

1

Laporan Praktikum CSS Layout

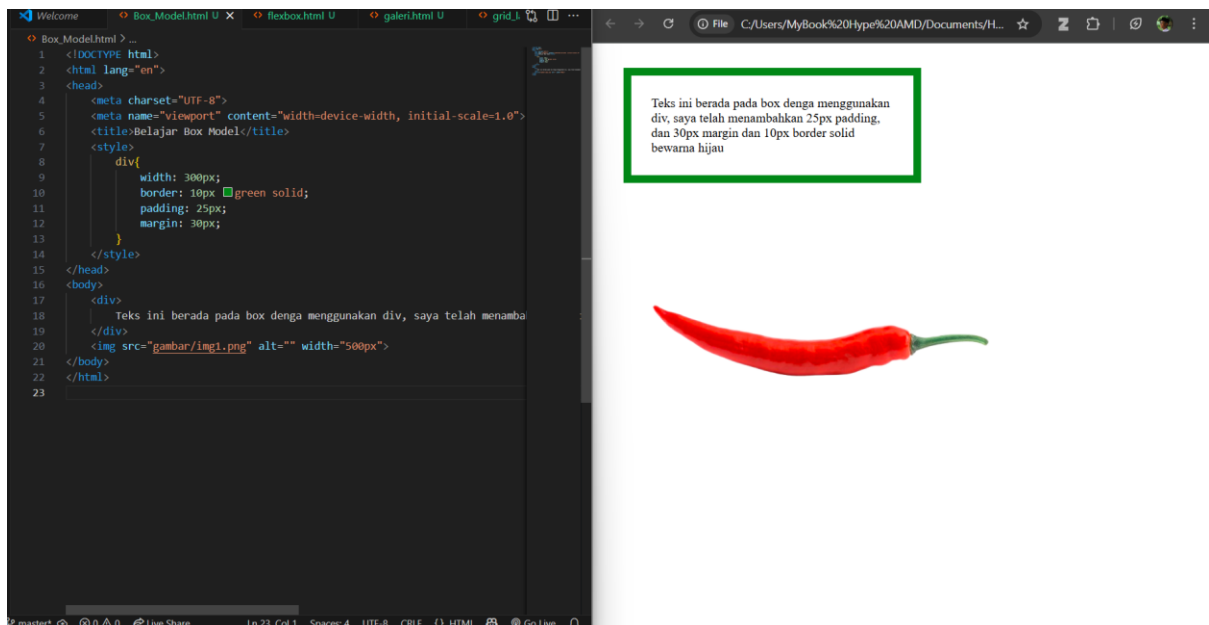
Nama	Habib Hajid Taqiuddin
NPM	244311043
Kelas	2B
Github	habibhajid

KEMAMPUAN AKHIR YANG DIRENCANAKAN

- Mampu menjelaskan Cascading Style Sheet (CSS) sebagai bahasa style web
- Mampu menjelaskan manfaat dari penggunaan CSS dalam teknologi web
- Mampu menjelaskan perbedaan HTML dengan CSS
- Mampu menjelaskan jenis, struktur, sintaks-sintaks CSS dan metode-metode implementasinya
- Mampu merancang sebuah web yang berbasis HTML dan CSS

No 1.

Hasil Tangkapan Layar :

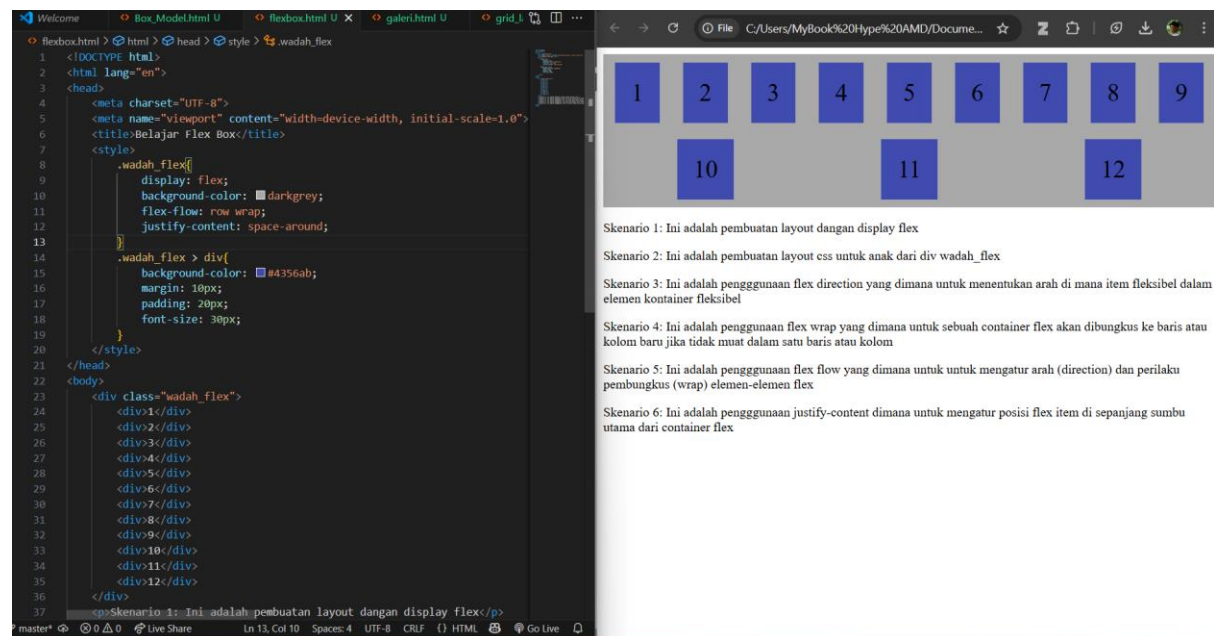


Deskripsi/Keterangan :

Kode HTML dan CSS tersebut membuat sebuah halaman web yang menampilkan kotak teks dan gambar cabai. Kotak teks dibuat menggunakan elemen `<div>` dengan lebar 300px, border 10px solid hijau, padding 25px (jarak dalam dari teks ke border), serta margin 30px (jarak luar dari elemen lainnya). Gambar yang ditampilkan memiliki lebar 500px. Hasilnya, teks berada dalam kotak dengan batas hijau yang jelas, sementara gambar berada di bawahnya.

