

# **Laporan Praktikum Pemrograman Web 1**

## **Modul 4 & 5**



**Nama : Asep haryana saputra**

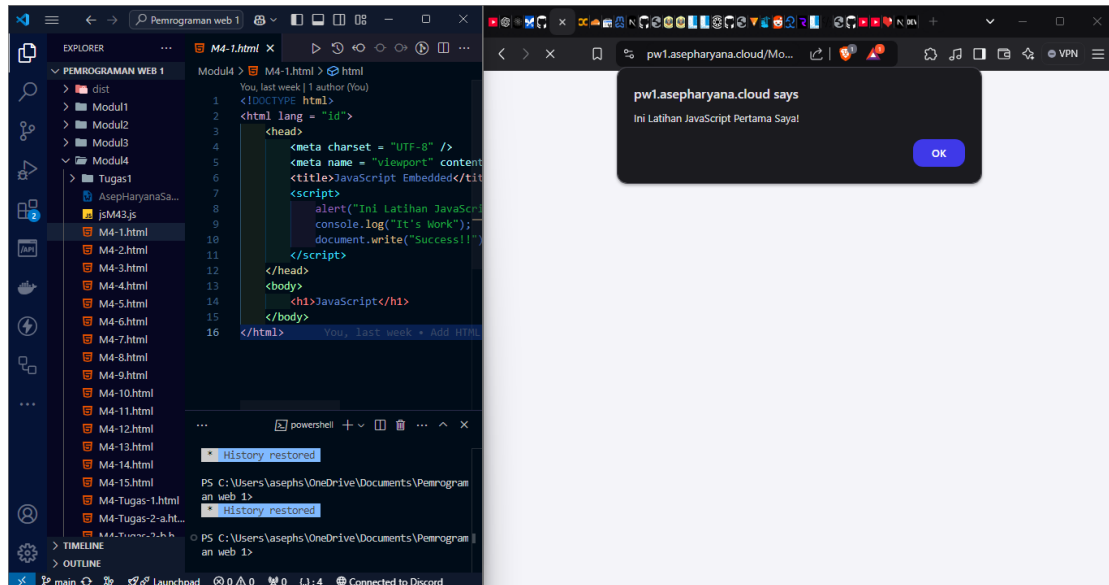
**NIM : 20230810043**

**Kelas : TINFC-2023-04**

**Teknik Informatika  
Fakultas Ilmu Komputer  
Universitas Kuningan**

# Praktikum

## Web1



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-1.html>

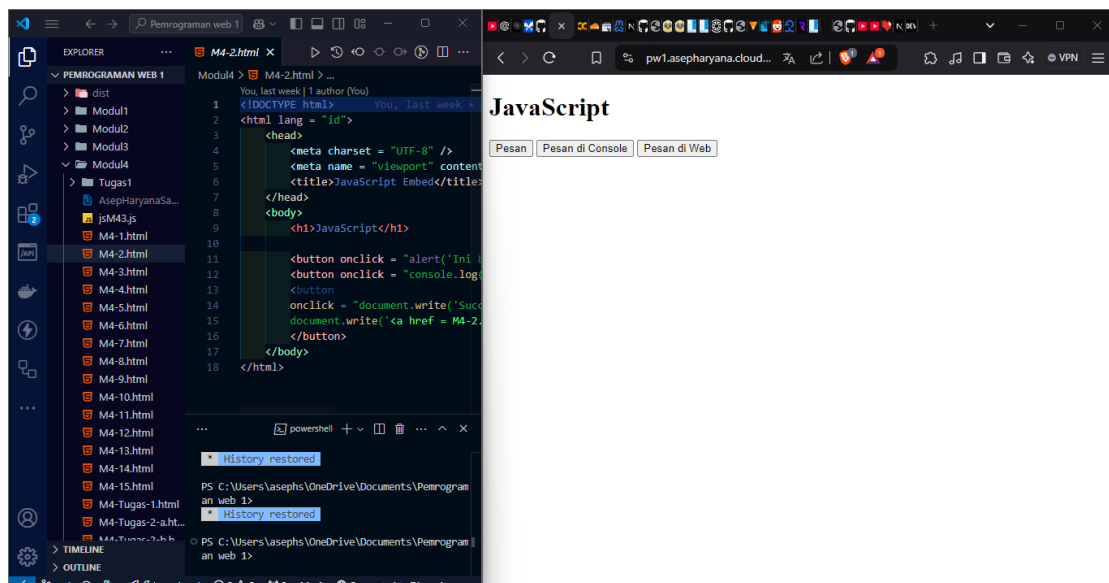
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-1.html>

### Penjelasan:

Menampilkan *alert box* dengan pesan "Ini Latihan JavaScript Pertama Saya!" saat halaman dimuat. Setelah pengguna menutup *alert box*, teks "Success!!" akan ditampilkan di halaman, menimpa tag `<h1>JavaScript</h1>` yang ada di dalam `<body>`. Selain itu, pesan "It's Work" akan tercetak di konsol web browser, yang berguna untuk debugging. Jadi, secara visual pengguna hanya melihat teks "Success!!", sementara di konsol developer akan terlihat pesan "It's Work"

## Web2



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-2.html>

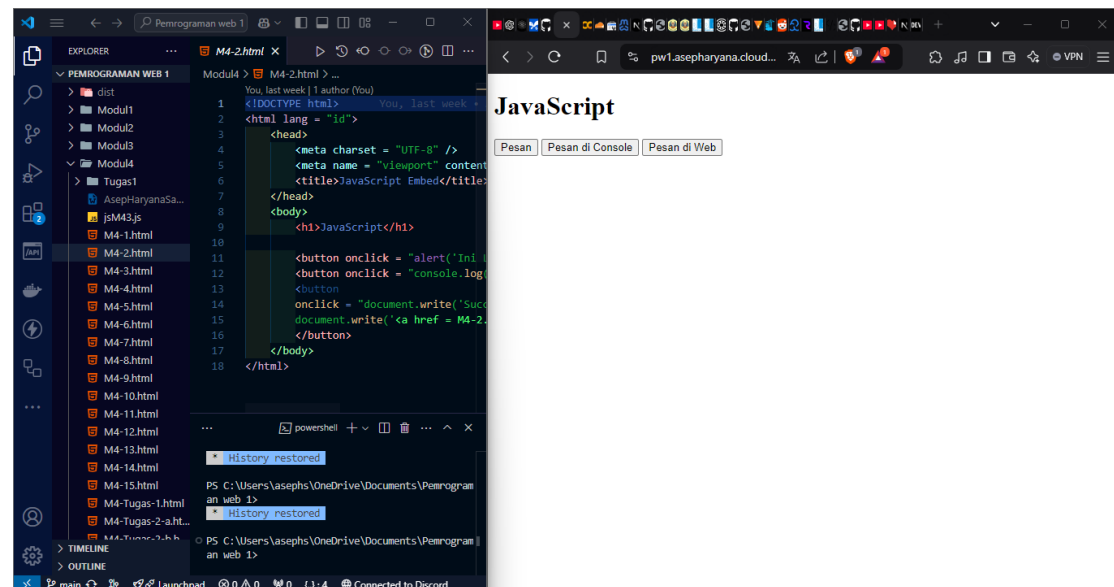
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-2.html>

### Penjelasan:

menampilkan tiga tombol. Tombol pertama, ketika diklik, akan memunculkan *alert box* dengan pesan "Ini Latihan JavaScript Pertama Saya!". Tombol kedua, ketika diklik, akan mencetak pesan "Its Work" di konsol web browser (tidak terlihat oleh pengguna biasa). Tombol ketiga, ketika diklik, akan menimpa seluruh isi halaman web dengan teks "Success" dan sebuah tautan bertuliskan "Kembali" yang mengarah ke file "M4-2.html".

