

**LATIHAN PERTEMUAN 2**

**PEMROGRAMAN WEB**



**NAMA : LILIYAN PRAMUDITA**

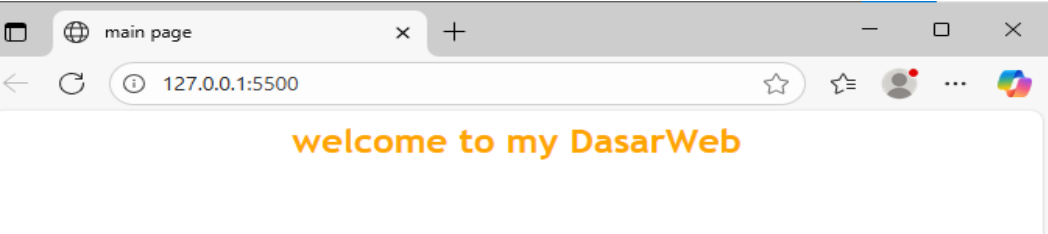
**KELAS : SIB 2C**

**NIM : 244107060096**

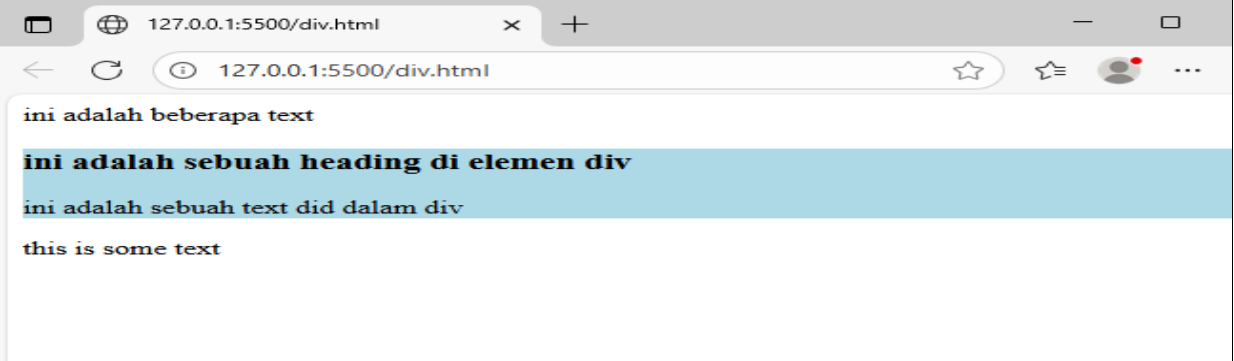
**TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG**

**2025**

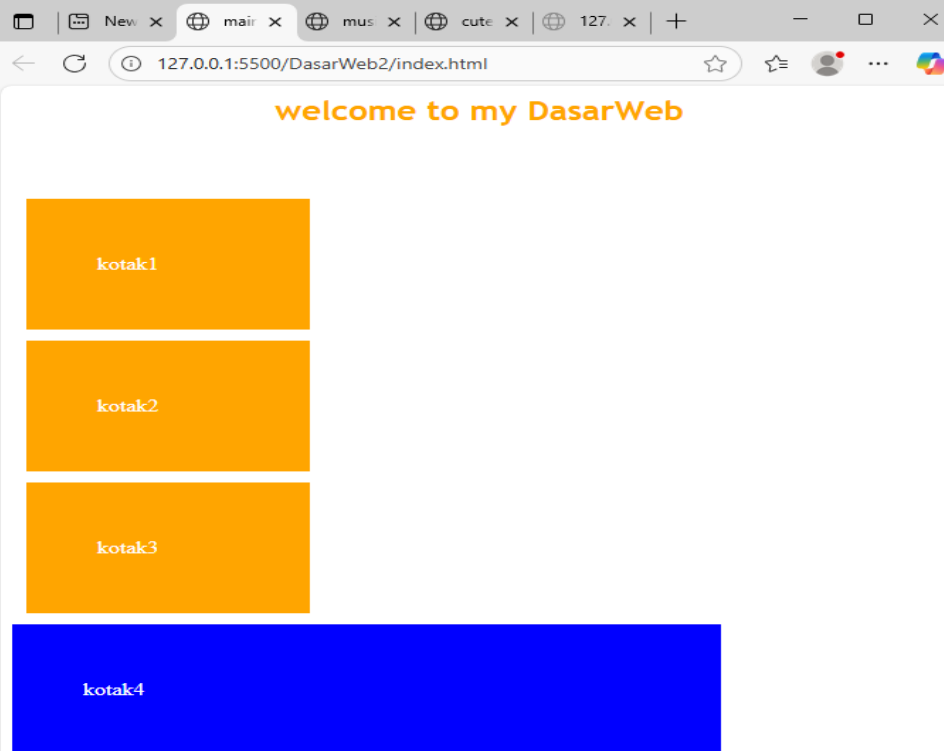
## Praktikum Bagian 1. Menghubungkan HTML dengan CSS

1	 A screenshot of a web browser window. The address bar shows '127.0.0.1:5500'. The page content displays the text 'welcome to my DasarWeb' in a bold, orange, sans-serif font.
---	--

## Praktikum 2: Penggunaan div

1	<p>(soal no 1)</p> <p>penggunaan &lt;div&gt; mempermudah membagi struktur halaman ke beberapa bagian, sehingga bisa diatur tampilannya dengan CSS.</p>  A screenshot of a web browser window showing a page with several lines of text. The text is: 'ini adalah beberapa text', 'ini adalah sebuah heading di elemen div' (highlighted in blue), 'ini adalah sebuah text did dalam div' (highlighted in blue), and 'this is some text'. The browser address bar shows '127.0.0.1:5500/div.html'.
---	--

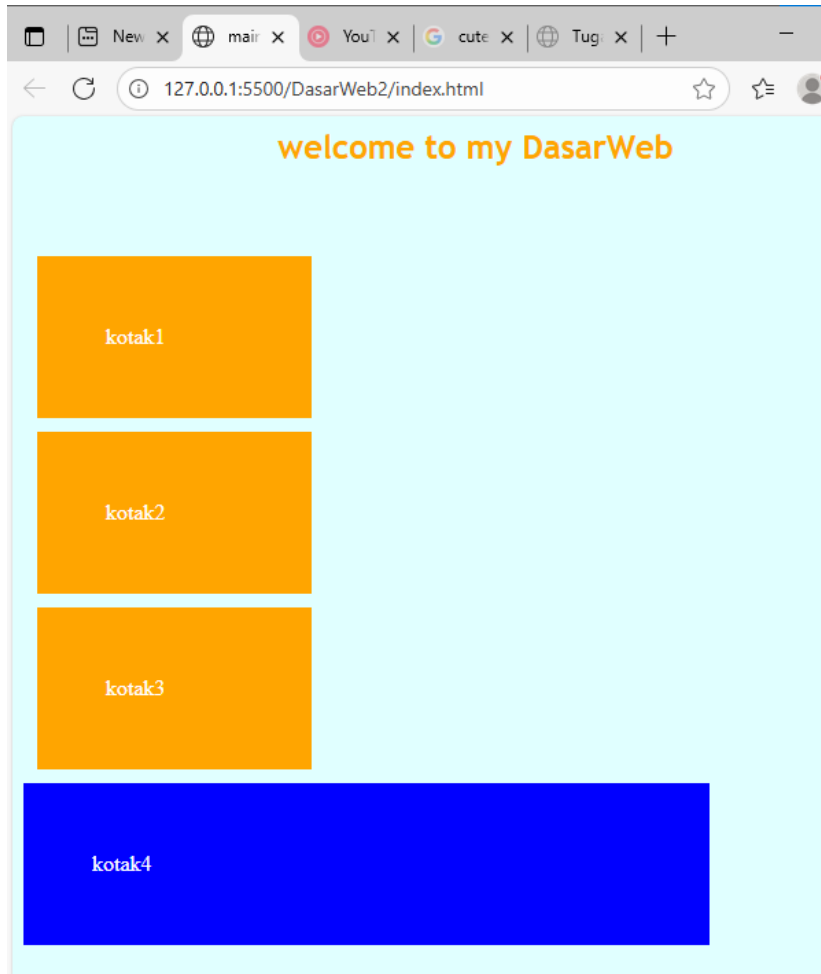
## Praktikum Bagian 3: Mengenal class dan id pada HTML

	<p>(soal no 2)</p> <p>tulisan “Header” jadi warna hijau karena ada aturan internal CSS, menunjukkan bahwa style bisa ditulis langsung di dalam file HTML selain di file CSS eksternal.</p>  A screenshot of a web browser window showing a page with four colored boxes. The first three boxes are orange and labeled 'kotak1', 'kotak2', and 'kotak3' respectively. The fourth box is blue and labeled 'kotak4'. The browser address bar shows '127.0.0.1:5500/DasarWeb2/index.html'.
--	--

## Praktikum Bagian 4: Mengubah *Background* Halaman Web dengan CSS

(soal no 3)

Perubahan warna pada background web.



