

Resume Pembelajaran Pengembangan Web

1. HTTP (Hypertext Transfer Protocol)

HTTP merupakan fondasi utama dalam komunikasi antara client (seperti browser) dan server. Protokol ini memungkinkan pertukaran data melalui berbagai metode, seperti:

- GET: Untuk mengambil data dari server.
- POST: Untuk mengirim data ke server.

Setiap respons dari server disertai dengan status code, seperti:

- 200 OK: Permintaan berhasil.
- 404 Not Found: Halaman/data tidak ditemukan.
- 500 Internal Server Error: Terjadi kesalahan pada server.

2. HTML (Hypertext Markup Language)

HTML adalah kerangka dasar dalam pembuatan halaman web, yang menentukan struktur dan konten.

Struktur dasar dokumen HTML:

```
<!DOCTYPE html>
<html>
<head>
  <title>Judul Halaman</title>
</head>
<body>
  <h1>Selamat Datang</h1>
  <p>Ini adalah paragraf contoh.</p>
</body>
</html>
```

Beberapa elemen penting:

- Heading (<h1> hingga <h6>)
- Paragraf (<p>)
- Tautan ()
- Gambar ()

Form HTML

Digunakan untuk mengumpulkan input pengguna, seperti login atau pendaftaran. Contoh:

```
<form action="proses.php" method="POST">
  <label for="nama">Nama:</label>
  <input type="text" id="nama" name="nama" required>
  <button type="submit">Kirim</button>
</form>
```

3. CSS (Cascading Style Sheets)

CSS memperindah tampilan HTML dengan mengatur warna, tata letak, dan animasi.

Contoh dasar CSS:

```
body {
  font-family: "Arial", sans-serif;
  background-color: #f4f4f4;
```

```

margin: 0;
padding: 20px;
}
h1 {
color: #0066cc;
text-align: center;
}

```

Tata Letak (Layout)

Teknik modern seperti Flexbox dan Grid memudahkan pembuatan desain responsif. Contoh Flexbox:

```

.container {
display: flex;
justify-content: space-between;
align-items: center;
}

```

4. JavaScript

JavaScript membuat halaman web menjadi interaktif dan dinamis.

Dasar JavaScript

- Variabel: let, const

- Tipe data: String, Number, Boolean

Contoh fungsi:

```

function sapa(nama) {
return `Halo, ${nama}!`;
}
console.log(sapa("Dani")); // Output: Halo, Dani!

```

Pemrograman Berorientasi Objek (OOP)

Menggunakan class untuk struktur yang lebih terorganisir:

```

class Pengguna {
constructor(nama, email) {
this.nama = nama;
this.email = email;
}

perkenalan() {
return `Nama saya ${this.nama}`;
}
}

```

Standard Library

Fungsi bawaan JavaScript:

- Array.map(): Memproses setiap elemen array.

- String.includes(): Memeriksa keberadaan substring.

5. Bootstrap

Bootstrap mempercepat pengembangan antarmuka dengan komponen siap pakai.

Fitur Utama

- Grid System: Membuat layout responsif.

- Komponen: Tombol, kartu, navbar.

Contoh tombol Bootstrap:

```
<button class="btn btn-primary">Klik Saya</button>
```

 Dokumentasi Lengkap: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>


6. PHP & MySQL

PHP adalah bahasa pemrograman server-side, sedangkan MySQL menyimpan data.

Koneksi Database

```
<?php
$koneksi = mysqli_connect("localhost", "user", "password", "nama_database");
if (!$koneksi) {
    die("Koneksi gagal: " . mysqli_connect_error());
}
?>
```

Sistem Login/Register

 Panduan Lengkap: <https://www.petanikode.com/php-login-register/>

7. Integrasi Teknologi

Contoh integrasi Bootstrap dengan PHP:

```
<!-- Form Login dengan Bootstrap -->
<form action="login.php" method="POST" class="p-4 shadow rounded">
    <div class="mb-3">
        <label class="form-label">Username</label>
        <input type="text" name="username" class="form-control" required>
    </div>
    <button type="submit" class="btn btn-success">Masuk</button>
</form>
```

Kesimpulan

Pengembangan web modern memadukan:

- Frontend: HTML, CSS, JavaScript, Bootstrap.
- Backend: PHP, MySQL.

Dengan menguasai teknologi ini, Anda dapat membangun aplikasi web yang fungsional dan estetik.

Referensi Utama

- <https://getbootstrap.com/>
- <https://www.petanikode.com/>