### Web3



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-3.html>

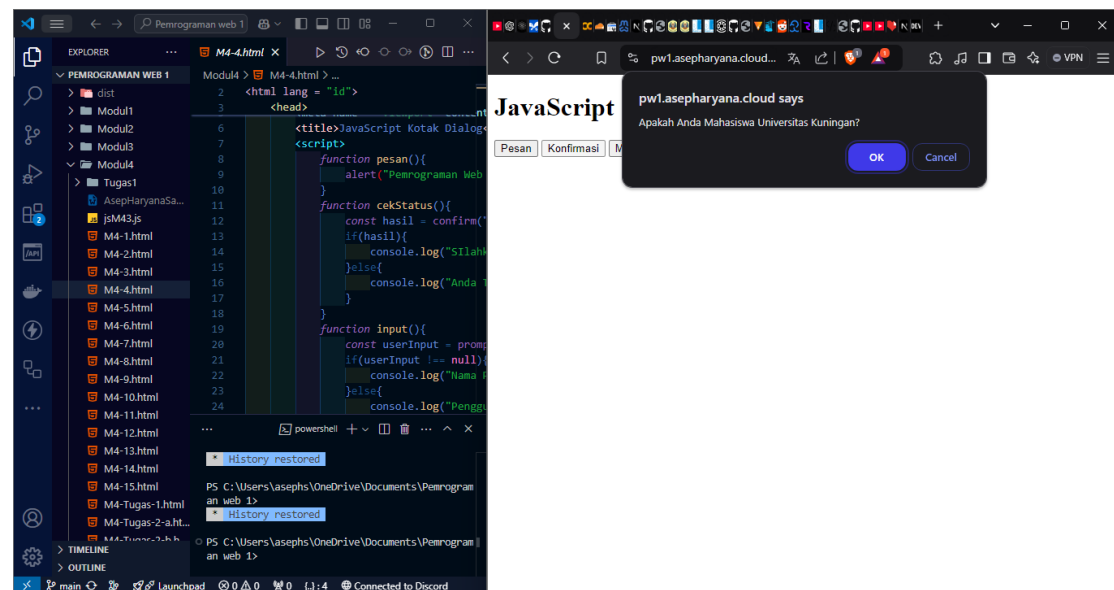
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-3.html>

### Penjelasan:

terdiri dari sebuah halaman HTML dan file JavaScript eksternal yang saling terhubung. Halaman HTML menampilkan tiga tombol, masing-masing dikonfigurasi untuk menjalankan fungsi JavaScript tertentu saat diklik. Koneksi ke JavaScript eksternal dibentuk melalui tag `<script src="jsM43.js"></script>` di bagian head HTML, yang memuat kode JavaScript dari file "jsM43.js". File JavaScript ini mendefinisikan tiga fungsi: `pesan()`, `pesanConsole()`, dan `pesanDok()`. Fungsi `pesan()` akan menampilkan *alert box* ketika tombol pertama diklik. Fungsi `pesanConsole()` akan menulis pesan ke konsol developer browser saat tombol kedua diklik, berguna untuk keperluan debugging. Terakhir, fungsi `pesanDok()` akan menulis teks dan tautan ke halaman HTML, menimpa konten yang ada, ketika tombol ketiga diklik. Dengan demikian, HTML menyediakan antarmuka pengguna berupa tombol, sementara JavaScript eksternal menangani logika dan aksi yang terjadi saat tombol-tombol tersebut diklik.

## Web4



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-4.html>

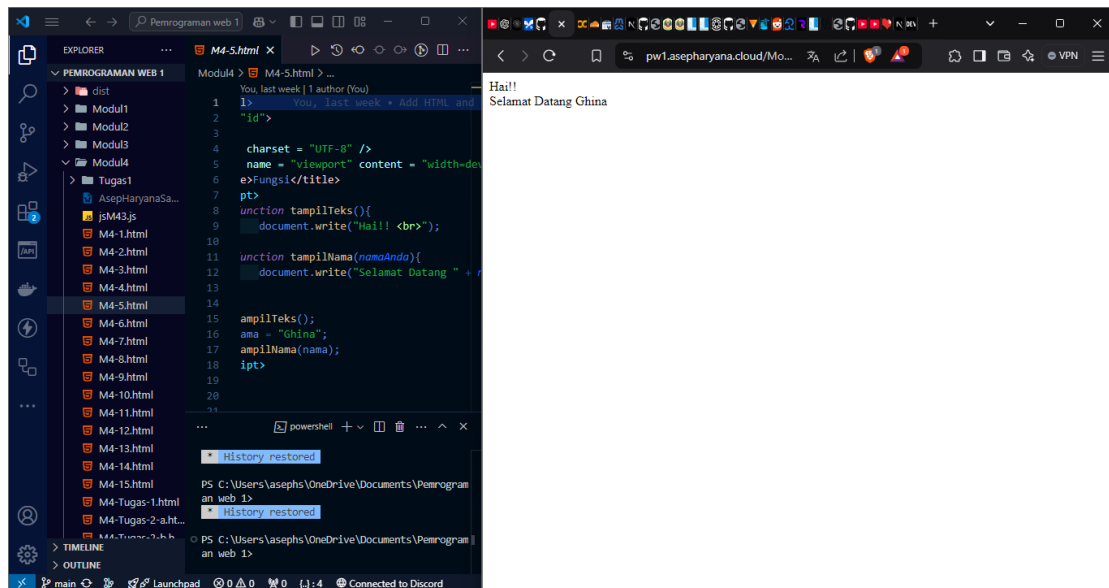
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-4.html>

### Penjelasan:

mendemonstrasikan tiga jenis kotak dialog JavaScript. Tombol pertama, saat diklik, menjalankan fungsi pesan() yang menampilkan *alert box* dengan pesan "Pemrograman Web 1". Tombol kedua, saat diklik, menjalankan fungsi cekStatus(). Fungsi ini menampilkan kotak konfirmasi (*confirm box*) yang menanyakan "Apakah Anda Mahasiswa Universitas Kuningan?". Jika pengguna menekan "OK", pesan "Silahkan Mengakses Materi yang Telah Disediakan" akan dicetak di konsol. Jika pengguna menekan "Cancel", pesan "Anda Tidak Berhak Mengakses!" akan dicetak di konsol. Tombol ketiga, saat diklik, menjalankan fungsi input(). Fungsi ini menampilkan kotak input (*prompt box*) yang meminta pengguna memasukkan namanya. Jika pengguna memasukkan nama dan menekan "OK", nama tersebut akan dicetak di konsol. Jika pengguna menekan "Cancel", pesan "Pengguna Membatalkan Input." akan dicetak di konsol.