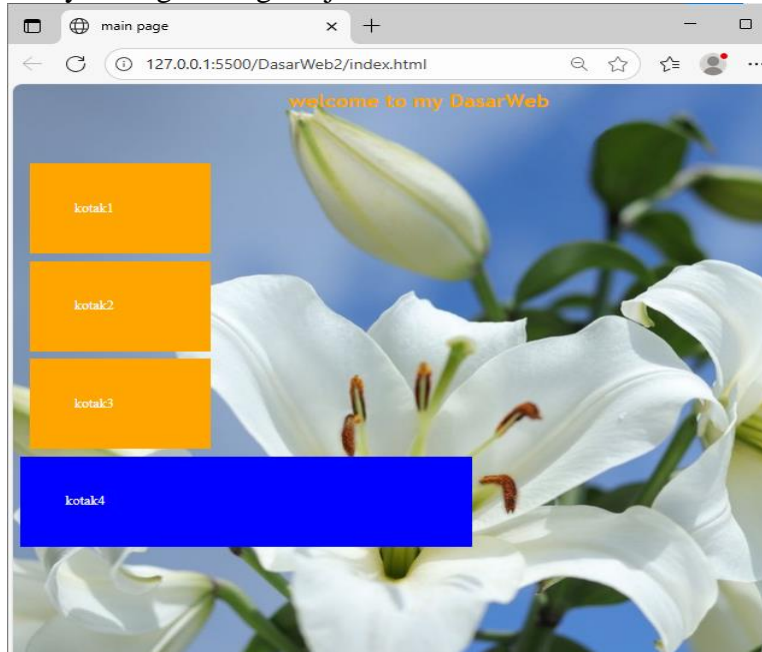
**Praktikum Bagian 5: Menggunakan *image* untuk *Background***

**(soal no 4)**

Background halaman berubah mengikuti gambar yang dipanggil, jadi kita bisa mengganti latar belakang dengan gambar sesuai kebutuhan.

elemen bergeser keluar dari posisi semula — terjadi ruang kosong di luar elemen (antara elemen dan elemen lain). Semakin besar nilai margin, semakin jauh elemen dari tetangganya.

Intinya: margin mengatur jarak luar elemen.



**(soal no 5)**

halaman sekarang punya gambar latar; tergantung properti yang dipakai gambar bisa terulang (repeat) atau menutup area (cover). Jika tidak di-set background-size, gambar biasanya diulang; kalau ukuran gambar kecil bisa terlihat motif berulang.



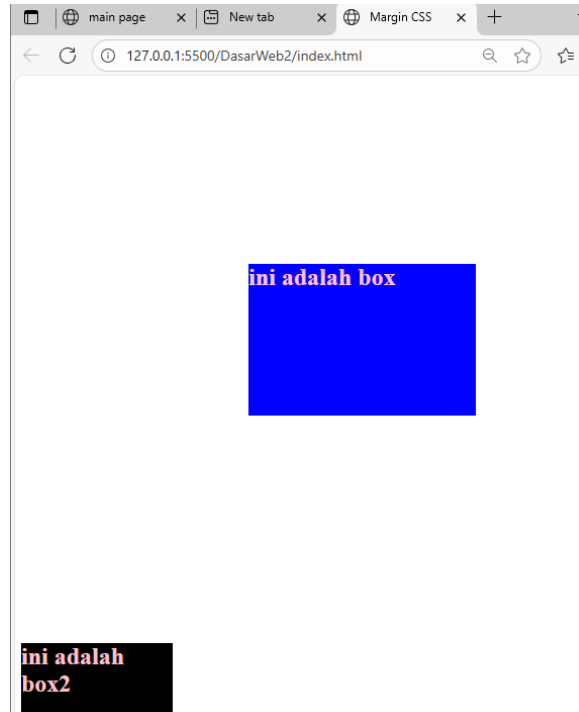
## Praktikum Bagian 6: Margin dan Padding pada CSS

### (soal no 6)

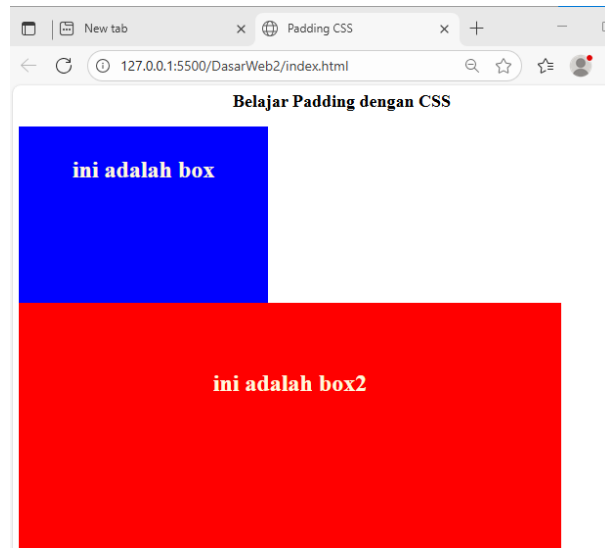
Setiap kotak punya jarak luar (kosong) 50px di semua sisi, jadi tidak menempel dengan elemen lain. kelas .box mendapat jarak luar dari elemen lain sesuai nilai margin yang dipakai. Jika contoh margin: 50px; maka ada 50px kosong di semua sisi elemen.



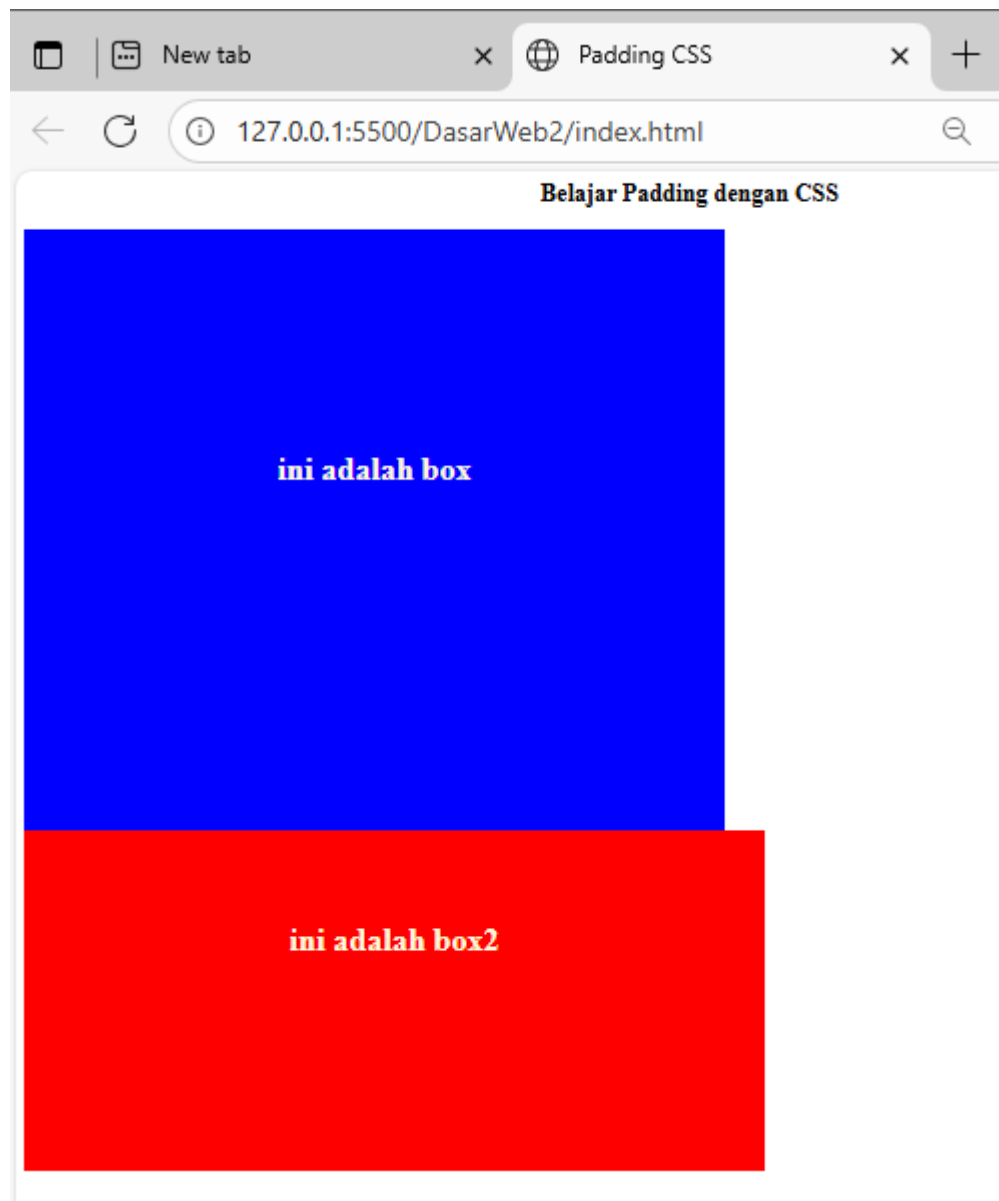
(soal no 7) Elemen berpindah jauh (300px) dari sekelilingnya, kadang terlihat “terlalu jauh” atau bahkan sebagian elemen keluar layar (terutama pada layar kecil). Kesimpulan: nilai margin besar menggeser posisi relatif ke parent/penyusun layout



(soal no 8) padding menambah ruang di dalam elemen — isi (teks/gambar) menjauh dari border/tepi elemen. Ukuran kotak bertambah



(soal no 9) padding besar membuat area isi sangat longgar dan box menjadi sangat besar — bisa menyebabkan overflow, merusak layout, atau membuat scrollbar muncul.