No 2.1

Hasil Tangkapan Layar :

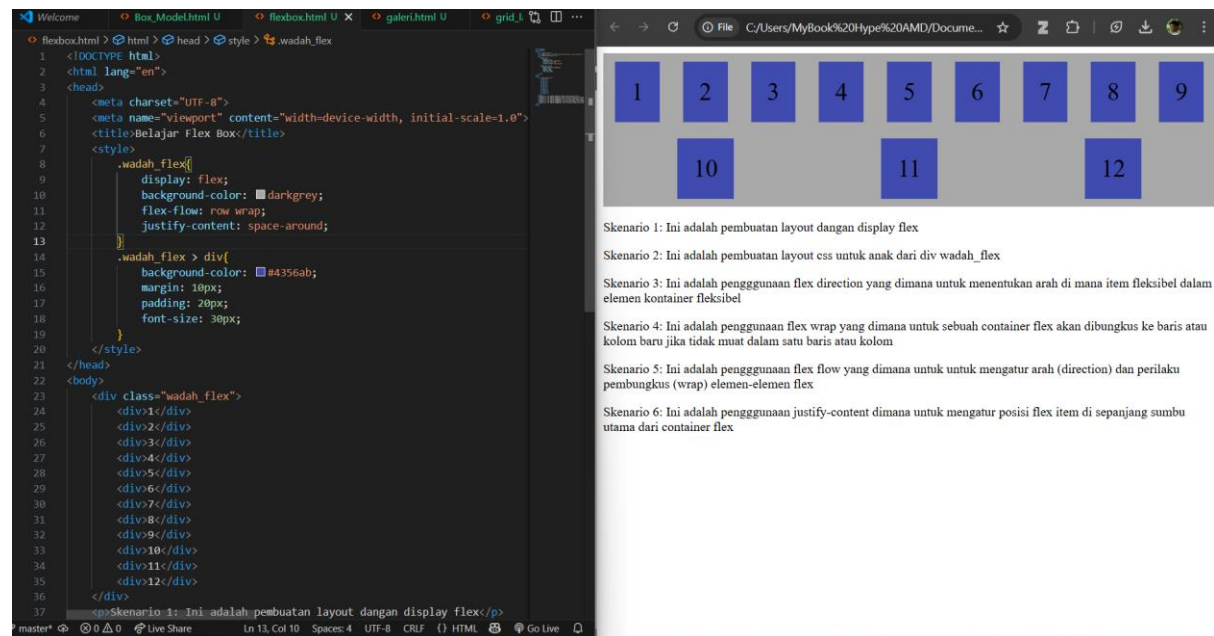


Deskripsi/Keterangan :

Skenario 1: Ini adalah pembuatan layout dengan display flex

No. 2.2

Hasil Tangkapan Layar :

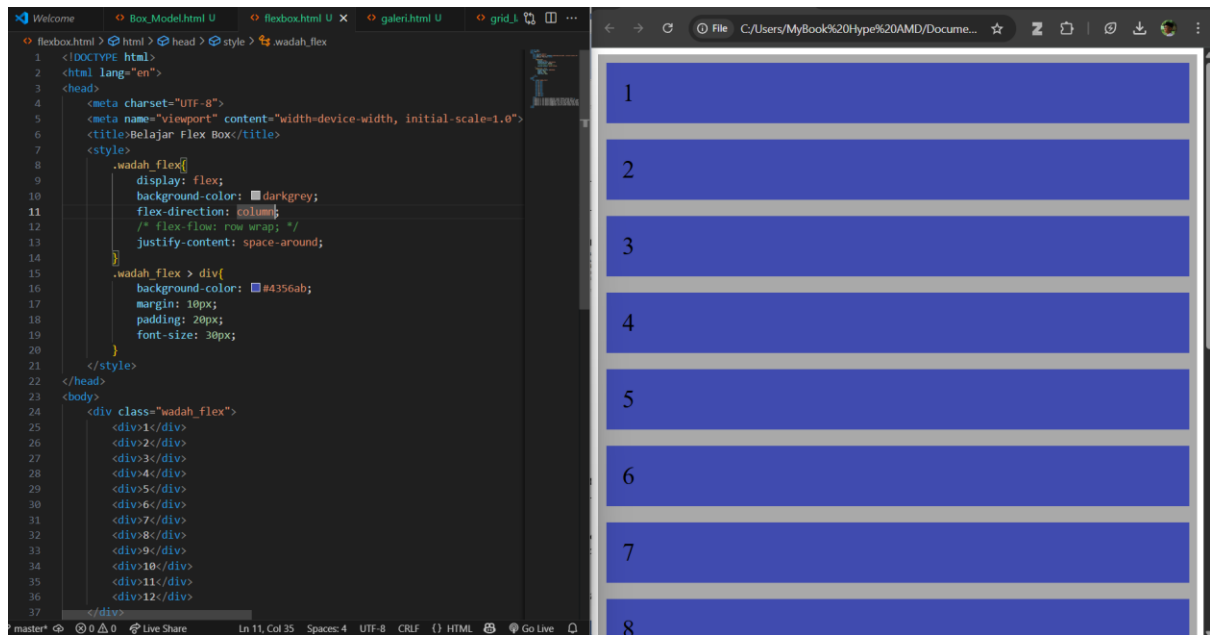


Deskripsi/Keterangan :

Skenario 2: Ini adalah pembuatan layout css untuk anak dari div wadah_flex

No. 2.3

Hasil Tangkapan Layar :