## Web5



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-5.html>

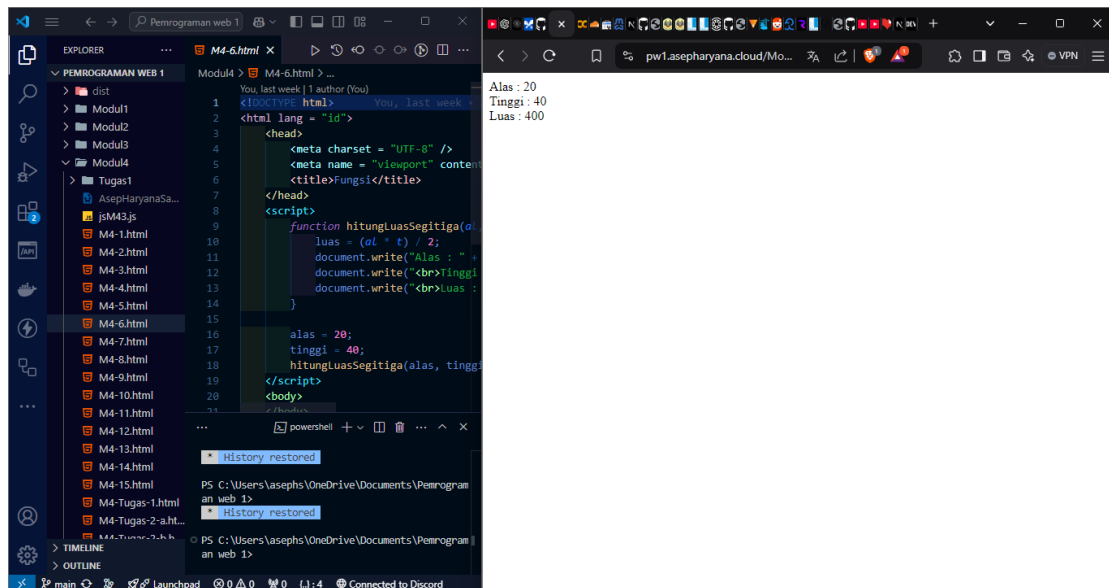
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-5.html>

### Penjelasan:

mendemonstrasikan penggunaan fungsi dalam JavaScript. Kode tersebut mendefinisikan dua fungsi: tampilTeks() dan tampilNama(namaAnda). Fungsi tampilTeks() menulis "Hai!!" diikuti baris baru ke halaman web. Fungsi tampilNama(namaAnda) menerima sebuah argumen namaAnda dan menulis "Selamat Datang " diikuti nilai dari namaAnda ke halaman web. Setelah mendefinisikan fungsi, kode tersebut langsung memanggil tampilTeks(), yang akan menampilkan "Hai!!<br>". Kemudian, variabel nama diinisialisasi dengan nilai "Ghina", dan fungsi tampilNama(nama) dipanggil dengan nama sebagai argumen, yang akan menampilkan "Selamat Datang Ghina" di halaman web. Hasil akhirnya, halaman web hanya akan menampilkan teks "Hai!!<br>Selamat Datang Ghina". Tag <h1>JavaScript</h1> dari body HTML ditimpa oleh document.write.

## Web6



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-6.html>

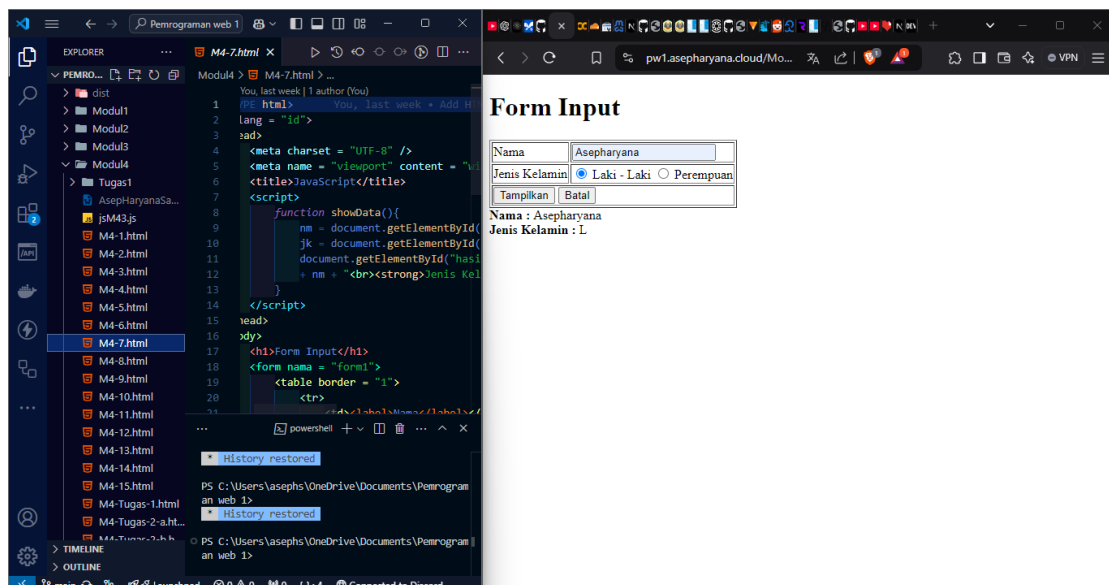
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-6.html>

### Penjelasan:

menghitung dan menampilkan luas segitiga menggunakan JavaScript. Di dalam tag `<script>`, terdapat fungsi `hitungLuasSegitiga(al, t)` yang menerima alas (`al`) dan tinggi (`t`) segitiga sebagai argumen. Fungsi ini menghitung luas segitiga dengan rumus  $(al * t) / 2$  dan kemudian menuliskan alas, tinggi, dan luas yang dihitung ke halaman web menggunakan `document.write()`. Di luar fungsi, variabel `alas` diinisialisasi dengan nilai 20 dan `tinggi` dengan nilai 40. Kemudian, fungsi `hitungLuasSegitiga()` dipanggil dengan `alas` dan `tinggi` sebagai argumen. Sebagai hasilnya, halaman web akan menampilkan: "Alas : 20<br>Tinggi : 40<br>Luas : 400".