## Praktikum Bagian 7: Pengaturan Font pada CSS

(soal no 10)

font-size → ukuran teks berubah (lebih besar/lebih kecil).

font-weight → tebal teks berubah (normal, bold, angka 100–900).

font-family → jenis huruf berubah (mis. Arial, Times New Roman, Roboto). Jika font tidak tersedia, browser pakai fallback.

font-style: italic; → teks miring.

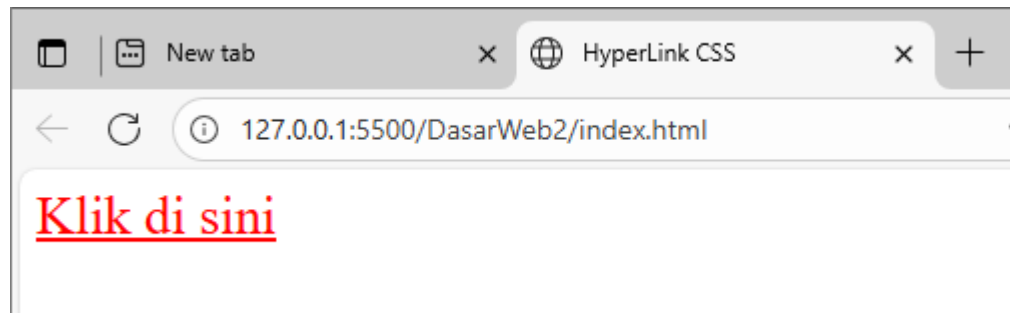
color → warna teks berubah.

Intinya: properti ini mengubah tampilan teks (kejelasan, hierarki, estetika).

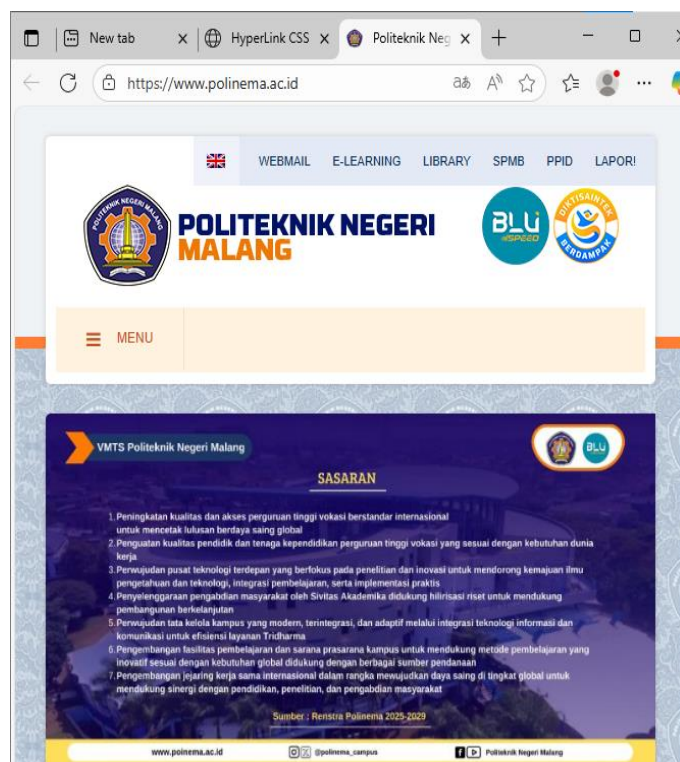


9

(soal no 11)



Saat diklik yang muncul

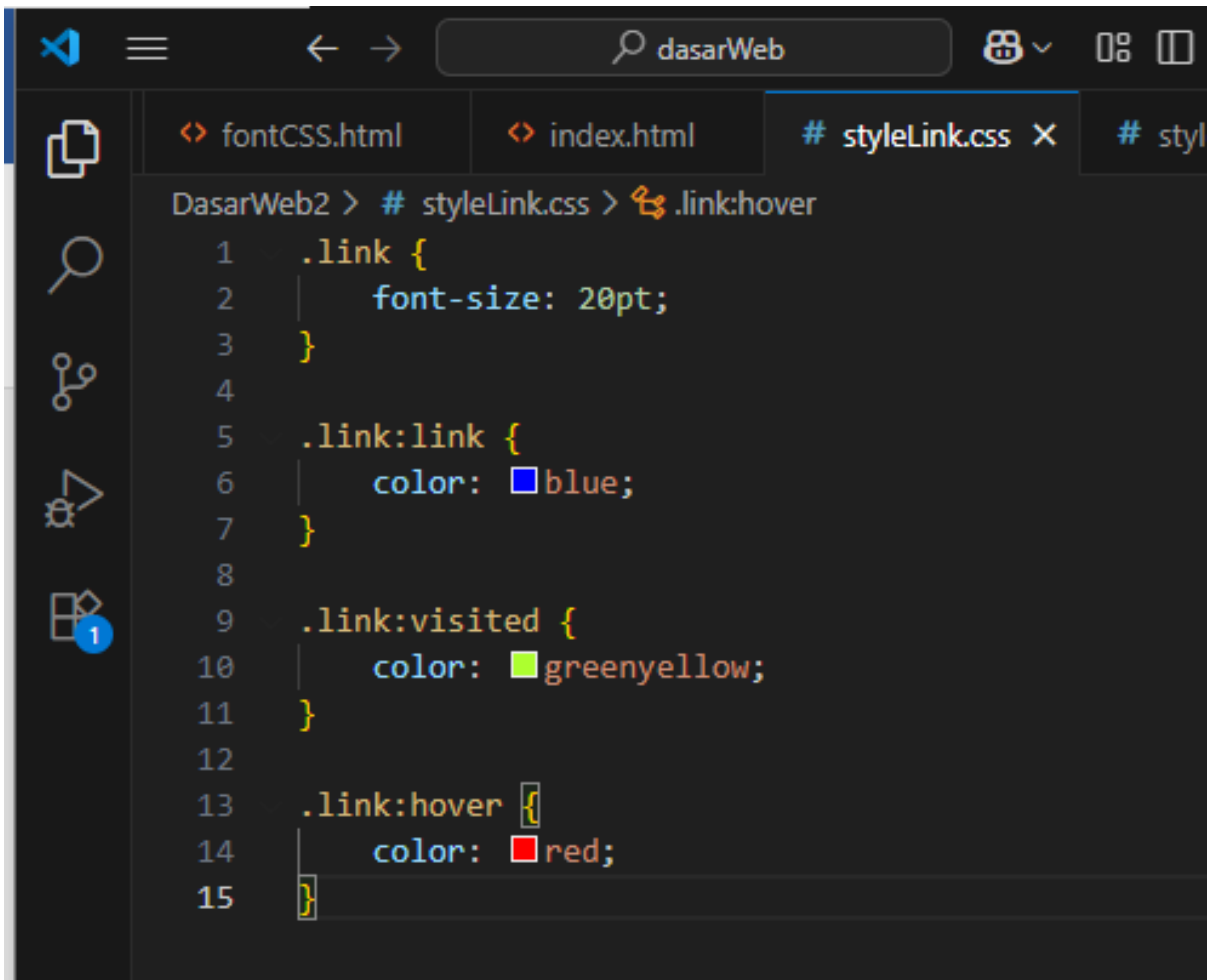


10

Tambahkan kode pada styleLink.css untuk mengubah warna link menjadi hijau muda (greenyellow) setelah link dikunjungi.







