**Jobsheet 2: CSS**

****

Muhammad Alif Febriansyah

2341720025

TI\_2E

**Praktikum bagian 1. Menghubungkan HTML dengan CSS**

1. Buat folder baru bernama minggu2. Buat file di dalamnya dan beri nama sebagai

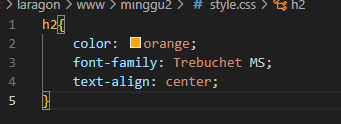
index.html.

1. Ketik ke dalam file index.html kode di bawah ini.



1. Buat satu file baru di folder minggu2, beri nama style.css. Ketik kode di bawah ini di

dalam file style.css.



1. Simpan file, lalu buka browser dan jalankan localhost/minggu2
2. Dalam struktur halaman web, index.html akan selalu menjadi halaman utama web. Jadi

jika ada index.html di direktori maka itu akan selalu dipanggil dan ditampilkan di

browser. Itulah mengapa pada langkah 6 hanya diketik halaman localhost/minggu2

tanpa nama file.

1. Dalam kode di dalam index.html ada tag <link> di kepala di mana atribut href

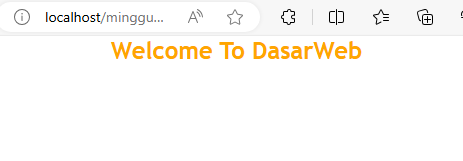
mengacu pada file style.css.

1. style.css berisi kode untuk mengatur tampilan di dalam halaman yang mengacu pada

file style.css, dalam praktikum ini index.html. Sehingga tampilan di dalam

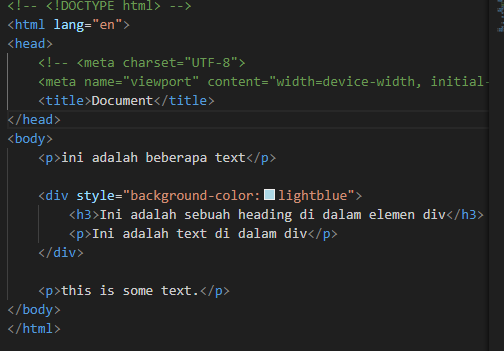
index.html akan sesuai dengan properti yang ditetapkan di style.css.

1. Begini tampilan html Ketika di style menggunakan css



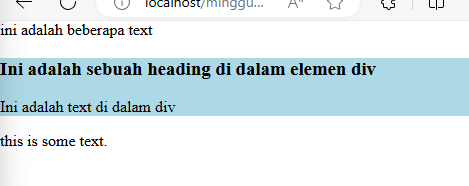
**Praktikum Bagian 2: Penggunaan div**

1. Buat file baru bernama div.html di dalam folder minggu2. Ketik kode pada langkah 2 di dalam div.html



1. Simpan file, lalu buka browser Anda dan jalankan localhost/minggu2/div.html
2. Apa yang Anda pahami dari menggunakan div pada file? Tuliskan pemahaman Anda.

(Pertanyaan No. 1)

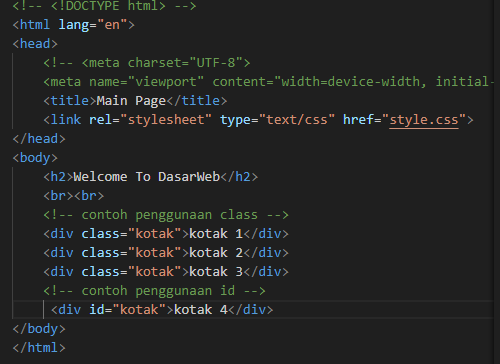


Jawab:

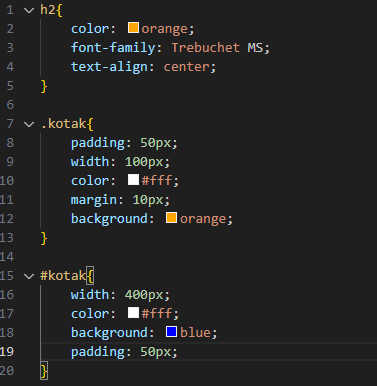
Div digunakan untuk mengelompokkan konten dalam suatu blok sehingga dapat diatur tata letak atau gaya visualnya menggunakan css.

**Praktis Bagian 3: Mengenal kelas dan id dalam HTML**

1. Lengkapi kode di dalam index.html dengan kode berikut.



1. Lengkapi kode di dalam file style.css sehingga menjadi kode di langkah 4



1. Simpan kedua file, lalu buka browser dan jalankan /refresh localhost/minggu2
2. Apa yang Anda pahami dari penggunaan class dan id di index.html? Tuliskan pemahaman Anda. (Pertanyaan No. 2)



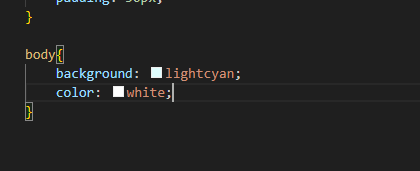
Jawab:

Class menerapkan gaya yang sama pada semua elemen yang memiliki class tersebut. Satu class dapat digunakan pada banyak elemen berbeda. Pemanggilan class menggunakan titik(.).

Id digunakan untuk menargetkan elemen tertentu secara spesifik, id ini hanya dapat digunakan sekali dalam satu halaman html. Pemanggilan id menggunakan hashtag (#).

**Praktikum Bagian 4: Mengubah Latar Belakang Halaman Web dengan CSS**

1. Tambahkan cuplikan kode pada langkah 2 di bawah ini ke file style.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2
2. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini. (Pertanyaan No. 3)

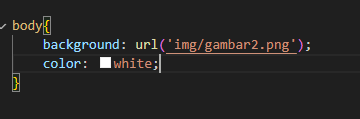


Jawab:

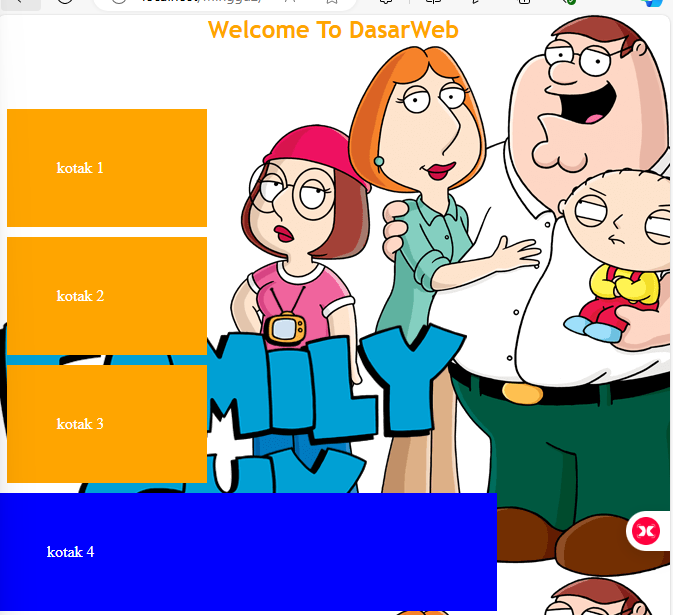
Dengan menambahkan properti background-image, gambar akan digunakan sebagai latar belakang elemen. Jika tidak diatur dengan benar, gambar mungkin tidak terlihat atau terpotong.

**Praktikum Bagian 5: Menggunakan gambar untuk latar belakang**

1. Di file style.css, ubah nilai dari properti background menjadi bunga2.jpg (atau nama file gambar sebagai background) seperti pada cuplikan kode pada langkah 2.



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2
2. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini. (Pertanyaan No. 4)



Jawab:

Setelah mengubah latar belakang menjadi gambar, gambar tersebut akan muncul di elemen sesuai pengaturannya.

