### **Bagian 1: Fondasi Styling (Contoh untuk Slide 3-6)**

Untuk bagian ini, kita akan menggunakan satu set file (index.html dan style.css) yang akan kita kembangkan secara bertahap.

#### **1. Contoh untuk Slide 4: Cara Menambahkan CSS (Metode Eksternal)**

Ini adalah setup awal. Peserta akan membuat dua file ini.

**File 1: index.html**

HTML

<!DOCTYPE **html**>  
<html lang="id">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Praktik CSS Dasar</title>  
 <link rel="stylesheet" href="style.css">  
</head>  
<body>  
 <h1>Selamat Datang di Praktik CSS</h1>  
 <p>Halaman ini akan kita gunakan untuk belajar dasar-dasar styling dengan CSS.</p>  
</body>  
</html>

**File 2: style.css**

CSS

/\* Ini adalah file CSS eksternal kita \*/  
  
/\* Mengubah warna latar belakang seluruh halaman \*/  
body {  
 background-color: #f2f2f2; /\* Warna abu-abu muda \*/  
}

**Cara Menjalankan & Hasil:**

* Simpan kedua file di dalam folder yang sama.
* Buka index.html di browser.
* **Hasil:** Halaman akan tampil dengan teks standar, namun latar belakangnya akan berwarna abu-abu muda, membuktikan bahwa file CSS sudah terhubung.

#### **2. Contoh untuk Slide 5: Sintaks & Selector Dasar**

Kita akan memodifikasi kedua file dari langkah sebelumnya.

Modifikasi index.html:

Tambahkan beberapa elemen baru agar kita bisa menargetkannya.

HTML

<!DOCTYPE **html**>  
<html lang="id">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Praktik CSS Dasar</title>  
 <link rel="stylesheet" href="style.css">  
</head>  
<body>  
 <h1>Selamat Datang di Praktik CSS</h1>  
 <p>Ini adalah paragraf biasa.</p>  
 <p class="highlight">Paragraf ini spesial karena memiliki class.</p>  
 <div id="footer-info">  
 <p>Ini adalah informasi di dalam div dengan ID unik.</p>  
 </div>  
</body>  
</html>

Modifikasi style.css:

Tambahkan aturan baru untuk setiap jenis selector.

CSS

/\* Aturan dari langkah sebelumnya \*/  
body {  
 background-color: #f2f2f2;  
 font-family: Arial, sans-serif; /\* Tambahkan font family \*/  
}  
  
/\* 1. Element Selector: menargetkan semua tag <h1> \*/  
h1 {  
 color: navy;  
 text-align: center;  
}  
  
/\* 2. Class Selector: menargetkan semua elemen dengan class="highlight" \*/  
.highlight {  
 background-color: yellow;  
 font-weight: bold;  
}  
  
/\* 3. ID Selector: menargetkan satu elemen dengan id="footer-info" \*/  
#footer-info {  
 font-style: italic;  
 color: #555; /\* Warna abu-abu tua \*/  
}

**Cara Menjalankan & Hasil:**

* Simpan perubahan pada kedua file.
* Refresh index.html di browser.
* **Hasil:**
  + Judul <h1> akan berwarna biru navy dan berada di tengah.
  + Paragraf kedua akan memiliki latar belakang kuning dan teks tebal.
  + Teks di bagian bawah akan menjadi miring dan berwarna abu-abu.

#### **3. Contoh untuk Slide 6: The Box Model**

Kita akan menambahkan satu elemen lagi ke HTML dan memberinya gaya Box Model di CSS.

Modifikasi index.html:

Tambahkan div baru di bawah h1.

HTML

<body>  
 <h1>Selamat Datang di Praktik CSS</h1>  
   
 <div class="box-model-demo">  
 Ini adalah kotak untuk demonstrasi Box Model. Perhatikan ruang di sekelilingnya.  
 </div>  
  
 <p>Ini adalah paragraf biasa.</p>  
 </body>

Modifikasi style.css:

Tambahkan aturan baru untuk .box-model-demo.