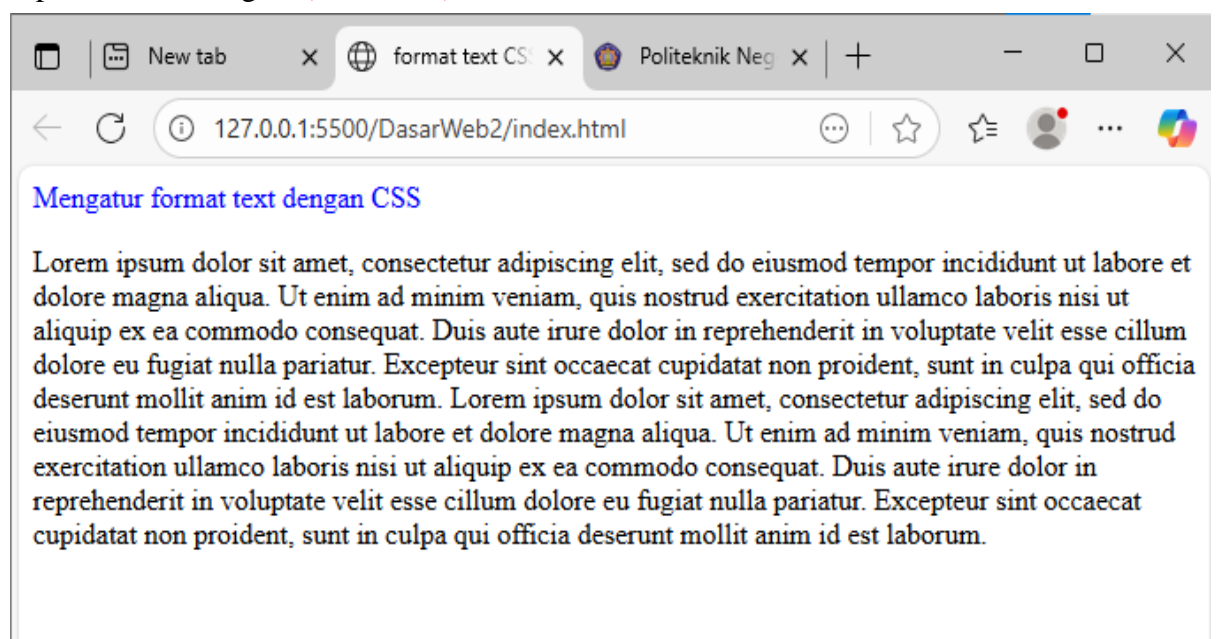
```

DasarWeb2 > # styleLink.css > .link:hover
1  .link {
2      font-size: 20pt;
3  }
4
5  .link:link {
6      color: blue;
7  }
8
9  .link:visited {
10     color: greenyellow;
11 }
12
13 .link:hover {
14     color: red;
15 }

```

## Praktikum Bagian 9: Mengatur Format Text dengan CSS

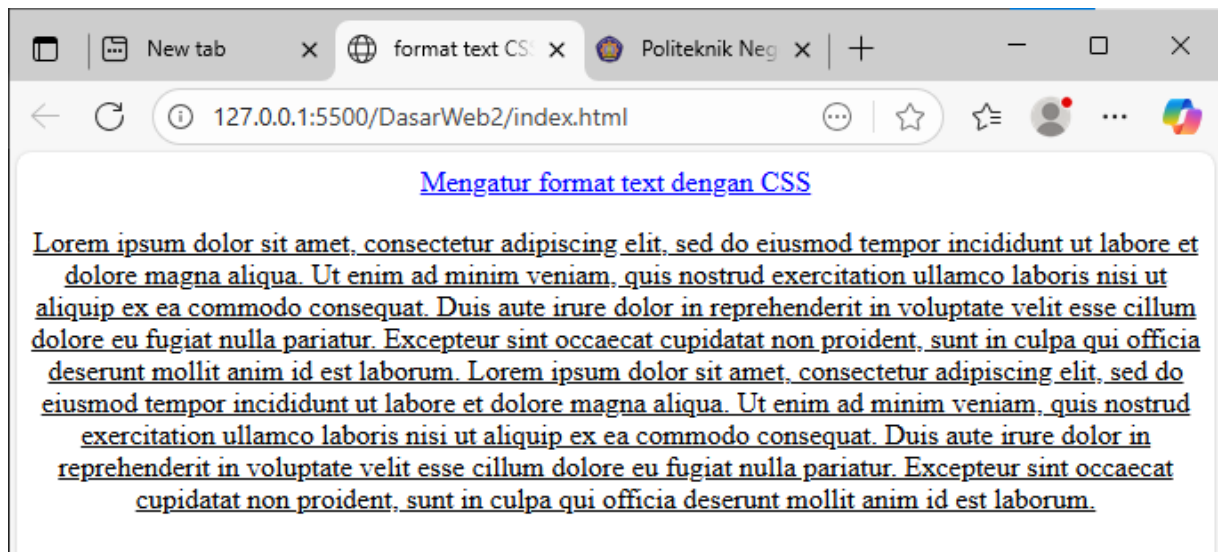
Awalnya teks mengikuti posisi default (static). Elemen tidak bisa diatur dengan top/left/bottom/right. (soal no 13)



11

kode program (soal no 14)

```
linkCSS.html index.html # styleText.css X
DasarWeb2 > # styleText.css > p
1  .tulisan_warna{
2      color: blue;
3  }
4  p{
5      text-align:center;
6      text-decoration:underline;
7  }
```

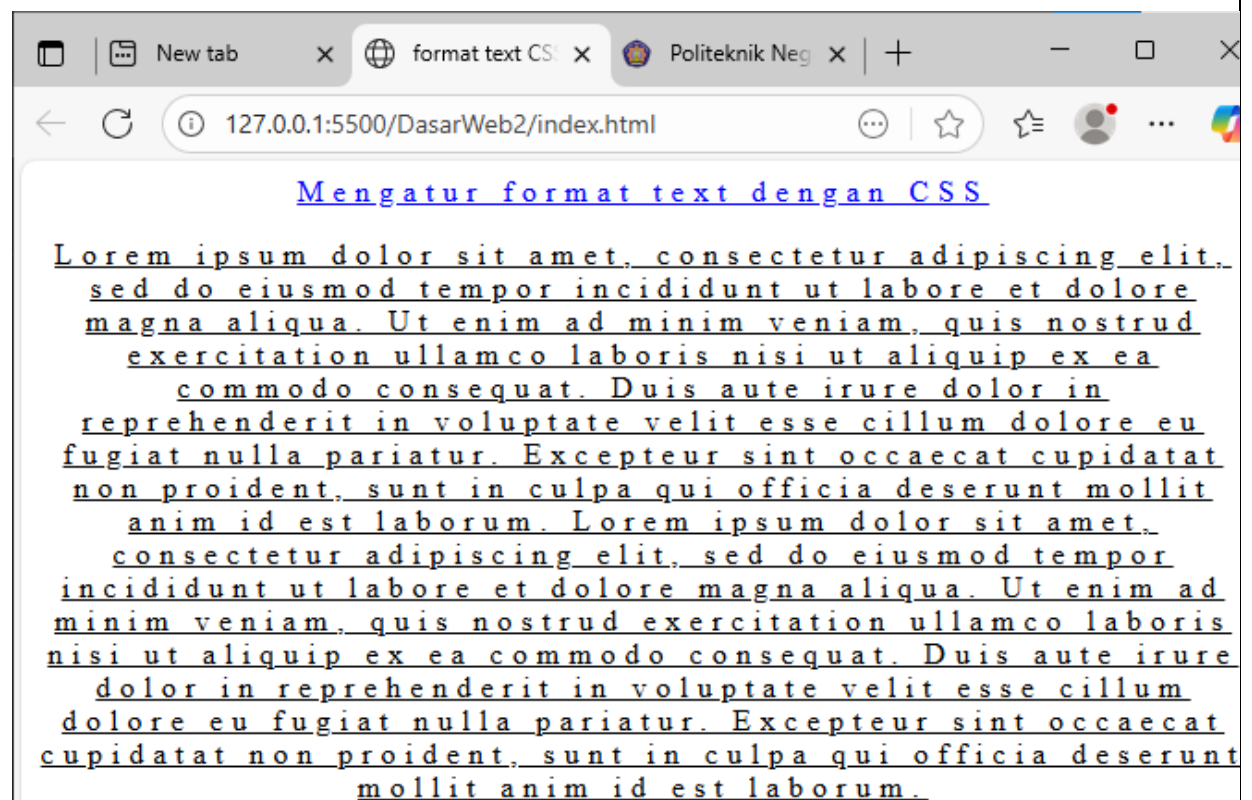


13

kode program(soal no 15)

```
linkCSS.html index.html # styleText.css X
DasarWeb2 > # styleText.css > p
1  .tulisan_warna{
2      color: blue;
3  }
4  p{
5      text-align:center;
6      text-decoration:underline;
7  }
```

```
DasarWeb2 > # styleText.css > ...
1  .tulisan_warna{
2    color: blue;
3  }
4  p{
5    text-align:center;
6    text-decoration:underline;
7  }
8  p{
9    letter-spacing:5px;
10 }
11
```

