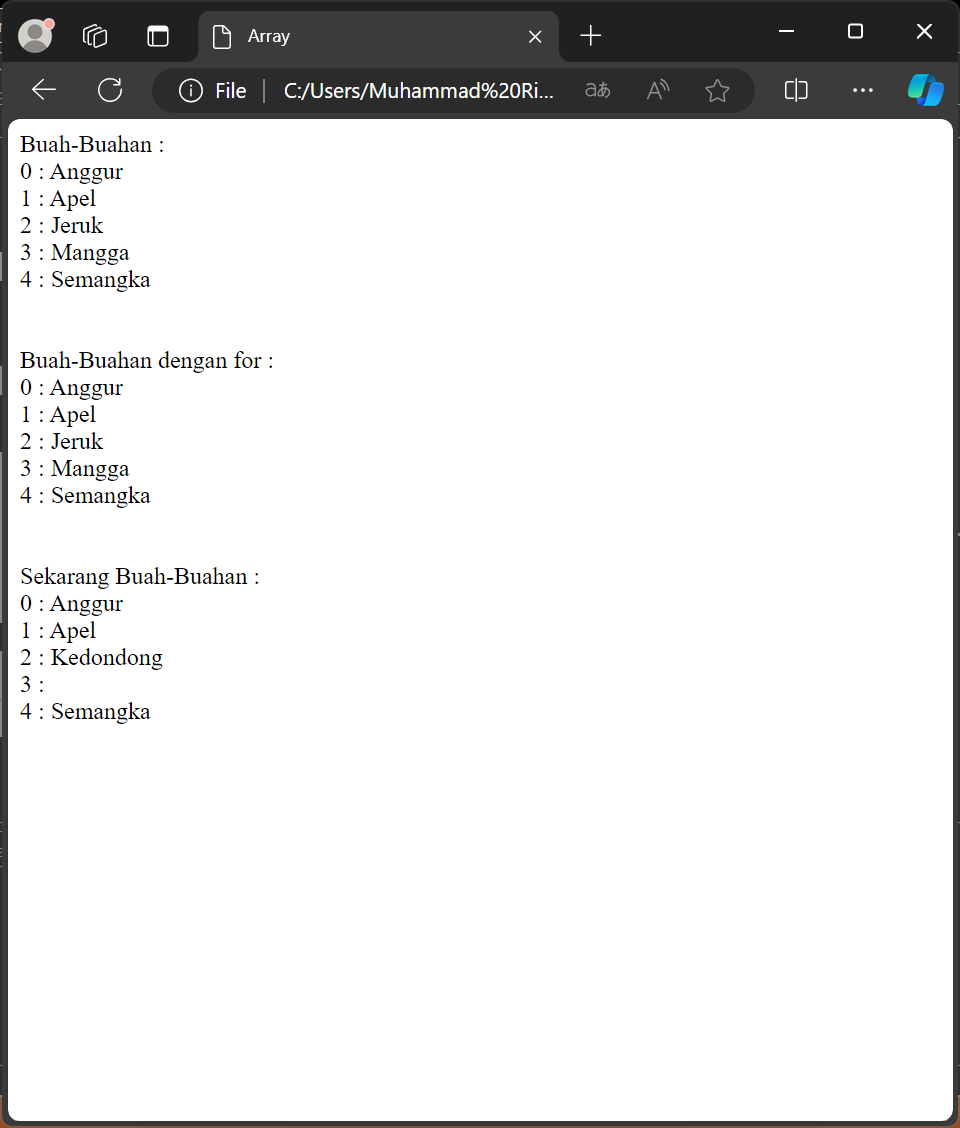
}

</script>

<body>

</body>

</html>



Analisis : Kode HTML di atas menampilkan penggunaan array di JavaScript dengan array buah yang berisi empat jenis buah. Elemen array diakses dan ditampilkan menggunakan document.write untuk menampilkan nilai setiap elemen berdasarkan indeks. Pada bagian pertama, semua elemen array ditampilkan satu per satu, kemudian elemen ke-4 diubah menjadi "Semangka". Selanjutnya, sebuah perulangan for digunakan untuk menampilkan seluruh isi array. Setelah itu, elemen ke-3 diubah menjadi "Kedondong", dan elemen ke-4 dihapus (set menjadi string kosong). Terakhir, array yang telah diperbarui ditampilkan kembali menggunakan perulangan for.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Array 1 Dimensi</title>