1. Tambahkan file gambar bernama field1.jpg dan tree1.jpg (atau nama file gambar sebagai latar belakang dan objek) ke folder img.
2. Ketik kode berikut di style.css.
3. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2
4. Kode Anda berjalan dengan baik jika terlihat seperti di bawah ini:
5. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini. (Pertanyaan No. 5)

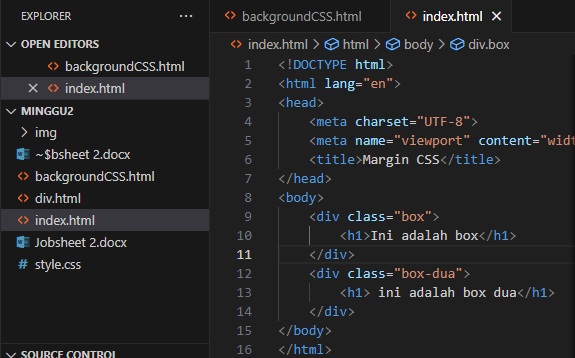
Jawab:

Margin menentukan jarak luar elemen, sedangkan padding menentukan jarak dalam elemen. Keduanya digunakan untuk mengatur ruang di antara elemen-elemen halaman.

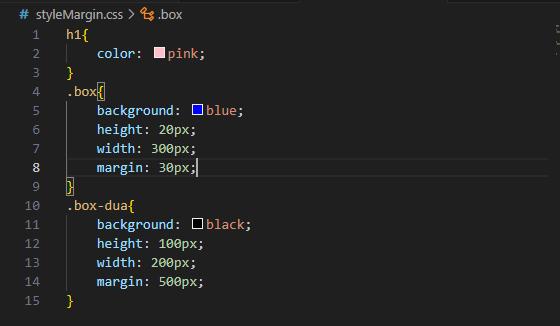
**Praktikum Bagian 6: Margin dan Padding pada CSS**

**Margin CSS**

1. Ganti nama index.html menjadi backgroundCSS.html
2. Buat file baru di dalam folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru .
3. Ketik kode berikut di index.html



1. Buat file baru bernama styleMargin.css di folder minggu2.



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2



1. Tuliskan apa yang Anda amati dari kode di atas. (Pertanyaan No. 6)

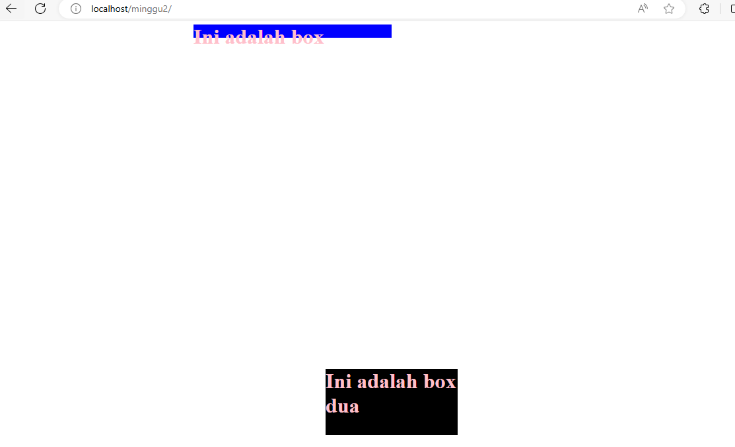
Jawab:

Saat margin diatur pada elemen .box, elemen tersebut bergerak menjauh dari elemen lainnya sesuai dengan nilai margin yang diberikan.

1. Ubah nilai margin di pengaturan .box menjadi margin: 300px;, amati perbedaannya.



Output



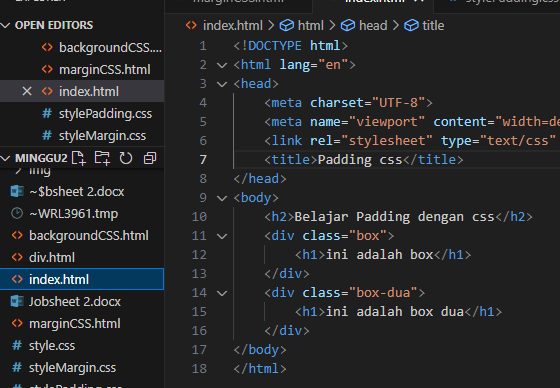
1. Tuliskan apa yang Anda amati dari kode di atas. (Pertanyaan No. 7)

Jawab:

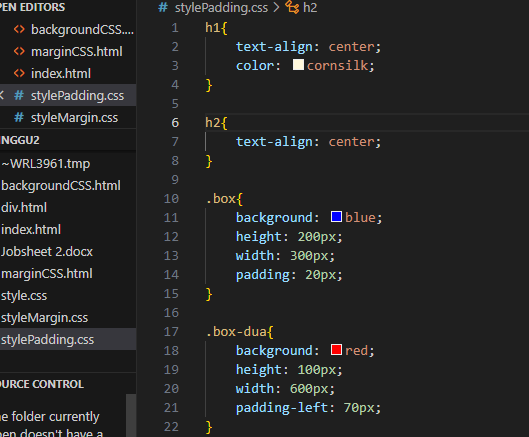
Setelah margin diubah menjadi 300px, elemen .box memiliki jarak yang sangat besar dari elemen-elemen lain atau dari batas halaman, menciptakan spasi yang jelas.

**Padding CSS**

1. Ganti nama index.html menjadi marginCSS.html
2. Buat file baru di folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru.
3. Ketik kode berikut untuk index.html



1. Buat file baru bernama stylePadding.css di folder minggu2
2. Ketik kode berikut untuk stylePadding.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2

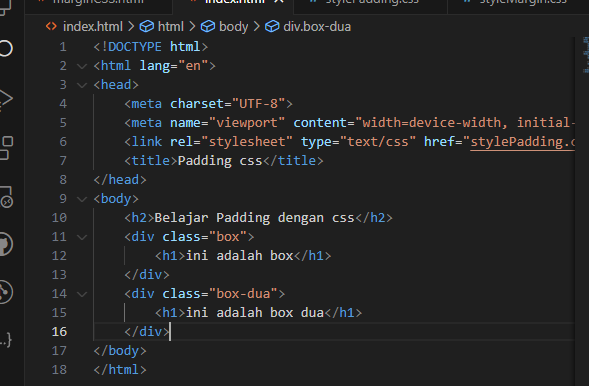


1. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini. (Pertanyaan No. 8)

Jawab:

Padding menambah ruang di dalam elemen, membuat konten seperti teks atau gambar di dalam elemen tersebut lebih jauh dari batas (border).

1. Ubah nilai padding dalam pengaturan .box menjadi padding=200px;



Output



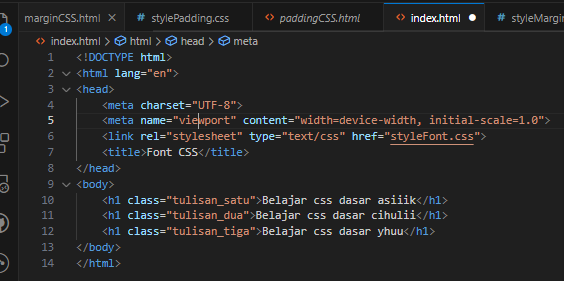
1. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini. (Pertanyaan No. 9)

Jawab:

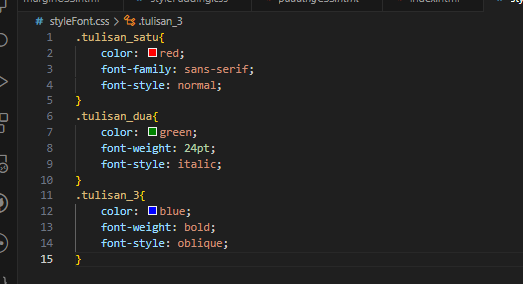
Saat padding diubah menjadi 200px, jarak antara konten elemen dengan batas elemen menjadi sangat besar, sehingga konten terlihat lebih jauh dari tepi elemen.

