

Nama : Kamilia Nurmalia Hayati

Kelas : C

NIM : 20230140108

HTTP (Hypertext Transfer Protocol)

- Definisi: Protokol komunikasi utama antara Client (browser) dan Server di web. Bekerja dengan model Request-Response dan bersifat Stateless.
- Komponen Kunci: GET (mengambil data), POST (mengirim data baru), Status Codes (200 OK, 404 Not Found), dan HTTPS (versi aman).

HTML (Hypertext Markup Language)

- Definisi: Bahasa markup standar untuk membuat struktur dan konten halaman web menggunakan elemen yang direpresentasikan oleh tag.
- Struktur Dasar: Terdiri dari <head> (metadata) dan <body> (konten visual).
- Elemen Umum: <h1>, <p>, <a>, , <div>.

CSS (Cascading Style Sheets)

- **Definisi:** Bahasa untuk mengatur **tampilan visual** (gaya dan layout) dari dokumen HTML, memisahkan presentasi dari struktur.
- **Konsep Kunci:**
 - **Selectors:** Cara memilih elemen HTML (h1, .class, #id).
 - **Box Model:** Setiap elemen adalah kotak yang terdiri dari **Content -> Padding -> Border -> Margin**.
 - **Layouting:** Menggunakan display: flex (Flexbox) untuk layout 1D dan display: grid (Grid) untuk layout 2D yang kompleks dan responsif.

Bootstrap

- **Definisi:** Framework CSS, HTML, dan JavaScript yang paling populer untuk mengembangkan situs web yang **responsif** dan **mobile-first**.
- **Tujuan:** Mempercepat proses pengembangan dengan menyediakan komponen UI siap pakai dan sistem grid yang kuat.
- **Fitur Utama:**
 - **Grid System:** Sistem layout berbasis baris (.row) dan kolom (.col-md-6) untuk membuat tata letak yang mudah beradaptasi di berbagai ukuran layar.
 - **Komponen Siap Pakai:** Tombol (.btn), Kartu (.card), Navbar (.navbar), Modal (.modal), dan Form yang sudah ditata.

- **Utility Classes:** Kelas-kelas pembantu untuk mengatur margin (.m-2), padding (.p-3), warna (.text-primary), dll., tanpa harus menulis CSS kustom.

HTML Form

- **Tujuan:** Mengumpulkan input dari pengguna.
- **Elemen Kunci:**
 - **<form>:** Wadah utama dengan atribut action (tujuan pengiriman) dan method (GET/POST).
 - **<input>:** Elemen input dengan berbagai type seperti text, password, dan submit. Atribut name digunakan sebagai kunci data saat dikirim.
 - Elemen lain seperti **<textarea>**, **<select>**, dan **<button>** digunakan untuk kebutuhan input yang lebih spesifik.

JavaScript Dasar & DOM

- **Definisi:** Bahasa scripting yang berjalan di browser untuk membuat halaman web menjadi **interaktif** dan **dinamis**.
- **DOM (Document Object Model):** Representasi struktur HTML yang memungkinkan JavaScript untuk memanipulasi konten dan gaya halaman.
 - **Seleksi & Manipulasi:** `document.getElementById()`, `document.querySelector()`, `.innerHTML`, `.style.color`.
 - **Event Handling:** Menanggapi aksi pengguna seperti `.addEventListener('click', ...)`.

JavaScript Standard Library (Topik 10)

- **Definisi:** Kumpulan objek dan fungsi bawaan yang tersedia di JavaScript tanpa memerlukan library eksternal.
- **Contoh Penting:**
 - **Math:** Objek untuk operasi matematika (`Math.random()`, `Math.floor()`).
 - **Date:** Objek untuk bekerja dengan tanggal dan waktu (`new Date()`).
 - **String Methods:** Fungsi untuk memanipulasi string (`.substring()`, `.toUpperCase()`, `.includes()`).
 - **Array Methods:** Fungsi untuk bekerja dengan array (`.map()`, `.filter()`, `.sort()`, `.push()`).
 - **JSON Object:** Untuk mengubah data antara format string dan objek JavaScript (`JSON.stringify()`, `JSON.parse()`).

Object-Oriented Programming (OOP) dalam JavaScript (Topik 9)

- **Konsep:** Paradigma pemrograman yang mengorganisir kode ke dalam "objek" yang memiliki **properti** (data) dan **metode** (fungsi). Tujuannya agar kode lebih terstruktur dan dapat digunakan kembali.
- **Implementasi:** Menggunakan sintaks class untuk membuat cetak biru objek.

```
class Car {
    constructor(name) { this.name = name; }
    start() { console.log(this.name + ' engine started. '); }
}
```

Komunikasi Frontend dan Backend

- **Konsep:** Proses di mana frontend (JavaScript di browser) berkomunikasi dengan backend (PHP di server) untuk mengirim atau mengambil data dari database (MySQL) tanpa me-refresh halaman.
- **Alur Kerja Umum:** Menggunakan **Fetch API** di JavaScript untuk mengirim request POST atau GET ke endpoint PHP, yang kemudian mengembalikan data dalam format **JSON**.

Navbar Autentikasi

- **Tujuan:** Menampilkan menu navigasi yang berbeda tergantung pada status login pengguna (misal: "Login/Register" untuk tamu, "Profile/Logout" untuk pengguna yang sudah login).
- **Implementasi:** Menggunakan kombinasi **Sesi PHP** di server dan **localStorage** di browser untuk menyimpan status login, lalu JavaScript akan mengubah tampilan navbar sesuai dengan status tersebut.

Proyek Akhir

- **Tujuan:** Mengaplikasikan semua konsep yang telah dipelajari untuk membangun sebuah aplikasi web fungsional secara berkelompok.
- **Lingkup Proyek:** Meliputi desain **Use Case Diagram** (menggunakan PlantUML), pembuatan **Source Code** lengkap, dan implementasi fitur-fitur seperti:
 - CRUD (Create, Read, Update, Delete) untuk entitas data seperti mobil dan berita.
 - Sistem autentikasi pengguna dengan level akses (User dan Admin).
 - Halaman dinamis yang menampilkan data dari database.
 - Antarmuka yang responsif menggunakan Bootstrap.