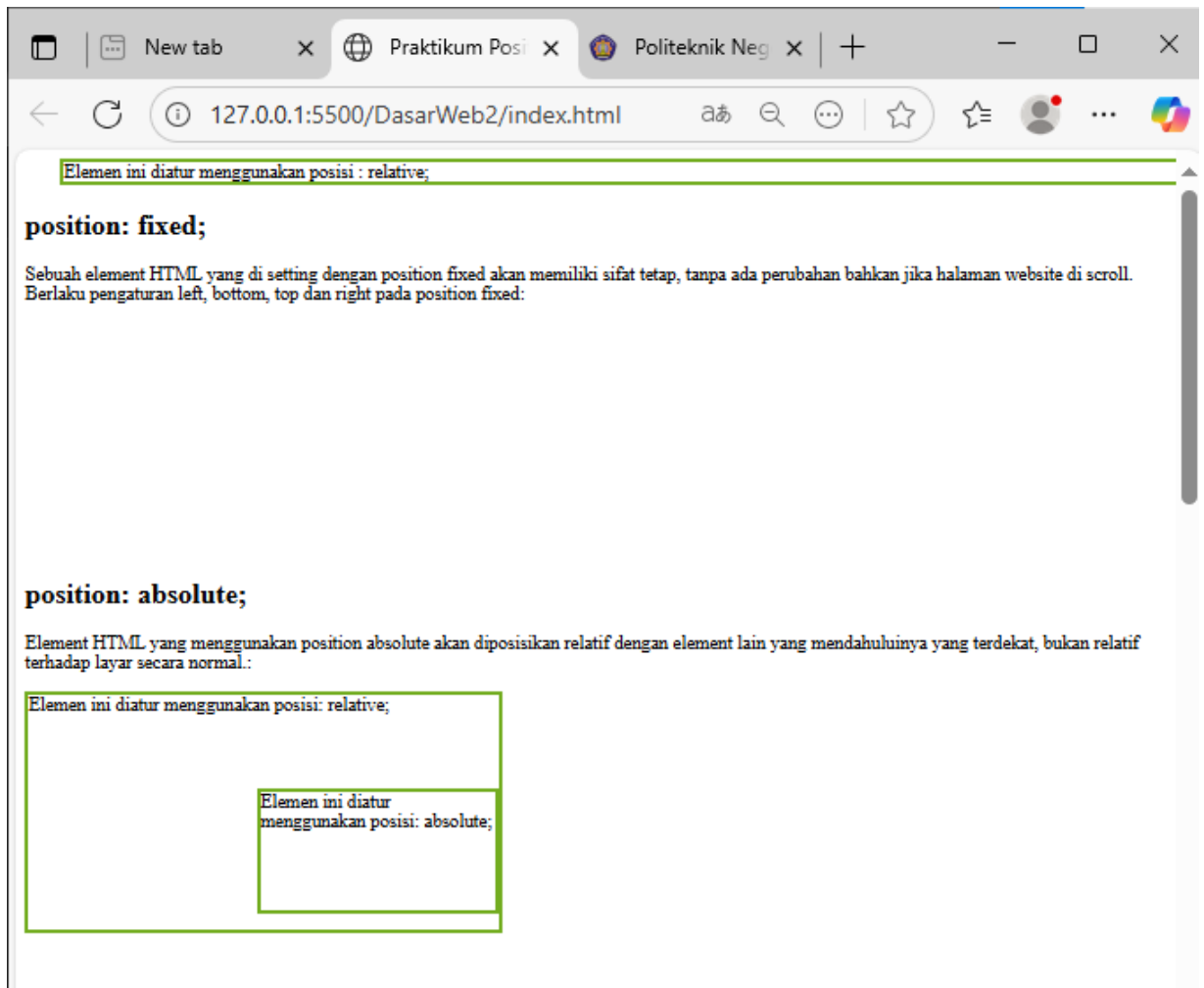


## Praktikum Bagian 10: Mengenal Position CSS

9

(soal no 17)

Menunjukkan perbedaan penggunaan posisi pada elemen HTML, di mana relative membuat elemen tetap di tempat semula tapi bisa digeser sedikit, fixed membuat elemen selalu menempel di layar meskipun halaman di-scroll, absolute membuat elemen diposisikan relatif terhadap elemen induk terdekat yang sudah diatur posisinya, sedangkan sticky awalnya mengikuti alur normal namun akan menempel di bagian atas layar ketika discroll sampai titik tertentu.



### position: sticky;

Cobalah untuk **scroll** untuk mengetahui bagaimana posisi sticky bekerja.

Catatan: IE/Edge 15 dan versi yang lebih lama tidak mendukung sticky.

**Hola Aku adalah sticky!**

Dalam contoh ini, elemen sticky menempel pada posisi paling atas halaman (top:0), ketika mencapai posisi scrollnya.

Cobalah untuk melakukan scroll.. Lorem ipsum dolor sit amet, illum definitiones no quo, maluisse concludaturque et eum, altera fabulas ut quo. Atqui causae gloriatur ius te, id agam omnis evertitur eum. Affert laboramus repudiandae nec et. Inciderint efficiantur his ad. Eum no molestiae voluptatibus.

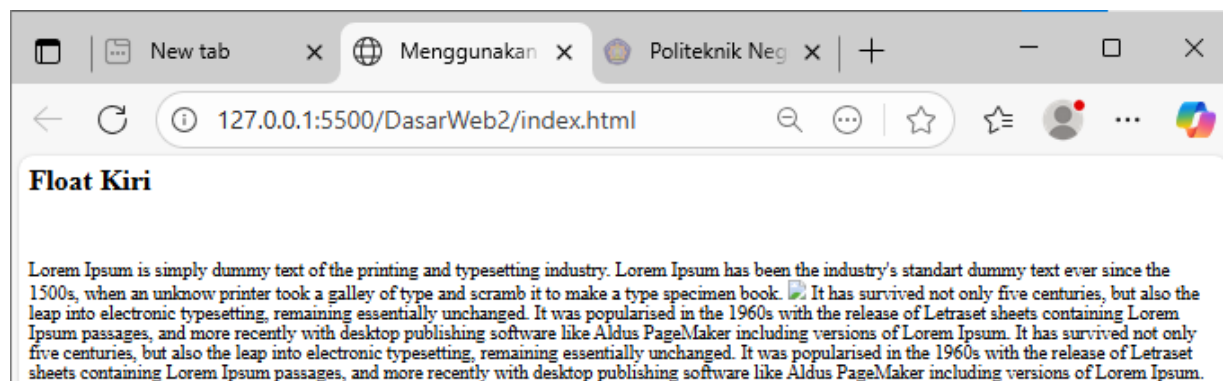
Cobalah untuk melakukan scroll.. Lorem ipsum dolor sit amet, illum definitiones no quo, maluisse concludaturque et eum, altera fabulas ut quo. Atqui causae gloriatur ius te, id agam omnis evertitur eum. Affert laboramus repudiandae nec et. Inciderint efficiantur his ad. Eum no molestiae voluptatibus.

Cobalah untuk melakukan scroll.. Lorem ipsum dolor sit amet, illum definitiones no quo, maluisse concludaturque et eum, altera fabulas ut quo. Atqui causae gloriatur ius te, id agam omnis evertitur eum. Affert laboramus repudiandae nec et. Inciderint efficiantur his ad. Eum no molestiae voluptatibus.

Elemen ini diatur menggunakan posisi: fixed;

## Praktikum Bagian 11: Menggunakan Float

Dari kode layout sederhana dengan header, navigasi, main, dan footer, saya amati tampilan web jadi lebih terstruktur dan rapi karena tiap bagian halaman dipisahkan dalam div yang berbeda, sehingga mudah dibedakan mana area kepala, menu, isi utama, dan kaki halaman. (soal no 18)



New tab x CSS Layouting x Politeknik Neg x

127.0.0.1:5500/DasarWeb2/index.html

Selamat Datang Di Website Kami

Daftar Link

[Link 1](#)  
[Link 2](#)  
[Link 3](#)  
[Link 4](#)

TEKNIK INFORMATIKA



Pada tahun 2010 berdasar kebutuhan masyarakat dan industri terkait Program Diploma IV bidang informatika maka Polinema mendirikan program studi baru Diploma IV (DIV) Teknik Informatika (TI). Pada awal berdirinya jumlah peserta didik Program D-IV TI hanya terdiri dari 46 Mahasiswa, namun pada tahun 2015 jumlah peserta didik Prodi D-IV TI telah mengalami peningkatan menjadi 545 Mahasiswa.

Perkembangan jumlah mahasiswa yang sangat baik di Program Studi D-III MI maupun D-IV TI mendorong pimpinan di Polinema untuk menyatukan kedua program studi tersebut dalam satu organisasi dari jurusan terpisah dari jurusan Teknik Elektro. Berdasarkan SK Direktur Nomor 53 dalam rangka meningkatkan mutu pengelolaan dan optimalisasi sumber daya dibentuklah Jurusan Teknologi Informasi (JTI) dengan prodi D-III MI dan D-IV TI.

Copyright 2020. Desain dan Pemrograman Web JTI Polinema

New tab x CSS Layouting x Politeknik Neg x


127.0.0.1:5500/DasarWeb2/index.html

Selamat Datang Di Website Kami

Daftar Link

[Link 1](#)  
[Link 2](#)  
[Link 3](#)  
[Link 4](#)

TEKNIK INFORMATIKA



Pada tahun 2010 berdasar kebutuhan masyarakat dan industri terkait Program Diploma IV bidang informatika maka Polinema mendirikan program studi baru Diploma IV (DIV) Teknik Informatika (TI). Pada awal berdirinya jumlah peserta didik Program D-IV TI hanya terdiri dari 46 Mahasiswa, namun pada tahun 2015 jumlah peserta didik Prodi D-IV TI telah mengalami peningkatan menjadi 545 Mahasiswa.