**Praktikum Bagian 7: Pengaturan Font di CSS**

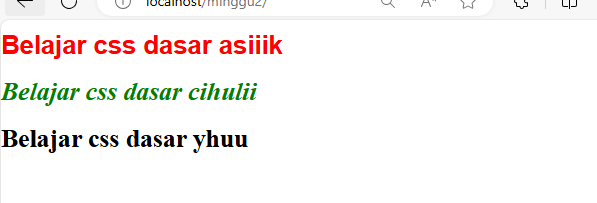
1. Ganti nama index.html menjadi paddingCSS.html
2. Buat file baru di folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru .
3. Ketik kode berikut pada index.html



1. Buat file baru bernama styleFont.css di folder minggu2
2. Ketik kode berikut ke styleFont.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2



1. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini (Pertanyaan No. 10)

Jawab:

CSS memungkinkan untuk mengatur ukuran (font-size), jenis (font-family), ketebalan (font-weight), dan gaya (font-style) font. Ini memungkinkan penyesuaian tampilan teks agar sesuai dengan desain web.

**Praktikum Bagian 8: Menyiapkan Hyperlink dengan CSS**

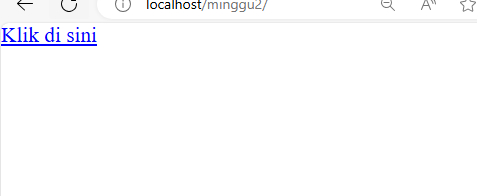
1. Ganti nama index.html menjadi fontCSS.html
2. Buat file baru di folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru.
3. Ketik kode berikut untuk index.html



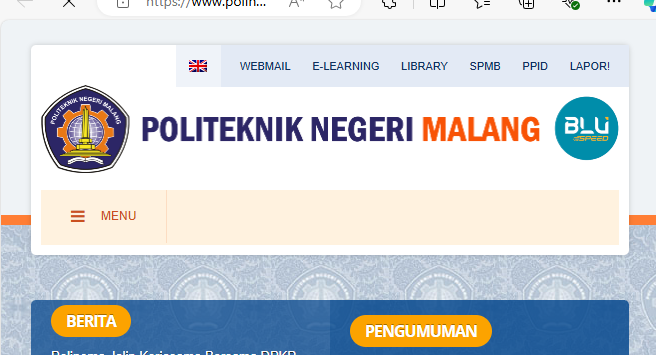
1. Buat file baru bernama styleLink.css di folder minggu2
2. Ketik kode berikut untuk styleLink.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2



Setelah di klik tautan

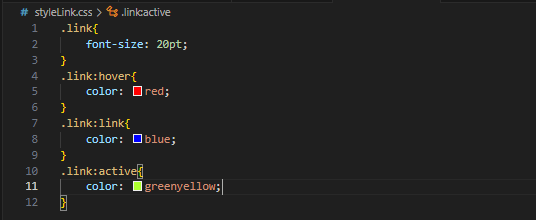


1. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini (Pertanyaan No. 11)

Jawab:

Hyperlink memiliki beberapa status: link biasa, link yang telah dikunjungi, link yang di-hover, dan link yang aktif. CSS dapat mengatur tampilan untuk masing-masing status, misalnya mengubah warna atau memberikan efek ketika link diklik atau di-hover.

1. Tambahkan kode ke styleLink.css untuk mengubah warna tautan menjadi greenyellow setelah tautan dikunjungi.



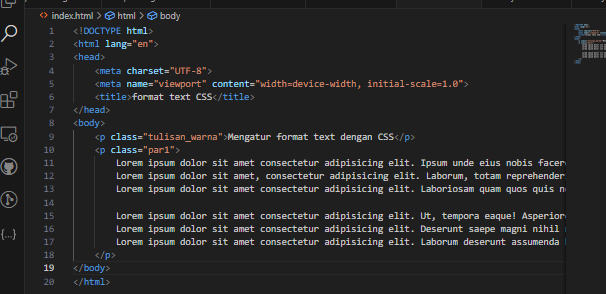
1. Tulis kode yang Anda tambahkan di sini (Pertanyaan No. 12)

Jawab

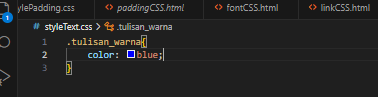
a:visited { color: greenyellow; }

**Praktikum Bagian 9: Memformat Text dengan CSS**

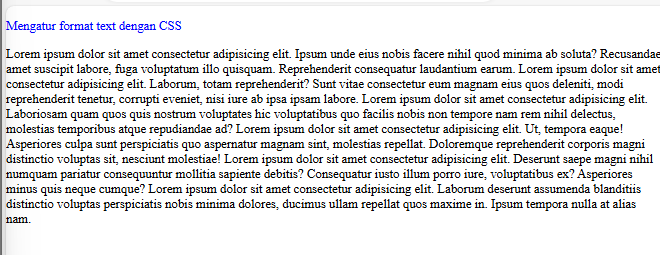
1. Ganti nama index.html menjadi linkCSS.html
2. Buat file baru di folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru .
3. Ketik kode berikut untuk index.html.



1. Buat file baru bernama styleText.css di folder minggu2
2. Ketik kode pada langkah 7 di bawah ini ke styleText.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2



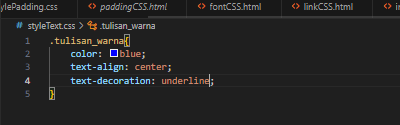
1. Catat pengamatan Anda dan tuliskan jawaban Anda di bawah ini (Pertanyaan No. 13)

Jawab:

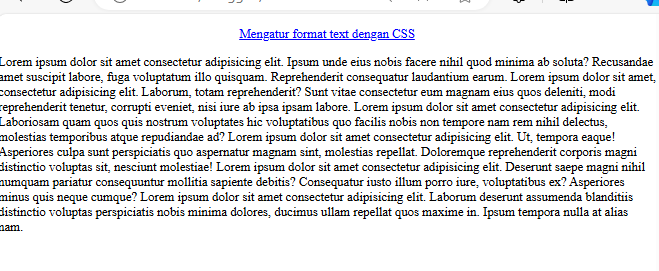
Menambahkan text-align: center; akan memusatkan teks. Dekorasi garis bawah (text-decoration: underline;) akan memberikan garis bawah pada teks.

1. Tambahkan kode ke styleText.css untuk mengubah posisi teks ke tengah dengan menambahkan text-align: center; dan menambahkan dekorasi dalam bentuk garis bawah.

Kode



Output



1. Tulis kode yang Anda tambahkan di sini (Pertanyaan No. 14)

Jawab:

p {

text-align: center;

}

1. Tambahkan kode ke styleText.css untuk menambahkan garis bawah ke teks.
2. Tulis kode yang Anda tambahkan di sini (Pertanyaan No. 15)

Jawab:

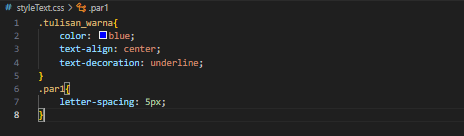
p {

text-decoration: underline;

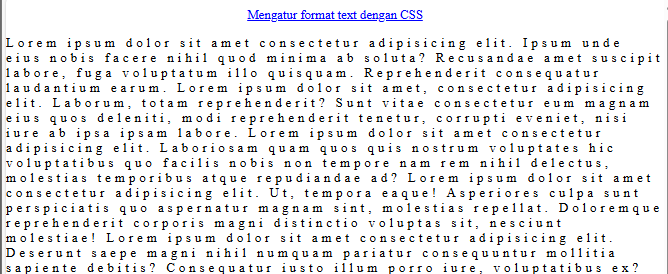
}

1. Tambahkan kode ke styleText.css untuk memberikan spasi antar karakter dalam paragraph yang ada menjadi 5px dengan letter-spacing.

