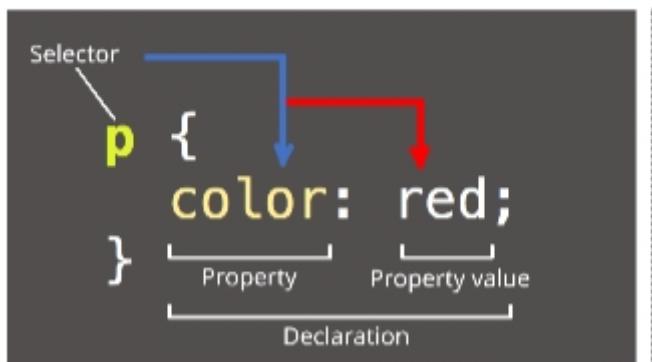


CATATAN CSS

Anatomi CSS

Anatomi CSS



Selector

Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya. Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.

Property

Property dalam CSS adalah karakteristik atau gaya yang diterapkan pada elemen HTML, seperti warna, ukuran font, atau margin. Properti bekerja dengan selector untuk mengatur tata letak dan penampilan elemen

Property value

Dalam CSS property value (nilai sifat) merujuk pada nilai konkret yang diberikan kepada suatu properti.

Percobaan I

Penjelasan

Dengan menggunakan CSS elemen html yang kita buat, bisa kita perantik seperti pada contoh dibawah mengganti warna textnya dengan warna merah.

Kode Program

```
<!DOCTYPE html>
<html>
  <head>
    <title>css</title>
  </head>
  <body>
    <p>SELAMAT DATANG di CSS</p>
    <p>APA YANG BISA SAYA BANTU</p>
  </body>
</html>
```

```
p {
color: red;
}
```

Hasil

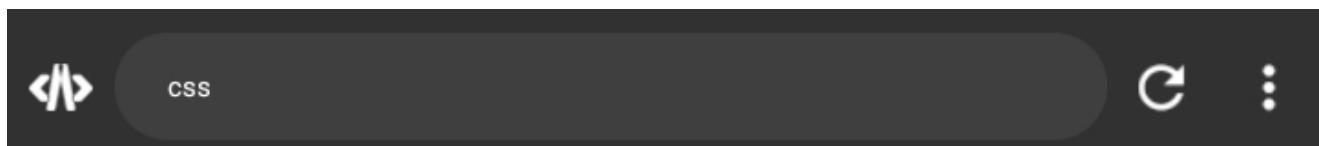
Before



SELAMAT DATANG di CSS

APA YANG BISA SAYA BANTU

After



SELAMAT DATANG di CSS

APA YANG BISA SAYA BANTU

Kesimpulan

Kode HTML menciptakan halaman web sederhana dengan dua elemen paragraf `<p>` yang berisi teks "WELCOME CSS!!!". Kode CSS yang terhubung dengan halaman tersebut menetapkan warna teks aqua untuk semua elemen `<p>`. Jadi, hasilnya akan menampilkan teks "WELCOME CSS!!!" dengan warna aqua di halaman web.

Percobaan II

Kode Program

```
Button {  
    Width: 150px;  
    Height: 50px;  
    Font-size: 20px;  
    Margin-top: 50px;  
    Color: white;  
}
```

Font-size

Penjelasan

Font-size adalah Property yang menentukan ukuran text pada property html,yang terdapat pada dalam kotak, yang bertuliskan klik aku!

```
Button {  
    Font-size: 20px  
}
```

Hasil gambar

Before



welcome css

klik aku !

After



welcome css

klik aku !

Kesimpulan

Font size berfungsi untuk mengatur ukuran Text pada Text dalam button

Margin-Top

Penjelasan

margin top adalah sebuah Property untuk memberikan jarak pada suatu elemen terhadap elemen yang ada di atasnya.

```
Button {  
margin-top:50px;  
}
```

Hasil Gambar

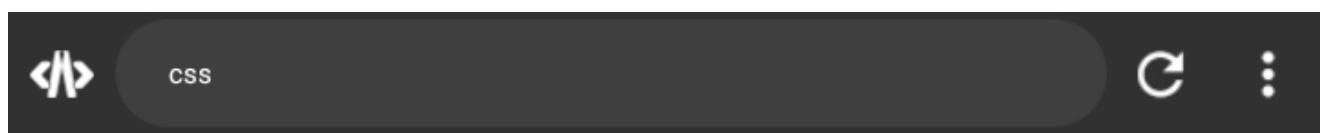
Before



welcome css

klik aku !

After



welcome css

klik aku !

Kesimpulan

Margin-top digunakan untuk memberikan jarak antara elemen dengan elemen di atasnya

Color

Penjelasan

Color adalah warna yang akan diberikan pada text yang dipanggil dengan Selector

```
Button {  
Color:green;  
}
```

Hasil Gambar

Before



welcome css

klik aku !

After



welcome css

klik aku !

Kesimpulan

Color di gunakan untuk memberikan gaya pada textnya.

Cara Pemanggilan CSS

Terdapat 3 cara untuk pemanggilan CSS yaitu secara internal, eksternal dan inline

Internal

Penjelasan

Internal adalah pemanggilan css secara satu file/ berada didalam file html yg ingin kita berikan style.

Adapun tempat penulisan CSS secara internal, kita dapat tuliskan di dalam <head>

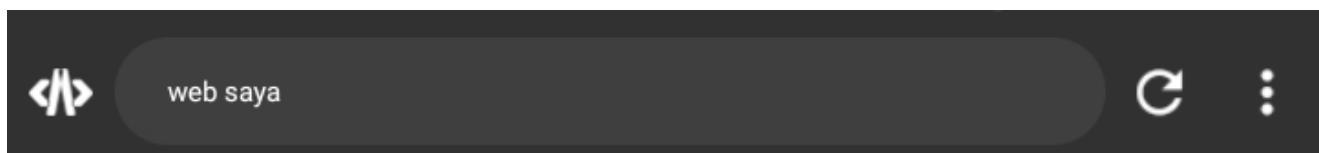
Kode Program

```
<!doctype html>
<html>
  <head>
    <title>web saya</title>

    <style>
      p {
        color: bluesky;
      }
    </style>

  </head>
  <body>
    <p>Selamat datang</p>
  </body>
</html>
```

Hasil



Selamat datang

Kesimpulan

Pemanggilan CSS secara internal dilakukan dengan hanya 1 file saja, yang dimana kode CSS kita taro di dalam `<style>` serta posisi `<style>` yang berada dalam `<head>`

Inline

Inline adalah pemanggilan css dengan cara didalam tag yang ingin kita berikan style.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>web saya</title>
  </head>
  <body>
    <p style="color:red;">Hallo world</p>
  </body>
</html>
```

Hasil



Haloo world

Kesimpulan

Pemanggilan secara inline dengan cara di dalam tag htmlnya.

External

External adalah pemanggilan dengan cara membuat file baru dengan format filenya `nama_file.css` serta memberikan `<link rel="stylesheet" href="nama_file.css">` dalam Tag head, Berbeda dengan internal dan inline yang cukup dengan 1 file saja.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>web saya</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <p>Hallo RPL</p>
  </body>
</html>
```

```
p {
Color: black;
}
```

Hasil



Hallo RPL

Kesimpulan

Pemanggilan CSS secara External dibutukan untuk membuat file baru dengan memberikan link dalam file htmlnya.

Selector Umum CSS

Elemen Selector

Penjelasan

Elemen selector adalah jenis selector dengan cara pemanggilan langsung menyebut tag htmlnya. Contoh jika menggunakan tag `<p>` itu akan memilih semua elemen paragraf pada html.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>web saya</title>
    <style>

      P {
        Color : blue;
      }

    </style>
  </head>
  <body>
    <p>Elemen Selector</p>
  </body>
</html>
```

Hasil



Elemen Selector

Kesimpulan

Dengan menggunakan elemen selector, Anda dapat menerapkan gaya ke semua elemen yang memiliki tag HTML yang sama, seperti `<p>` untuk paragraf atau `<h1>` untuk heading level 1.

Class Selector

Penjelasan

Digunakan untuk memberikan gaya pada elemen HTML yang memiliki kelas tertentu. Kelas ini didefinisikan di dalam tag HTML dengan atribut class. Satu elemen dapat memiliki banyak kelas, dan kelas dapat digunakan oleh beberapa elemen cara Pemanggilan Class dengan cara memberikan titik sebelum nama classnya.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>web saya</title>
    <style>
      .text {
        color: red;
      }
    </style>
  </head>
  <body>
    <p class="text">Class Selector</p>
  </body>
</html>
```

Hasil



Class Selector

Kesimpulan

Class Selector dalam CSS memungkinkan Anda untuk menargetkan elemen berdasarkan nilai atribut `class` yang telah ditetapkan untuk elemen tersebut.

Id Selector

Penjelasan

ID dalam CSS merujuk pada identifikasi unik dari sebuah elemen HTML. Setiap elemen dapat diberi ID yang berbeda, dan ID tersebut harus unik di dalam satu halaman web. ID digunakan untuk memberikan gaya atau memanipulasi elemen secara spesifik, pemanggilan id memberikan tanda # sebelum nama id nya dalam css.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>web saya</title>
    <style>
      #text {
        color: red;
      }
    </style>
  </head>
  <body>
    <p id="text">Id Selector</p>
  </body>
</html>
```

Hasil



web saya



Id Selector

Kesimpulan

Id Selector dalam CSS digunakan untuk menargetkan elemen berdasarkan nilai atribut id yang unik untuk setiap elemen.

Text

Text-align

Penjelasan

text-align dalam CSS digunakan untuk mengatur penataan horizontal teks dalam sebuah elemen HTML. Beberapa property value seperti :

- right Text dipojok kanan
- left Text dipojok kiri
- center Text di tengah
- justify Text rata kiri dan kanan

Kode Program

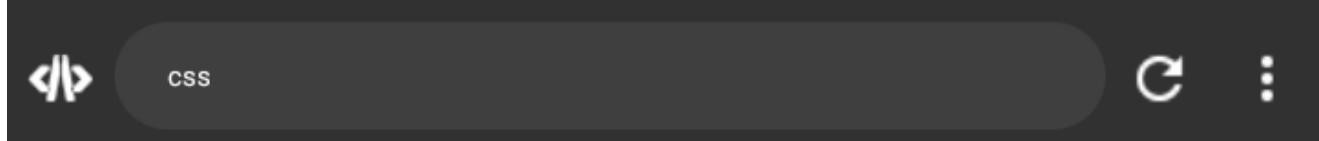
```
<!doctype html>
<html>
  <head>
    <title>css</title>
    <style>
      p {
        text-align: center;
      }
    </style>
```

```
</head>
<body>
  <p>TEXT ALIGN</p>

</body>
</html>
```

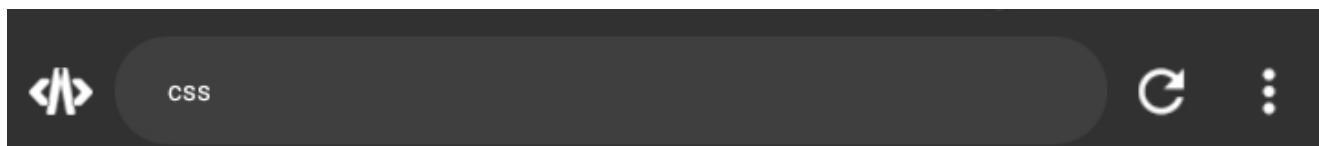
Hasil

Before



TEXT ALIGN

After



TEXT ALIGN

Kesimpulan

Jika ingin mengatur tata letak sebuah Text dalam website, kita bisa gunakan Property text-align

Text-decoration

Penjelasan

Text decoration digunakan untuk memberi dekorasi pada sebuah Text, seperti memberi garis bawah,atas,tengah serta hilangkan dekorasi(none).

Beberapa contoh property value nya :

- `underline` : Menambahkan garis bawah pada teks.
- `overline` : Menambahkan garis di atas teks.
- `line-through` : Menambahkan garis tengah melintasi teks.
- `none` : Menghapus semua dekorasi teks.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        text-decoration:line-through ;
      }
    </style>
  </head>
  <body>
    <p>TEXT DECORATION</p>

  </body>
</html>
```

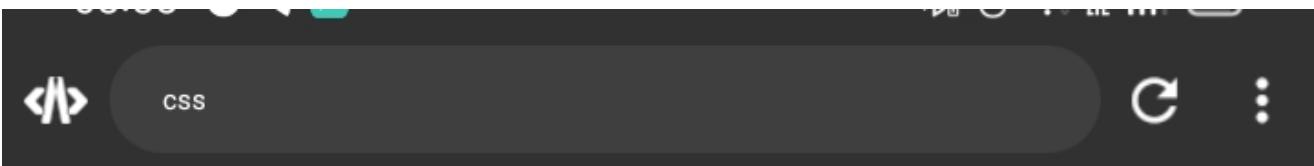
Hasil

Before



TEXT DECORATION

after



TEXT DECORATION

Kesimpulan

Text-decoration digunakan untuk memberikan dekorasi pada Text dimana dekorasi tersebut berupa garis bawah,tengah,dan atas.