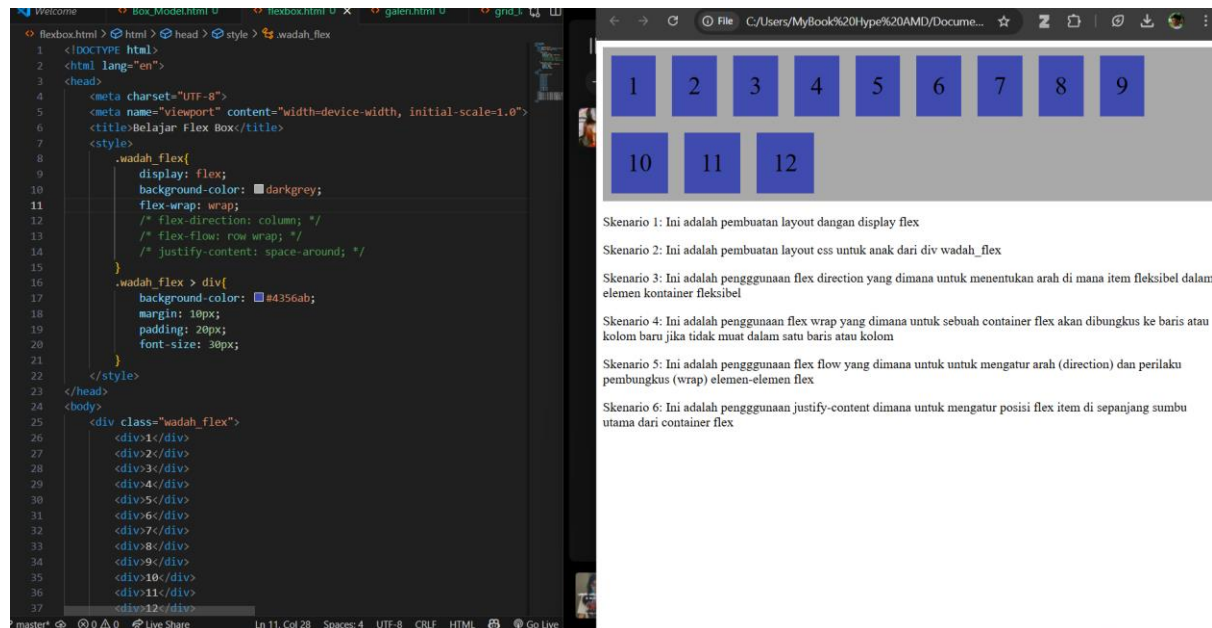


Deskripsi/Keterangan :

Skenario 3: Ini adalah penggunaan flex direction yang dimana untuk menentukan arah di mana item fleksibel dalam elemen kontainer fleksibel

No. 2.4

Hasil Tangkapan Layar :

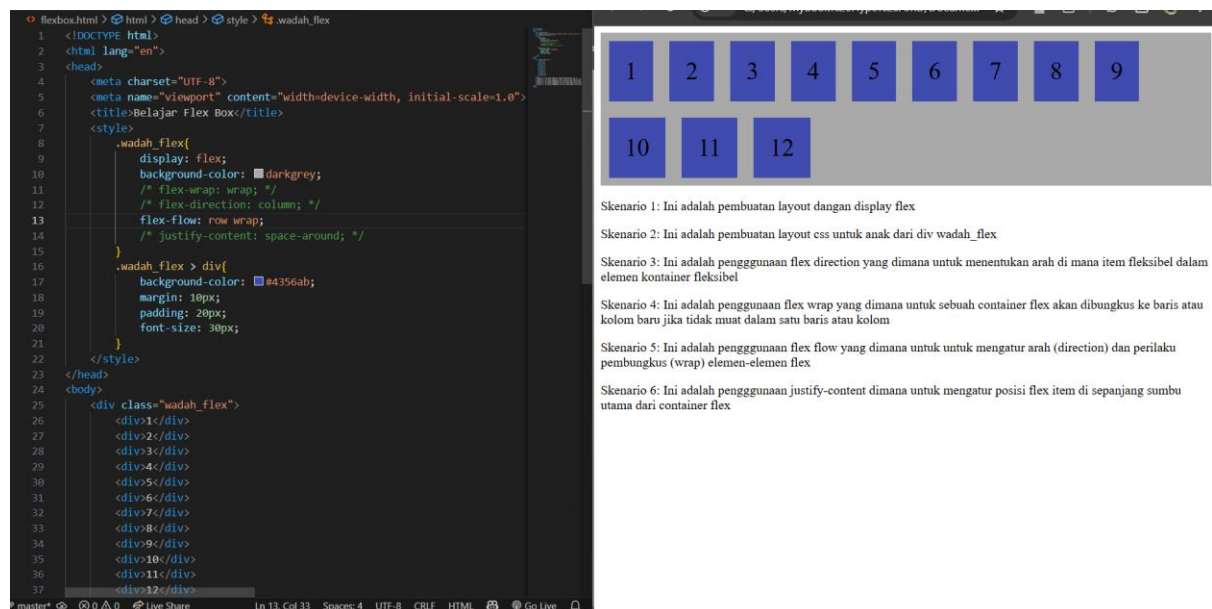


Deskripsi/Keterangan :

Skenario 4: Ini adalah penggunaan flex wrap yang dimana untuk sebuah container flex akan dibungkus ke baris atau kolom baru jika tidak muat dalam satu baris atau kolom

No. 2.5

Hasil Tangkapan Layar :



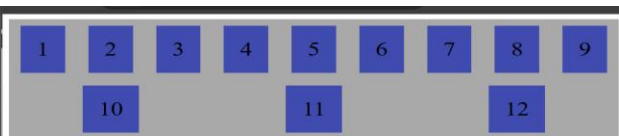
Deskripsi/Keterangan :