Kode



Output



1. Tulis kode yang Anda tambahkan di sini (Pertanyaan No. 16)

Jawab:

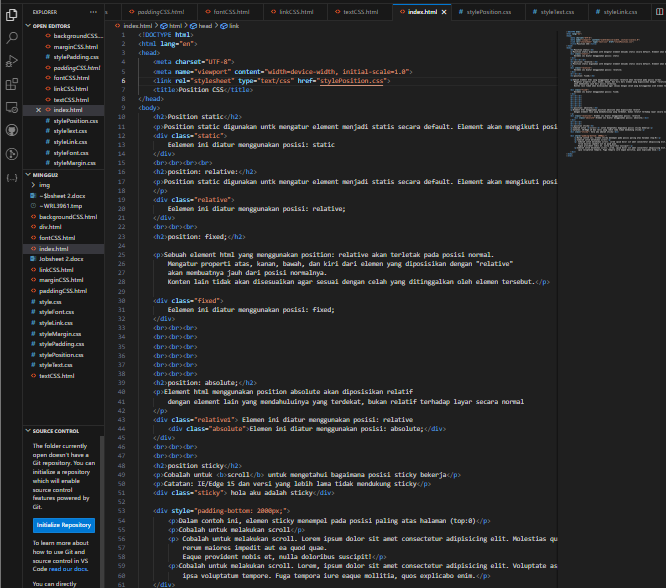
.par1 {

letter-spacing: 5px;

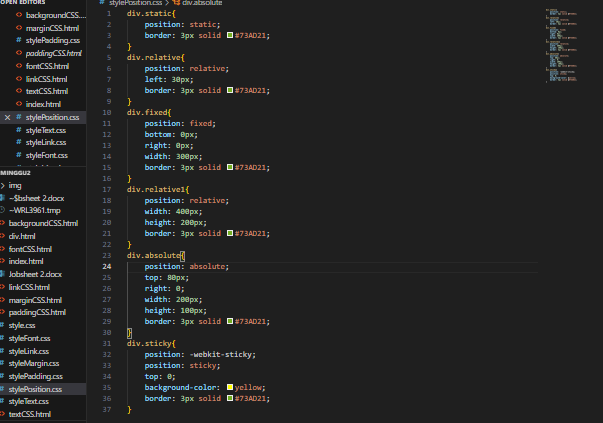
}

**Praktikum Bagian 10: Mengenal Posisi CSS**

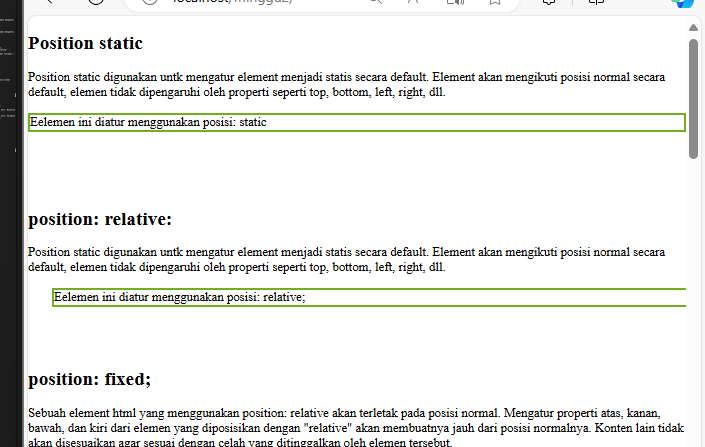
1. Ganti nama index.html menjadi textCSS.html
2. Buat file baru di folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru.
3. Ketik kode berikut untuk index.html



1. Buat file baru bernama stylePosition.css di folder minggu2
2. Ketik kode berikut untuk stylePosition.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2



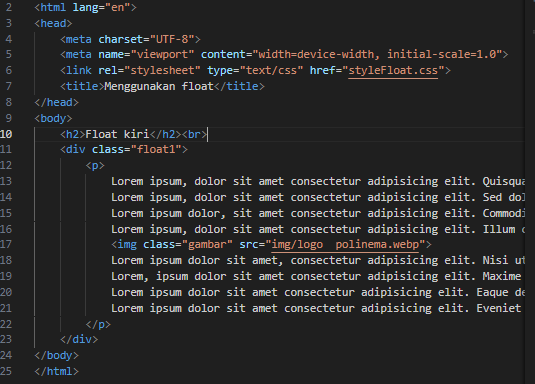
1. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini (Pertanyaan No. 17)

Jawab:

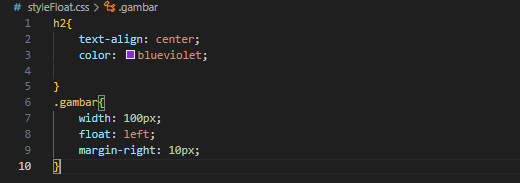
Pengaturan posisi dengan properti seperti static, relative, absolute, dan fixed akan mempengaruhi bagaimana elemen ditempatkan di halaman dan bagaimana interaksi dengan elemen lain.

**Praktikum Bagian 11: Menggunakan Float**

1. Ganti nama index.html menjadi positionCSS.html
2. Buat file baru di folder minggu2 bernama index.html. Ini berarti Anda memiliki file index.html baru.
3. Ketik kode berikut untuk index.html



1. Buat file baru bernama styleFloat.css di folder minggu2
2. Ketik kode berikut untuk styleFloat.css



1. Simpan file, lalu buka browser dan jalankan /refresh localhost/minggu2



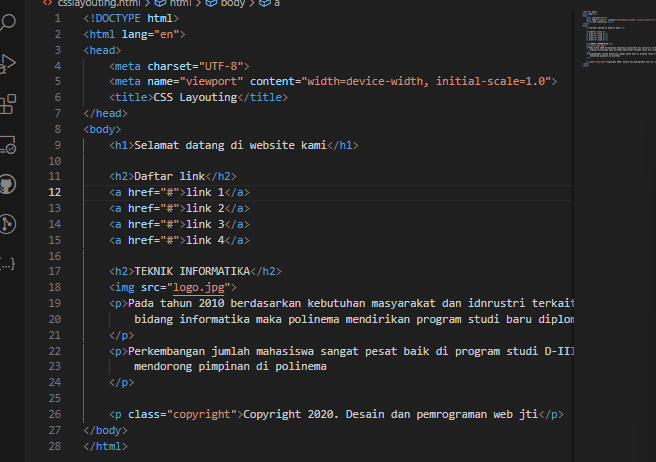
1. Catat pengamatan Anda dan tulis jawaban Anda di bawah ini. (Pertanyaan No. 18)

Jawab:

CSS layout seperti display, position, dan float membantu dalam menyusun elemen di halaman web agar terlihat lebih terstruktur dan responsif.