<style>

body{

font-family : Arial, sans-serif;

margin : 20px;

}

#output{

margin-top : 20px;

}

</style>

</head>

<body>

<h1>Contoh Penggunaan Array 1 Dimensi</h1>

<input type = "text" id = "inputValue" placeholder="Masukan Nama Buah-Buahan : "/>

<button onclick = "tambah()">Tambah (Push)</button>

<button onclick = "hapus()">Hapus (Pop)</button>

<div id = "output"></div>

<script>

let buah = [];

function tambah(){

const input = document.getElementById("inputValue").value;

if(input){

buah.push(input);

document.getElementById("inputValue").value = "";

tampilkan();

}

}

function hapus(){

if(buah.length > 0){

buah.pop();

tampilkan();

} else {

alert("Buah-Buahan Sudah Kosong!");

}

}

function tampilkan(){

const output = document.getElementById("output")

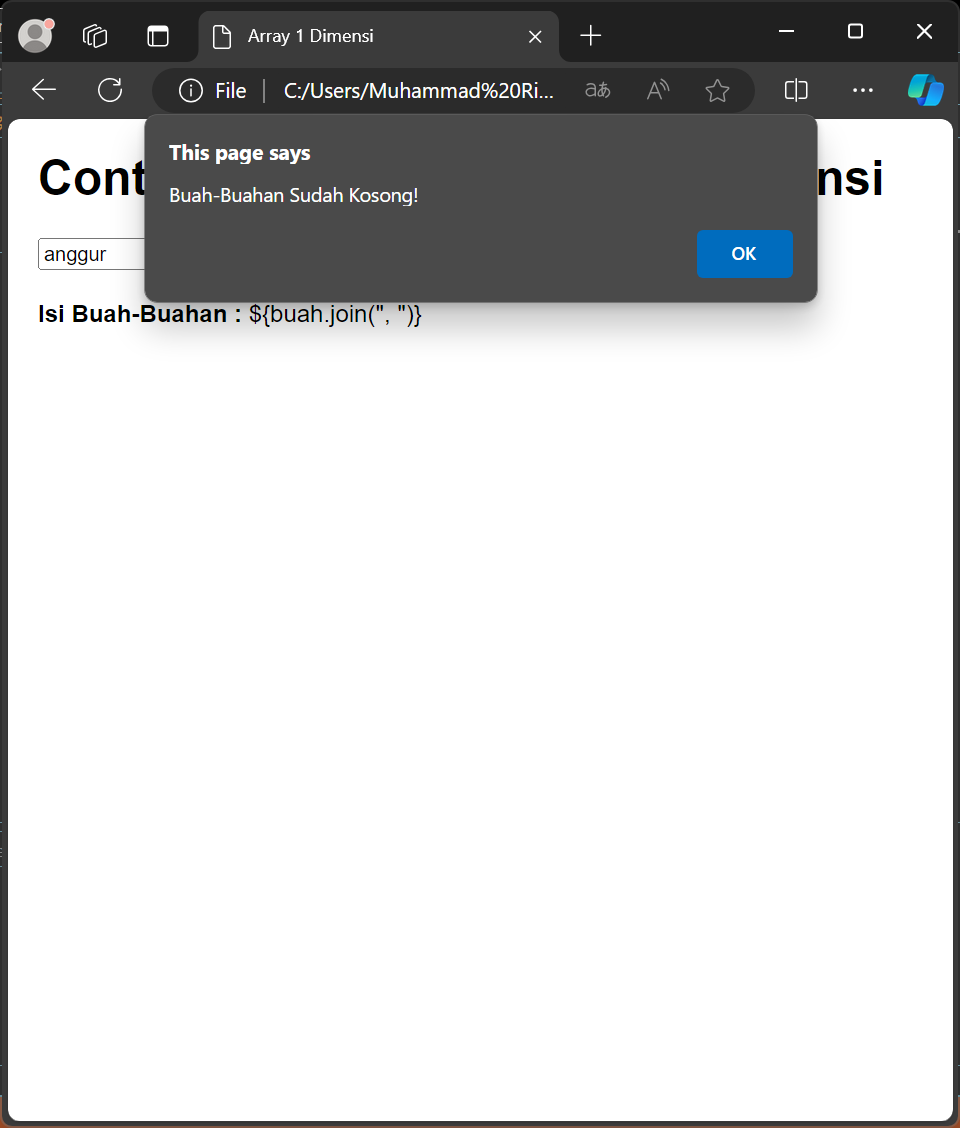
output.innerHTML = '<strong>Isi Buah-Buahan : </strong> ${buah.join(", ")}';