Text-transfrom

Penjelasan

Text-transfrom digunakan untuk mengubah gaya kapitalisasi Text pada suatu elemen.

Beberapa contoh Properti value nya :

- `capitalize` : membuat karakter pertama pada setiap Text menjadi kapital
- `uppercase` : mengubah semua karakter menjadi huruf kapital
- `lowercase` : mengubah semua karakter menjadi huruf kecil
- `none` : tidak ada perubahan

Kode Program

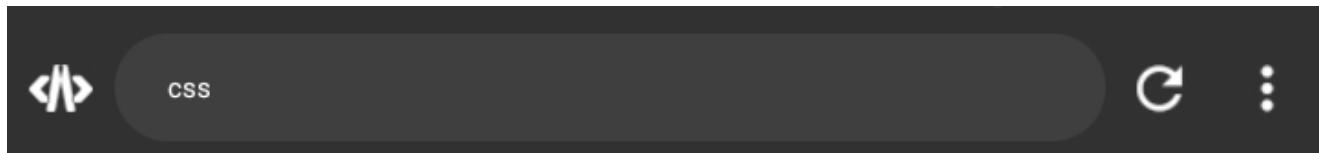
```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        text-transform: lowercase ;
      }
    </style>
  </head>
  <body>

    <p>TEXT TRANSFORM</p>
```

```
</body>  
</html>
```

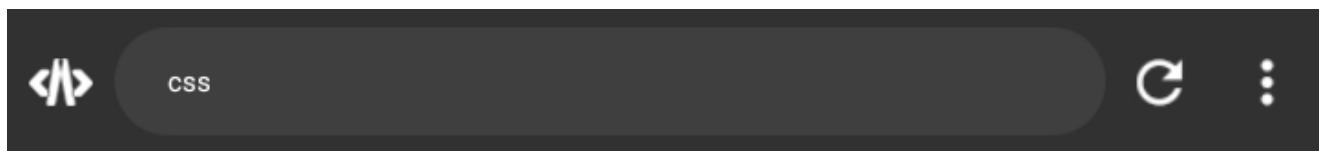
Hasil

Before



TEXT TRANSFORM

After



text transform

Kesimpulan

Digunakan untuk mengubah Text, apakah menjadi huruf kapital atau huruf kecil

Text-indent

Penjelasan

Text indent digunakan untuk menentukan jumlah ruang atau jarak disetiap awal baris sebuah blok Text.

Beberapa contoh property value nya :

- `20px` : akan memberikan jarak pada baris pertama Text sebanyak 20px dari kiri elemen.
- `0` : tidak ada perubahan pada suatu blok Text.

Kode Program

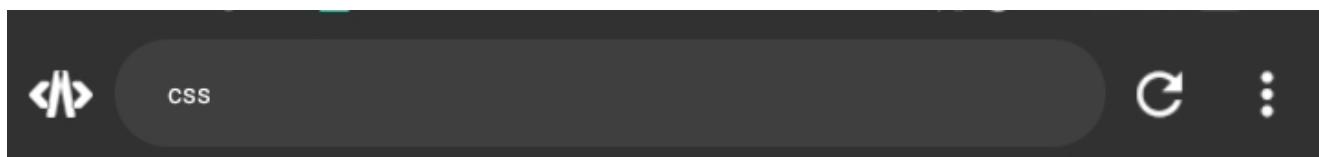
```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        text-indent: 20px;
      }
    </style>
  </head>
  <body>

    <p>Text indent digunakan untuk menentukan jumlah ruang atau jarak
    disetiap awal baris sebuah blok Text.</p>

  </body>
</html>
```

Hasil

Before



Text indent digunakan untuk menentukan jumlah ruang atau jarak disetiap awal baris sebuah blok Text.

After



Text indent digunakan untuk menentukan jumlah ruang atau jarak disetiap awal baris sebuah blok Text.

Kesimpulan

Text indent biasa digunakan ketika kita ingin membuat paragraf yang dimana baris pertamanya memiliki jarak tertentu dengan dari kirinya

Letter-spacing

Penjelasan

Letter spacing berfungsi untuk mengatur jarak setiap karakter(huruf) dalam Text.

Beberapa contoh property value nya :

- 2px : meningkatkan jarak antar karakter
- -1px : mengurangi jarak antar karakter
- normal : tidak ada perubahan jarak.

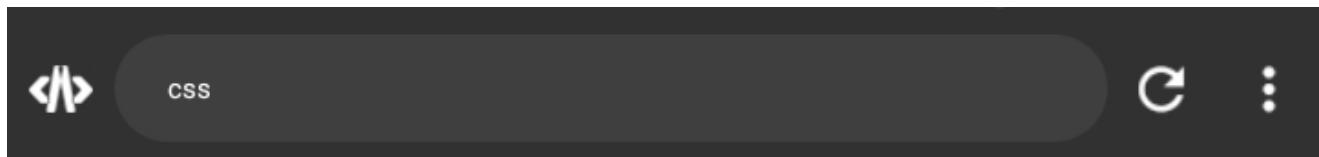
Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        letter-spacing:5px;
      }
    </style>
  </head>
  <body>
    <p>LETTER SPACING</p>

  </body>
</html>
```

Hasil

Before



LETTER SPACING

After



LETTER SPACING

Kesimpulan

Letter spacing digunakan ketika ingin memberi jarak antara karakter.

Line-height

Penjelasan

Line-height Property yang digunakan untuk mengatur tinggi baris dalam sebuah elemen, dimana Property value nya dapat berupa piksel, persentase atau nilai tanpa satuan untuk mengatur tinggi barisnya.

Kode Program

```
<html>
<head>
```

```
<title>css</title>
<style>
p {
  line-height: 40px;
}
</style>
</head>
<body>


Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya.Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.</p>


</body>
</html>
```

Hasil

Before



Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya.Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.

After



Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya.Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.

Kesimpulan

Line-height dalam CSS adalah properti yang mengontrol tinggi baris dalam sebuah elemen, mempengaruhi jarak vertikal antara garis teks.

Word-spacing

Penjelasan

Word-spacing digunakan untuk mengatur jarak antar kata dalam sebuah blok Text. Dimana jika ingin menambahkan jarak berikan nilai positif ataupun sebaliknya jika ingin mengurangi jarak antar kalimat, cukup beri Property value dengan nilai negatif.

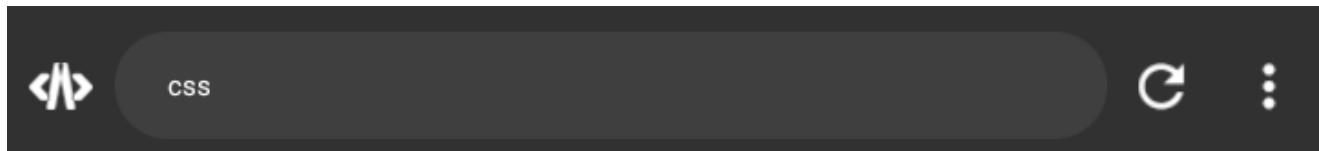
Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        word-spacing: 20px;
      }
    </style>
  </head>
  <body>
    <p>Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya.Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.</p>
```

```
</body>  
</html>
```

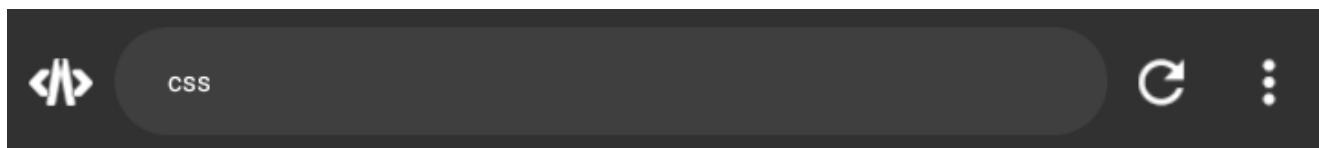
Hasil

Before



Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya. Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.

After



Selector CSS di gunakan untuk memilih elemen HTML yang akan diberi gaya. Dengan menggunakan selector, Anda dapat menargetkan satu atau lebih elemen HTML untuk menerapkan properti CSS.

Kesimpulan

`word-spacing` dalam CSS adalah properti yang memungkinkan Anda mengatur jarak antara kata-kata dalam teks, dengan nilai positif menambahkan spasi dan nilai negatif mengurangi spasi. Properti ini membantu mengontrol tata letak teks.

Background

Background-image

Penjelasan

Background-image Property CSS yang digunakan untuk menetapkan gambar sebagai latar belakang elemen html.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>css</title>
    <style>
      p {
        background-image: url('2.jpg');
      }
    </style>
  </head>
  <body>
    <p>MATERI BACKGROUND</p>
  </body>
</html>
```

Hasil



css



Kesimpulan

Background image dalam CSS adalah properti yang memungkinkan Anda menetapkan gambar sebagai latar belakang elemen HTML.

Background-size

Penjelasan

Background-size Property CSS yang digunakan untuk mengatur ukuran sebuah gambar yang dijadikan background.

Beberapa contoh property value nya :

- `length` : Menentukan ukuran dalam satuan piksel, contoh: `background-size: 200px 150px`.
- `percentage` : Menentukan ukuran dalam persentase, contoh: `background-size: 50% 75%`.
- `cover` : Memastikan latar belakang mencakup seluruh elemen, tetapi mungkin memotong bagian gambar.
- `contain` : Memastikan seluruh gambar latar belakang ditampilkan di dalam elemen, mungkin dengan spasi kosong.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      body {
        background-image: url('2.jpg');
        background-size: 200px 400px;
        background-repeat: no-repeat;
      }
    </style>
  </head>
  <body>
    <p>Materi background</p>
  </body>
</html>
```

Hasil



css



Materi background



Kesimpulan

Background size property css yang digunakan untuk mengatur ukuran background gambar yang ada pada elemen html.

Background-repeat

Penjelasan

background-repeat untuk mengontrol cara gambar latar (background image) diulang di dalam elemen HTML. Properti ini dapat memiliki beberapa nilai, termasuk:

