

LAPORAN PRAKTIKUM  
PRAKTIKUM PEMROGRAMAN WEB  
PERTEMUAN 5  
SLICING



Disusun oleh:

Nama : Aditya Lucky Zulkarnaen  
NIM : 24/537764/SV/24449  
Kelas : PLB1  
Dosen Pengampu : Faza Maula Azif, S.Kom., M.Cs.

PROGRAM STUDI D-IV TEKNOLOGI REKAYASA PERANGKAT LUNAK  
DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA  
SEKOLAH VOKASI  
UNIVERSITAS GADJAH MADA  
YOGYAKARTA  
2025

## PERTEMUAN 5

### SLICING

#### A. Link

- Link deployment : <https://adityazulkarnaen.github.io/PPW/>
- Link repo : <https://github.com/AdityaZulkarnaen/PPW>

#### B. Penjelasan

##### 1. Home Page (#home)

- a. Fungsi: Halaman utama (home) berfungsi sebagai landing page yang memperkenalkan website. Bagian ini biasanya berisi elemen hero, teks selamat datang, dan tombol untuk mengarahkan pengguna ke bagian lain dari website.
- b. Elemen Utama:
  - Hero Section: Bagian ini memiliki elemen utama berupa judul (h1), deskripsi (p), dan tombol aksi (button).
  - Latar Belakang: Warna atau gambar latar belakang digunakan untuk meningkatkan estetika halaman.
- c. Kode Penting:

##### HTML

```
<section id="home" class="home">
  <div class="hero">
    <h1>Welcome to My Website</h1>
    <button>Get Started</button>
  </div>
</section>
```

##### CSS

```
.home {
  display: flex;
  justify-content: center;
```

```
align-items: center;
flex-direction: column;
}
```

d. Penjelasan:

- display: flex; flex-direction: column; digunakan untuk menata elemen secara vertikal di tengah halaman.
- background-color: #f5f5f7; memberikan tampilan bersih dan profesional.
- button memiliki efek warna biru untuk menarik perhatian pengguna.

2. About Page (#about)

a. Fungsi: Memberikan informasi tentang website atau perusahaan.

b. Elemen Utama:

- Judul (h2): Memberikan pengenalan tentang bagian ini.
- Deskripsi (p): Menjelaskan sejarah, visi, dan misi dari pemilik website.

c. Kode Penting:

HTML

```
<section id="about">
  <h2>About Us</h2>
  <p>Kami adalah perusahaan yang berdedikasi untuk...</p>
</section>
```

CSS

```
#about {
  text-align: center;
  padding: 50px;
}
```

d. Penjelasan:

- padding: 50px; memberikan ruang di sekitar teks agar terlihat lebih nyaman.
- text-align: center; memastikan teks berada di tengah halaman.

### 3. Benefits Page (#benefits)

- a. Fungsi: Menampilkan manfaat utama dari layanan atau produk.
- b. Elemen Utama:
  - Ikon manfaat dengan teks deskripsi.
- c. Kode Penting:

#### HTML

```
<section id="benefits">
  <h2>Why Choose Us?</h2>
  <div class="benefit-item">
    
    <p>Fast and Reliable</p>
  </div>
</section>
```

#### CSS

```
.benefit-item {
  display: flex;
  flex-direction: column;
  align-items: center;
}
```

- d. Penjelasan:
  - display: flex; flex-direction: column; align-items: center; memastikan ikon dan teks berada dalam satu kolom.
  - width: 50px; height: 50px; mengatur ukuran ikon agar seragam.

### 4. Testimonials Page (#testimonials)

- a. Fungsi: Menampilkan ulasan dari pelanggan.
- b. Elemen Utama:

- Nama pelanggan, rating bintang, dan ulasan singkat.

c. Kode Penting:

HTML

```
<section id="testimonials">
  <h2>What Our Clients Say</h2>
  <div class="testimonial">
    <p>"Layanan sangat memuaskan!"</p>
    <span>- John Doe</span>
  </div>
</section>
```

CSS

```
.testimonial {
  border-left: 4px solid #2979FF;
  padding-left: 10px;
}
```

d. Penjelasan:

- border-left: 4px solid #2979FF; memberikan efek garis samping untuk testimoni.
- margin-bottom: 20px; memberi jarak antar testimoni agar lebih rapi.