}

</script>

</body>

</html>



Analisis : Kode HTML di atas menampilkan contoh penggunaan array satu dimensi dalam JavaScript, di mana pengguna dapat menambah atau menghapus elemen dari array buah. Terdapat dua tombol: "Tambah (Push)" untuk menambahkan nilai ke dalam array dan "Hapus (Pop)" untuk menghapus elemen terakhir dari array. Nilai yang dimasukkan oleh pengguna melalui input text ditambahkan ke dalam array menggunakan metode push(), dan elemen terakhir dapat dihapus dengan metode pop(). Setelah setiap perubahan, array yang diperbarui ditampilkan di dalam elemen <div> dengan id output. Ada juga pengecekan untuk memastikan array tidak kosong sebelum melakukan penghapusan. Namun, terdapat kesalahan pada kode tampilkan(), di mana penggunaan template literal pada innerHTML seharusnya menggunakan tanda backticks (`), bukan tanda kutip biasa.



<!DOCTYPE html>

<html lang = "id">

<head>

<meta charset = "UTF-8" />

<meta name = "viewport" content = "width=device-width", initial-scale = 1.0>

<title>Array 1 Dimensi</title>

<style>

body{

font-family : Arial, sans-serif;

margin : 20px;

}

#output{

margin-top : 20px;

}

</style>

</head>

<body>

<h1>Contoh Array 2 Dimensi dan 3 Dimensi</h1>

<button onclick = "tampilkanArray2D()">Tampilkan Array 2D</button>

<button onclick = "tampilkanArray3D()">Tampilkan Array 3D</button>

<div id = "output"></div>

<script>

const array2D = [

["Anggur", "Apel", "Jeruk"],

["Mangga", "Pisang", "Kiwi"],

["Semangka", "Nanas", "Stroberi"]

];

const array3D = [

[

["Anggur", "Apel"],

["Jeruk", "Mangga"]

],

[

["Pisang", "Kiwi"],

["Semangka", "Nanas"]

],

[

["Stroberi", "Blueberry"],

["Raspberry", "Cherry"]

]

];

function tampilkanArray2D(){

const output = document.getElementById("output");

let result = "<strong>Isi Array 2D : </strong><br>";

array2D.forEach((row) => {

result += row.join(", ") + "<br>";

});

output.innerHTML = result;

}

function tampilkanArray3D(){

const output = document.getElementById("output");

let result = "<strong>Isi Array 3D : </strong><br>";

array3D.forEach((matrix) => {

matrix.forEach((row) => {

result += row.join(", ") + "<br>";

});

result += "<br>";

});