- `repeat` akan mengulang gambar secara horizontal dan vertikal
- `no-repeat` gambar tidak akan di ulang.
- `repeat-x`: Gambar hanya diulang secara horizontal.
- `repeat-y`: Gambar hanya diulang secara vertikal.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      body {
        background-image: url('2.jpg');
        background-size: 200px 400px;
        background-repeat: no-repeat;
      }
    </style>
  </head>
  <body>
    <p>Materi background</p>
  </body>
</html>
```

Hasil

Before

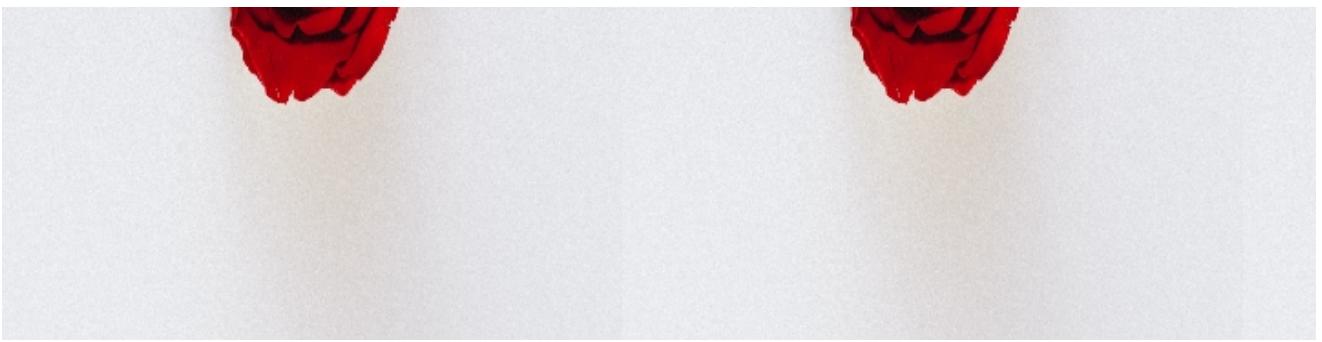


css

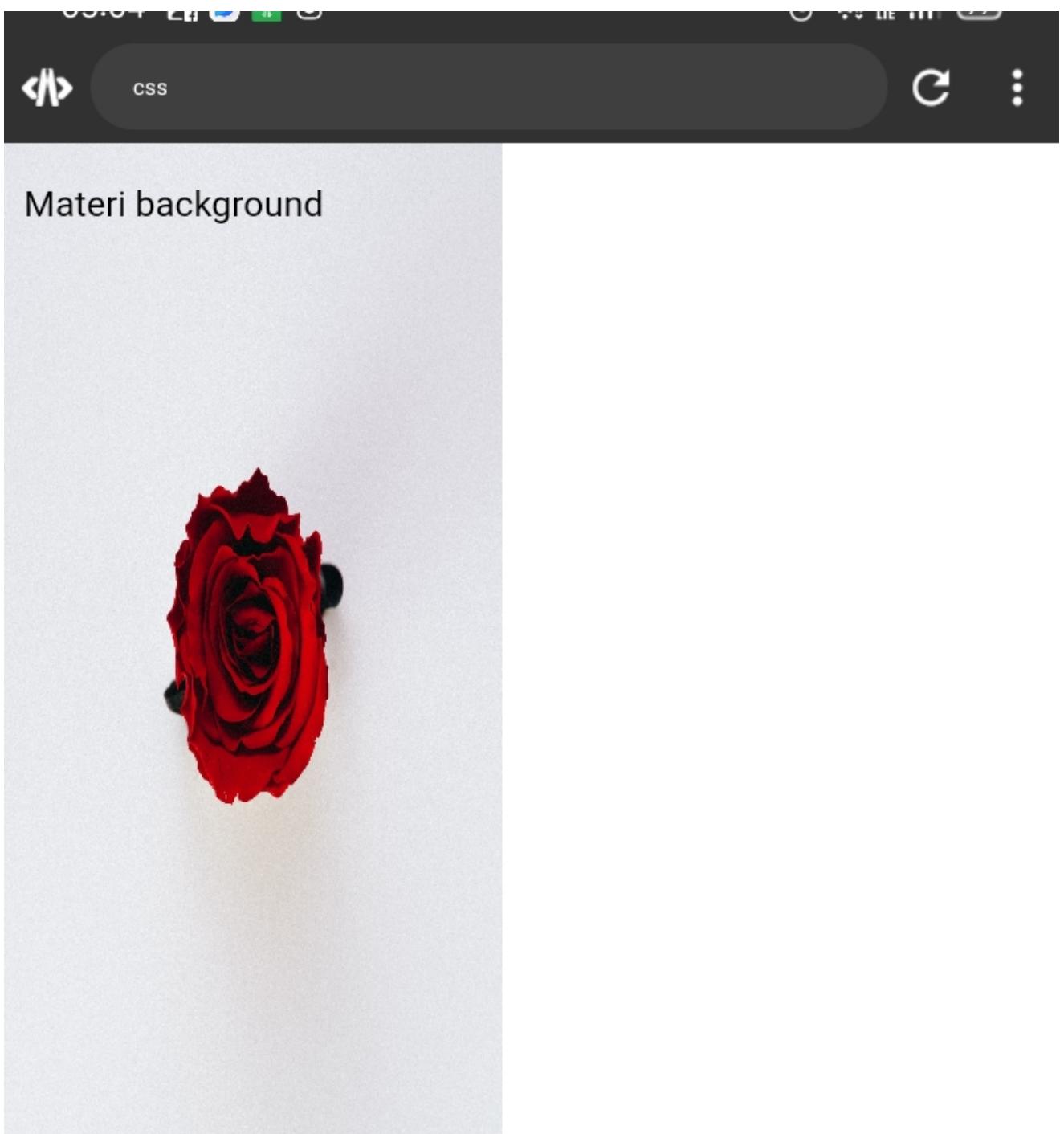


Materi background





After



Kesimpulan

Background repeat digunakan jika ingin mengatur cara latar gambar diulang dalam elemen html.

Background-attachment

Penjelasan

`background-attachment` dalam CSS adalah properti yang digunakan untuk mengontrol apakah gambar latar (background image) akan tetap diam atau bergulir bersamaan dengan konten saat pengguna meng gulir halaman. Properti ini memiliki dua nilai utama:

- `scroll` : Gambar latar belakang akan bergulir bersamaan dengan konten saat halaman digulir.
- `fixed` : Gambar latar belakang akan tetap diam relatif terhadap tampilan, sehingga saat halaman digulir, gambar tersebut tetap pada posisi yang sama di layar.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        background-image: url('2.jpg');
        height: 100vh;
        background-size: 100px;
        background-attachment: fixed;
      }
    </style>
  </head>
  <body>
    <p>Materi background</p>

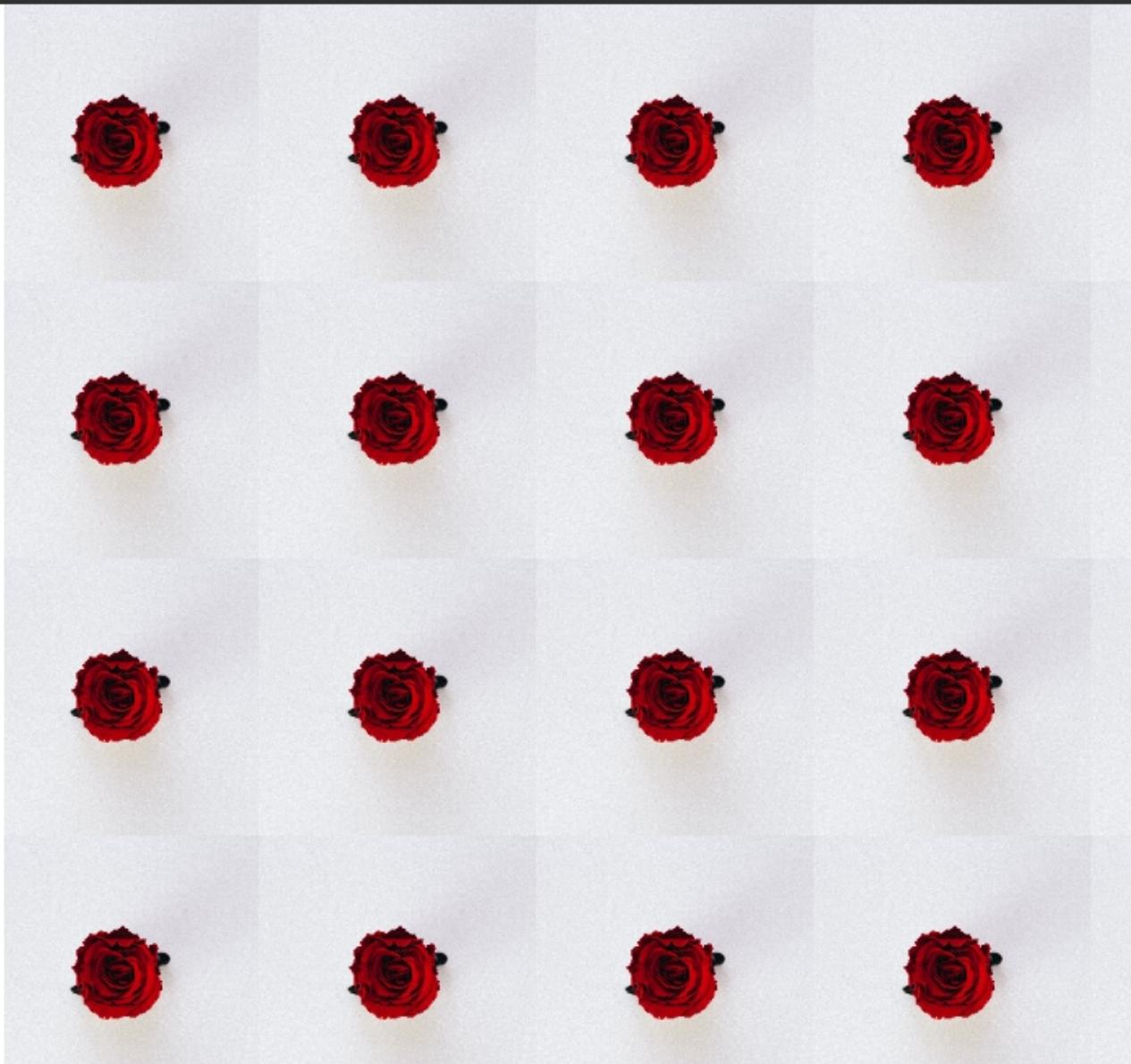
    <p>Materi background</p>

  </body>
</html>
```

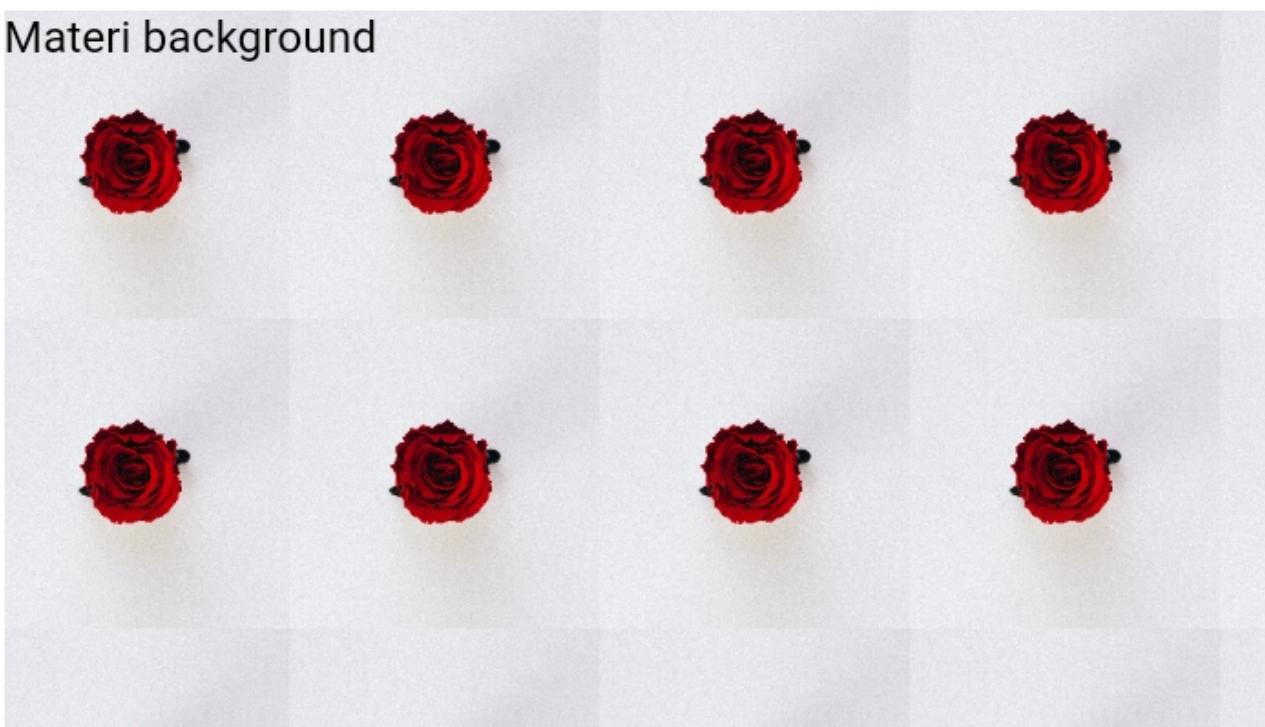
Hasil



CSS



Materi background





Kesimpulan

`background-attachment` dalam CSS adalah properti yang mengontrol apakah gambar latar belakang akan tetap diam atau bergulir bersamaan dengan konten saat pengguna meng gulir halaman.

Background-position

Penjelasan

`Background position` adalah Property CSS yang digunakan untuk menentukan posisi awal atau letak gambar latar belakang di elemen html. Property ini menentukan 2 nilai, nilai pertama horizontal dan nilai kedua vertikal.