Skenario 5: Ini adalah penggunaan flex flow yang dimana untuk mengatur arah (direction) dan perilaku pembungkus (wrap) elemen-elemen flex

No. 2.6

Hasil Tangkapan Layar :

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Belajar Flex Box</title>
7   <style>
8     .wadah_flex{
9       display: flex;
10      background-color: #darkgrey;
11      /* flex-wrap: wrap; */
12      /* flex-direction: column; */
13      flex-flow: row wrap;
14      justify-content: space-around;
15    }
16    .wadah_flex > div{
17      background-color: #4356ab;
18      margin: 10px;
19      padding: 20px;
20      font-size: 30px;
21    }
22  </style>
23 </head>
24 <body>
25   <div class="wadah_flex">
26     <div>1</div>
27     <div>2</div>
28     <div>3</div>
29     <div>4</div>
30     <div>5</div>
31     <div>6</div>
32     <div>7</div>
33     <div>8</div>
34     <div>9</div>
35     <div>10</div>
36     <div>11</div>
37     <div>12</div>
38   </div>
39 </body>
40 </html>
```



Skenario 1: Ini adalah pembuatan layout dengan display flex

Skenario 2: Ini adalah pembuatan layout css untuk anak dari div wadah_flex

Skenario 3: Ini adalah penggunaan flex direction yang dimana untuk menentukan arah di mana item fleksibel dalam elemen kontainer fleksibel

Skenario 4: Ini adalah penggunaan flex wrap yang dimana untuk sebuah container flex akan dibungkus ke baris atau kolom baru jika tidak muat dalam satu baris atau kolom

Skenario 5: Ini adalah penggunaan flex flow yang dimana untuk mengatur arah (direction) dan perilaku pembungkus (wrap) elemen-elemen flex

Skenario 6: Ini adalah penggunaan justify-content dimana untuk mengatur posisi flex item di sepanjang sumbu utama dari container flex

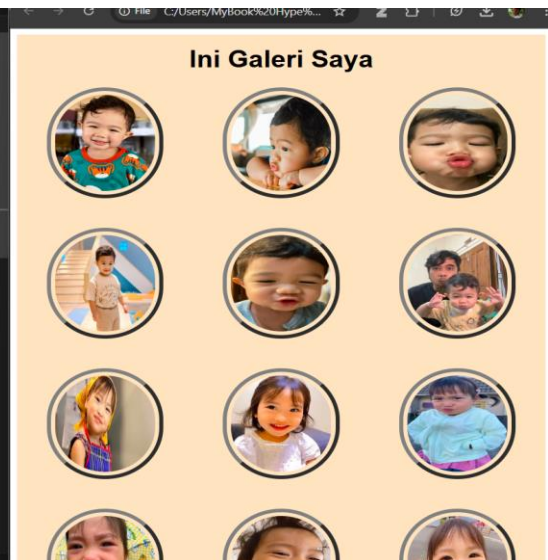
Deskripsi/Keterangan :

Skenario 6: Ini adalah penggunaan justify-content dimana untuk mengatur posisi flex item di sepanjang sumbu utama dari container flex

No. 3

Hasil Tangkapan Layar :

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Galeri Saya</title>
7   <style>
8     .container{
9       display: flex;
10      background-color: #bisque;
11      flex-direction: row;
12      flex-wrap: wrap;
13      justify-content: space-around;
14    }
15    .container img{
16      width: 120px;
17      height: 130px;
18      margin: 20px;
19      padding: 5px;
20      border: 5px solid #gray;
21      border-radius: 200px;
22      border-style: outset;
23    }
24  </style>
25 </head>
26 <body>
27   <div style="background-color: #bisque; text-align: center; margin: 0; font-family:
28     <div class="container">
29       
30       
31       
32       
33       
34       
35       
36       
37       
38     </div>
39   </div>
40 </body>
41 </html>
```

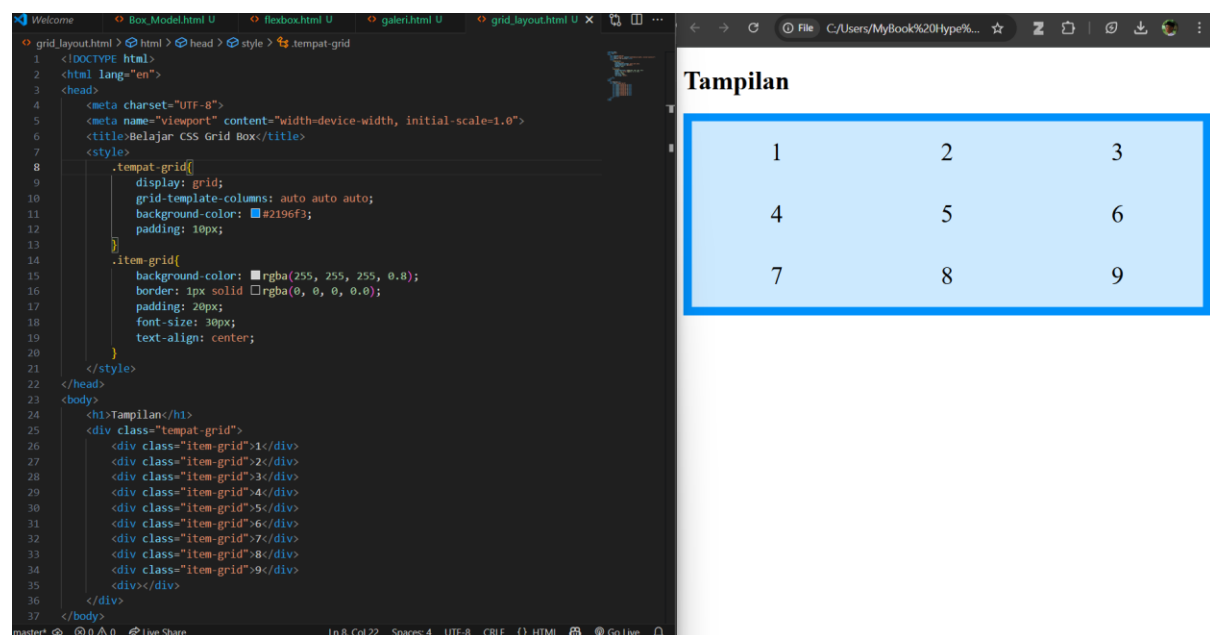


Deskripsi/Keterangan :