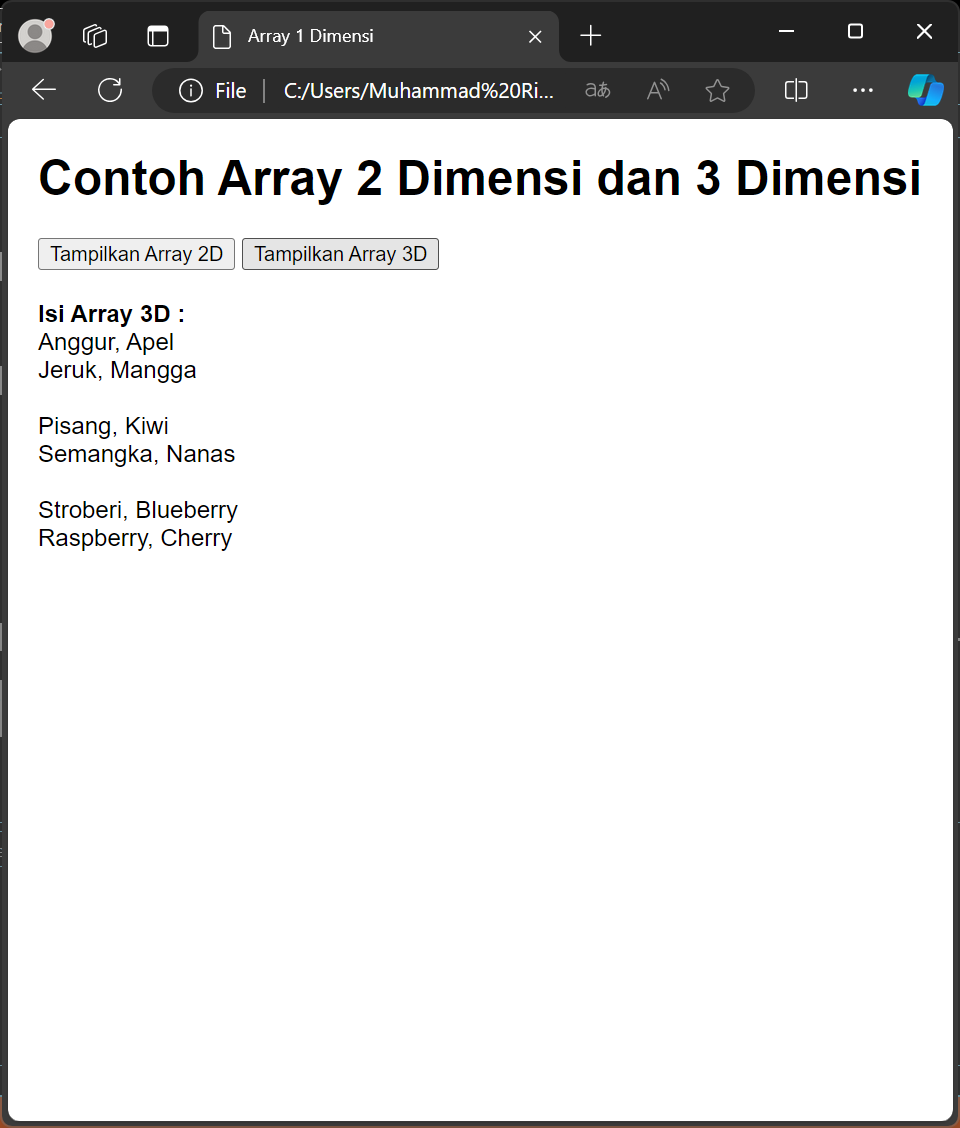
output.innerHTML = result;

}

</script>

</body>

</html>



Analisis : Kode HTML di atas menunjukkan penggunaan array dua dimensi (2D) dan tiga dimensi (3D) dalam JavaScript. Array 2D berisi tiga sub-array yang masing-masing mewakili daftar buah-buahan, sedangkan array 3D berisi tiga array yang masing-masing berisi dua array yang mewakili kelompok buah. Dua tombol di halaman memungkinkan pengguna untuk menampilkan isi array 2D atau 3D. Fungsi tampilkanArray2D() menampilkan isi array 2D dengan menggunakan metode forEach untuk mengiterasi setiap baris dan menampilkan elemen-elemen dalam satu baris. Fungsi tampilkanArray3D() bekerja serupa, namun dengan tambahan iterasi dua tingkat untuk mengakses elemen-elemen dalam array 3D. Hasilnya ditampilkan dalam elemen <div> dengan id output.

**Tugas**

1. Login.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Login Form</title>

</head>

<body>

    <h2>Login</h2>

    <form id="loginForm">

        <label for="username">Username:</label>

        <input type="text" id="username" required><br><br>

        <label for="password">Password:</label>

        <input type="password" id="password" required><br><br>

        <button type="button" onclick="validateLogin()">Login</button>

    </form>

    <script>

        function validateLogin() {

            const username = document.getElementById('username').value;

            const password = document.getElementById('password').value;

            if (username === "admin" && password === "123") {

                alert("Login Sukses");

                window.location.href = "index.html";

            } else {

                alert("Login Gagal");

                window.location.href = "Login.html";