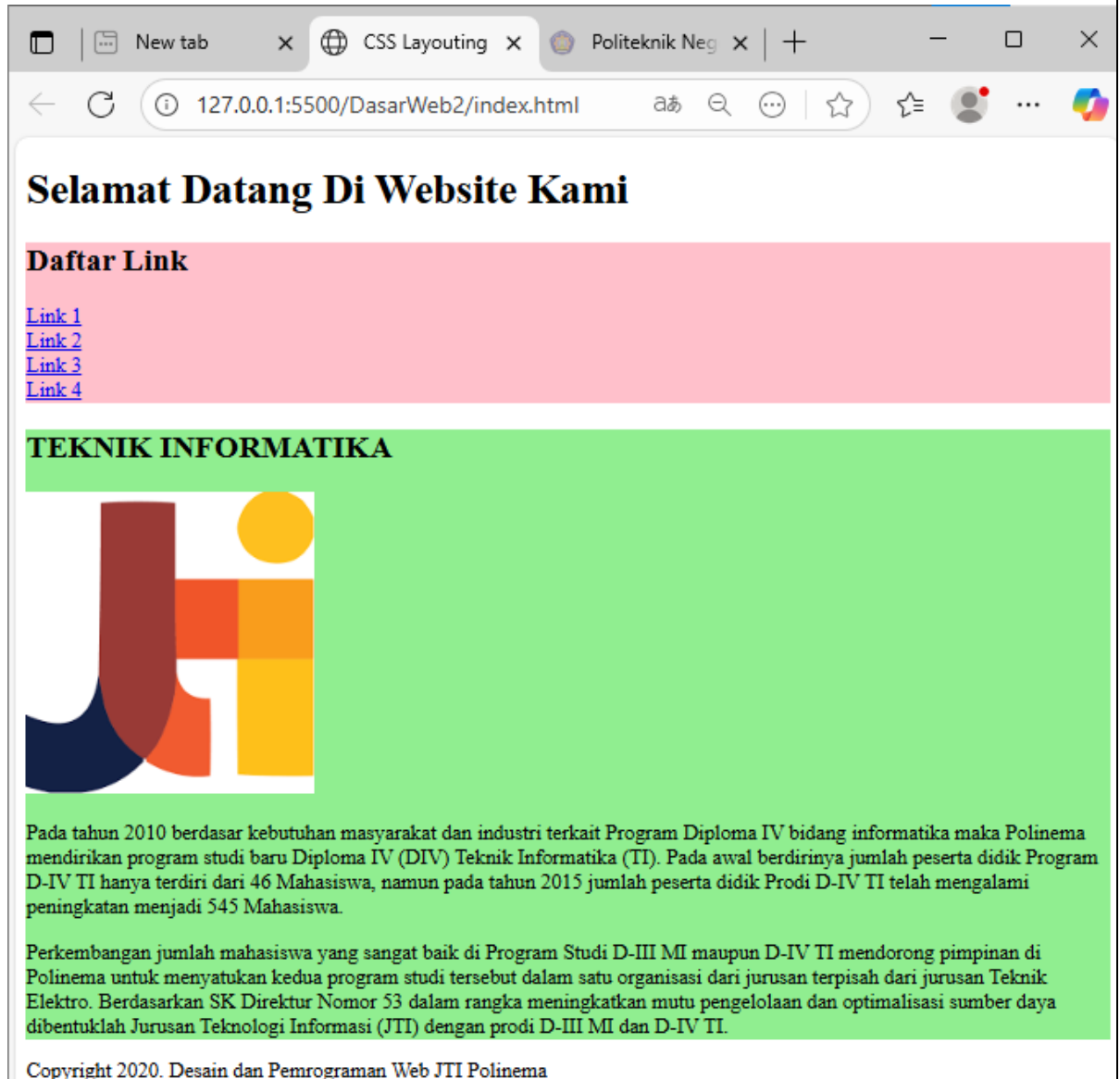
Perkembangan jumlah mahasiswa yang sangat baik di Program Studi D-III MI maupun D-IV TI mendorong pimpinan di Polinema untuk menyatukan kedua program studi tersebut dalam satu organisasi dari jurusan terpisah dari jurusan Teknik Elektro. Berdasarkan SK Direktur Nomor 53 dalam rangka meningkatkan mutu pengelolaan dan optimalisasi sumber daya dibentuklah Jurusan Teknologi Informasi (JTI) dengan prodi D-III MI dan D-IV TI.

Copyright 2020. Desain dan Pemrograman Web JTI Polinema

Saya amati kalau elemen link <a> tampil berjejer dalam satu baris karena sifatnya inline, sedangkan elemen <div> otomatis pindah ke baris baru karena sifatnya block, jadi hasil keduanya jelas berbeda.

(soal 19)

Setelah ditambah style background kuning pada navigasi dan hijau pada main, saya amati area menu dan isi utama lebih jelas dibedakan dengan warna, sehingga tampilan web lebih mudah dibaca. (soal 20)



### Praktikum Bagian – 13. Display

3

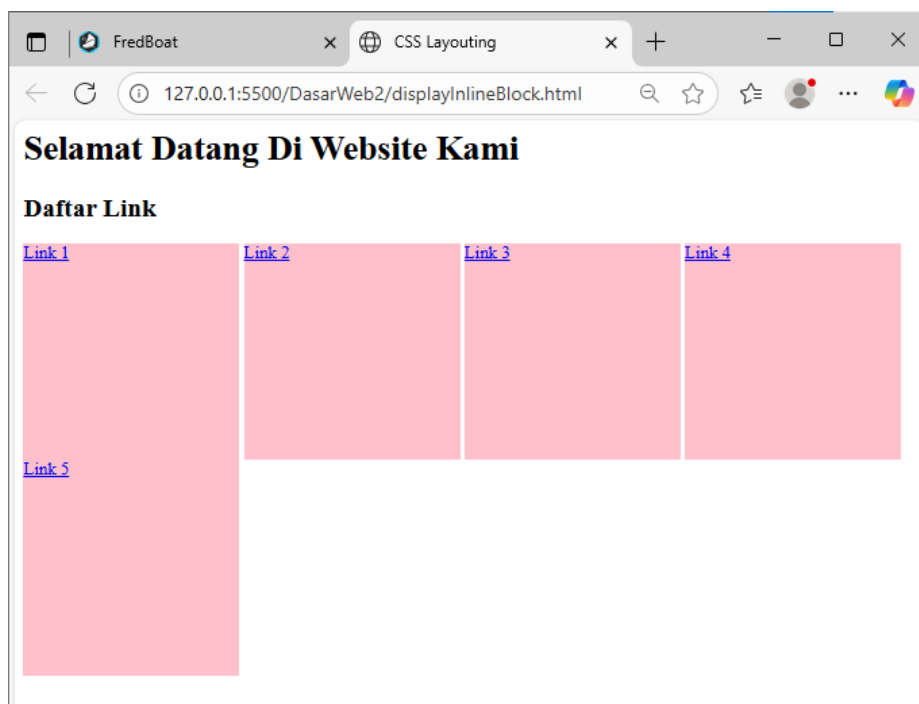
(Soal 21)

Menghapus barisnya, jadi semua link terkumpul dalam satu baris



### Praktikum Bagian – 14. Display *inline-block*

Setelah elemen link diubah menjadi display inline-block dengan lebar dan tinggi tertentu, saya amati link tetap sejajar dalam satu baris, tetapi sekarang bisa diatur ukuran kotaknya sehingga terlihat seperti tombol menu. (Soal 22)





## Praktikum Bagian – 15. Display block

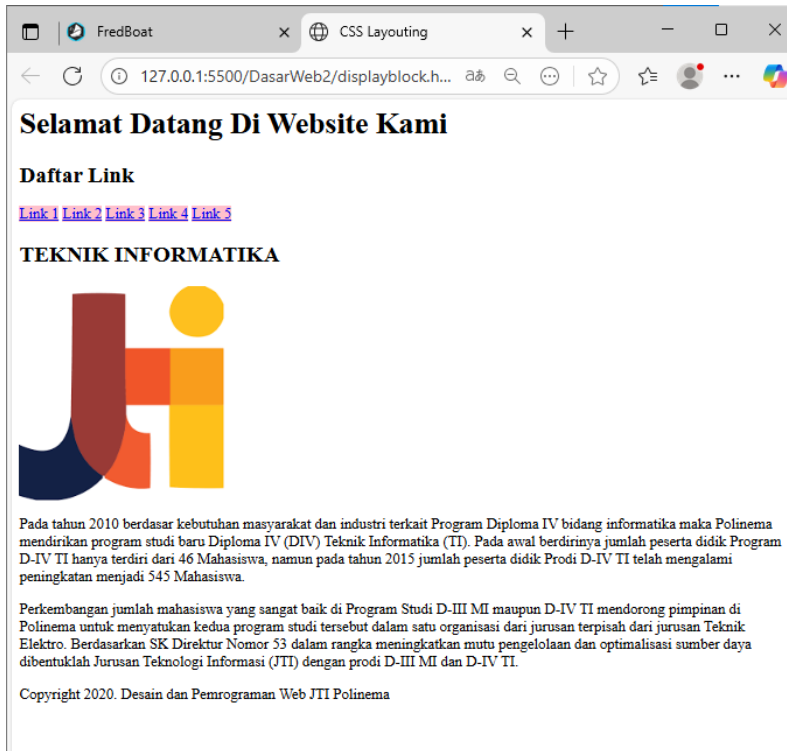
### (Soal 23)

Elemen block otomatis memenuhi lebar layar. Saat resize, elemen menyesuaikan. Semua teks dengan background hijau menutupi 1 baris penuh.