Beberapa contoh property value nya:

- `center` menempatkan gambar di sisi tengah elemen
- `Left` menempatkan gambar disisi kiri elemen
- `Right` menempatkan gambar di sisi kanan elemen
- `top` menempatkan gambar di sisi atas elemen
- `bottom` menempatkan gambar di sisi bawah elemen

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        background-image: url('2.jpg');
        height: 100vh;
```

```
background-size:100px;  
background-position: top;  
background-repeat: no-repeat;  
}  
</style>  
</head>  
<body>  
    <p>Materi background</p>  
</body>  
</html>
```

Hasil



Materi background



Kesimpulan

Background position digunakan untuk mengatur posisi gambar yang ada dalam elemen html, seperti contoh di atas menggunakan top, gambar akan berada di atas elemen

Font

Font-size

Penjelasan

Font-size digunakan untuk menentukan ukuran Text pada suatu elemen html. Beberapa contoh property value nya seperti piksel (px), poin (pt), persentase (%), em.

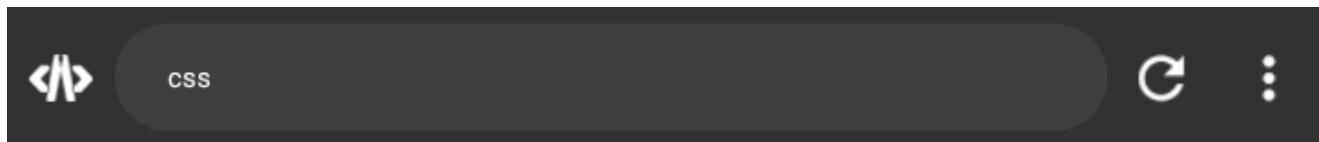
Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        font-size: 30px;
      }
    </style>
  </head>
  <body>

    <p>Materi font</p>

  </body>
</html>
```

Hasil



Materi font

Kesimpulan

Font size berfungsi untuk mengatur ukuran Text yang ada pada elemen html.

Font-style

Penjelasan

Font style Property CSS yang digunakan untuk menentukan gaya penulisan Text.

Beberapa contoh property value nya :

- normal menampilkan Text dengan gaya standar default .
- italic menampilkan Text dengan gaya miring.

Kode Program

```
<!doctype html>
<html>
  <head>
    <title>css</title>
    <style>
      p {
        font-style: italic ;
      }
    </style>
  </head>
  <body>

    <p>Materi font</p>

  </body>
</html>
```

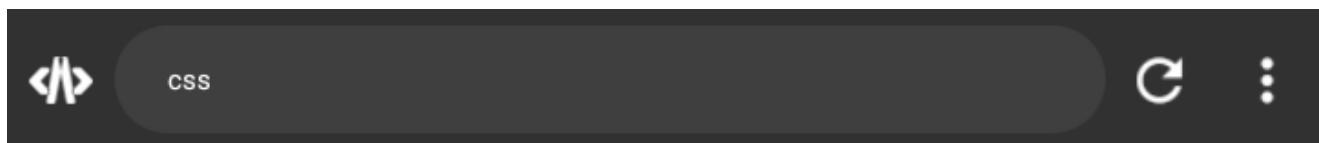
Hasil

Before



Materi font

After



Materi font

Kesimpulan

Font style Property CSS yang digunakan untuk menentukan gaya penulisan Text.

Font-weight

Penjelasan

Font weight Property CSS yang digunakan untuk mengatur ketebalan Text dalam suatu elemen html.

Beberapa contoh property value nya :

- Normal menampilkan Text dengan gaya standar default
- Bold menampilkan Text dengan menambahkan ketebalan text
- Lighter menampilkan Text dengan gaya Text yang tipis
- Bolder menampilkan Text dengan gaya menambahkan ketebalan pada Text.

Kode Program

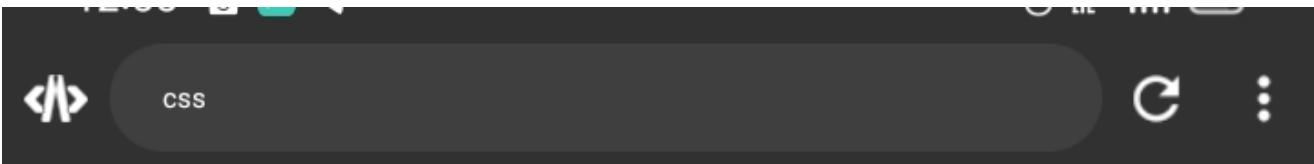
```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        font-weight: bold;
      }
    </style>
  </head>
  <body>

    <p>Materi font</p>

  </body>
</html>
```

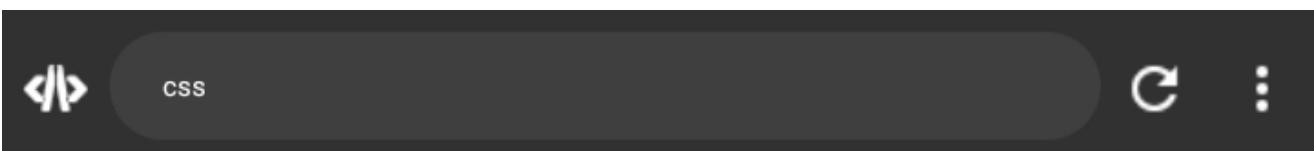
Hasil

Before



Materi font

After



Materi font

Kesimpulan

Font-weight digunakan untuk mengatur pada suatu Text dalam elemen html.

Font-family

Penjelasan

Font family Property CSS yang digunakan untuk menentukan jenis font atau kumpulan font yang akan digunakan dalam elemen html. Properti ini menerima satu atau lebih nama font, dengan nama font utama diikuti oleh alternatif jika font utama tidak tersedia.

Kode Program

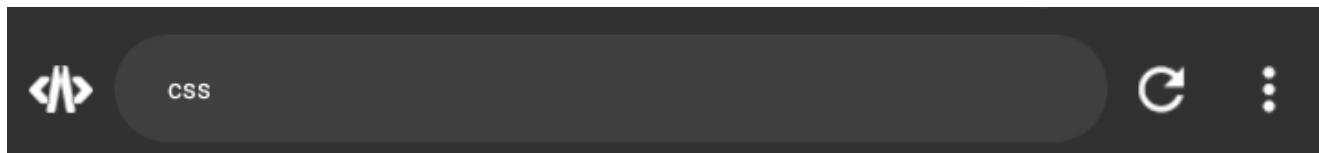
```
<html>
  <head>
    <title>css</title>
    <style>
      p {
        font-family:Times New Roman;
```

```
        }
    </style>
</head>
<body>

<p>Materi font</p>

</body>
</html>
```

Hasil



Materi font

Kesimpulan

Font-family digunakan jika ingin merubah font default pada elemen html.

Box Model

Border

Border-width

Penjelasan

border-width dalam CSS adalah properti yang digunakan untuk menentukan ketebalan dari batas (border) pada suatu elemen html. Property value nya dapat berupa piksel.misalnya, border-width: 2px 4px 6px 8px; untuk mengatur secara berurutan untuk atas, kanan, bawah, dan kiri.

Kode Program

```
<html>
<head>
```

```
<title>css</title>
<style>
  button {
    border-width: 5px;
  }
</style>
</head>
<body>
  <button>Klik</button>
</body>
</html>
```

Hasil

Before



After



Kesimpulan

Border width Property CSS yang digunakan untuk mengatur ketebalan garis pada suatu elemen html.

Border-style

Penjelasan

`border-style` dalam CSS adalah properti yang digunakan untuk menentukan gaya dari batas (border) pada suatu elemen HTML.

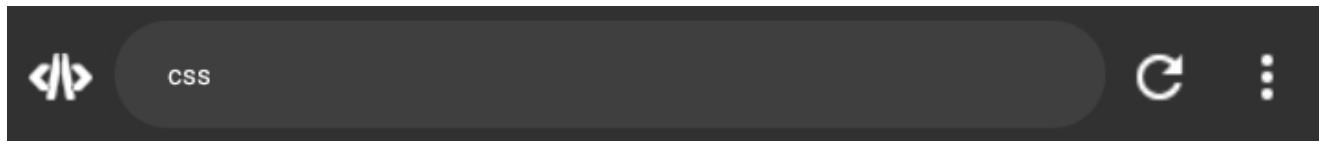
Beberapa contoh property value nya:

- `solid` : Batas dengan garis lurus dan penuh.
- `dashed` : Batas dengan garis putus-putus.
- `double` : Batas dengan dua garis.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        border-style: double;
      }
    </style>
  </head>
  <body>
    <button>Klik</button>
  </body>
</html>
```

Hasil



Kesimpulan

`Border style` digunakan jika ingin mengatur garis sesuai dengan keinginan kita pada suatu elemen html

Border-color

Penjelasan

Border color Property CSS yang digunakan mengatur warna pada Border suatu elemen html.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        border-color: red;
      }
    </style>
  </head>
  <body>

    <button>Klik</button>

  </body>
</html>
```

Hasil



Kesimpulan

Border color berfungsi untuk memberikan warna pada garis yang ada pada elemen html.

Border-radius

Penjelasan

border-radius dalam CSS adalah properti yang digunakan untuk memberikan sudut melengkung pada batas (border) elemen HTML. `border-radius: 10px 20px 30px 40px;` untuk mengatur secara berurutan untuk sudut atas kiri, atas kanan, bawah kanan, dan bawah kiri.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        background-color: purple;
        border-color: red;
        height: 100px;
        width: 200px;
        border-radius: 50px;
      }
    </style>
  </head>
  <body>
    <button>Klik</button>
  </body>
</html>
```

Hasil

Before



After



css



Klik

Kesimpulan

Border radius digunakan untuk mengatur sudut lengkung pada suatu elemen html.

Padding

Padding-left

penjelasan

Padding left property css yang digunakan untuk memberikan jarak antara batas kiri elemen dengan isi kontennya. Property value Padding juga dapat berupa satuan piksel(px), persen(%) atau em.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        padding-left: 80px;
        padding-bottom: 40px;
        padding-right: 80px;
        padding-top: 40px;
      }
    </style>
  </head>
  <body>
```

```
<button>Klik</button>
</body>
</html>
```

Hasil

Before



After



Kesimpulan

Padding-left digunakan jika ingin membuat jarak dari batas kiri elemen dengan konten yang ada didalamnya.

Padding-bottom

Penjelasan

Padding `bottom` property css yang digunakan untuk memberikan jarak antara batas bawah elemen dengan isi kontennya. Property value Padding juga dapat berupa satuan piksel(px), persen(%) atau em.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        padding-left: 80px;
        padding-bottom: 40px;
        padding-right: 80px;
        padding-top: 40px;
      }
    </style>
  </head>
  <body>
    <button>Klik</button>
  </body>
</html>
```

Hasil

Before



After



Kesimpulan

Padding `bottom` digunakan jika ingin membuat jarak dari batas bawah elemen dengan konten yang ada didalamnya.

Padding-right

Penjelasan

Padding `right` property css yang digunakan untuk memberikan jarak antara batas kanan elemen dengan isi kontennya. Property value Padding juga dapat berupa satuan piksel(px), persen(%) atau em.

Kode Program

```
<html>  
  <head>  
    <title>css</title>  
    <style>  
      button {  
        padding-left: 80px;  
        padding-bottom: 40px;  
        padding-right: 80px;  
        padding-top: 40px;  
      }  
  
    </style>  
  </head>
```

```
<body>
  <button>Klik</button>
</body>
</html>
```

Hasil

Before



After



Kesimpulan

Padding-right digunakan jika ingin membuat jarak dari batas kanan elemen dengan konten yang ada didalamnya.

Padding-top

Penjelasan

Padding top property css yang digunakan untuk memberikan jarak antara batas atas elemen dengan isi kontennya. Property value Padding juga dapat berupa satuan piksel(px), persen(%) atau em.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        padding-left: 80px;
        padding-bottom: 40px;
        padding-right: 80px;
        padding-top: 40px;
      }
    </style>
  </head>
  <body>
    <button>Klik</button>
  </body>
</html>
```

Hasil

Before



After



Kesimpulan

Padding top digunakan jika ingin membuat jarak dari batas atas elemen dengan konten yang ada didalamnya.