            }

        }

    </script>

</body>

</html>

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Selamat Datang</title>

</head>

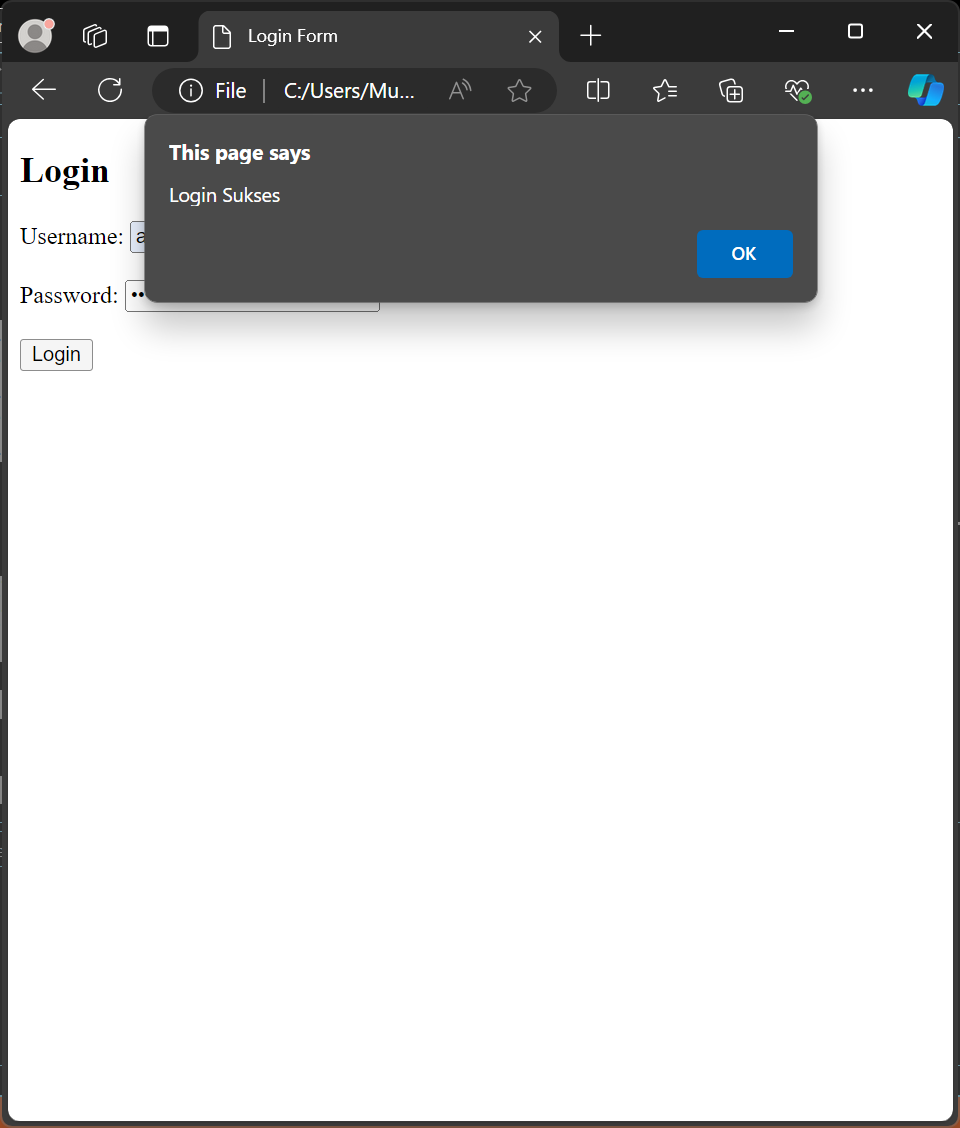
<body>

    <h1>Selamat Datang</h1>

    <p>Selamat datang di halaman web kami!</p>

</body>

</html>



**Analisis :**

Kode HTML di atas adalah formulir login sederhana yang memanfaatkan JavaScript untuk validasi sisi klien. Formulir ini memiliki dua input utama, yaitu username dan password, dengan validasi yang mengarahkan pengguna ke halaman "index.html" jika login berhasil atau kembali ke "Login.html" jika gagal. Validasi dilakukan dengan membandingkan input dengan kredensial statis ("admin" untuk username dan "123" untuk password), serta memberikan notifikasi menggunakan alert`.