Kode HTML dan CSS tersebut membuat halaman galeri dengan latar belakang berwarna bisque. Judul "Ini Galeri Saya" ditampilkan di tengah atas halaman. Gambar dalam galeri ditampilkan dalam format grid fleksibel menggunakan flexbox dengan display: flex, flex-wrap: wrap, dan justify-content: space-around. Setiap gambar memiliki ukuran 120x130px, border abu-abu 5px dengan gaya outset, serta border-radius besar sehingga berbentuk lingkaran. Hasilnya adalah tampilan galeri foto dengan gambar berbentuk bulat yang tersusun rapi.

No. 4.1

Hasil Tangkapan Layar :

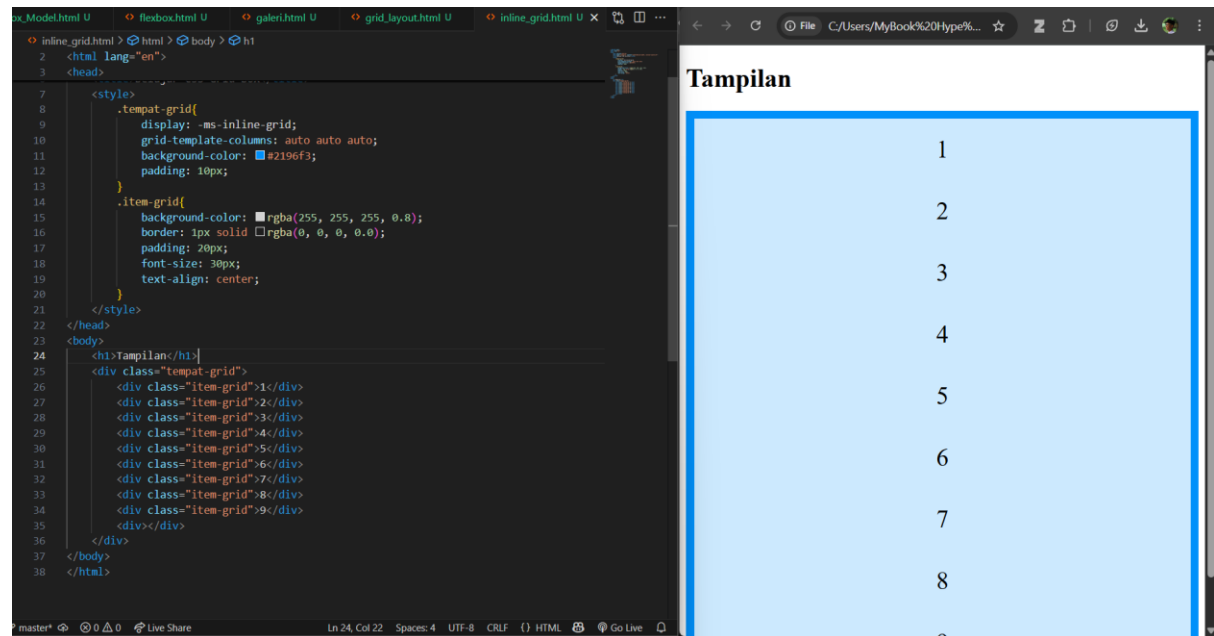


Deskripsi/Keterangan :

Kode tersebut menggunakan CSS Grid untuk membuat tata letak tiga kolom otomatis dengan elemen berisi angka 1-9 dan satu div tambahan. Setiap item grid memiliki background, border, padding, serta teks yang terpusat agar tampilan lebih rapi

No. 4.2

Hasil Tangkapan Layar :

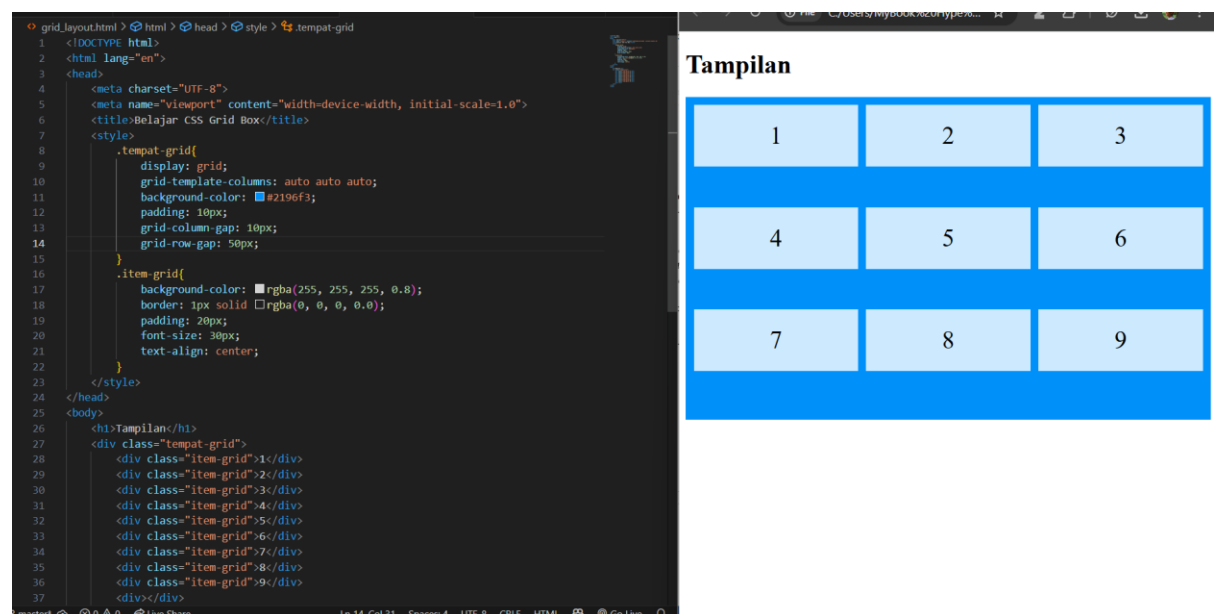


Deskripsi/Keterangan :