Margin

Margin-left

Penjelasan

Margin-left adalah ruang yang berada di sisi kanan elemen yang digunakan untuk memberikan jarak antara batas kiri elemen dengan elemen disekitarnya.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        height:60px;
        width: 120px;
        margin-left:30px;
      }
    </style>
  </head>
```

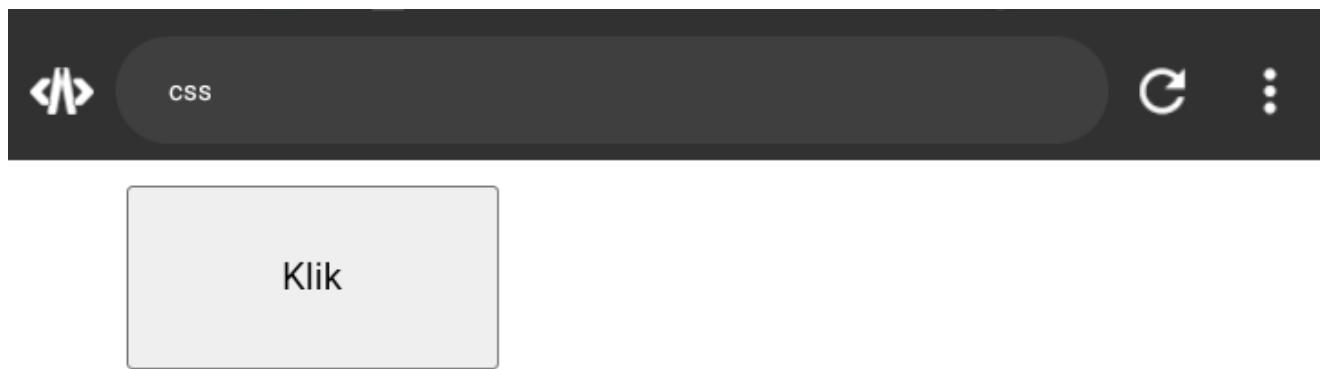
```
<body>
  <button>klik</button>
</body>
</html>
```

Hasil

Before



After



Kesimpulan

Margin left digunakan jika ingin mengatur jarak antara batas elemen kiri dengan blok elemen html.

Margin-right

Penjelasan

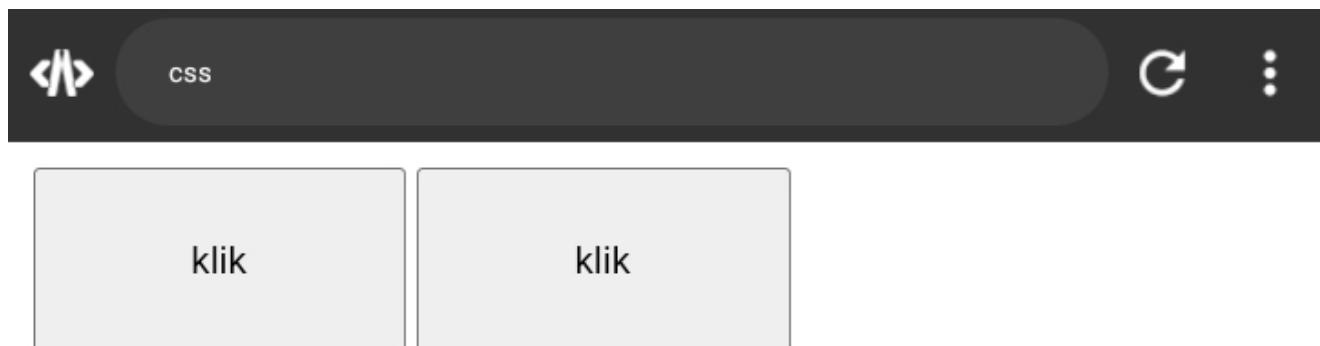
`margin-right` adalah ruang yang berada di sisi kanan elemen yang digunakan untuk memberikan jarak antara batas kanan elemen dengan elemen disekitarnya.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        height:60px;
        width: 120px;
        margin-right: 30px;
      }
    </style>
  </head>
  <body>
    <button>klik</button>
    <button>klik</button>
  </body>
</html>
```

Hasil

Before



After



klik

klik

Kesimpulan

Margin right digunakan jika ingin mengatur jarak antara elemen yang dikanan dengan elemen yang berada di kirinya.

Margin-Top

Penjelasan

margin-top adalah ruang yang berada di sisi atas elemen yang digunakan untuk memberikan jarak antara batas atas elemen dengan elemen diatasnya.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        height:60px;
        width: 120px;
        margin-top: 30px;
      }
    </style>
  </head>
  <body>
    <button>klik</button>

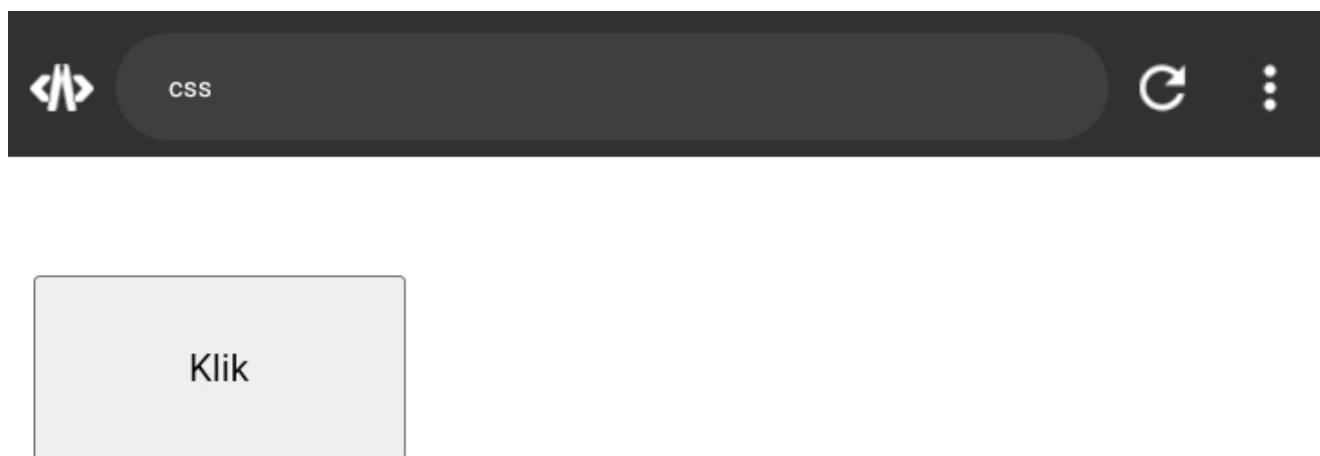
  </body>
</html>
```

Hasil

before



After



Kesimpulan

`Margin top` digunakan jika ingin mengatur jarak antara batas atas elemen dengan elemen yang diatasnya.

Margin-bottom

Penjelasan

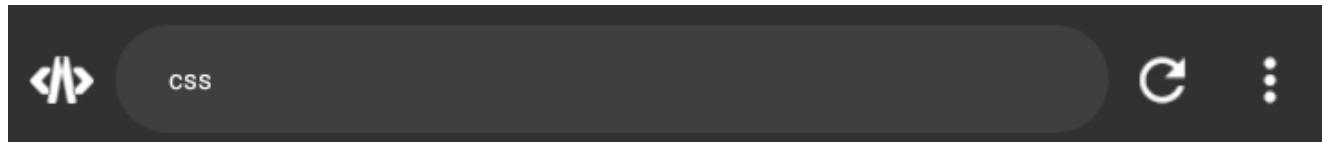
`Margin-bottom` adalah ruang yang berada di sisi bawah elemen yang digunakan untuk memberikan jarak antara batas bawah elemen dengan elemen dibawahnya.

Kode Program

```
<html>
  <head>
    <title>css</title>
    <style>
      button {
        height:60px;
        width: 120px;
        margin-bottom: 30px;
      }
    </style>
  </head>
  <body>
    <button>klik</button><br>
    <button>klik</button>
  </body>
</html>
```

Hasil

Before



After



klik

Klik

Kesimpulan

Margin bottom digunakan jika ingin mengatur jarak antara batas elemen bawah dengan elemen dibawahnya.

STUDI KASUS BOX MODEL

Deskripsi

- body : Mengatur latar belakang halaman menjadi warna ungu (background-color: purple;).
- span : Mengatur teks yang berada dalam elemen dengan warna putih (color: white;), ukuran font 30px (font-size: 30px;), dan jenis font Courier (font-family: Courier;).
- button : Mengatur gaya tombol dengan teks berwarna oranye (color: orangered;), latar belakang ungu (background-color: purple;), dan garis batas oranye (border-color: orangered;). Tombol memiliki lebar 100px dan tinggi 50px, dengan margin kiri 100px dan margin bawah 1px.
- img : Mengatur gambar dengan lebar 130px dan tinggi 130px (width: 130px; height: 130px;), diberi sudut melengkung untuk sudut bulat (border-radius: 100px;), dan garis batas hitam (border-color: black; border-style: solid;). Gambar ini memiliki margin kiri 50px dan margin atas -80px, yang menyebabkan posisinya naik sedikit dari posisi normal.

Kode Program

```
<!DOCTYPE html>
<html>
  <head>
    <title>CSS</title>
    <style>
body{
background-color: purple;
}

span{
  color: white;
  font-size: 30px;
  font-family: Courier;
}

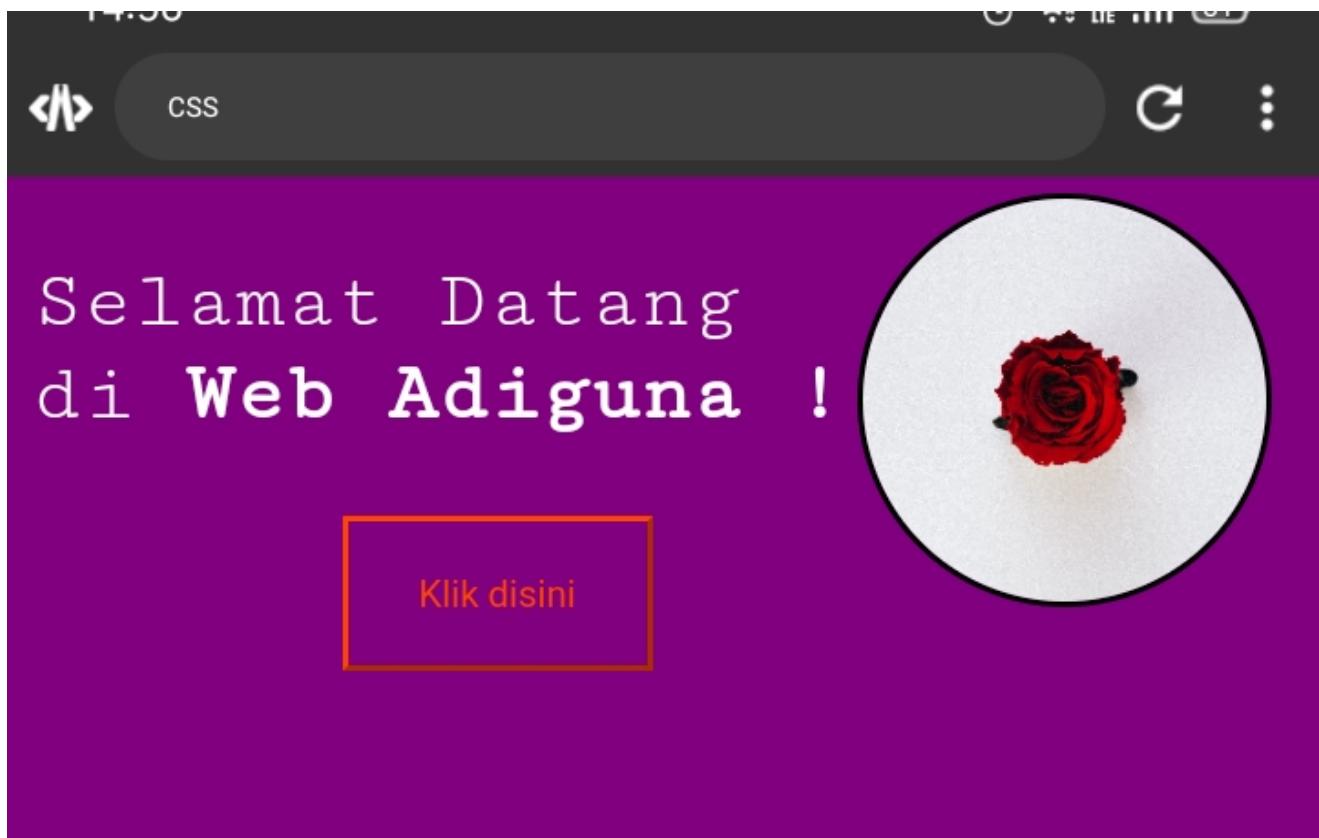
button{
  color: orangered;
  background-color: purple;
  border-color: orangered;
  width: 100px;
  height: 50px;
  margin-left: 100px;
  margin-bottom: 1px;
}

img {
  width: 130px;
  height: 130px;
  border-radius: 100px;
  border-color: black;
  border-style: solid;
  margin-left: 50px;
  margin-top: -80px;
}

</style>
</head>
<body>
<br>
<span>
  Selamat Datang
</span>
<br>
<span>
  di <b>Web Adiguna !</b><br>
<button>Klik disini</button>
</span>

</body>
</html>
```

Hasil



Pseudo class

HOVER

Penjelasan

`hover` adalah salah satu pseudo-class yang digunakan untuk menentukan gaya elemen ketika pengguna mengarahkan kursor mouse ke elemen tersebut

Kode Program