1. TambahData.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Tambah Data</title>

</head>

<body>

<h2>Tambah Data Kamar</h2>

<form id="roomForm">

<label for="roomType">Tipe Kamar:</label>

<input type="text" id="roomType" required><br><br>

<label for="price">Harga:</label>

<input type="number" id="price" required><br><br>

<button type="button" onclick="validateRoom()">Tambah Kamar</button>

</form>

<h2>Tambah Data Fasilitas</h2>

<form id="facilityForm">

<label for="facilityName">Nama Fasilitas:</label>

<input type="text" id="facilityName" required><br><br>

<button type="button" onclick="validateFacility()">Tambah Fasilitas</button>

</form>

<script>

function validateRoom() {

const roomType = document.getElementById('roomType').value;

const price = document.getElementById('price').value;

if (roomType && price > 0) {

alert("Data Kamar Ditambahkan");

} else {

alert("Isi semua data dengan benar.");

}

}

function validateFacility() {

const facilityName = document.getElementById('facilityName').value;

if (facilityName) {

alert("Data Fasilitas Ditambahkan");

} else {

alert("Nama fasilitas harus diisi.");

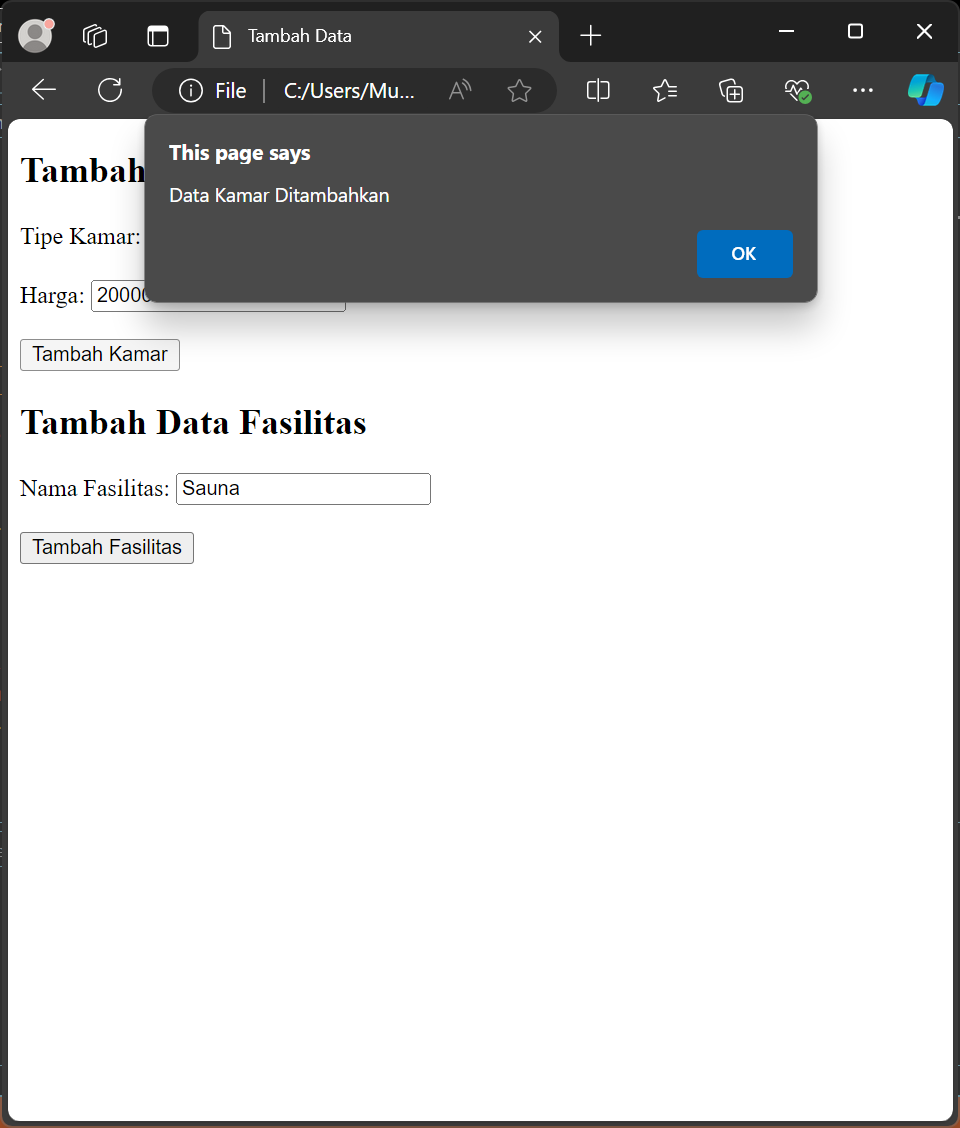
}

}

</script>

</body>

</html>



**Analisis :**

Formulir ini berfungsi untuk menambah data kamar hotel dengan elemen input yang cukup lengkap. Gaya CSS sederhana tetapi efektif, menonjolkan fungsi dengan perbedaan warna pada tombol, serta menampilkan form secara terpusat.