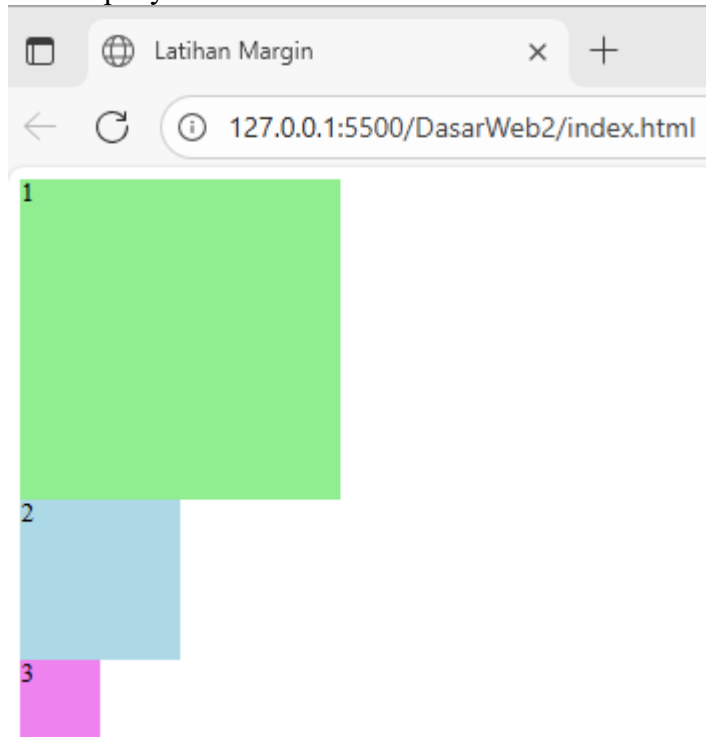
### (Soal 24)

Setelah style main ditambahkan, saya amati area konten utama lebih jelas tampilannya karena ada perbedaan warna, menegaskan sifat elemen block yang menempati seluruh lebar halaman.



**(Soal 25)**

Dari kode box model sederhana, saya amati setiap elemen ditampilkan dalam bentuk kotak dengan bagian isi (content) yang bisa diberi warna atau teks, menunjukkan bahwa semua elemen HTML punya “box” bawaan.

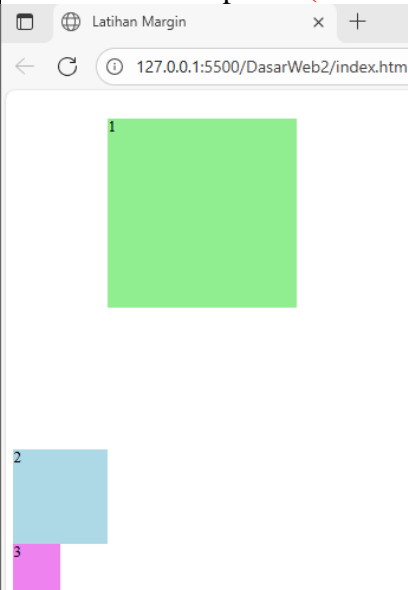


Selanjutnya adalah memberi *margin*, dimana *margin* adalah *area transparan* yang ada di sekitar kotak. Tambahkan *style* pada *style.css* **margin-top** sebesar 100px pada **class.satu** simpan dan kemudian jalankan pada web browser. Capture dan amati hasilnya

**(Soal 26)**



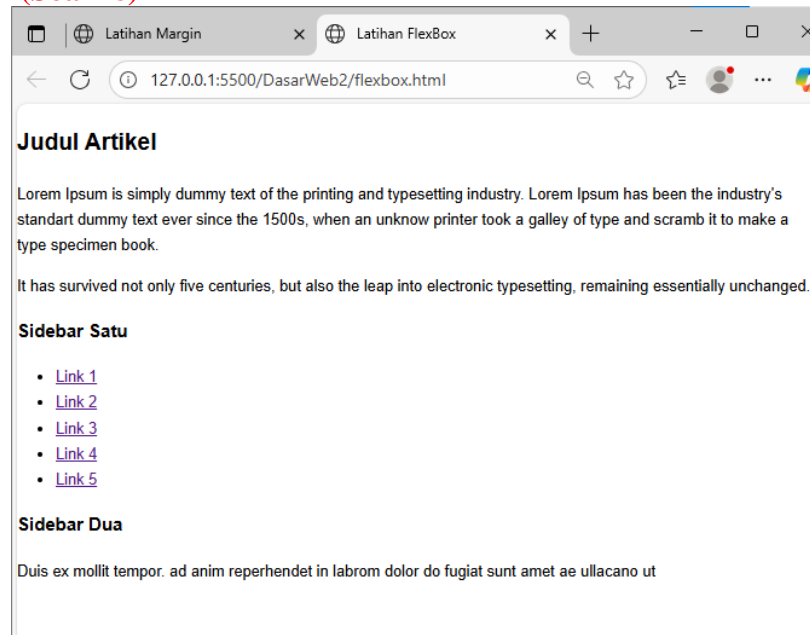
Ketika saya menambahkan margin lain seperti margin-left, margin-right, dan margin-bottom, kotak jadi punya ruang kosong di semua sisi, bukan hanya atas, sehingga terbukti margin dipakai untuk mengatur jarak luar elemen dari setiap sisi agar posisinya lebih rapi dan tidak berhimpitan. **(Soal 27)**



## Praktikum Bagian – 17. *Flex Box*

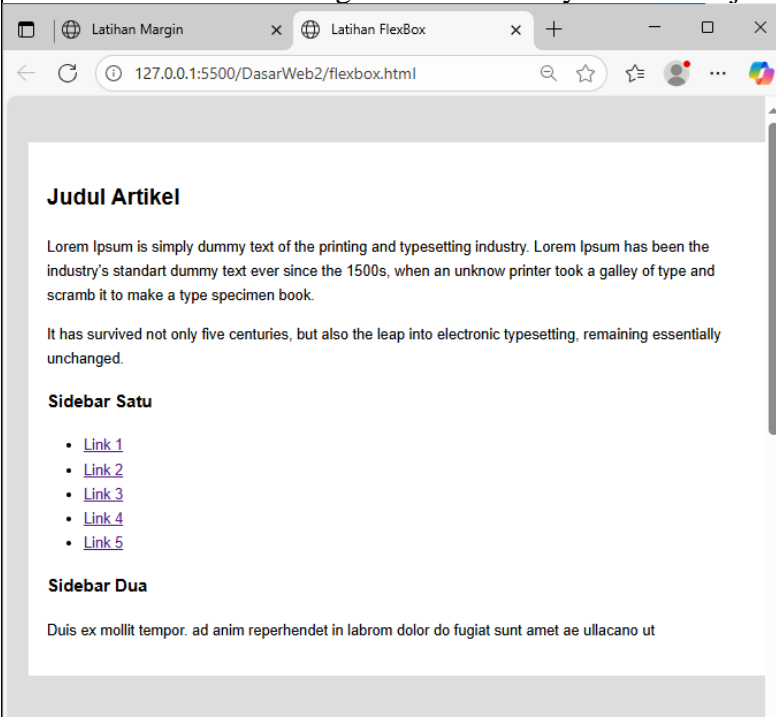
Setelah kode flexbox dijalankan pertama kali, saya lihat item di dalam container masih tersusun ke bawah secara default, jadi kesannya masih seperti elemen biasa, dan ini membuktikan bahwa elemen belum berperilaku flex sebelum diberi aturan khusus.