## Web7



**Kode:**

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-7.html>

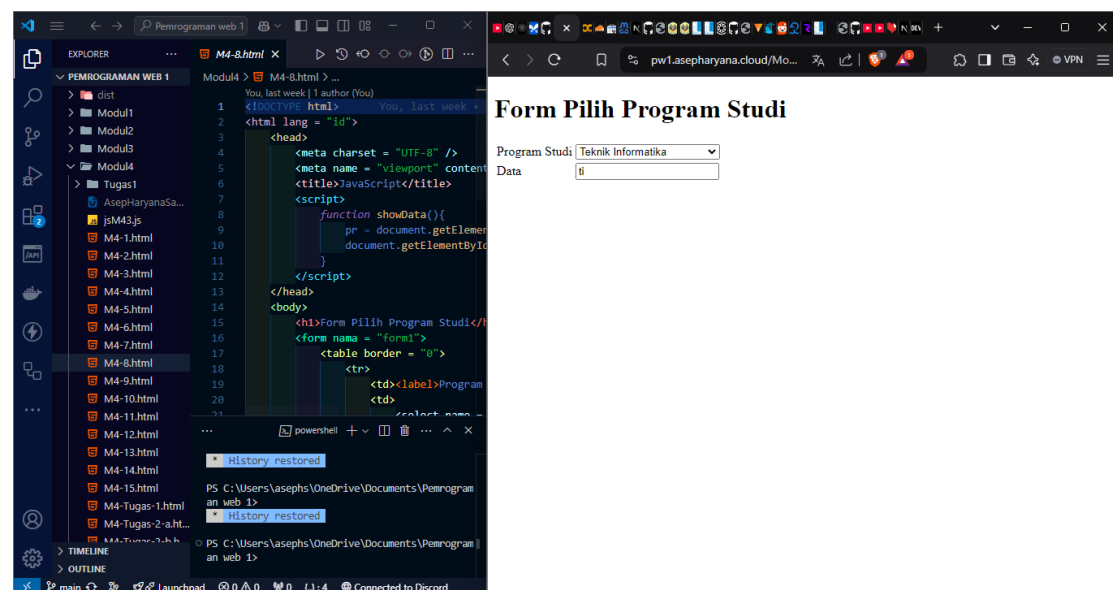
**Demo:**

<https://pw1.asepharyana.cloud/Modul4/M4-7.html>

**Penjelasan:**

membuat sebuah form sederhana untuk memasukkan nama dan jenis kelamin, lalu menampilkan data yang dimasukkan tersebut. Form tersebut berisi dua input: sebuah teks field untuk nama (dengan id "nama") dan dua radio button untuk jenis kelamin (dengan id "jenKel" dan nilai "L" untuk Laki-laki dan "P" untuk Perempuan). Terdapat dua tombol: "Tampilkan" dan "Batal". Tombol "Batal" mereset form ke nilai awal. Tombol "Tampilkan" menjalankan fungsi JavaScript showData().

Fungsi showData() mengambil nilai dari input "nama" dan "jenKel" menggunakan document.getElementById().value. Kemudian, fungsi ini memperbarui konten dari elemen HTML dengan id "hasil" dengan teks yang menampilkan nama dan jenis kelamin yang telah dimasukkan. Hasilnya ditampilkan dalam format yang diformat dengan tag <strong>. Jadi, ketika pengguna mengisi form dan mengklik "Tampilkan", data yang dimasukkan akan ditampilkan di bawah form di dalam elemen div dengan id "hasil".

**Web8****Kode:**

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-8.html>

**Demo:**

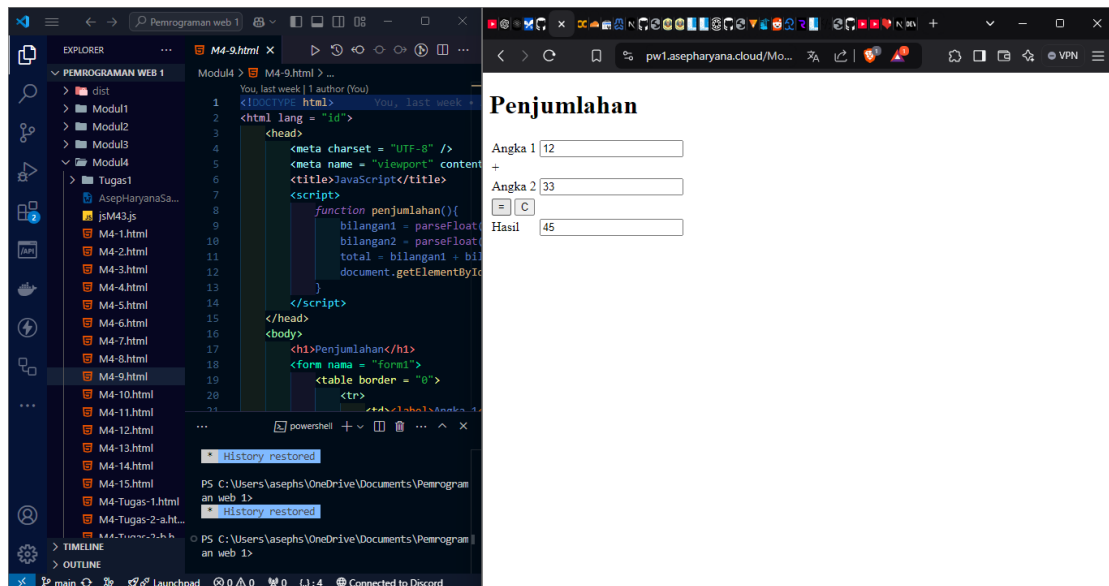
<https://pw1.asepharyana.cloud/Modul4/M4-8.html>

**Penjelasan:**

menampilkan form untuk memilih program studi dan menampilkan pilihan tersebut di kolom teks lain. Formulir ini memiliki elemen <select> dengan id "prodi" yang berisi beberapa pilihan program studi. Ketika pengguna memilih sebuah opsi dari dropdown, event onchange akan memicu fungsi JavaScript showData().

Fungsi `showData()` mengambil nilai yang dipilih dari dropdown "prodi" menggunakan `document.getElementById("prodi").value` dan menempatkan nilai tersebut ke dalam elemen input teks dengan id "programStudi" menggunakan `document.getElementById("programStudi").value = pr`. Elemen input teks "programStudi" memiliki atribut `readonly`, yang mencegah pengguna untuk mengeditnya secara langsung. Dengan demikian, teks box "programStudi" akan selalu menampilkan pilihan program studi yang dipilih pengguna dari dropdown. Nilai awal teks box adalah kosong karena opsi default dropdown adalah "-- Pilih --" yang memiliki nilai "Belum ada yang dipilih!"

## Web9



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-9.html>

### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-9.html>

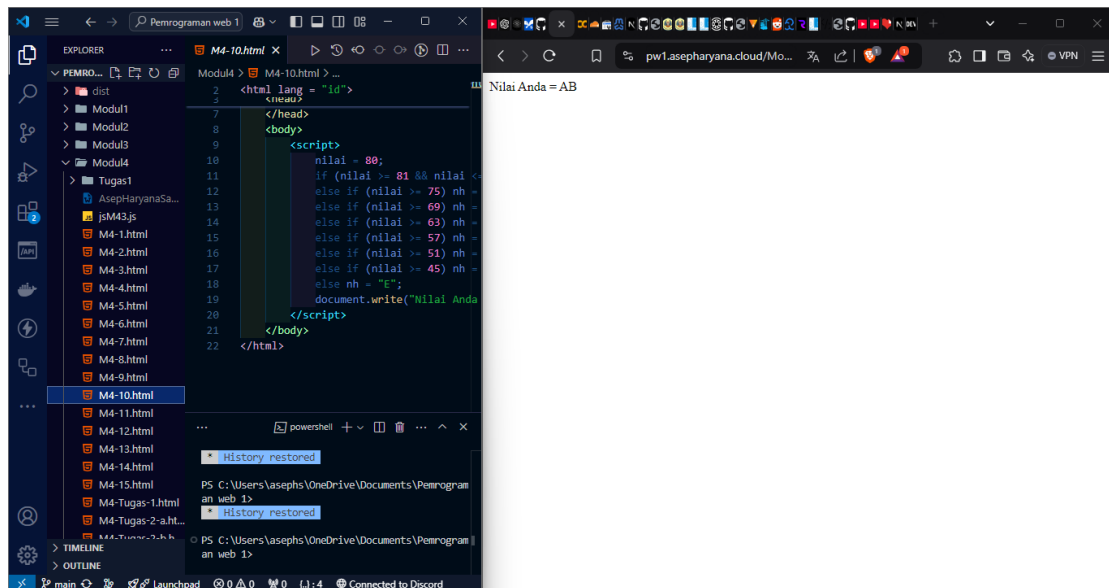
### Penjelasan:

menciptakan kalkulator penjumlahan sederhana. Pengguna dapat memasukkan dua angka di dalam dua *input field* bertipe "number" (dengan id "angka1" dan "angka2"). Ketika tombol "=" (yang bertipe "button") diklik, fungsi JavaScript `penjumlahan()` dijalankan.