**Praktikum Bagian 12. Tata Letak CSS**

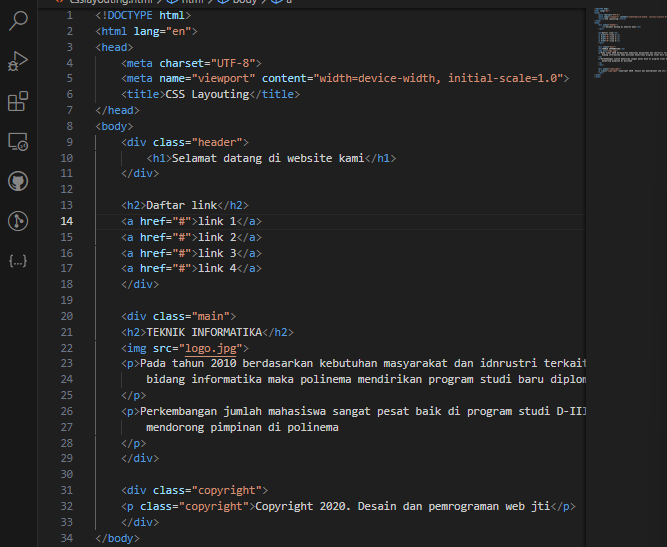
1. Buat file baru di folder minggu2 dan beri nama csslayouting.html
2. Ketik kode berikut untuk csslayouting.html



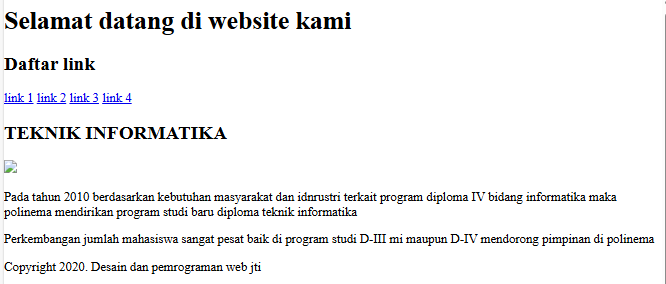
Output



1. Kemudian buat <div> pada halaman html yang sama seperti pada kode di bawah ini



Output

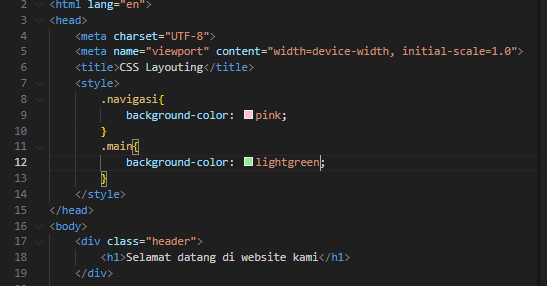


1. Amati apakah hasil dari kedua program tersebut sama atau berbeda, jelaskan (Pertanyaan No. 19)

Jawab:

CSS layout seperti display, position, dan float membantu dalam menyusun elemen di halaman web agar terlihat lebih terstruktur dan responsif.

1. Tambahkan style ke kelas navigasi dan kelas main seperti pada kode di bawah ini



Output



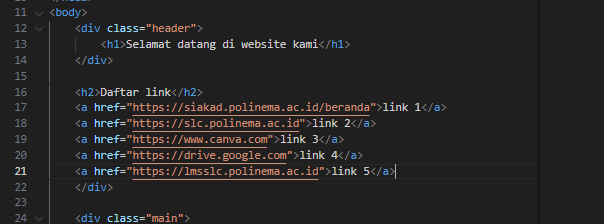
1. Jalankan kode dan screen shoot hasilnya. Jelaskan apa yang terjadi (Pertanyaan No. 20)

Jawab:

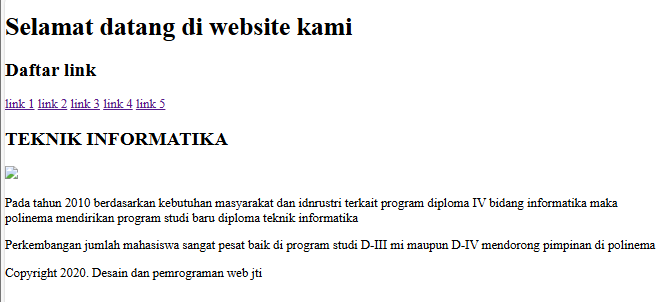
Elemen inline tidak menambahkan baris baru dan hanya mempengaruhi ruang horizontal. Jika diatur dengan display: inline, properti tinggi dan lebar tidak akan berpengaruh.

**Praktikum Bagian 13. Tampilan Sebaris**

1. Hapus style di langkah 9 Praktikum Bagian 12. Tambahkan 5 link ke kelas navigasi seperti yang ditunjukkan pada kode di bawah ini.



Output



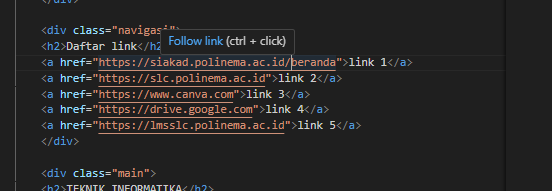
1. Jalankan kode dan screen shoot hasilnya. Jelaskan apa yang terjadi (Pertanyaan No. 21)

Jawab:

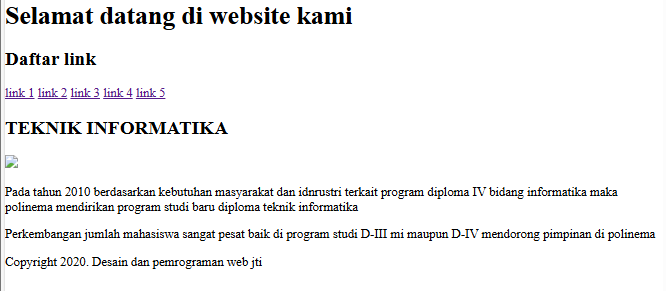
Dengan menggunakan display: inline-block, kita bisa mengatur tinggi dan lebar elemen sebaris, memungkinkan kontrol yang lebih besar pada tampilan elemen.

**Praktikum Bagian 13. Tampilan Sebaris**

1. Hapus style di langkah 9 Praktikum Bagian 12. Tambahkan 5 link ke kelas navigasi seperti yang ditunjukkan pada kode di bawah ini.



Output



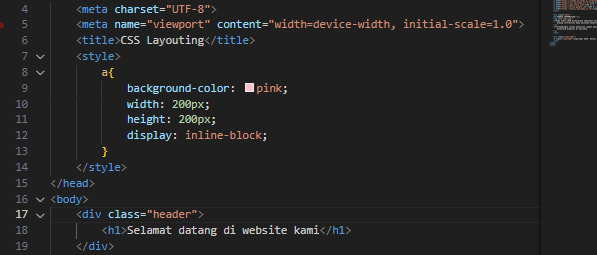
1. Jalankan kode dan screen shoot hasilnya. Jelaskan apa yang terjadi (Pertanyaan No. 21)

Jawab:

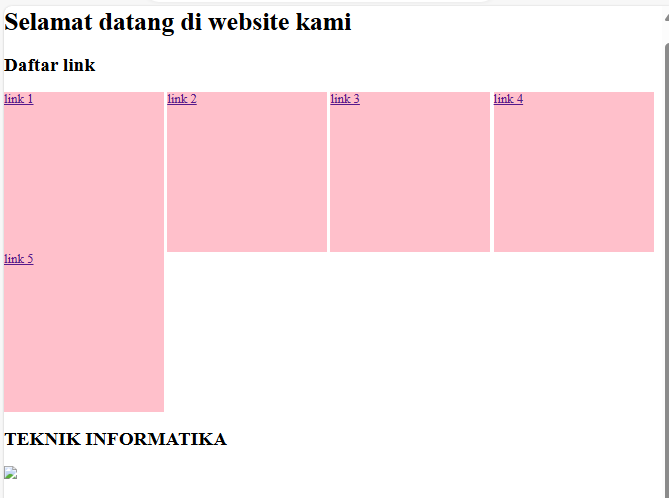
Dengan menggunakan display: inline-block, kita bisa mengatur tinggi dan lebar elemen sebaris, memungkinkan kontrol yang lebih besar pada tampilan elemen.