Kode tersebut menggunakan CSS Grid untuk membuat layout tiga kolom, tetapi terdapat kesalahan pada `display: -ms-inline-grid;`, yang menyebabkan elemen ditampilkan dalam satu kolom. Setiap item dalam grid memiliki background putih transparan, border, padding, serta teks berukuran besar yang terpusat, dengan latar belakang utama berwarna biru.

No. 4.3

Hasil Tangkapan Layar :

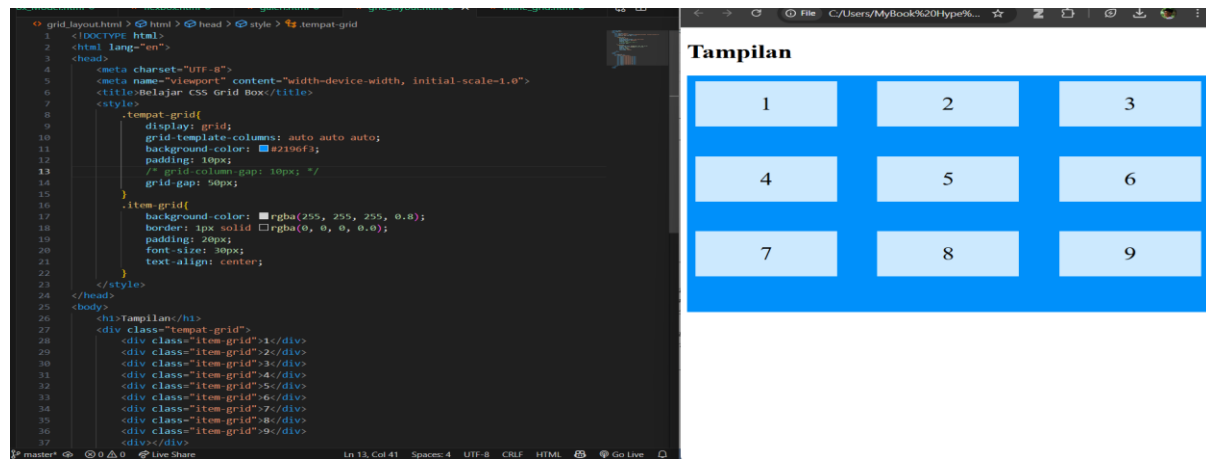


Deskripsi/Keterangan :

Kode tersebut menggunakan CSS Grid untuk menyusun elemen dalam tiga kolom dengan jarak antar kolom 10px dan antar baris 10px. Setiap item dalam grid memiliki background putih transparan, teks berukuran besar yang terpusat, serta ditampilkan dalam wadah berwarna biru dengan padding.

No. 4.4

Hasil Tangkapan Layar :

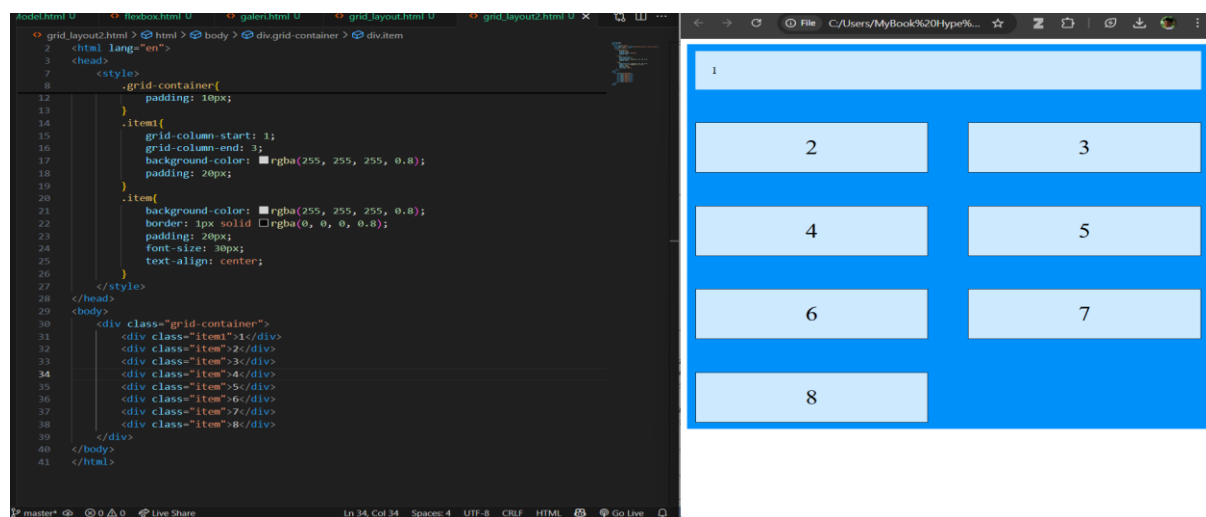


Deskripsi/Keterangan :

Kode tersebut menggunakan CSS Grid untuk menata elemen dalam tiga kolom dengan jarak antar item sebesar 50px menggunakan `grid-gap`. Setiap item dalam grid memiliki latar belakang putih transparan, teks yang terpusat, dan ditampilkan dalam wadah berwarna biru dengan padding.