Fungsi `penjumlahan()` mengambil nilai dari kedua *input field*, mengonversinya menjadi angka floating-point menggunakan `parseFloat()`, menjumlahkan keduanya, dan menempatkan hasilnya ke dalam *input field* "hasil" (yang bertipe "number" dan memiliki atribut `readonly`, sehingga tidak dapat diedit langsung oleh pengguna). Tombol "C" bertipe "reset", yang akan membersihkan semua *input field* pada formulir kembali ke nilai awalnya. Singkatnya, kode ini menyediakan antarmuka bagi pengguna untuk memasukkan dua angka, dan setelah mengklik tombol "=", hasil penjumlahan kedua angka tersebut akan ditampilkan di *input field* "hasil"



## Web10



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-10.html>

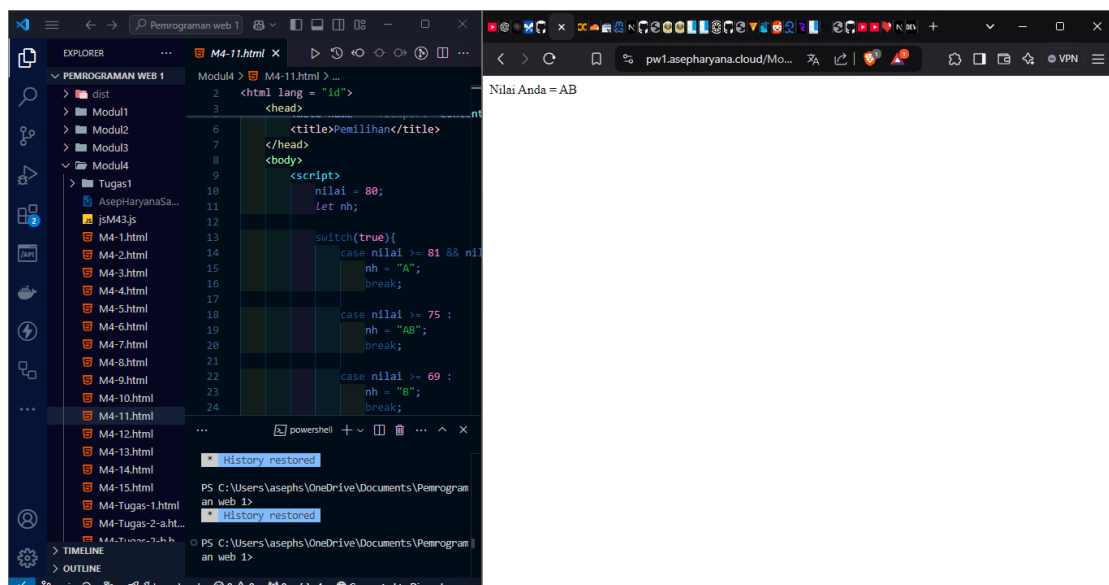
### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-10.html>

### Penjelasan:

mendemonstrasikan struktur kontrol pemilihan (if-else if) dalam JavaScript. Kode tersebut menginisialisasi variabel nilai dengan 80. Kemudian, serangkaian pernyataan if-else if digunakan untuk menentukan nilai huruf berdasarkan nilai angka. Karena nilai adalah 80, kondisi nilai  $\geq 75$  terpenuhi, sehingga nh diset menjadi "AB". Pernyataan else if selanjutnya dilewati. Akhirnya, kode tersebut menulis "Nilai Anda = AB" ke halaman web menggunakan document.write().

## Web11



**Kode:**

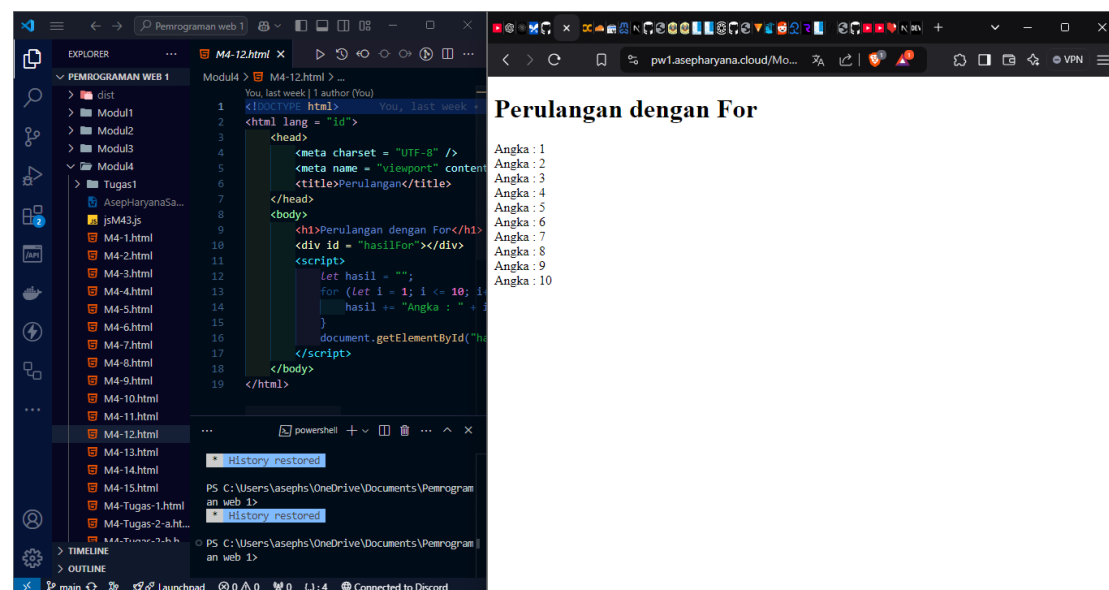
<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-11.html>

**Demo:**

<https://pw1.asepharyana.cloud/Modul4/M4-11.html>

**Penjelasan:**

mendemonstrasikan konversi nilai angka ke nilai huruf, tetapi menggunakan pernyataan switch daripada if-else if. Variabel nilai diinisialisasi dengan 80. Pernyataan switch(true) mengevaluasi setiap case dengan membandingkannya dengan true. Pada case nilai  $\geq 81$  && nilai  $\leq 100$ , kondisi tersebut salah karena nilai adalah 80. Eksekusi berlanjut ke case nilai  $\geq 75$ . Kondisi ini benar, sehingga nh diset menjadi "AB" dan pernyataan break keluar dari switch. case selanjutnya diabaikan. Akhirnya, kode menulis "Nilai Anda = AB" ke halaman web menggunakan document.write(). Hasilnya sama dengan contoh sebelumnya yang menggunakan if-else if, tetapi dengan pendekatan yang berbeda. Penggunaan switch(true) memungkinkan pengecekan kondisi yang lebih kompleks secara berurutan.