**Praktikum Bagian 14. Tampilkan blok sebaris**

1. Tambahkan weight, height dan display ke style dalam elemen a seperti yang ditunjukkan pada kode di bawah ini



Output



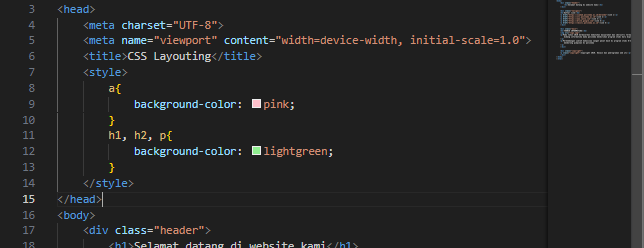
1. Jalankan kode dan screen shoot hasilnya. Jelaskan apa yang terjadi. (Pertanyaan No. 22)

Jawab:

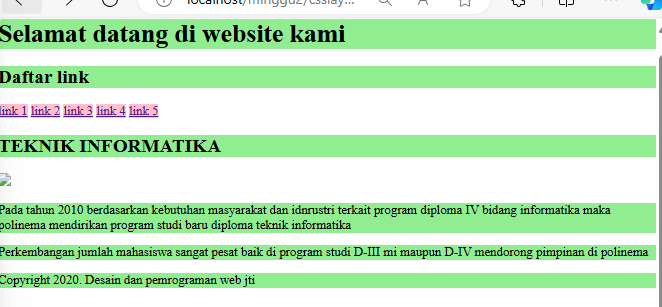
Dengan inline-block, elemen memiliki properti block-level tetapi tetap dalam satu baris dengan elemen lain. Ini berguna untuk pengaturan layout yang lebih fleksibel.

**Praktikum Bagian 15. Display Block**

1. Style elemen h1, h2 dan p dengan background color light-green seperti yang ditunjukkan pada kode di bawah ini



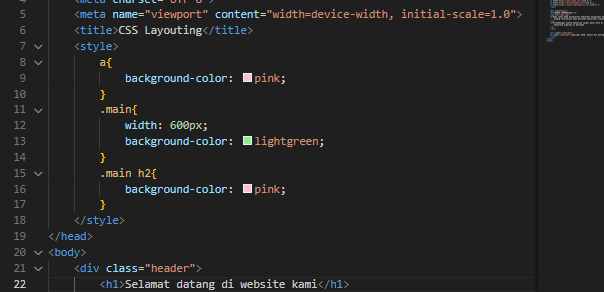
1. Jalankan program dan coba ubah ukuran halaman web. Screen shoot hasilnya, amati dan jelaskan apa yang terjadi. (Pertanyaan No. 23)



Jawab:

Ketika ukuran halaman web diubah, elemen-elemen dengan display: block akan beradaptasi dengan lebar halaman, dan tata letak dapat berubah secara dinamis.

1. Kemudian tambahkan gaya ke kelas main seperti pada kode di bawah ini



1. Jalankan program. Screen shoot hasilnya, amati dan jelaskan apa yang terjadi. (Pertanyaan No. 24)



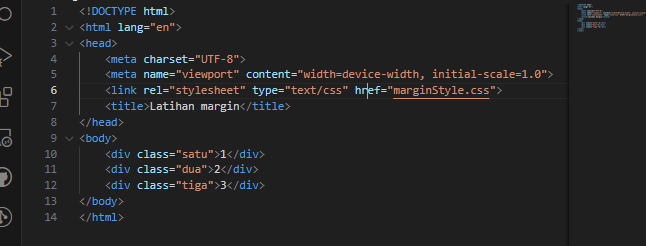
Jawab:

Gaya tambahan pada .main akan mengubah tata letak dan tampilan elemen, seperti perubahan ukuran, warna, atau posisi elemen utama dalam halaman.

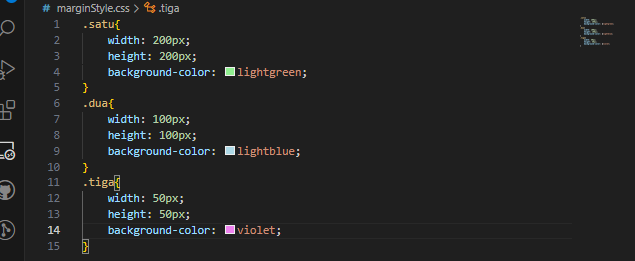
**Praktikum Bagian 16: Model Kotak: Margin**

1. Buat 2 file seperti yang ditunjukkan pada gambar di bawah ini. File pertama dengan nama margin.html dan file kedua marginstyle.css

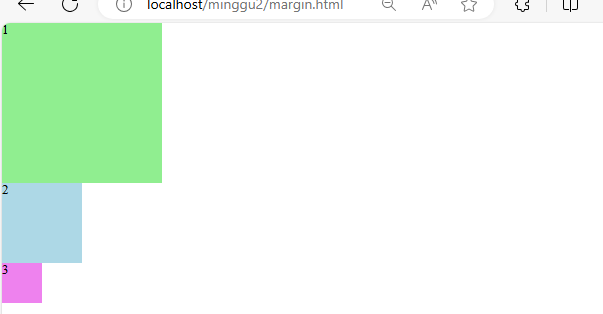
File margin.html



marginStyle.css



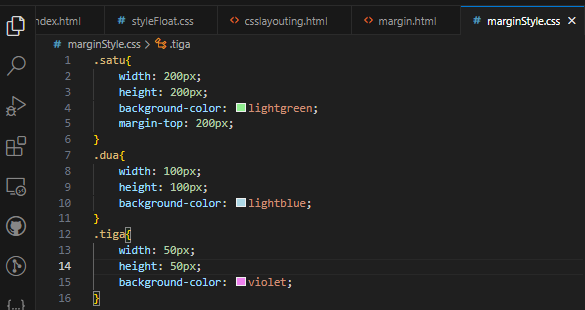
1. Jalankan program, amati hasilnya dan jelaskan ( Pertanyaan No. 25)



Jawab:

Margin memberikan ruang di luar elemen, dan dapat digunakan untuk menciptakan jarak antara elemen-elemen halaman.

1. Selanjutnya adalah memberikan margin, di mana margin adalah area transparan di sekitar kotak. Tambahkan style ke marginstyle.css dengan margin-top 100px pada class .satu, simpan dan kemudian jalankan di browser web. Amati dan jelaskan pengamatan Anda (Pertanyaan No. 26)



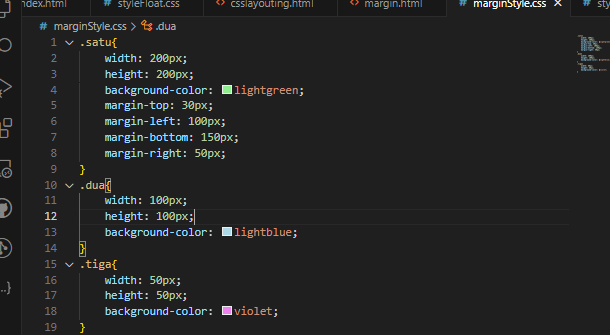
Output



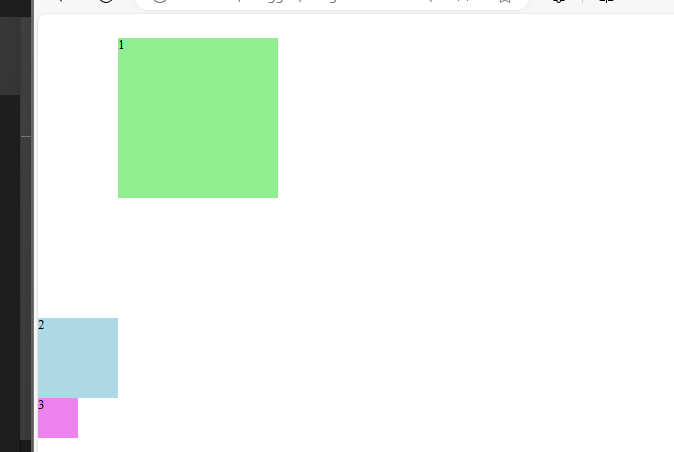
Jawab:

Elemen dengan margin-top: 100px akan memiliki jarak tambahan dari elemen di atasnya, mendorong elemen ke bawah halaman.