```
<html>
  <head>
    <title>hover</title>
    <style>
      button {
        height: 50px;
        width: 100px;
        background-color: red;
      }

      button:hover {
```

```
height: 60px;  
width: 120px;  
background-color: magenta;  
}  
</style>  
</head>  
<body>  
  <button>Klik</button>  
</body>  
</html>
```

Hasil

Before



After



Kesimpulan

`Hover` digunakan jika ingin menambahkan gaya pada suatu elemen html, dimana jika cursor diarahkan ke elemen maka otomatis elemen tersebut berubah bentuk sesuai dengan gaya yang dikasih.

ACTIVE

Penjelasan

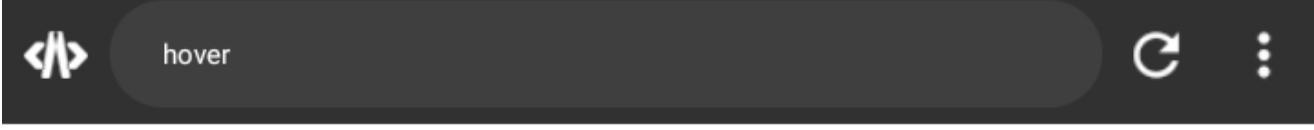
`Active` adalah pseudo class yang digunakan untuk menentukan gaya elemen ketika elemen tersebut sedang aktif/diklik

Kode Program

```
<head>
    <title>hover</title>
    <style>
        button {
            height: 50px;
            width: 100px;
            background-color: red;
        }

        button:active {
            height: 60px;
            width: 120px;
            background-color: lightpink;
        }
    </style>
</head>
<body>
    <button>Klik</button>
</body>
</html>
```

Hasil



Klik



Tidak dapat memberikan hasil Afternya, karena hanya tampil jika diklik saja, beda dengan hover yang hanya perlu geser cursor ke elemen untuk melihat perubahannya.

Kesimpulan

`Active` digunakan jika ingin menambahkan gaya pada elemen yang akan terlihat perubahannya setelah diklik.

VISITED

Penjelasan

`Visited` adalah pseudo class yang digunakan untuk menentukan gaya pada elemen yang telah di kunjungi oleh user. umumnya digunakan untuk merubah tampilan tautan yang di klik agar user bisa membedakan mana link yang sudah dikunjungi dan yang belum dikunjungi

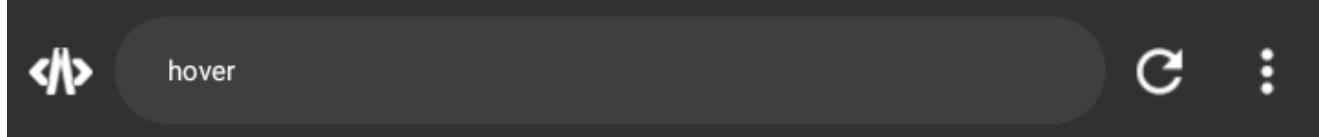
Kode Program

```
<html>
<head>
    <title>hover</title>
    <style>
        a {
            color: black;
        }
        a:visited {
            color: lightpink;
        }
    </style>
</head>
<body>
    <a href="#">Klik</a>
</body>
</html>
```

```
</style>
</head>
<body>
    <a href="https://www.w3schools.com/html/html_colors.asp">klik disini</a>
</body>
</html>
```

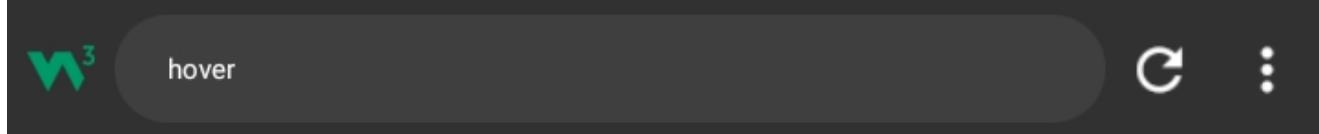
Hasil

Before



klik disini

After



klik disini

Kesimpulan

`Visited` pseudo Class ini digunakan untuk memberikan gaya pada tampilan link yang sudah dikunjungi.

LINK

Penjelasan

Link Pseudo Class yang digunakan untuk memberikan gaya pada elemen/link yang belum dikunjungi, kebalikannya dari Visited.

Kode Program

```
<html>
  <head>
    <title>unvisited</title>
    <style>
      a:link {
        font-size: 30px;
        color:lightpink;
      }
    </style>
  </head>
  <body>
    <a href="https://developer.mozilla.org/">klik aja</a><br>
    <a href="https://www.w3schools.com/">baca disini</a><br>
    <a href="https://css-tricks.com/">coba klik</a>
  </body>
</html>
```

Hasil

Before



klik aja
baca disini
coba klik

After



klik aja
baca disini
coba klik

Kesimpulan

Jadi fungsi dari Pseudo Class `link` ini memberikan gaya pada tautan yang belum dikunjungi.

Transition

Penjelasan

Transition dalam css adalah sebuah mekanisme untuk mengendalikan perubahan pada sebuah elemen secara halus.

- `Transition-property` menentukan property apa yang akan diberi efek Transition, seperti `width, color, atau all(semua property)`.
- `transition-duration` menentukan waktu berapa lama efek Transitionnya di ukur dalam detik seperti `0,5s`.
- `transition-timing-function` menentukan bagaimana kecepatan perubahan nilai properti selama transisi. Seperti
 - `ease` : lambat diawal, cepat di tengah, dan lambat di akhir
 - `ease-in` : dimulai dengan lambat dan semakin cepat seiring waktu
 - `ease-out` : cepat diawal dan lambat diakhir
 - `ease-in-out` : Kombinasi dari ease-in dan ease-out, menciptakan efek transisi mulai lambat, cepat di tengah, dan melambat kembali
 - `linear` : memberikan perubahan langsung secara konstan
- `Transition delay` adalah properti dalam CSS yang digunakan untuk menunda mulainya efek transisi pada suatu elemen setelah perubahan nilai properti terjadi.

Kode Program

```
<html>
  <head>
    <title>transisi</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

    button:hover {
      height: 60px;
      width: 110px;
      background-color: magenta;
      transition: all 0.5s ease-in-out 0.3s;
    }
  </style>
  <body>
    <button>coba klik</button>
  </body>
</html>
```

Hasil

Before



After



coba klik

Kesimpulan

Transition Property CSS yang digunakan untuk memberikan efek perubahan secara halus dalam sebuah elemen html.

STUDI KASUS TRANSITION

Deskripsi

- body : Mengatur latar belakang halaman menjadi warna ungu (background-color: purple;).
- span : Mengatur teks yang berada dalam elemen `` dengan warna putih (color: white;), ukuran font 30px (font-size: 30px;), dan jenis font Courier (font-family: Courier;).
- Button : Mengatur gaya tombol dengan teks berwarna oranye (color: orangered;), latar belakang ungu (background-color: purple;), dan garis batas oranye (border-color: orangered;). Tombol memiliki lebar 100px dan tinggi 50px, dengan margin kiri 100px dan margin bawah 1px.
 - Pada hover (button:hover), latar belakang tombol berubah menjadi oranye (background-color: orangered;), warna teks menjadi hitam (color: black;), dan efek transisi (transition: all 0.5s ease-in-out;) diterapkan untuk memberikan animasi perubahan gaya.
- img : Mengatur gambar dengan lebar 130px dan tinggi 130px (width: 130px; height: 130px;), diberi sudut melengkung untuk sudut bulat (border-radius: 100px;), dan garis batas hitam (border-color: black; border-style: solid;). Gambar ini memiliki margin kiri 50px dan margin atas -80px, yang menyebabkan posisinya naik sedikit dari posisi normal.

Kode Program

```
<!DOCTYPE html>
<html>
  <head>
    <title>CSS</title>
    <style>
      body{
        background-color: purple;
      }
      span{
        color: white;
        font-size: 30px;
        font-family: Courier;
      }
      button{
        color: orangered;
        background-color: purple;
        border-color: orangered;
        width: 100px;
        height: 50px;
        margin-left: 100px;
        margin-bottom: 1px;
      }
      button:hover {
        background-color: orangered;
        transition: all 0.5s ease-in-out;
        color: black;
      }
      img {
        width: 130px;
        height: 130px;
        border-radius: 100px;
        border-color: black;
        border-style: solid;
        margin-left: 50px;
        margin-top: -80px;
      }
    </style>
  </head>
  <body>
    <br>
    <span>
      Selamat Datang
    </span>
    <br>
    <span>
```

```
di <b>Web Adiguna !</b><br>
<button>Klik disini</button>
</span>

</body>
</html>
```

Hasil



Transform

Scale

Penjelasan

digunakan untuk mengubah tampilan suatu elemen, seperti memindahkannya, memutarinya, atau mengubah ukurannya. nilai `Scale` didefinisikan dengan menyebut faktor perubahannya seperti `Scale(2)` akan memperbesar elemen menjadi 2 kali lipat.

Kode Program

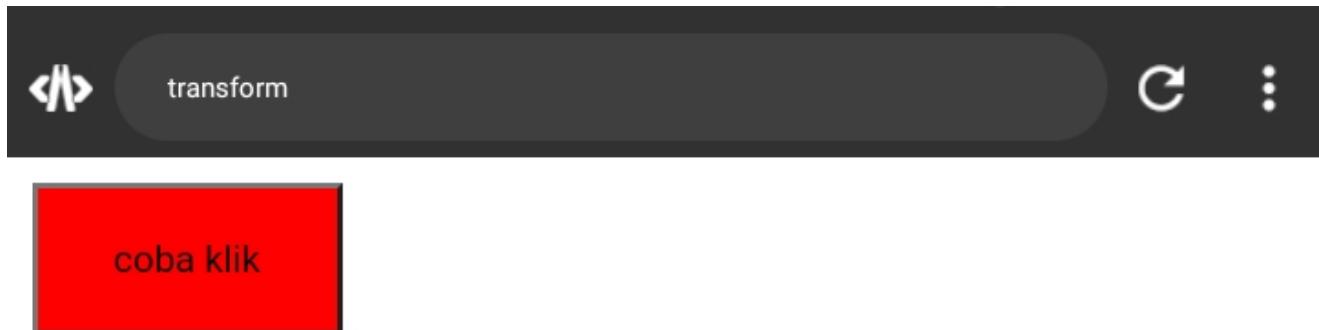
```
<html>
<head>
```

```
<title>transform</title>
</head>
<style>
button {
    height: 50px;
    width: 100px;
    background-color: red;
}

button:hover {
    height: 60px;
    width: 110px;
    background-color: magenta;
    transform:scale(0.5);
}
</style>
<body>
    <button>coba klik</button>
</body>
</html>
```

Hasil

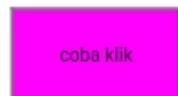
Before



After



transform



Kesimpulan

Properti `scale` ini mengubah ukuran dari elemen dengan memperbesar (scaling up) atau memperkecil (scaling down) ukuran elemen berdasarkan faktor yang diberikan.

ScaleX

Penjelasan

ScaleX Property CSS yang digunakan untuk mengubah skala elemen hanya pada sumbu horizontal saja. Misalnya, transform: `scaleX(2)` akan membuat elemen menjadi dua kali lebih lebar dari ukurannya aslinya.

Kode Program

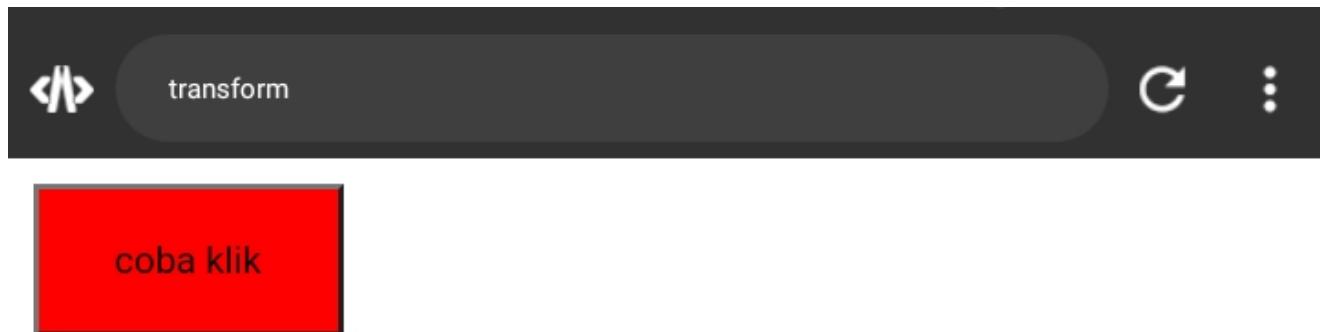
```
<html>
  <head>
    <title>transform</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

    button:hover {
      height: 60px;
      width: 110px;
      background-color: magenta;
      transform:scaleX(0.5);
    }
  </style>
</html>
```