No. 5.1

Hasil Tangkapan Layar :

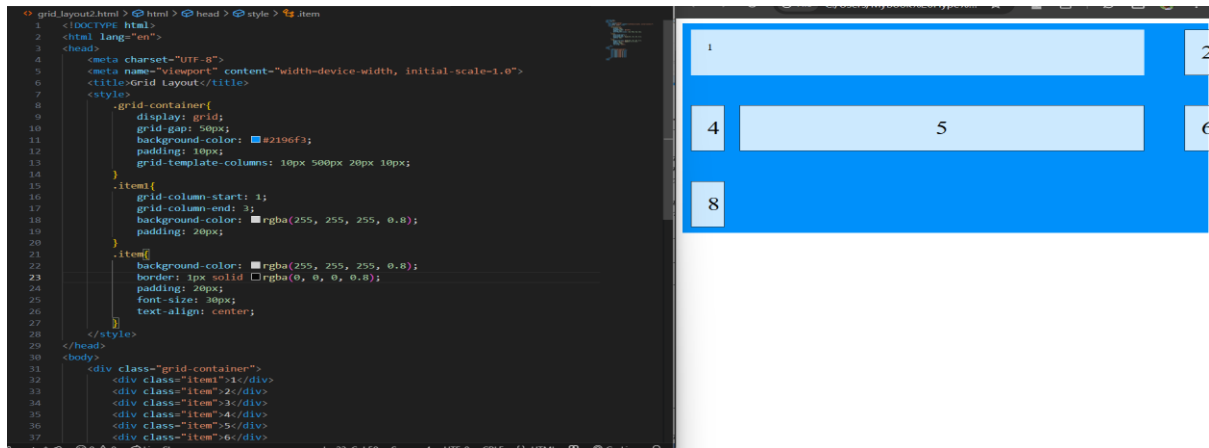


Deskripsi/Keterangan :

Kode menggunakan CSS Grid untuk membuat tata letak dengan elemen pertama melebar ke dua kolom. Setiap item memiliki background putih transparan, border, padding, dan teks terpusat. Grid memiliki jarak antar elemen sebesar 50px.

No. 5.2

Hasil Tangkapan Layar :

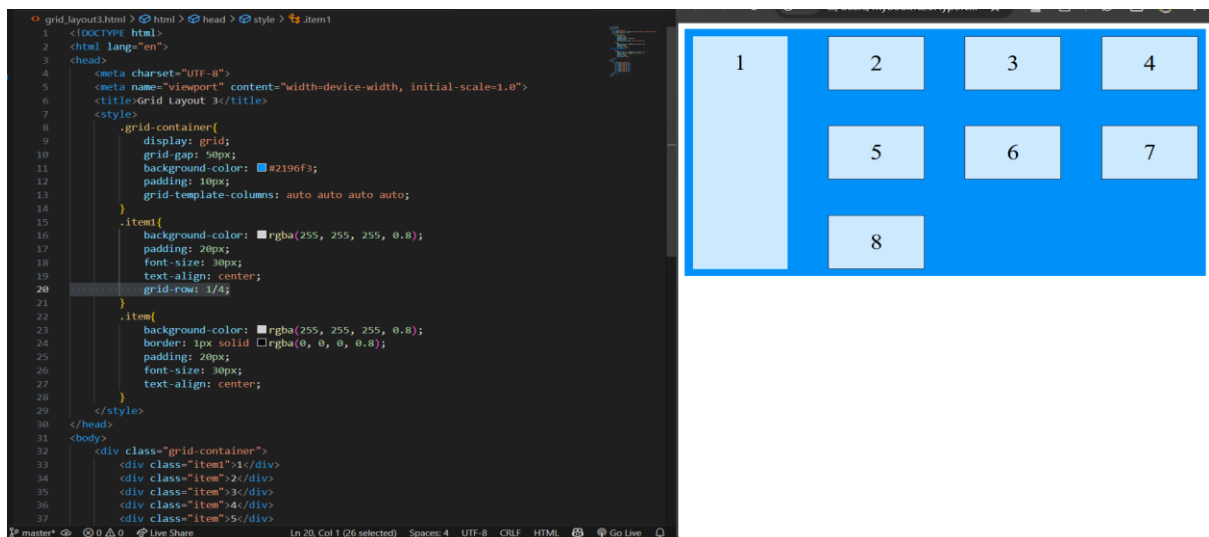


Deskripsi/Keterangan :

perubahan grid-template-columns mempengaruhi lebar kolom, sehingga tata letak elemen berubah sesuai dengan nilai yang diberikan.

No. 6.1

Hasil Tangkapan Layar :