1. Kemudian tambahkan ukuran margin lain seperti pada kode berikut



1. Jalankan program. Screen shoot hasilnya, amati dan jelaskan apa yang terjadi. (Pertanyaan No. 27)

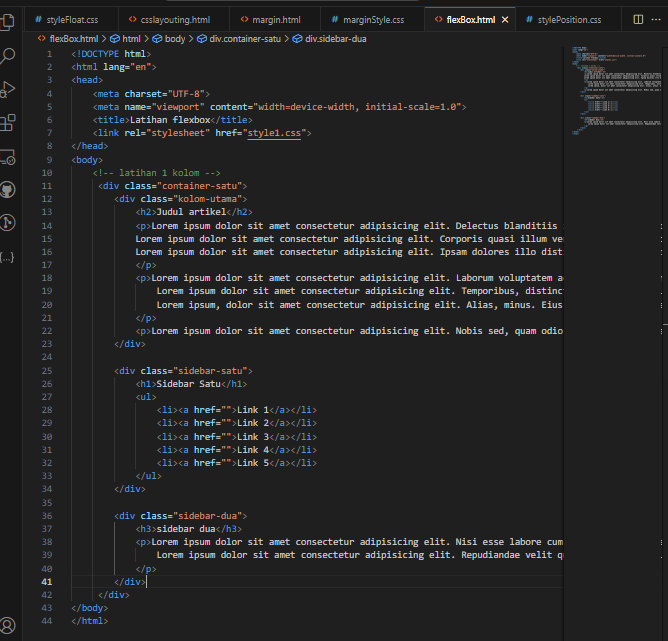


Jawab:

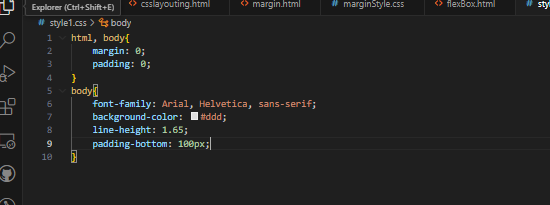
Perubahan margin lainnya akan menyebabkan perubahan dalam posisi elemen pada halaman, baik secara horizontal maupun vertikal.

**Praktikum Bagian 17. Flex Box**

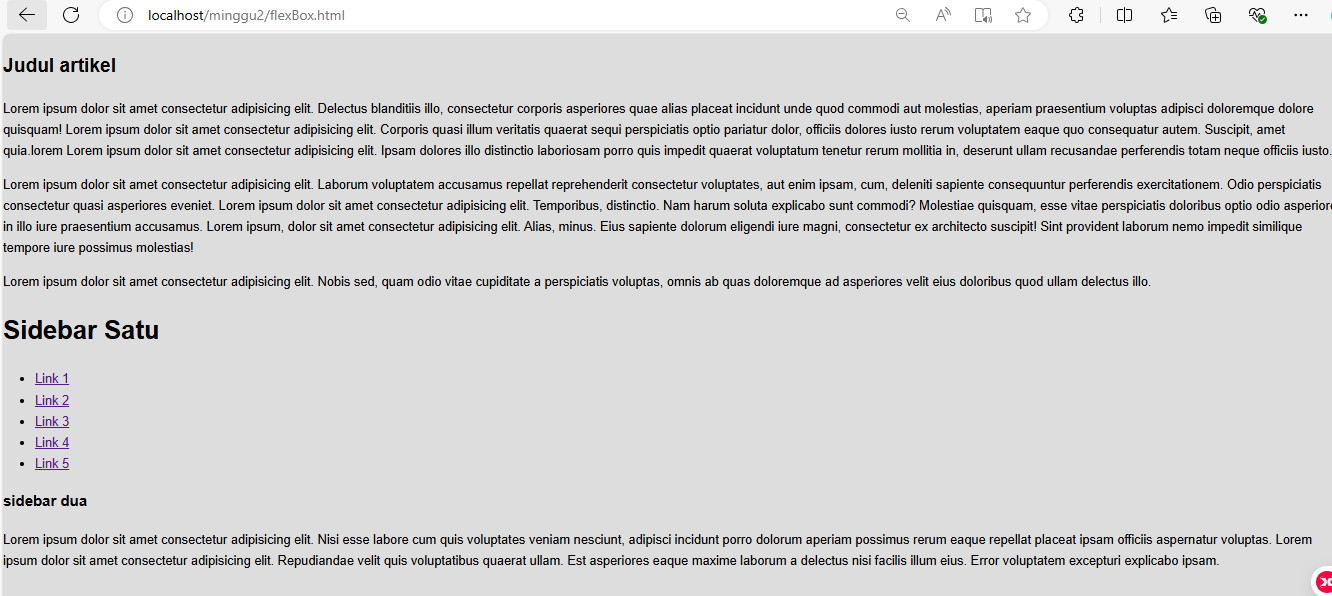
1. Ketik kode di bawah ini lalu simpan file dengan nama flexbox.html



1. Dan untuk file style1.css sebagai berikut



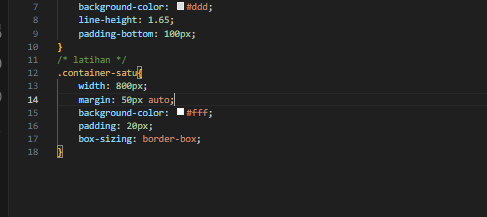
1. Jalankan program, amati hasilnya dan jelaskan ( Pertanyaan No. 28)



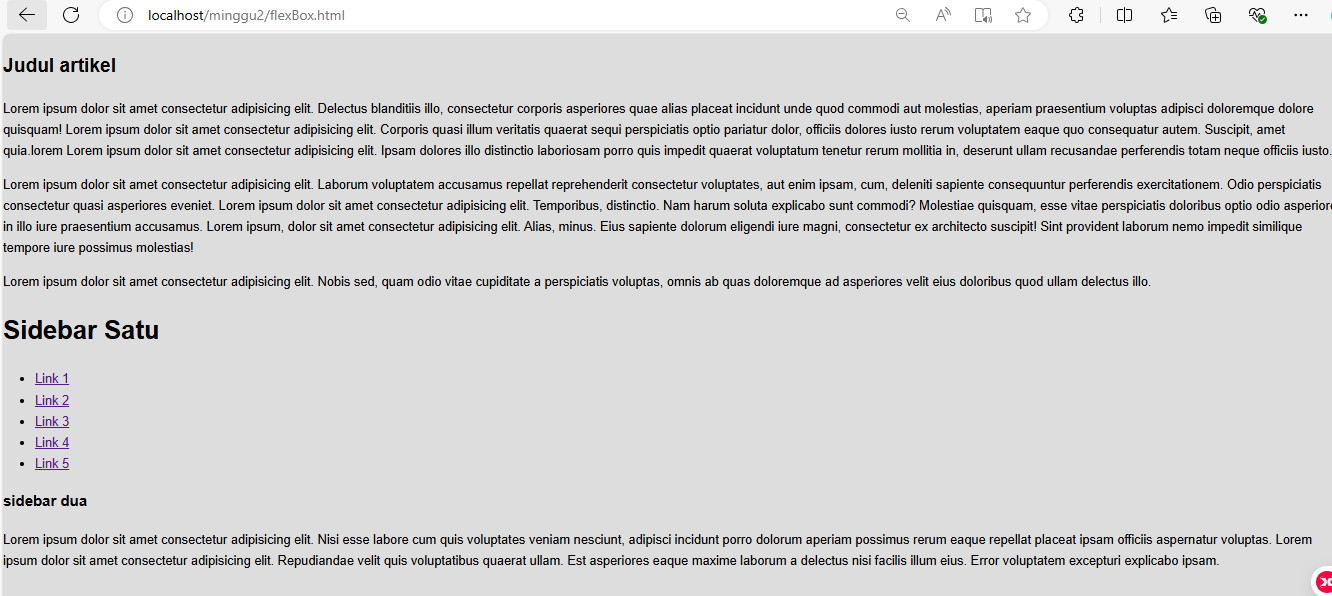
Jawab:

Flexbox mempermudah pengaturan elemen dalam satu dimensi (baris atau kolom), dengan properti seperti justify-content dan align-items untuk pengaturan penjajaran.

