

Resume PDW Akhir

Nama: Syandy Arda Syahnuari

NIM: 20230140148

Ringkasan Materi Pemrograman Web: JavaScript, PHP, dan MySQL

Dokumen ini merangkum beberapa konsep fundamental dalam pengembangan web, mulai dari dasar-dasar JavaScript, penerapan Object-Oriented Programming (OOP) di dalamnya, hingga penggunaan praktis PHP dan MySQL untuk membangun aplikasi.

1. Pengenalan JavaScript Dasar

JavaScript, yang diciptakan oleh Brendan Eich di Netscape pada tahun 1995, adalah bahasa pemrograman yang dirancang untuk menjadikan halaman web lebih dinamis dan interaktif. Penting untuk dipahami bahwa JavaScript berbeda dengan Java, meskipun namanya serupa. Saat ini, kemampuannya tidak hanya terbatas pada browser, tetapi juga dapat berjalan di sisi server berkat Node.js. Sebelum mendalami JavaScript, sangat dianjurkan untuk memiliki pemahaman yang kuat tentang HTML dan CSS.

Lingkungan dan Penulisan Kode Untuk memulai pengembangan dengan JavaScript, Anda hanya memerlukan peramban web modern seperti Chrome atau Firefox serta editor teks seperti Visual Studio Code. Kode JavaScript bisa dituliskan langsung di dalam file HTML atau, sebagai praktik terbaik, ditempatkan pada file

.js terpisah.

Tipe Data dan Variabel JavaScript memiliki beberapa tipe data esensial, antara lain:

- **Number:** Untuk angka.
- **Boolean:** Untuk nilai kebenaran (true atau false).
- **String:** Untuk teks.
- **Array:** Untuk kumpulan data.
- **Object:** Untuk struktur data berbasis pasangan kunci-nilai (key-value).
- Tipe data khusus seperti

Undefined, Null, dan NaN (Not a Number).

Variabel dideklarasikan menggunakan

let untuk nilai yang bisa berubah dan const untuk nilai konstan. Penggunaan

var sudah tidak direkomendasikan lagi.

Operator, Kontrol Logika, dan Perulangan Bahasa ini menyediakan operator aritmatika, perbandingan, dan logika untuk memanipulasi data. Untuk debugging,

`console.log()` menjadi alat yang sangat berguna. JavaScript juga mendukung struktur kontrol program seperti:

- **Percabangan:** Menggunakan `if-else`, `switch`, dan operator ternary.
- **Perulangan:** Menggunakan `for`, `while`, dan `do-while`.

Fungsi dan Fitur Lanjutan Fungsi adalah blok kode yang dapat digunakan kembali, dideklarasikan dengan kata kunci

`function` atau melalui sintaksis *arrow function* (`=>`) yang lebih ringkas. Fitur lanjutan seperti

destructuring mempermudah pemecahan data dari array atau objek ke dalam variabel terpisah. Untuk menjaga kualitas kode, "strict mode" dapat diaktifkan guna menerapkan aturan penulisan yang lebih ketat.

Setelah menguasai dasar-dasar ini, pengembang disarankan untuk melanjutkan ke topik seperti JavaScript OOP, Modules, Document Object Model (DOM), dan pemrograman asinkron.

2. JavaScript Object-Oriented Programming (OOP)

OOP adalah sebuah paradigma yang mengorganisir kode di sekitar konsep "objek", yaitu entitas yang memiliki properti (data) dan metode (fungsi). Class berfungsi sebagai cetak biru (blueprint) untuk menciptakan objek-objek tersebut.

Dari Constructor ke Class Meskipun awalnya bersifat prosedural, JavaScript telah berkembang untuk mendukung OOP secara penuh. Sebelum standar ES6, class dibuat menggunakan

Constructor Function. Namun, sejak ES6, kata kunci

`class` diperkenalkan untuk menyediakan sintaksis yang lebih bersih dan mudah dipahami. Fitur

`class` ini mencakup constructor, method, serta pewarisan (inheritance) menggunakan kata kunci `extends`.

Fitur Lanjutan dan Penanganan Error JavaScript modern terus berkembang dengan fitur seperti

private class field (menggunakan awalan `#`) yang hanya bisa diakses dari dalam class itu sendiri, serta *static method* dan *property* yang melekat pada class, bukan pada instans objeknya.

Penanganan error (error handling) adalah aspek krusial dalam OOP. JavaScript memungkinkan pengembang untuk "melempar" (

`throw`) error dan menangkapnya menggunakan blok `try-catch`, bahkan membuat tipe error kustom dengan mewarisi dari class `Error` bawaan.

3. JavaScript Standard Library

JavaScript dilengkapi dengan pustaka standar yang kaya akan fungsi dan metode bawaan untuk berbagai keperluan. Ini mencakup fungsionalitas untuk manipulasi tipe data dasar seperti

Number, String, dan Array (termasuk implementasi struktur data seperti antrean dan tumpukan). Terdapat pula metode untuk mengelola

Object, seperti `Object.freeze()` dan `Object.assign()`, serta utilitas untuk bekerja dengan format JSON.

Untuk kebutuhan yang lebih kompleks, tersedia tipe data seperti:

- **BigInt:** Untuk menangani angka yang sangat besar.
- **Date:** Untuk bekerja dengan waktu dan tanggal.
- **Map & Set:** Struktur data modern untuk koleksi data unik.
- **RegExp:** Untuk pencocokan pola pada teks.

Fitur canggih lainnya termasuk

Proxy dan Reflect untuk meta-programming, serta utilitas praktis untuk encoding URL dan konversi Base64.

4. Praktik: Aplikasi CRUD dengan PHP Native dan MySQL

Bagian ini merangkum panduan praktis untuk membangun aplikasi CRUD (Create, Read, Update, Delete) menggunakan PHP Native yang terhubung ke database MySQL.

Pengaturan Lingkungan Proses pengembangan dimulai dengan persiapan lingkungan kerja, yang meliputi instalasi server lokal seperti Laragon dan versi PHP yang sesuai.

Struktur Backend Struktur backend diorganisir secara rapi. Sebuah file konfigurasi (

database.php) dibuat untuk mengelola koneksi ke database menggunakan PDO (PHP Data Objects) yang lebih aman. Logika bisnis ditempatkan dalam sebuah

class model (misalnya, User.php) yang berisi metode-metode untuk setiap operasi CRUD (create, read, update, delete). Skema database, termasuk tabel

users dengan kolom seperti id, nama, dan email, dibuat melalui antarmuka seperti phpMyAdmin.

Integrasi Frontend dan Backend Integrasi dilakukan dengan menyisipkan kode PHP di dalam file HTML:

- **index.php:** Menampilkan daftar pengguna dalam sebuah tabel dengan mengambil data dari database.
- **edit.php:** Menyediakan formulir untuk memperbarui data pengguna, yang datanya diambil berdasarkan parameter ID dan diproses melalui metode POST.
- **delete.php:** Sebuah file terpisah yang menerima ID pengguna untuk dihapus, memanggil metode `delete()` dari class model, lalu mengarahkan pengguna kembali ke halaman utama.