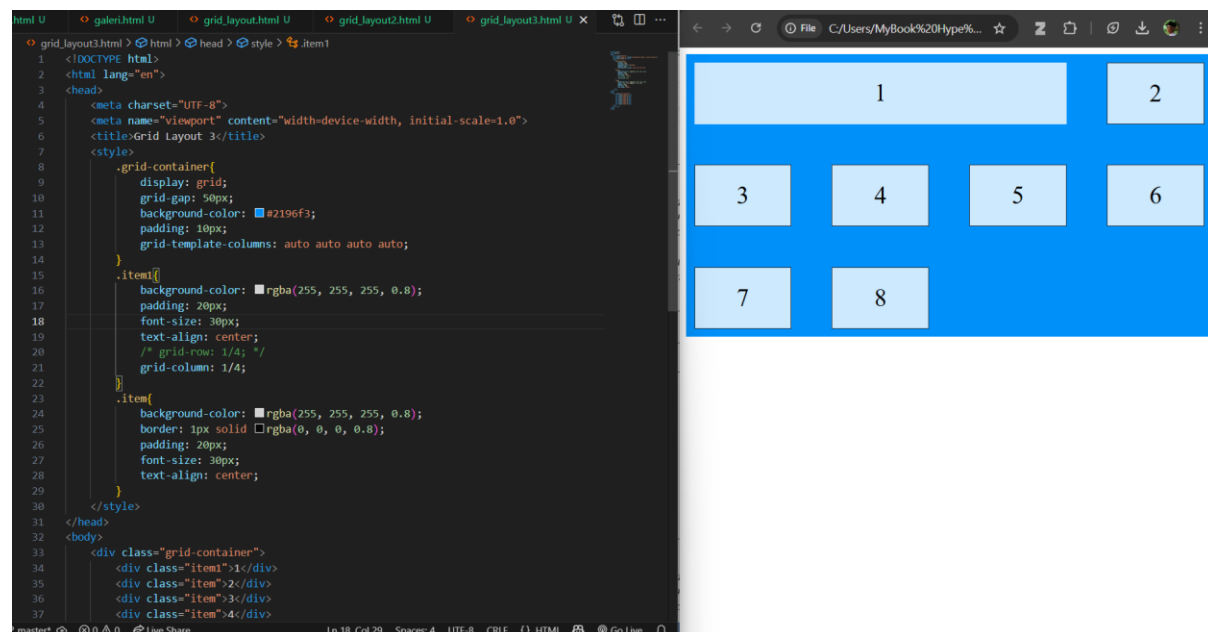


Deskripsi/Keterangan :

Kode HTML di atas membangun antarmuka website menggunakan CSS untuk tata letak dan gaya teks. Header memiliki background biru, teks besar, dan padding untuk ruang dalam. Navigasi menu ditampilkan sebagai daftar dengan border dashed biru dan elemen li yang mengapung (float: left) agar sejajar horizontal. Bagian utama berisi logo, teks institusi, serta deskripsi dengan warna merah untuk penekanan. Footer menampilkan informasi nama, NIM, dan kelas dengan background biru muda. Hasilnya, tampilan web lebih rapi, menarik, dan mudah dinavigasi.

No. 6.2

Hasil Tangkapan Layar :

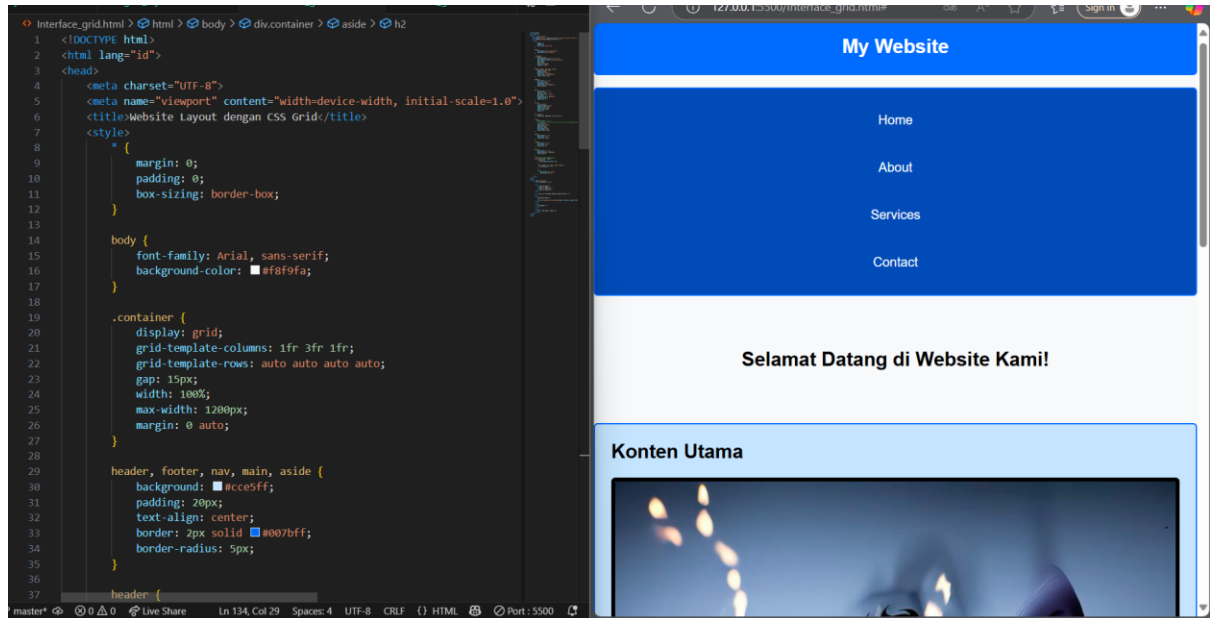


Deskripsi/Keterangan :

Kode di atas menggunakan CSS Grid untuk membuat tata letak halaman dengan beberapa item dalam sebuah kontainer grid. .grid-container memiliki empat kolom dengan jarak antar elemen 50px dan latar belakang biru. Elemen pertama (.item1) membentang dari kolom 1 hingga 4, sehingga lebih lebar dibanding elemen lainnya. Item lainnya (.item) memiliki latar belakang putih transparan dengan teks di tengah. Pada hasil tampilan di gambar, item pertama berada di atas dengan ukuran lebih besar, sementara item lainnya tersusun dalam beberapa baris di bawahnya.

No. 7

Hasil Tangkapan Layar :



Deskripsi/Keterangan :

Kode di atas menggunakan CSS Grid untuk membuat layout website yang responsif. `.container` membagi halaman menjadi tiga kolom utama: sidebar kiri (1fr), konten utama (3fr), dan sidebar kanan (1fr). Elemen header, nav, hero, dan footer membentang penuh (grid-column: 1 / -1). Navigasi memiliki menu horizontal dengan efek hover. Konten utama (main) berisi teks dan gambar, sementara aside digunakan untuk sidebar tambahan. Warna biru digunakan untuk header, navigasi, dan footer, menciptakan tampilan profesional. Pada layar kecil, layout berubah menjadi satu kolom untuk meningkatkan keterbacaan.