```
        }
    </style>
<body>
    <button>coba klik</button>
</body>
</html>
```

Hasil

Before



After



Kesimpulan

`scaleX` adalah cara yang efektif untuk mengubah ukuran horizontal elemen dalam tata letak web.

Rotate

Penjelasan

Rotate Property CSS yang digunakan untuk memutar elemen disekitar titik pusatnya. Nilai rotate dapat dinyatakan dalam derajat(deg) seperti `rotate(45deg)`.

Kode Program

```
<html>
  <head>
    <title>transform</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

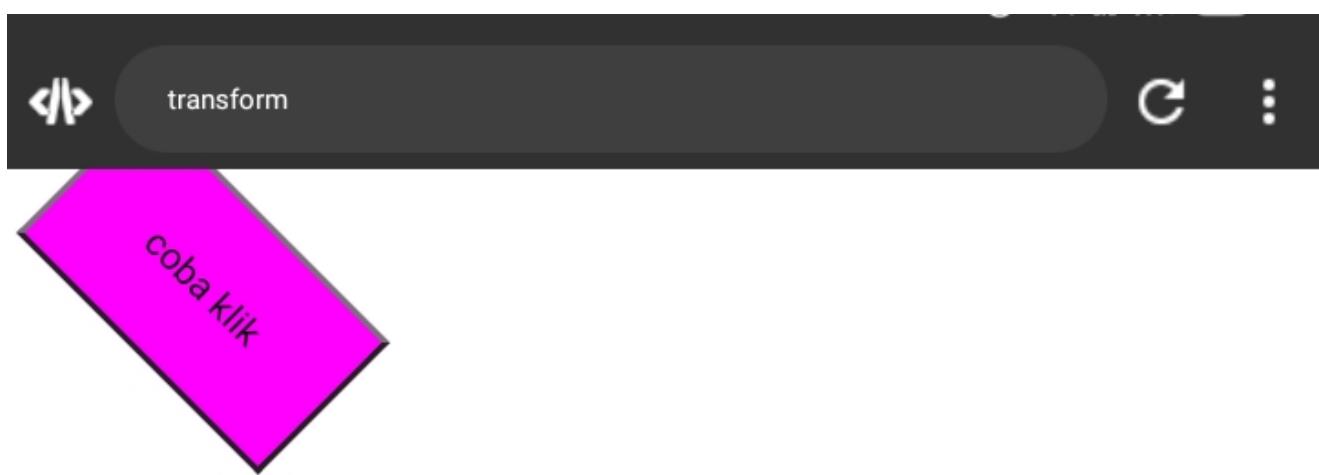
    button:hover {
      height: 60px;
      width: 110px;
      background-color: magenta;
      transform:rotate(45deg);
    }
  </style>
  <body>
    <button>coba klik</button>
  </body>
</html>
```

Hasil

Before



After



Kesimpulan

`rotate` ia akan berputar searah jarum jam jika nilai positif digunakan, dan berlawanan arah jarum jam jika nilai negatif digunakan

SkewX

Penjelasan

`SkewX` digunakan untuk memiringkan elemen sepanjang horizontal. Nilai `skewX` dinyatakan dengan derajat(deg) seperti `skewX(30deg)`.

Kode Program

```
<html>
  <head>
    <title>transform</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

    button:hover {
      height: 60px;
      width: 110px;
      background-color: magenta;
      transform:skewX(30deg);
    }
  </style>
  <body>
    <button>coba klik</button>
  </body>
</html>
```

Hasil

Before



After



transform



Kesimpulan

Properti `skewX` mengizinkan kita untuk menentukan sudut miring (`skew angle`) dalam derajat. Ketika kita menggunakan `skewX(angle)`, elemen akan dimiringkan sepanjang sumbu horizontal sebesar sudut yang ditentukan.

Skew

Penjelasan

Skew digunakan untuk memiringkan elemen baik secara horizontal maupun vertikal dalam satu nilai. Nilai skew dapat dinyatakan dalam derajat(deg) seperti `skew(45deg, 20deg)` nilai yang pertama untuk horizontal sedangkan nilai yang kedua untuk vertikal.

Kode Program

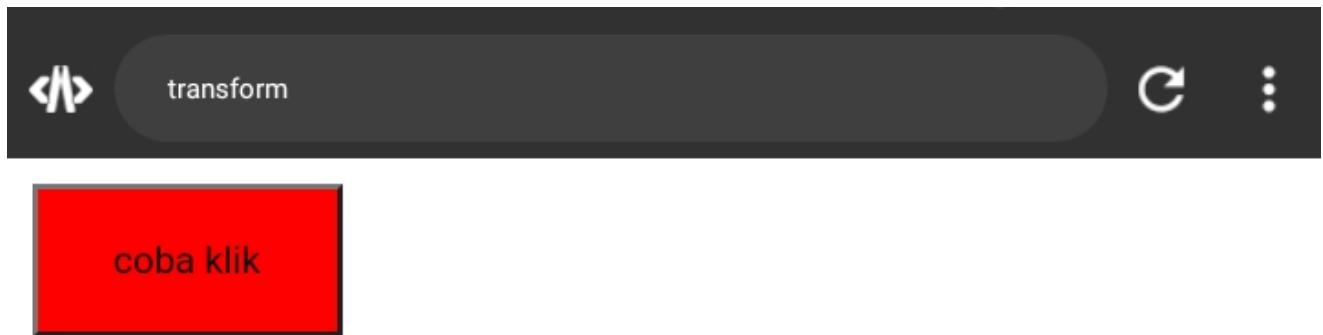
```
<html>
  <head>
    <title>transform</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

    button:hover {
      height: 60px;
    }
  </style>
</head>
<body>
  <button>coba klik</button>
</body>
</html>
```

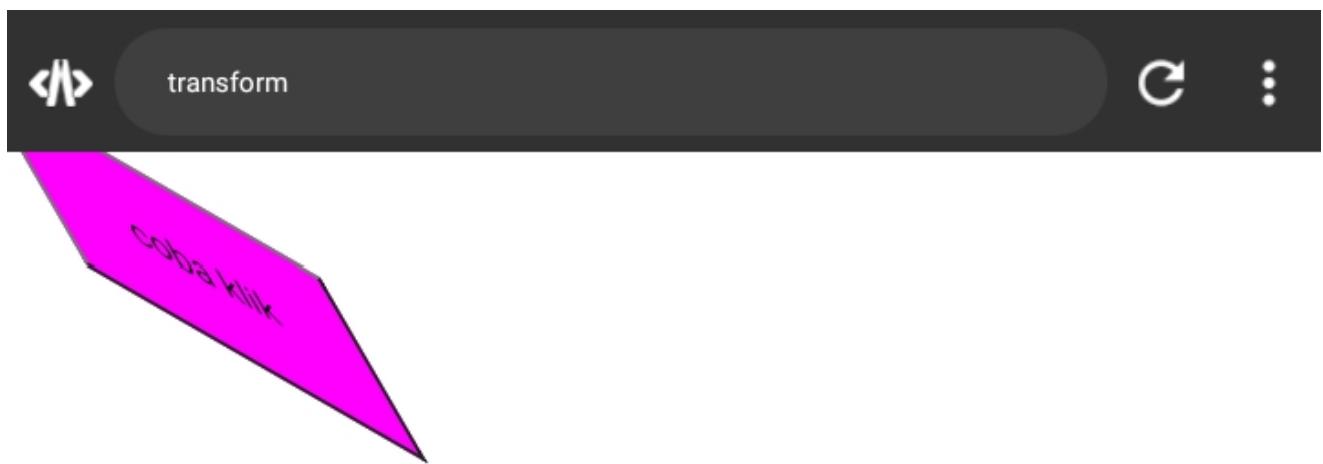
```
width: 110px;  
background-color: magenta;  
transform: skew(30deg, 30deg);  
}  
</style>  
<body>  
  <button>coba klik</button>  
</body>  
</html>
```

Hasil

Before



After



Kesimpulan

Properti `skew` memungkinkan kita untuk menentukan dua sudut miring (skew angles), satu untuk sumbu X dan satu untuk sumbu Y, dalam derajat.

Translate

Penjelasan

`Translate` digunakan untuk memindahkan elemen dari posisi awalnya. Nilai translate dapat berupa piksel(px) atau persen(%), seperti `translate(100px, 50px)`, nilai awalnya akan memindahkan elemen 100px ke kanan dan nilai yang kedua digunakan untuk memindahkan kebawah sebanyak 50px, jika ingin geser kekiri kita beri nilai awalnya dengan minus, begitupun jika kita ingin geser keatas hanya perlu tambahkan minus pada nilai yang kedua.

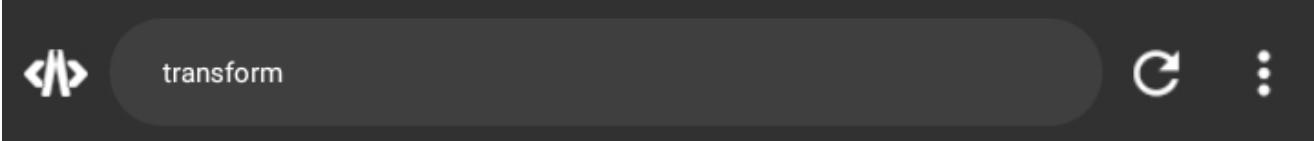
Kode Program

```
<html>
  <head>
    <title>transform</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

    button:hover {
      height: 60px;
      width: 110px;
      background-color: magenta;
      transform:translate(100px, 50px);
    }
  </style>
  <body>
    <button>coba klik</button>
  </body>
</html>
```

Hasil

Before



coba klik

After



coba klik

Kesimpulan

Properti `translate` memungkinkan kita untuk menentukan pergeseran (offset) dalam piksel atau persentase terhadap ukuran elemen terkait.

Matrix

Penjelasan

`Matrix` adalah salah satu fungsi transformasi yang digunakan untuk mengubah tata letak elemen HTML.

Contoh property value nya seperti `matrix(1, -0.3, 0.6, 1, 50, 20)`.

Kode Program

```
<html>
  <head>
    <title>transform</title>
  </head>
  <style>
    button {
      height: 50px;
      width: 100px;
      background-color: red;
    }

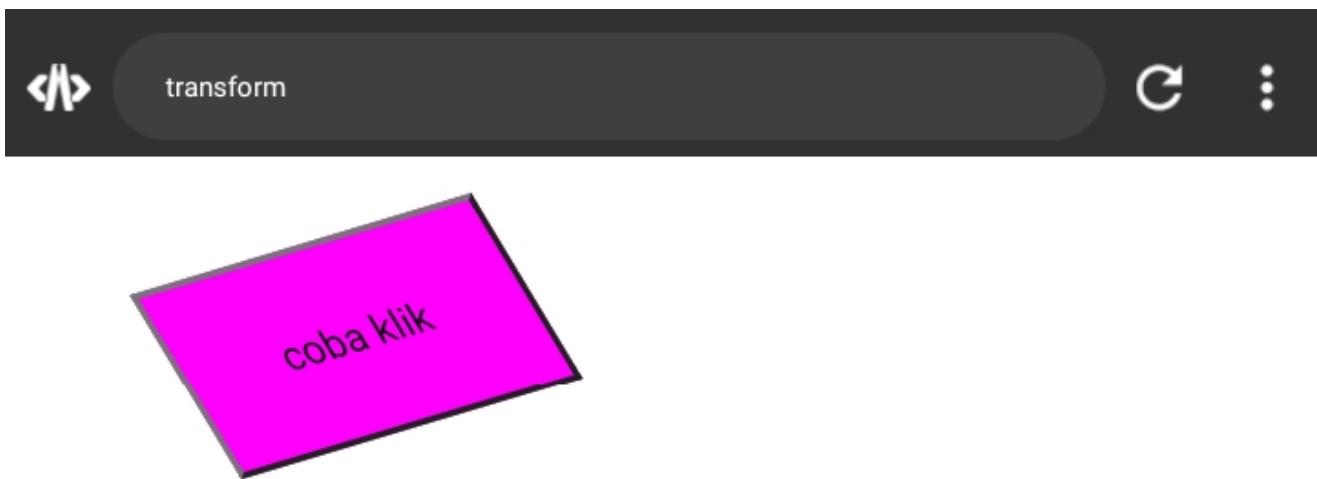
    button:hover {
      height: 60px;
      width: 110px;
      background-color: magenta;
      transform: matrix(1, -0.3, 0.6, 1, 50, 20);
    }
  </style>
  <body>
    <button>coba klik</button>
  </body>
</html>
```

Hasil

Before



After



Kesimpulan

Penggunaan `matrix` memungkinkan kita untuk mengontrol transformasi dengan tingkat kebebasan yang tinggi, tetapi sintaksnya memerlukan pemahaman tentang operasi matriks dan pengaruhnya terhadap elemen yang ditransformasi.

FlexBox

FLEX CONTAINER

Display Flex

Penjelasan

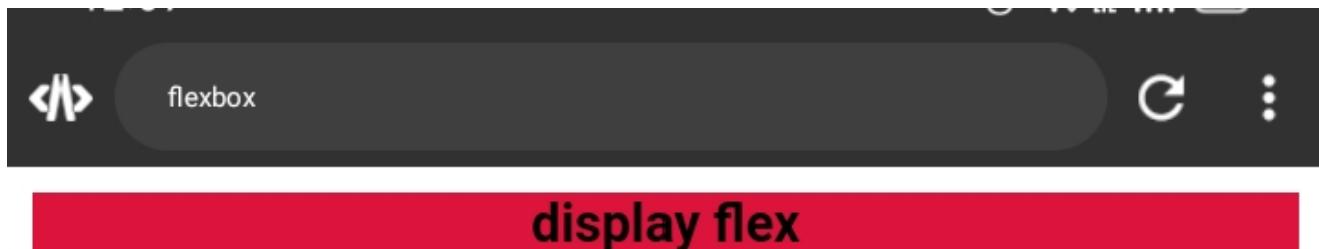
`display: flex;` adalah sebuah properti CSS yang digunakan untuk mengatur sebuah container agar anak-anak elemennya (child elements) menjadi flex items dan mengaktifkan model layout Flexbox. Ini memberikan kontrol yang sangat fleksibel dalam pengaturan tata letak elemen di dalam container, baik secara horizontal maupun vertikal.

Kode Program