5. FAQ Page (#faq)

a. Fungsi: Menjawab pertanyaan umum tentang layanan.

b. Elemen Utama:

- Pertanyaan dan jawaban dalam format dropdown atau accordion.

c. Kode Penting:

HTML

```
<section id="faq">
```

```
<h2>Frequently Asked Questions</h2>
<div class="faq-item">
  <h3>How does it work?</h3>
  <p>We provide services by...</p>
</div>
</section>
```

CSS

```
.faq-item {
  border-bottom: 1px solid #ddd;
  padding: 10px 0;
}
```

d. Penjelasan:

- border-bottom: 1px solid #ddd; memberikan pemisah antar pertanyaan.
- Bisa dikembangkan dengan JavaScript untuk membuat fitur dropdown.

## 6. Contact Page (#contact)

- Fungsi: Halaman ini menyediakan cara bagi pengunjung untuk menghubungi pemilik website melalui formulir kontak.
- Elemen Utama:
  - Formulir dengan input nama, email, dan pesan
  - Tombol Kirim
- Kode penting

HTML

```
<section id="contact">
  <h2>Contact Us</h2>
  <form>
    <input type="text" placeholder="Your Name">
    <input type="email" placeholder="Your Email">
```

```
<textarea placeholder="Message"></textarea>
<button type="submit">Send</button>
</form>
</section>
```

CSS

```
form {
  display: flex;
  flex-direction: column;
  gap: 10px;
}
```

d. Penjelasan:

- flex-direction: column; memastikan input diatur secara vertikal.
- gap: 10px; memberikan jarak antar elemen formulir.

## 7. Footer (footer)

a. Fungsi:

- Bagian ini menampilkan informasi hak cipta dan tautan ke media sosial.

b. Kode penting:

HTML

```
<footer>
  <p>© 2025 My Website. All Rights Reserved.</p>
</footer>
```

CSS

```
footer {
  text-align: center;
  padding: 20px;
```

```
background-color: #23315A;
color: white;
}
```

### C. Quick Assessment

1. Mengapa penggunaan elemen semantik penting dalam pengembangan web?

Penggunaan elemen semantik dalam pengembangan web sangat penting karena membantu meningkatkan keterbacaan kode, SEO, dan aksesibilitas. Elemen seperti `<header>`, `<nav>`, `<section>`, dan `<footer>` memberikan struktur yang jelas, memudahkan mesin pencari memahami konten, serta membantu pembaca layar dalam navigasi. Dengan mengikuti standar HTML5, pengembang dapat membuat website yang lebih terorganisir, ramah pengguna, dan mudah dipelihara.

2. Bagaimana cara meningkatkan keterbacaan dan aksesibilitas kode menggunakan elemen semantik terhadap HTML

berikut?

```
<div id="top">
<div class="menu">...</div>
<div class="content">
  <div class="article">...</div>
  <div class="article">...</div>
</div>
<div class="bottom">...</div>
</div>
```

Kode HTML tersebut dapat ditingkatkan keterbacaan dan aksesibilitasnya dengan mengganti elemen `<div>` yang tidak bermakna dengan elemen semantik HTML5 sebagai berikut:

```
<header>
  <nav>...</nav>
</header>
<main>
```



```
<section>
  <article>...</article>
  <article>...</article>
</section>
</main>
<footer>...</footer>
```

- `<div id="top">` diganti dengan `<header>` untuk menandakan bagian kepala halaman.
- `<div class="menu">` diganti dengan `<nav>` karena berisi menu navigasi.
- `<div class="content">` diganti dengan `<section>` karena berisi bagian utama dari halaman.
- `<div class="article">` diganti dengan `<article>` karena mewakili konten mandiri.
- `<div class="bottom">` diganti dengan `<footer>` karena merupakan bagian bawah halaman.

3. Apa yang dimaksud dengan selector `:hover`? Sebagai contoh: `button:hover`

Selector `:hover` adalah pseudo-class CSS yang mengubah gaya elemen saat kursor diarahkan ke atasnya. Misalnya, pada `button:hover { background-color: darkblue; }`, tombol akan berubah warna saat pengguna mengarahkannya dengan mouse, meningkatkan interaktivitas dan pengalaman pengguna.