CSS

/\* (Aturan CSS lainnya tetap sama) \*/  
  
/\* Demonstrasi Box Model \*/  
.box-model-demo {  
 background-color: lightblue; /\* Warna konten agar terlihat jelas \*/  
   
 width: 300px; /\* Lebar area konten \*/  
   
 padding: 20px; /\* Jarak di DALAM border (tebal 20px di semua sisi) \*/  
   
 border: 5px solid navy; /\* Border biru navy setebal 5px \*/  
   
 margin: 40px auto; /\* Jarak di LUAR border (40px atas-bawah, auto kiri-kanan untuk menengahkan) \*/  
}

**Cara Menjalankan & Hasil:**

* Simpan perubahan dan refresh browser.
* **Hasil:** Akan muncul sebuah kotak biru muda di tengah halaman. Anda bisa jelaskan kepada peserta:
  + Teks berada di dalam area background-color (content).
  + Ada ruang kosong antara teks dan border (padding).
  + Ada garis biru tebal (border).
  + Ada ruang kosong besar di luar border yang mendorong elemen lain (margin).

### **Bagian 2: Tata Letak Modern (Contoh untuk Slide 8-9)**

Untuk bagian ini, kita akan membuat set file baru yang lebih fokus pada layout.

#### **1. Contoh untuk Slide 8: Flexbox (Navbar)**

**File 1: flexbox-navbar.html**

HTML

<!DOCTYPE **html**>  
<html lang="id">  
<head>  
 <title>Praktik Flexbox</title>  
 <link rel="stylesheet" href="flexbox-style.css">  
</head>  
<body>  
  
 <nav class="navbar">  
 <div class="logo">MyLogo</div>  
 <ul class="nav-links">  
 <li><a href="#">Home</a></li>  
 <li><a href="#">Produk</a></li>  
 <li><a href="#">Tentang</a></li>  
 <li><a href="#">Kontak</a></li>  
 </ul>  
 </nav>  
  
 <div class="konten-halaman">  
 <h1>Ini adalah Konten Halaman</h1>  
 </div>  
  
</body>  
</html>

**File 2: flexbox-style.css**

CSS

body {  
 margin: 0;  
 font-family: sans-serif;  
}  
  
.navbar {  
 display: flex; /\* Mengaktifkan Flexbox \*/  
 justify-content: space-between; /\* Item didorong ke ujung: logo di kiri, link di kanan \*/  
 align-items: center; /\* Menengahkan item secara vertikal \*/  
 background-color: #333;  
 padding: 10px 20px;  
}  
  
.logo {  
 color: white;  
 font-size: 24px;  
 font-weight: bold;  
}  
  
.nav-links {  
 display: flex; /\* Jadikan <ul> ini flex container juga \*/  
 list-style: none; /\* Hilangkan bullet points \*/  
 margin: 0;  
 padding: 0;  
}  
  
.nav-links li a {  
 color: white;  
 text-decoration: none;  
 padding: 15px;  
 display: block;  
}  
  
.nav-links li a:hover {  
 background-color: #555;  
}

**Cara Menjalankan & Hasil:**

* Buat kedua file baru ini di folder yang sama dan buka flexbox-navbar.html.
* **Hasil:** Sebuah bilah navigasi (navbar) hitam yang responsif sederhana, di mana logo berada di kiri dan daftar link berada di kanan, semuanya sejajar rapi.