1. FormPemesananHotel.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Form Pemesanan Hotel</title>

    <link rel="stylesheet" href="../Modul4/formhotel.css" />

</head>

<body>

    <div class="form-container">

        <h1>Form Pemesanan Hotel</h1>

        <form method="get" onsubmit="calculateTotal(event)">

            <div class="form-group">

                <label for="idPemesan">ID Pemesan : </label>

                <input type="text" id="idPemesan" name="idPemesan" maxlength="100" required />

            </div>

            <div class="form-group">

                <label for="namaPemesan">Nama Pemesan : </label>

                <input type="text" id="namaPemesan" name="namaPemesan" maxlength="100" required />

            </div>

            <div class="form-group">

                <label for="noIdentitas">Nomor Identitas : </label>

                <input type="number" id="noIdentitas" name="noIdentitas" maxlength="60" required />

            </div>

            <div class="form-group">

                <label for="tipeKamar">Tipe Kamar : </label>

                <select id="tipeKamar" name="tipeKamar" required>

                    <option value="" selected>---Pilih---</option>

                    <option value="Standard">Standard</option>

                    <option value="Deluxe">Deluxe</option>

                    <option value="Family">Family</option>

                </select>

            </div>

            <div class="form-group">

                <label for="tglPesanan">Tanggal Pesanan : </label>

                <input type="date" id="tglPesanan" name="tglPesanan" required />

            </div>

            <div class="form-group">

                <label for="durasiMenginap">Durasi Menginap : </label>

                <input type="number" id="durasiMenginap" name="durasiMenginap" maxlength="3" required />

                <span> Hari</span>

            </div>

            <div class="form-group">

                <label for="breakfast">Termasuk Breakfast </label>

                <input type="checkbox" id="breakfast" name="breakfast" value="Ya" /> Ya

            </div>

            <div class="form-group">

                <label for="totalBiaya">Total Biaya : </label>

                <input type="text" id="totalBayar" name="totalBiaya" readonly />

            </div>

            <div class="button-group">

                <button type="submit" class="save-button">Simpan</button>

                <button type="reset" class="cancel-button">Batal</button>

            </div>

        </form>

    </div>

    <script>

        function calculateTotal(event) {

            event.preventDefault(); // Mencegah pengiriman formulir default

            const tipeKamar = document.getElementById("tipeKamar").value;

            const durasiMenginap = parseInt(document.getElementById("durasiMenginap").value) || 0;

            const breakfast = document.getElementById("breakfast").checked;

            let hargaKamar = 0;

            if (tipeKamar === "Standard") {

                hargaKamar = 300000;

            } else if (tipeKamar === "Deluxe") {

                hargaKamar = 500000;

            } else if (tipeKamar === "Family") {

                hargaKamar = 700000;

            }

            let total = hargaKamar \* durasiMenginap;

            // Diskon jika durasi menginap lebih dari 3 hari

            if (durasiMenginap > 3) {

                total \*= 0.9;

            }

            // Tambahkan biaya breakfast

            if (breakfast) {

                total += 80000;

            }

            // Tampilkan total biaya di input Total Bayar

            document.getElementById("totalBayar").value = "Rp " + total.toLocaleString();

        }

    </script>

</body>

</html>

formhotel.css

body {

    background-color : #ffffff;

    display : flex;

    justify-content : center;

    align-items : center;

    height : 80vh;}

.form-container {

    background-color : #ffffff;

    padding : 20px;

    width : 100%;}

h1 {

    color : #000000;}

.form-group {

    display : flex;

    margin-bottom : 15px;}

label {

    width : 160px;

    margin-right : 10px;

    color : #000000;}

input[type = "text"], input[type = "number"], input[type = "date"], select {

    flex-grow : 1;

    padding : 10px;

    border-radius : 5px;

    border : 1px solid #848484;

    box-sizing : border-box;}

.button-group {

    text-align : center;

    margin-top : 20px;}

button {

    padding : 10px 20px;

    font-size : 16px;

    border-radius : 5px;

    cursor : pointer;