**(Soal 28)**



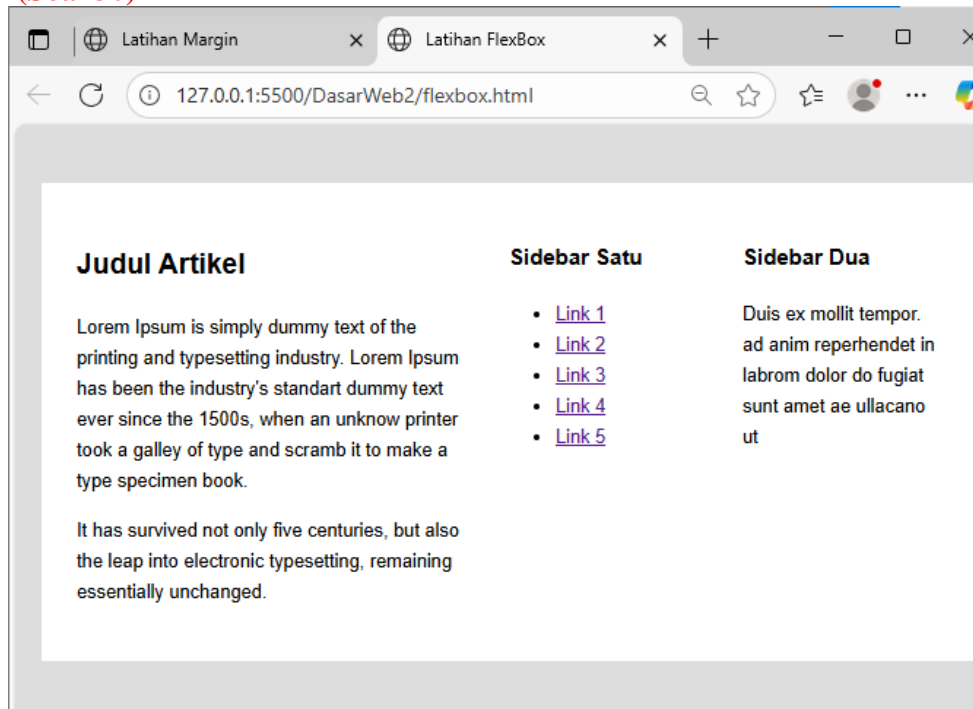
**(Soal 29)**

Setelah saya tambahkan style ke .container-satu, tampilannya lebih teratur karena ada pengaturan tambahan pada kontainer, misalnya jarak antar item lebih rapi, yang menunjukkan bahwa kontainer bisa mengatur isi di dalamnya sebelum dijadikan flex.



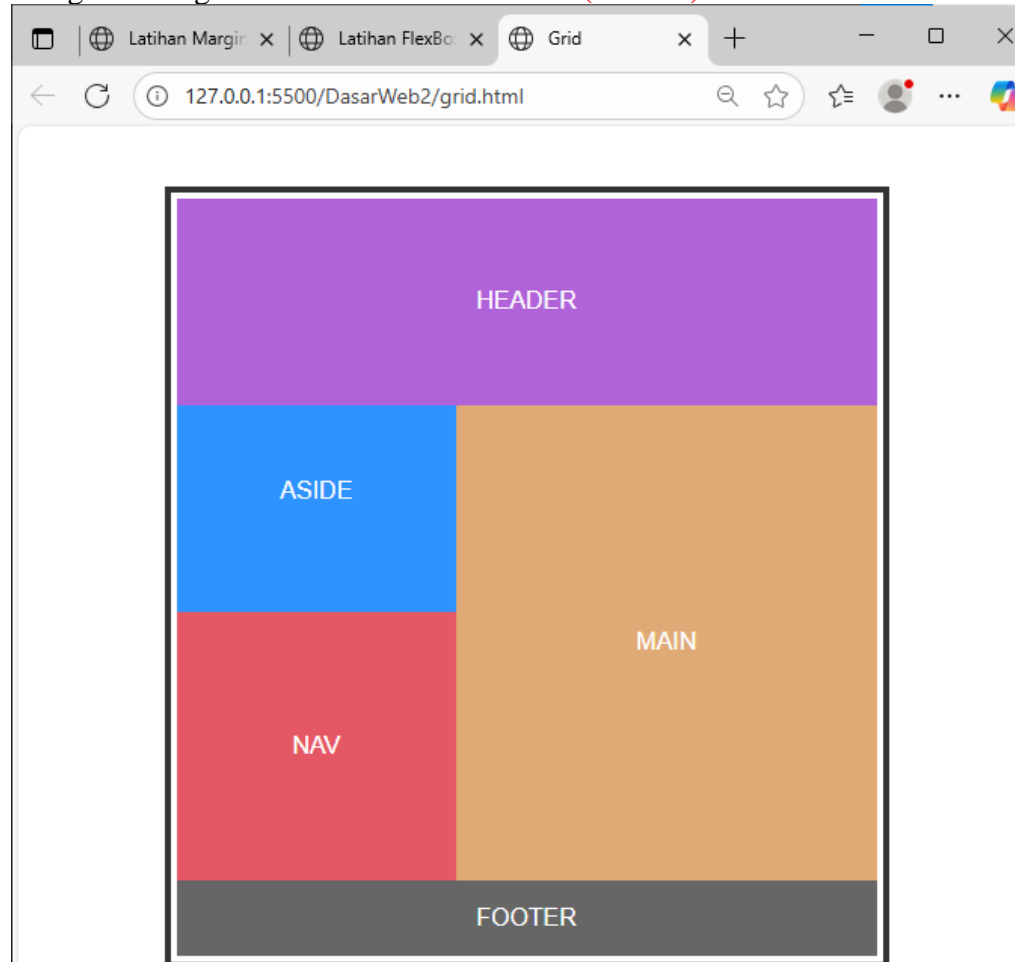
Ketika saya tambahkan `display:flex;` pada .container-satu, semua item langsung tersusun rapi dalam satu baris secara horizontal, membuktikan bahwa flexbox membuat tata letak lebih fleksibel dan mudah menata item dalam satu garis tanpa harus menggunakan float atau posisi manual.

**(Soal 30)**



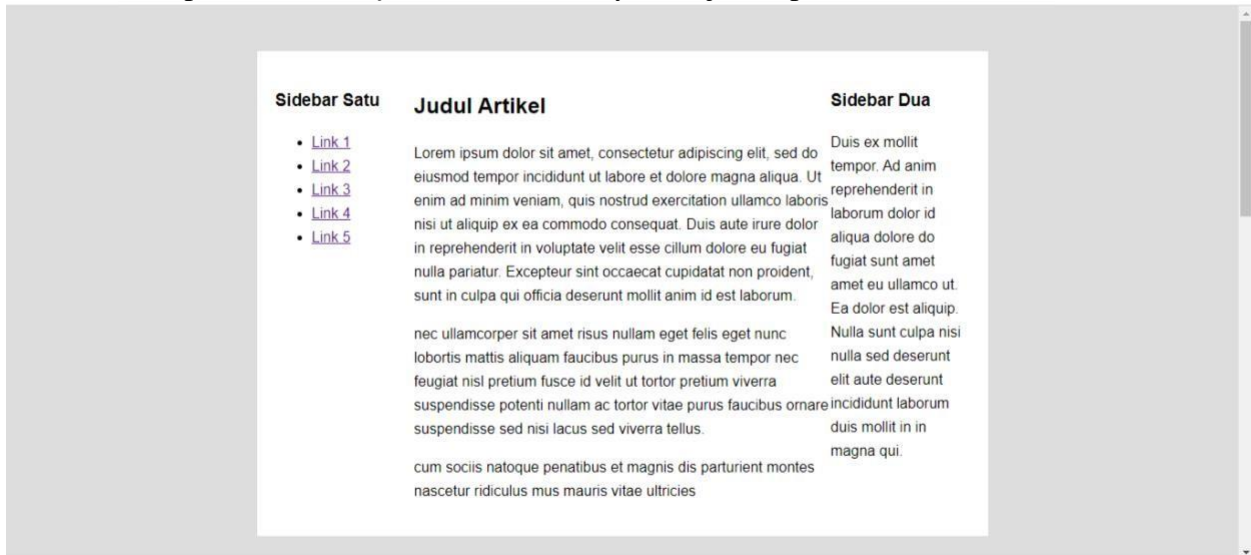
## Praktikum Bagian – 18. CSS Grid

Setelah kode grid dijalankan, elemen-elemen langsung terbagi menjadi kotak-kotak dalam baris dan kolom seperti tabel, dan saya amati bahwa CSS Grid memudahkan membagi halaman menjadi layout dua dimensi (baris dan kolom) yang rapi tanpa harus mengatur margin atau float secara manual. **(Soal 31)**



## Tugas Jobsheet 2:

1. Buatlah tampilan dari hasil *flex-box* sebelumnya menjadi seperti ini



2. Tambahkan gambar dan aplikasikan permainan warna untuk backgroundnya supaya menjadi lebih menarik

Jadi saya mengubah tampilan menjadi lebih menarik dan estetik. Perubahan utama dilakukan pada file `styleFlexTugas.css` dengan memodifikasi warna latar belakang (`body` dan `.container-satu`) menjadi palet warna pink. Selain itu, ditambahkan properti `border-radius` untuk membuat sudut container menjadi membulat dan `box-shadow` untuk memberikan efek bayangan yang halus, sehingga tampilan tidak lagi kaku. Warna-warna teks judul (`h2`, `h3`) dan tautan (`a`) juga disesuaikan agar senada.