1. Tambahkan gaya ke kelas container-satu pada file style1.css seperti pada kode di bawah ini



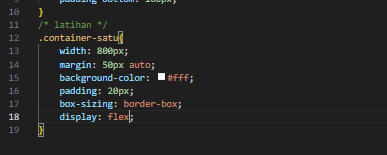
1. Jalankan program. Screen shoot hasilnya, amati dan jelaskan apa yang terjadi. (Pertanyaan No. 29)



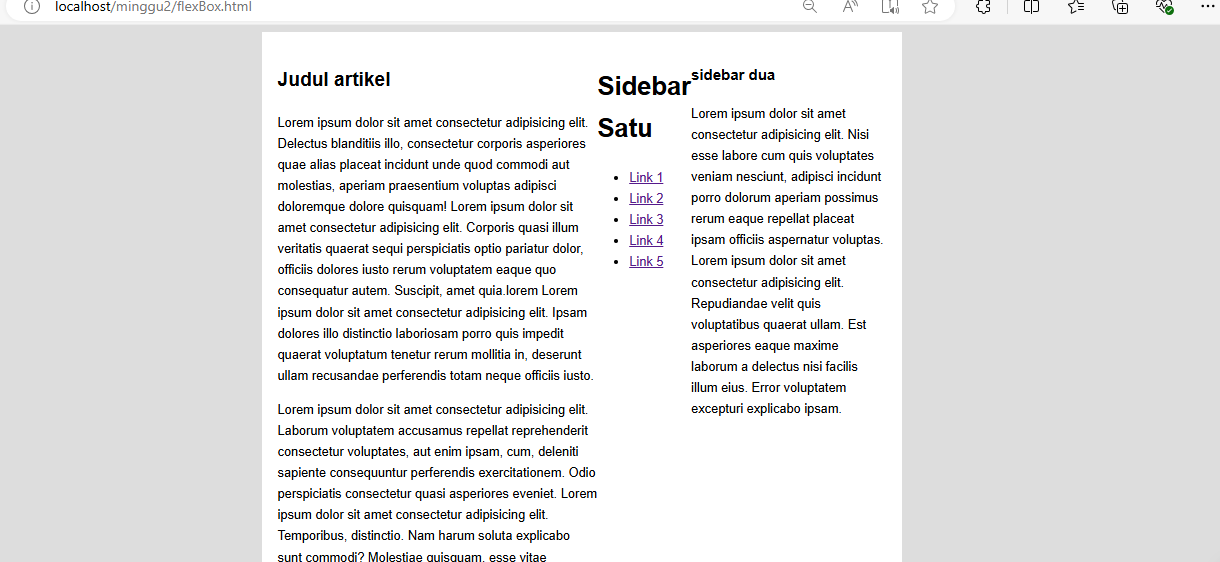
Jawab:

Gaya tambahan pada flex container akan mengatur distribusi elemen anaknya, memungkinkan kontrol yang lebih presisi pada tata letak elemen.

1. Tambahkan property display pada selector container-satu dengan value flex. Capture dan jelaskan hasilnya (Pertanyaan No. 30)



Output

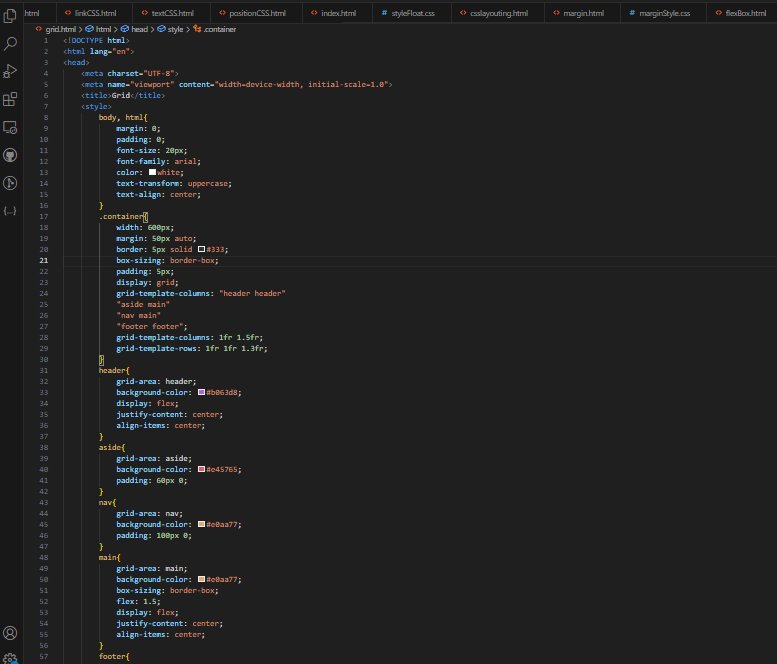


Jawab:

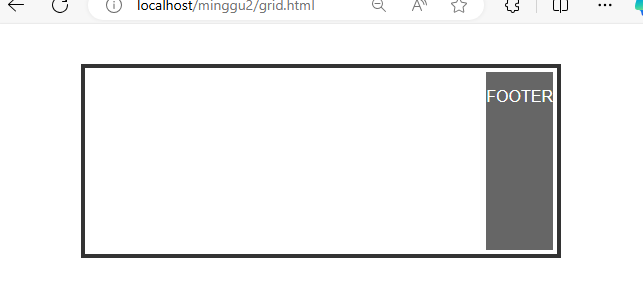
Dengan menambahkan display: flex, elemen-elemen di dalam container akan diatur dalam baris atau kolom sesuai dengan properti flex yang ditentukan.

**Praktikum Bagian 18. Grid CSS**

1. Ketik kode di bawah ini lalu beri nama dengan grid.html



1. Jalankan program, amati hasilnya dan jelaskan. (Pertanyaan No. 31)

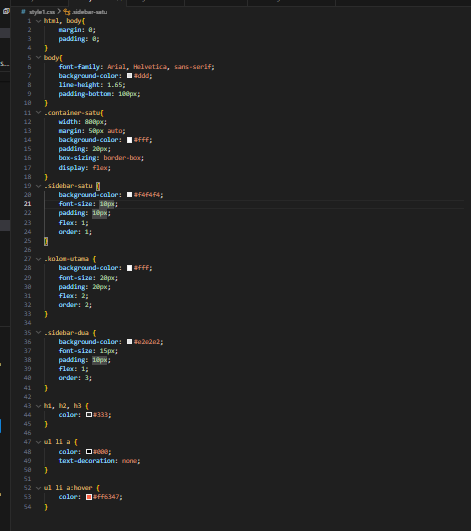


Jawab:

CSS Grid memungkinkan pengaturan layout dalam dua dimensi (baris dan kolom), membuat tata letak halaman menjadi lebih fleksibel dan terstruktur.

**Lembar Tugas 2**

1. Buatlah tampilan dari hasil flex-box sebelumnya menjadi seperti ini.
2. Tambahkan gambar dan terapkan permainan warna ke latar belakang agar lebih menarik. Kata kunci; Gunakan properti Flex dan Order pada masing-masing pemilih itemnya. Kemudian screen shoot kode dan hasilnya.



Output