    margin : 5px;}

.save-button {

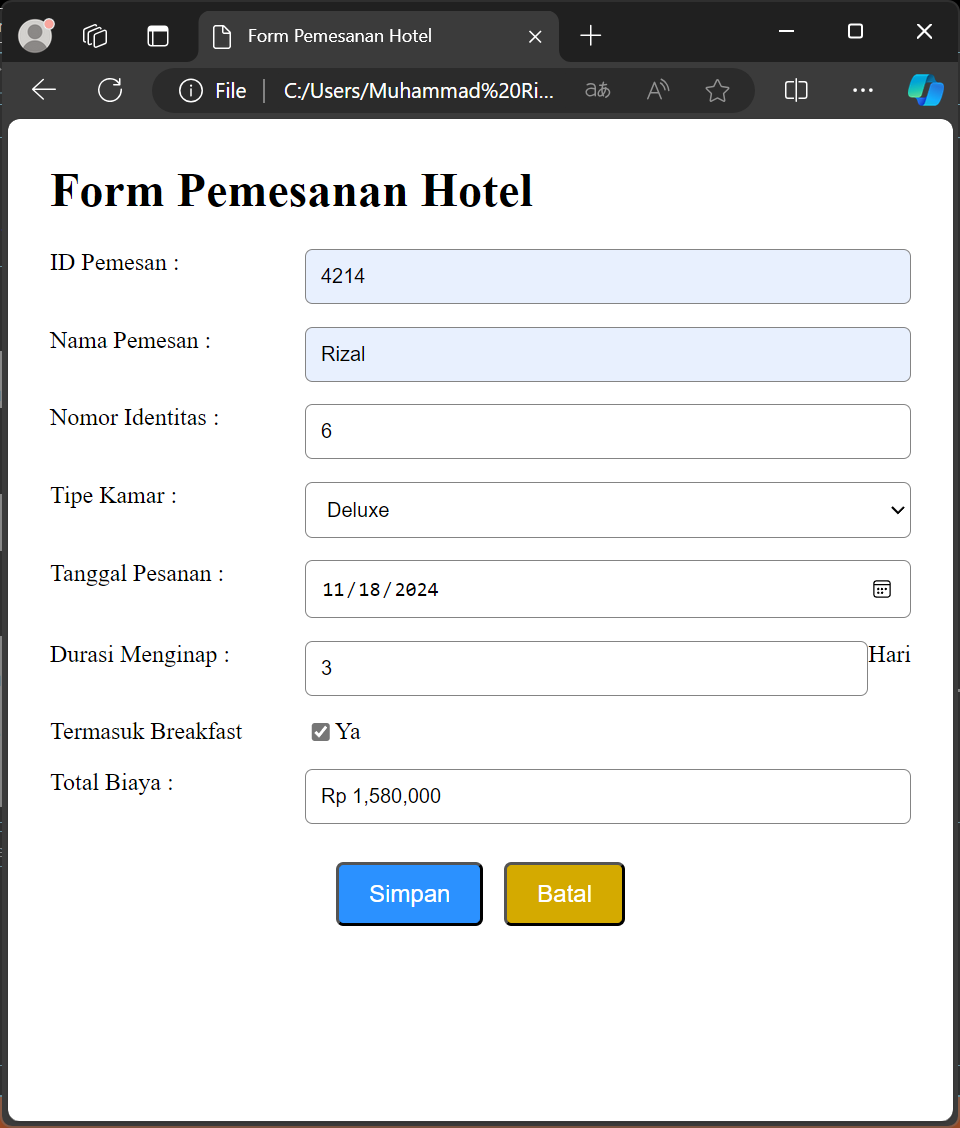
    background-color : #2b91ff;

    color : rgb(255, 255, 255);}

.cancel-button {

    background-color : #d4aa00;

    color : rgb(255, 255, 255);}



**Analisis :**

Formulir pemesanan hotel diatas memiliki struktur yang terorganisasi dengan baik, mencakup elemen-elemen penting seperti input teks, dropdown, dan checkbox. Validasi dasar dengan atribut required memastikan beberapa data utama seperti nama, nomor identitas, dan durasi menginap tidak dibiarkan kosong. Logika perhitungan total biaya sudah mencakup fitur diskon otomatis untuk durasi menginap lebih dari 3 hari dan biaya tambahan untuk opsi breakfast. Namun, dropdown tipe kamar memiliki ketidakkonsistenan nilai antara HTML (STD, DLX, FMY) dan logika di JavaScript (Standard, Deluxe, Family), yang dapat menyebabkan error.