**Web12****Kode:**

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-12.html>

**Demo:**

<https://pw1.asepharyana.cloud/Modul4/M4-12.html>

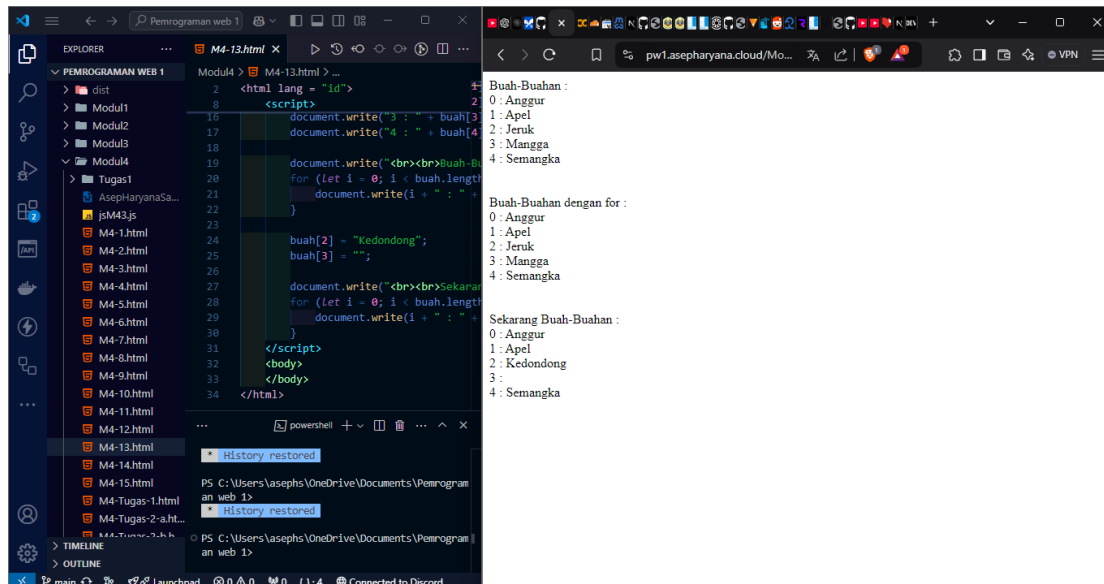
**Penjelasan:**

mendemonstrasikan perulangan for dalam JavaScript. Kode tersebut menginisialisasi variabel string kosong bernama hasil. Perulangan for berjalan dari  $i = 1$  hingga  $i \leq 10$ , menambahkan 1 ke i setiap iterasi.

Di dalam perulangan, string "Angka : " diikuti nilai i dan tag `<br>` (baris baru) ditambahkan ke variabel hasil. Setelah perulangan selesai, hasil berisi string dengan angka 1 hingga 10, masing-masing pada baris baru.

Akhirnya, kode tersebut menggunakan `document.getElementById("hasilFor").innerHTML = hasil`; untuk menempatkan konten string hasil ke dalam elemen HTML dengan id "hasilFor". Dengan demikian, halaman web akan menampilkan judul "Perulangan dengan For" diikuti daftar angka 1 hingga 10, masing-masing pada baris baru, di dalam elemen div.

## Web13



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-13.html>

### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-13.html>

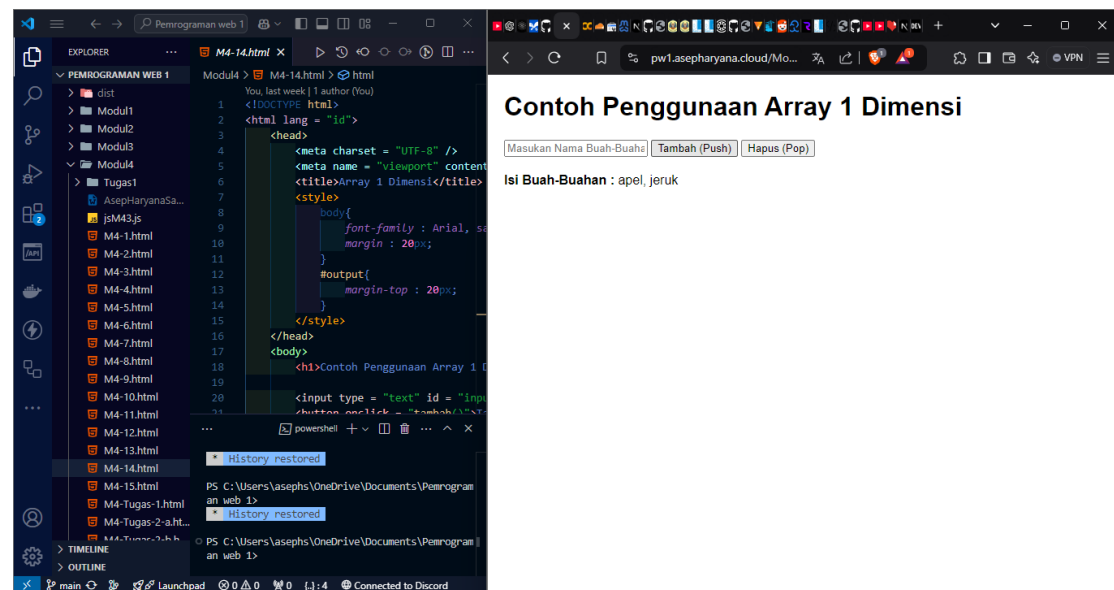
### Penjelasan:

mendemonstrasikan penggunaan array dalam JavaScript. Pertama, sebuah array bernama buah diinisialisasi dengan empat buah: "Anggur", "Apel", "Jeruk", dan "Mangga". Kemudian, elemen kelima ("Semangka") ditambahkan ke array pada indeks 4.

Kode tersebut kemudian menampilkan isi array buah dua kali: pertama secara manual dengan `document.write()` untuk setiap elemen, dan kedua menggunakan perulangan `for` untuk menampilkan setiap elemen beserta indeksnya.

Setelah itu, nilai pada indeks 2 dan 3 diubah. "Jeruk" diganti dengan "Kedondong", dan "Mangga" diganti dengan string kosong. Kode kemudian menampilkan isi array buah lagi menggunakan perulangan `for` untuk menunjukkan perubahan yang telah terjadi. Jadi, output di halaman web akan menunjukkan isi array pada tiga tahap: inisial, setelah penambahan "Semangka", dan setelah modifikasi elemen array. Ini menggambarkan bagaimana elemen array dapat diakses dan dimodifikasi menggunakan indeksnya.

## Web14



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-14.html>

### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-14.html>

### Penjelasan:

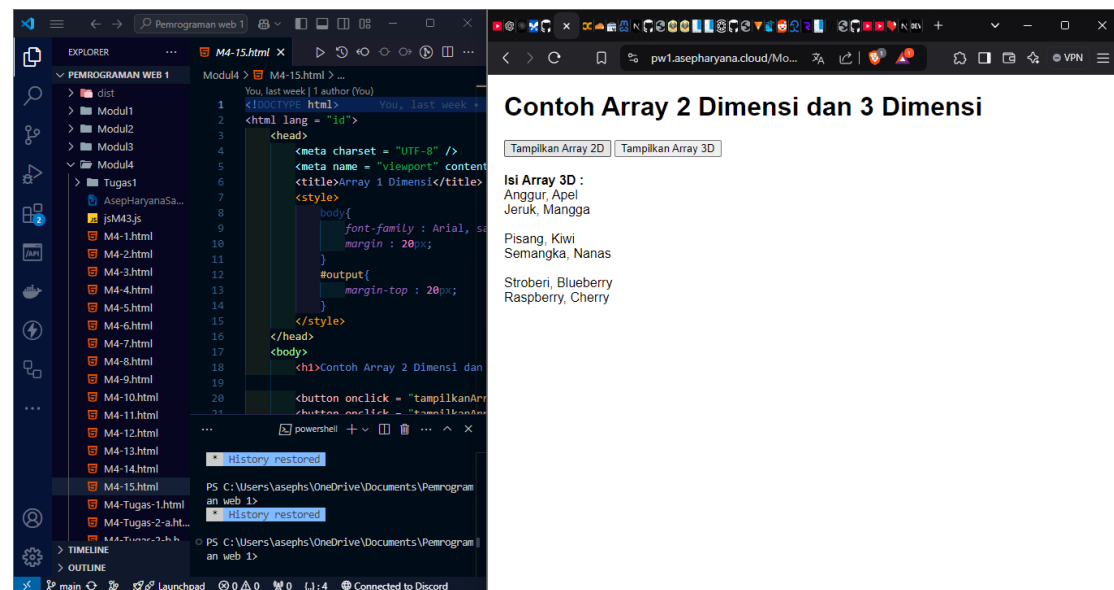
mendemonstrasikan manipulasi array satu dimensi dengan menambahkan dan menghapus elemen menggunakan metode push() dan pop(). Halaman web menampilkan judul, input teks untuk memasukkan nama buah, tombol "Tambah" dan "Hapus", serta area output untuk menampilkan isi array.