4. Perhatikan navigasi berikut. Underline pada navigasi tidak terlihat rapi karena memiliki panjang yang melebihi teksnya. Bagaimana cara mengubah panjang underline pada menu home menjadi sama panjang dengan text navigasi? Sertakan HTML dan CSS-nya.



Untuk membuat underline sesuai dengan panjang teks navigasi, kita bisa menggunakan `::after` dengan `display: inline-block` dan `width: 100%`.

HTML

```
<nav>
  <ul>
    <li><a href="#" class="nav-link">Home</a></li>
    <li><a href="#" class="nav-link">Features</a></li>
    <li><a href="#" class="nav-link">Media</a></li>
    <li><a href="#" class="nav-link">News</a></li>
  </ul>
</nav>
```

## CSS

```
nav ul {
  list-style: none;
  display: flex;
  gap: 20px;
  padding: 0;
  background-color: #0a1f44;
}

.nav-link {
  position: relative;
  text-decoration: none;
  color: white;
  font-size: 18px;
  padding: 10px;
}

.nav-link::after {
  content: "";
  display: block;
  width: 100%; /* Sesuai panjang teks */
}
```

```
height: 2px;
background-color: white;
position: absolute;
left: 0;
bottom: -5px;
transform: scaleX(0);
transition: transform 0.3s ease-in-out;
}

.nav-link:hover::after {
  transform: scaleX(1); /* Muncul saat hover */
}
```

5. Jika item dalam Flexbox tidak terlihat atau berada di luar elemen/layar, properti apa yang perlu diperiksa untuk memperbaiki masalah tersebut?

Jika item dalam Flexbox tidak terlihat atau berada di luar elemen/layar, periksa properti overflow pada elemen induk karena overflow: hidden; dapat menyembunyikan konten yang keluar. Selain itu, pastikan flex-wrap: wrap; diaktifkan jika item terlalu banyak agar bisa berpindah ke baris berikutnya. Cek juga nilai width, max-width, dan justify-content untuk memastikan item memiliki ruang yang cukup dan tidak terlempar ke luar area tampilan.

6. Perhatikan kode CSS berikut:

```
/* Style test */
.container {
  display: flex;
  height: 100vh;
}
.item {
  width: 100px;
  height: 100px;
  background-color: blue;
}
```

```
/* End of style test */
```

Elemen "item" tidak terpusat di tengah. Apa yang perlu diperbaiki?

Untuk memusatkan elemen .item di tengah .container, tambahkan properti justify-content dan align-items ke .container. Justify content center memusatkan .item secara horizontal dan Align items center secara vertikal.

7. Pada gambar berikut, bagaimana cara menempatkan Logo di ujung kiri dan Navigasi di ujung kanan? Pastikan logo dan navigasi sejajar. Sertakan HTML dan CSS-nya.



#### HTML

```
<nav class="navbar">
  
  <ul class="nav-menu">
    <li><a href="#">Home</a></li>
    <li><a href="#">Features</a></li>
    <li><a href="#">Media</a></li>
    <li><a href="#">News</a></li>
  </ul>
</nav>
```

#### CSS

```
.navbar {
  display: flex;
  justify-content: space-between; /* Memisahkan logo dan navigasi */
  align-items: center; /* Memastikan sejajar vertikal */
  padding: 10px 20px;
  background-color: #0a2a43; /* Warna latar opsional */
}
```

```

}

.logo {
  height: 50px; /* Sesuaikan ukuran logo */
}

.nav-menu {
  list-style: none;
  display: flex;
  gap: 20px; /* Memberi jarak antar menu */
}

.nav-menu li a {
  text-decoration: none;
  color: white;
  font-size: 18px;
}

```

- display: flex; pada .navbar membuat logo dan menu navigasi sejajar.
- justify-content: space-between; menempatkan logo di kiri dan navigasi di kanan.
- align-items: center; memastikan elemen sejajar secara vertikal.
- gap: 20px; memberi jarak antar item navigasi.