#### **2. Contoh untuk Slide 9: Grid (Layout Halaman)**

**File 1: grid-layout.html**

HTML

<!DOCTYPE **html**>  
<html lang="id">  
<head>  
 <title>Praktik Grid Layout</title>  
 <link rel="stylesheet" href="grid-style.css">  
</head>  
<body class="grid-container">  
  
 <header class="header">HEADER</header>  
 <aside class="sidebar">SIDEBAR</aside>  
 <main class="main">MAIN CONTENT</main>  
 <footer class="footer">FOOTER</footer>  
  
</body>  
</html>

**File 2: grid-style.css**

CSS

body {  
 margin: 0;  
 font-family: sans-serif;  
}  
  
.grid-container {  
 display: grid;  
 height: 100vh; /\* Tinggi 100% dari viewport \*/  
   
 /\* Mendefinisikan kolom dan baris \*/  
 grid-template-columns: 200px 1fr; /\* Kolom 1: 200px, Kolom 2: sisa ruang \*/  
 grid-template-rows: auto 1fr auto; /\* Baris 1 & 3: tinggi sesuai konten, Baris 2: sisa ruang \*/  
   
 /\* Mendefinisikan area \*/  
 grid-template-areas:  
 "header header"  
 "sidebar main"  
 "footer footer";  
 gap: 10px; /\* Jarak antar area \*/  
}  
  
/\* Menempatkan setiap elemen ke dalam areanya \*/  
.header { grid-area: header; }  
.sidebar { grid-area: sidebar; }  
.main { grid-area: main; }  
.footer { grid-area: footer; }  
  
/\* Styling tambahan agar terlihat jelas \*/  
.header, .sidebar, .main, .footer {  
 background-color: lightskyblue;  
 padding: 20px;  
 text-align: center;  
 border: 1px solid navy;  
}

**Cara Menjalankan & Hasil:**

* Buat kedua file baru ini dan buka grid-layout.html.
* **Hasil:** Layout halaman penuh yang klasik. Header dan Footer membentang penuh di atas dan bawah, dengan Sidebar di kiri dan Main Content di kanan.

### **Bagian 3: Desain Responsif (Contoh untuk Slide 12-13)**

Kita akan **memodifikasi contoh Grid Layout** di atas.

#### **Modifikasi grid-style.css (untuk Slide 12)**

Tambahkan blok @media di bagian paling bawah grid-style.css.

CSS

/\* (Semua kode CSS dari grid-style.css sebelumnya tetap ada) \*/  
  
  
/\* --- Bagian Responsif --- \*/  
@media (max-width: 768px) {  
 .grid-container {  
 /\* Ubah definisi grid untuk layar kecil \*/  
 grid-template-columns: 1fr; /\* Hanya ada satu kolom \*/  
 grid-template-rows: auto auto 1fr auto; /\* Tambah satu baris untuk sidebar \*/  
   
 /\* Susun ulang area secara vertikal \*/  
 grid-template-areas:  
 "header"  
 "sidebar"  
 "main"  
 "footer";  
 }  
}

**Cara Menjalankan & Hasil:**

* Simpan perubahan pada grid-style.css.
* Buka grid-layout.html di browser.
* Lihat dalam mode layar lebar, lalu persempit jendela browser.
* **Hasil:** Saat jendela browser menyempit di bawah 768px, layout dua kolom akan secara otomatis berubah menjadi satu kolom, dengan sidebar pindah ke bawah header.

#### **Modifikasi grid-layout.html (untuk Slide 13)**

Pastikan tag <meta name="viewport"> ada di dalam <head>.

HTML

<!DOCTYPE **html**>  
<html lang="id">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Praktik Grid Layout</title>  
 <link rel="stylesheet" href="grid-style.css">  
</head>  
<body class="grid-container">  
  
 <header class="header">HEADER</header>  
 <aside class="sidebar">SIDEBAR</aside>  
 <main class="main">MAIN CONTENT</main>  
 <footer class="footer">FOOTER</footer>  
  
</body>  
</html>

Penjelasan untuk Peserta:

Jelaskan bahwa tanpa tag <meta name="viewport"> ini, browser di HP akan mencoba menampilkan versi desktop dengan cara di-zoom out, sehingga media query tidak akan berfungsi seperti yang diharapkan.