Fungsi tambah() mengambil nilai dari input teks, menambahkannya ke array buah menggunakan push(), membersihkan input teks, dan memanggil fungsi tampilkan() untuk memperbarui tampilan.

Fungsi hapus() menghapus elemen terakhir dari array buah menggunakan pop() jika array tidak kosong dan kemudian memperbarui tampilan. Jika array kosong, fungsi ini menampilkan pesan alert.

Fungsi tampilkan() memperbarui konten elemen div dengan id "output" untuk menampilkan isi array buah dalam format string yang dipisahkan koma. Intinya, pengguna dapat memasukkan nama buah, menambahkannya ke array dengan tombol "Tambah", dan menghapus buah terakhir yang ditambahkan dengan tombol "Hapus". Isi array yang selalu diperbarui ditampilkan di bawah tombol

## Web15



### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-15.html>

### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-15.html>

### Penjelasan:

mendemonstrasikan penggunaan array dua dimensi dan tiga dimensi dalam JavaScript. Terdapat dua tombol: "Tampilkan Array 2D" dan "Tampilkan Array 3D". Masing-masing tombol, ketika diklik, akan menjalankan fungsi JavaScript yang sesuai untuk menampilkan isi array di dalam elemen div dengan id "output".

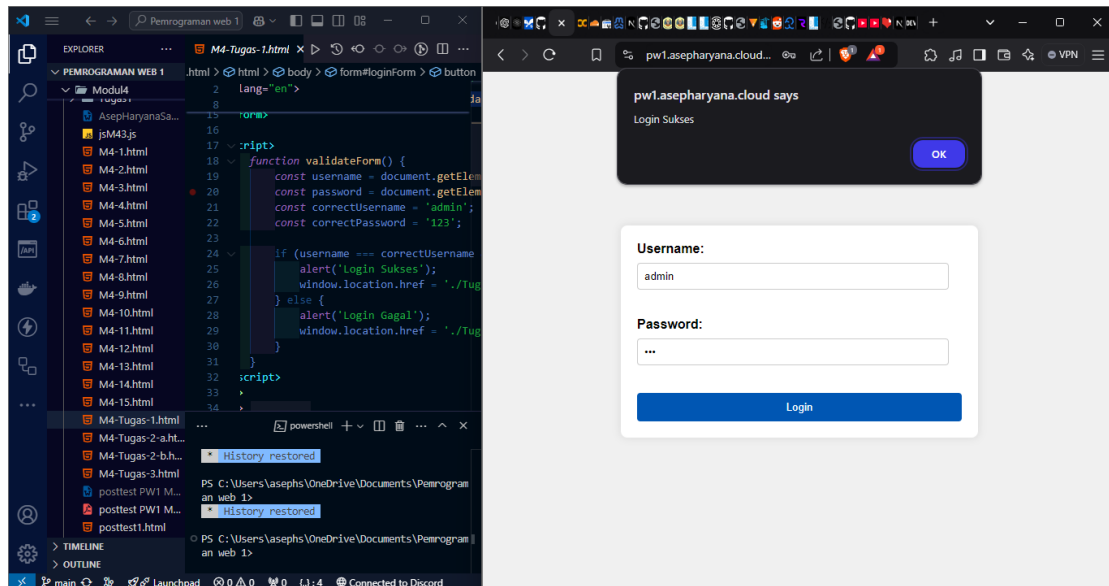
array2D diinisialisasi sebagai array dua dimensi yang berisi tiga array di dalamnya, masing-masing mewakili baris dan berisi nama-nama buah. Fungsi tampilkanArray2D() mengiterasi setiap baris dalam array2D menggunakan forEach dan menggabungkan elemen-elemen dalam setiap baris menjadi satu string yang dipisahkan koma, kemudian menambahkannya ke variabel result beserta tag <br> untuk membuat baris baru.

array3D diinisialisasi sebagai array tiga dimensi. Bayangkan seperti kubus yang berisi array-array dua dimensi. Fungsi tampilkanArray3D() mengiterasi setiap "matriks" (array 2D) dalam array3D, lalu mengiterasi setiap baris dalam "matriks" tersebut, dan menggabungkan elemen-elemen dalam setiap baris menjadi satu string yang dipisahkan koma. Setiap "matriks" dipisahkan oleh baris baru ganda untuk memberikan pemisahan visual.

Singkatnya, kode ini menampilkan representasi string dari array dua dimensi dan tiga dimensi di halaman web ketika tombol yang sesuai diklik.

# Tugas

1.



## Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-Tugas-1.html>

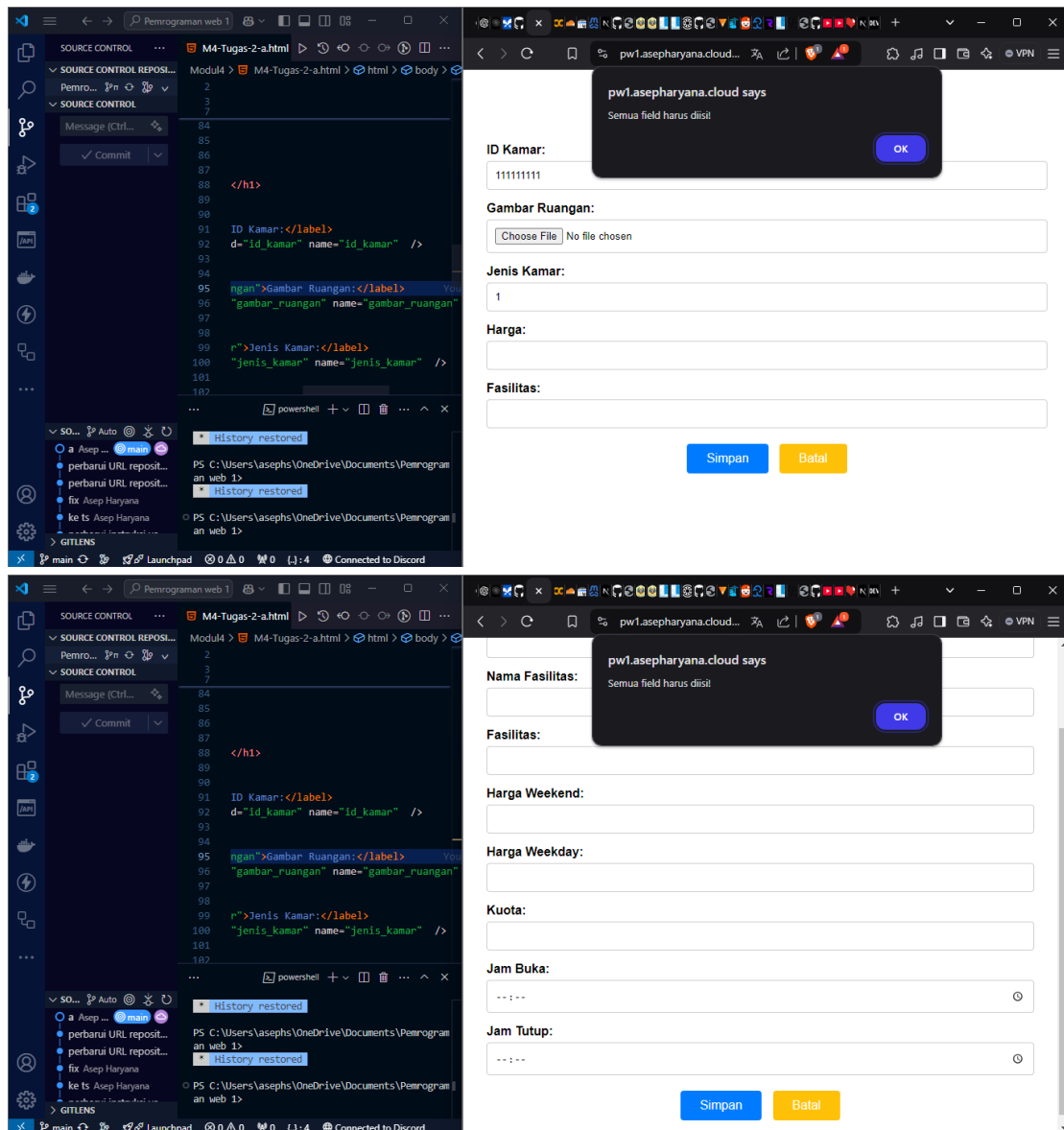
## Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-Tugas-1.html>

## Penjelasan:

form login sederhana dengan validasi username dan password menggunakan JavaScript. Formulir tersebut meminta pengguna untuk memasukkan username dan password. Ketika tombol "Login" diklik, fungsi validateForm() dijalankan. Fungsi ini membandingkan username dan password yang dimasukkan dengan nilai yang telah ditentukan ('admin' dan '123'). Jika username dan password cocok, maka akan muncul pesan "Login Sukses" dan halaman akan dialihkan ke "./Tugas1/index.html". Jika username atau password tidak cocok, akan muncul pesan "Login Gagal" dan halaman dialihkan ke "./Tugas1/login.html". Kode CSS yang disertakan memberikan style pada formulir dan elemen-elemennya untuk tampilan yang lebih baik, dengan memusatkan formulir di tengah layar dan memberikan warna serta padding

2.



#### Kode:

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-Tugas-2-a.html>  
<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-Tugas-2-b.html>

#### Demo:

<https://pw1.asepharyana.cloud/Modul4/M4-Tugas-2-a.html>  
<https://pw1.asepharyana.cloud/Modul4/M4-Tugas-2-b.html>

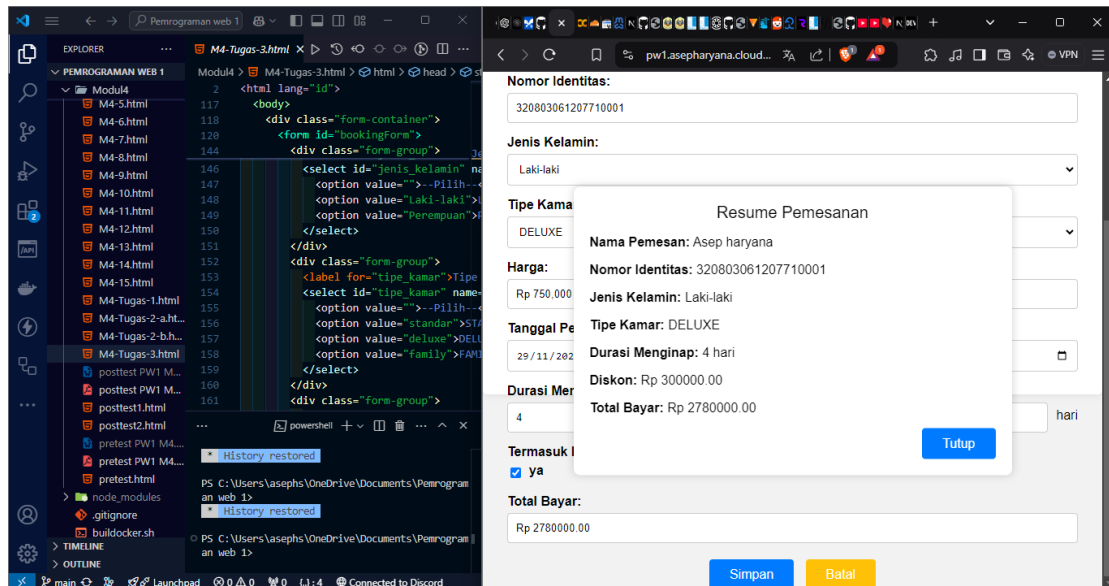
#### Penjelasan:

**Formulir Tambah Data Kamar:** Formulir ini meminta input untuk ID Kamar, Gambar Ruangan, Jenis Kamar, Harga, dan Fasilitas. JavaScript-nya memvalidasi apakah semua field telah diisi dan apakah harga kamar adalah angka positif. Jika validasi gagal, akan muncul pesan alert dan pengiriman formulir dicegah.

**Formulir Tambah Data Fasilitas:** Formulir ini lebih detail, meminta input untuk ID Fasilitas, Nama Fasilitas, Fasilitas, Harga Weekend, Harga Weekday, Kuota, Jam Buka, dan Jam Tutup. JavaScript-nya memvalidasi pengisian semua field, memastikan harga dan kuota

berupa angka, dan memastikan harga serta kuota tidak negatif. Sama seperti formulir kamar, jika validasi gagal, pesan alert ditampilkan dan pengiriman formulir dihentikan.

3.



**Kode:**

<https://github.com/MythEclipse/Pemrograman-web-1/blob/main/Modul4/M4-Tugas-3.html>

**Demo:**

<https://pw1.asepharyana.cloud/Modul4/M4-Tugas-3.html>

**Penjelasan:**

formulir pemesanan kamar hotel dengan fitur perhitungan harga dan tampilan resume pemesanan dalam modal. Formulir ini mengumpulkan informasi seperti ID pemesanan, nama, nomor identitas, jenis kelamin, tipe kamar, tanggal pesan, durasi menginap, dan pilihan sarapan. Logika JavaScript yang tertanam menangani beberapa hal: Pertama, harga kamar otomatis terisi berdasarkan tipe kamar yang dipilih. Kedua, total bayar dihitung secara dinamis berdasarkan tipe kamar, durasi menginap (dengan diskon untuk menginap lebih dari 3 hari), dan pilihan sarapan. Ketiga, validasi sederhana memastikan semua field wajib diisi dan durasi menginap valid. Terakhir, ketika formulir disubmit, sebuah modal akan muncul menampilkan resume pemesanan yang berisi informasi yang telah dimasukkan pengguna, termasuk diskon yang didapat. Modal ini dapat ditutup dengan tombol "Tutup". Secara keseluruhan, kode ini menciptakan pengalaman pemesanan yang interaktif dan informatif bagi pengguna.