```
<!DOCTYPE html>
<html>
  <head>
    <title>flexbox</title>
    <style>
```

```
h3 {  
    display: flex;  
    justify-content: center;  
    background-color: crimson;  
}  
</style>  
</head>  
<body>  
    <h3>display flex</h3>  
</body>  
</html>
```

Hasil



Kesimpulan

`display: flex;` mengaktifkan model layout Flexbox di sebuah container CSS, yang memungkinkan pengaturan tata letak yang fleksibel untuk elemen-elemen di dalamnya.

FLEX DIRECTION

Penjelasan

`Flex direction` digunakan untuk mengatur arah tata letak elemen dalam sebuah flexbox container.

Beberapa contoh property value nya :

- `row` : Flex item disusun dalam satu baris dari kiri ke kanan.
- `row-reverse` : Sama seperti `row`, tetapi urutan item dibalik, dari kanan ke kiri.
- `column` : Flex item disusun dalam satu kolom dari atas ke bawah.

- `column-reverse` : Sama seperti column, tetapi urutan item dibalik, dari bawah ke atas

Kode Program

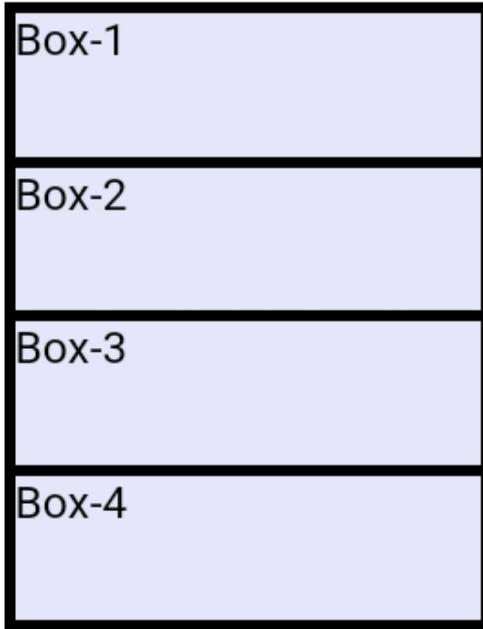
```
<html>
  <head>
    <title>flexbox</title>
  </head>
  <style>
    .container {
      display: flex;
      flex-direction: column;
      border: 2px solid black;
      height: 200px;
      width: 400px;
    }

    .item {
      background-color: lavender;
      border: 2px solid black;
      height: 50px;
      width: 150px;
    }
  </style>
  <body>
    <div class="container">
      <div class="item">Box-1</div>
      <div class="item">Box-2</div>
      <div class="item">Box-3</div>
      <div class="item">Box-4</div>
    </div>
  </body>
</html>
```

Hasil



flexbox



Kesimpulan

`Flex direction` berfungsi ketika kita ingin mengatur tataletak item yang berada didalam container.

FLEX WRAP

Penjelasan

`Flex wrap` dalam flexbox untuk mengontrol apakah item-item flexbox dalam sebuah container akan melintasi baris atau tidak ketika ruang yang tersedia tidak cukup untuk menampung semua item tersebut dalam satu baris atau kolom.

Beberapa Property Value nya :

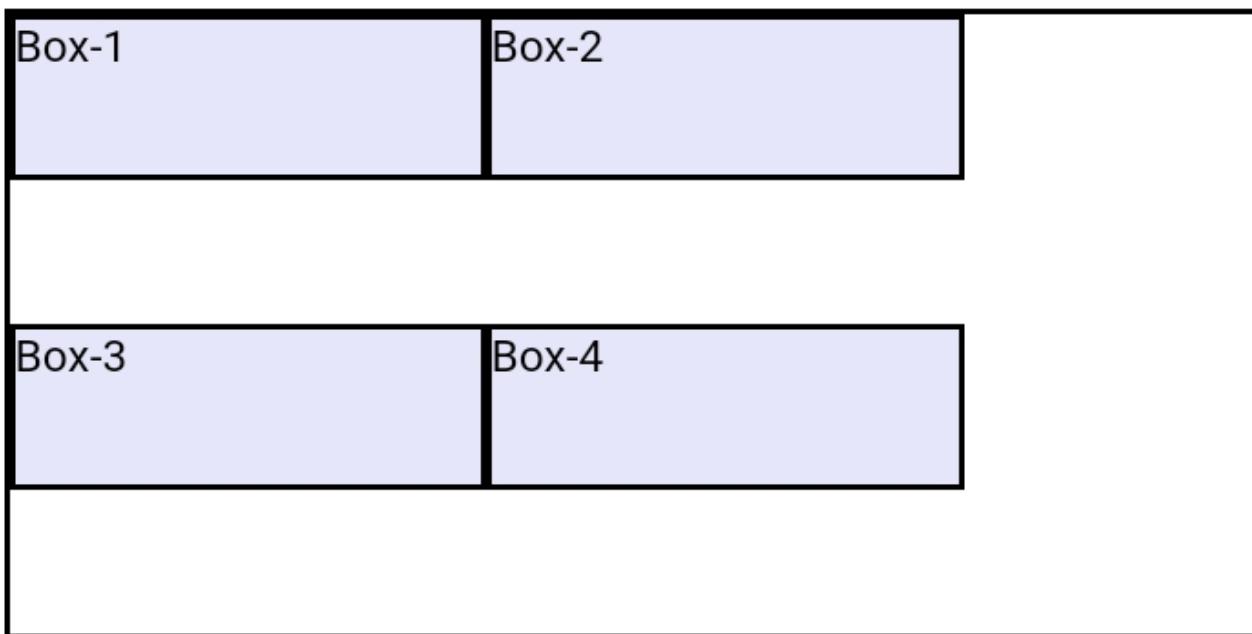
- `wrap` berfungsi ketika dalam 1 baris tidak dapat lagi di tampung maka akan membuat baris baru di bawahnya
- `nowrap` item-item akan berada dalam satu baris atau kolom, meskipun ruang tidak cukup.
- `wrap-reverse` sama dengan `wrap` cuman kebalikannya saja.

Kode Program

```
<html>
  <head>
    <title>flexbox</title>
  </head>
  <style>
    .container {
      display: flex;
      flex-wrap: wrap;
      border: 2px solid black;
      height: 200px;
      width: 400px;
    }

    .item {
      background-color: lavender;
      border: 2px solid black;
      height: 50px;
      width: 150px;
    }
  </style>
  <body>
    <div class="container">
      <div class="item">Box-1</div>
      <div class="item">Box-2</div>
      <div class="item">Box-3</div>
      <div class="item">Box-4</div>
    </div>
  </body>
</html>
```

Hasil



Kesimpulan

Flex wrap dominan digunakan ketika item yang berada dalam 1 baris sudah tidak cukup ditampung oleh container, maka item selanjutnya akan membuat baris baru.

ALIGN ITEMS

Penjelasan

Align items Property CSS yang digunakan untuk mengatur tataletak elemen secara vertikal dalam flex container.

Beberapa contoh Property Value nya :

- Flex-start item akan diletakkan di awal container
- Flex-end item akan diletakkan di akhir container
- center item akan diletakkan di tengah container
- stretch item akan meregang mengisi seluruh tinggi container
- base-line item akan diletakkan digaris dasar dari Text mereka.

Kode Program

```
<html>
<head>
```

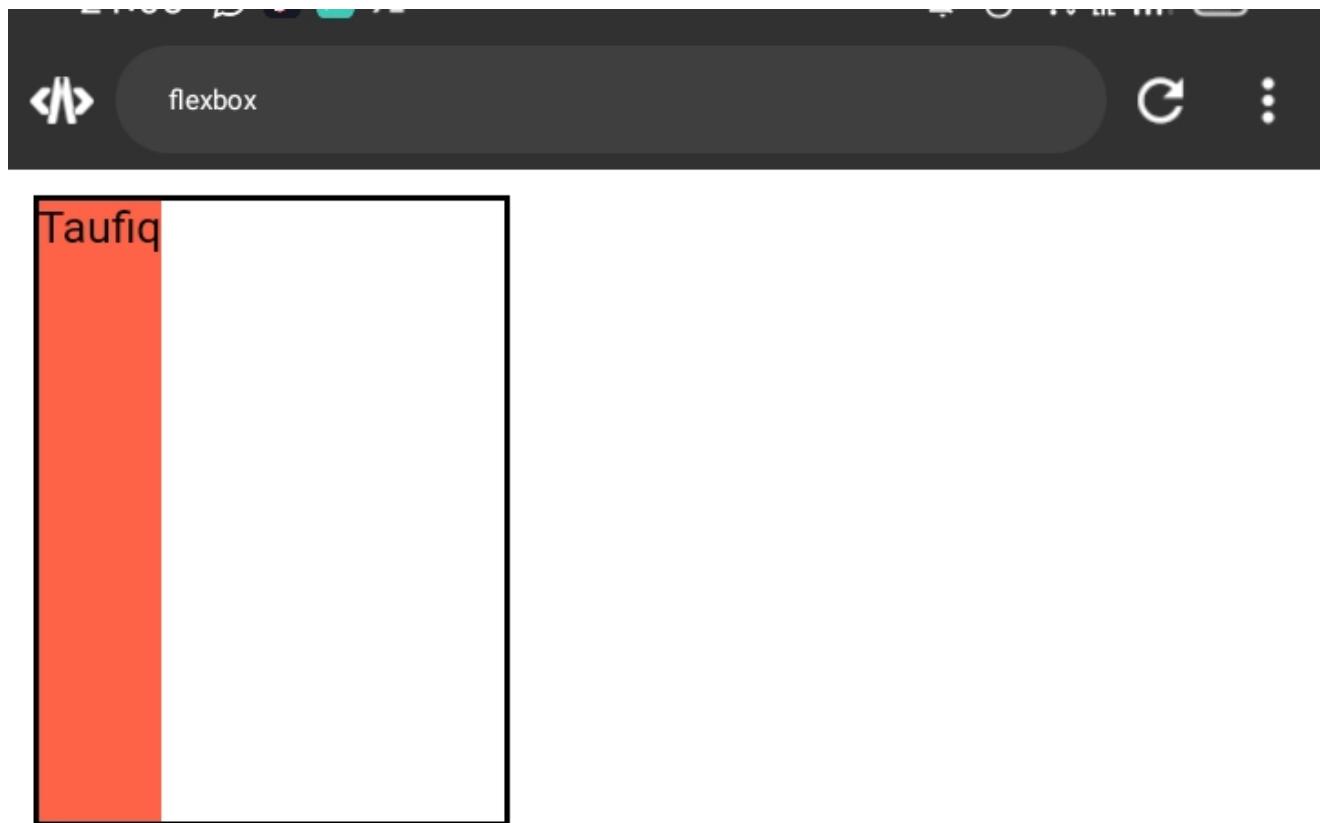
```
<title>flexbox</title>
</head>
<style>
  .container {
    display: flex;
    height: 200px;
    width: 150px;
    border: 2px solid black;
    align-items: stretch;
  }

  .item {
    background-color: tomato;
  }

</style>
<body>
  <div class="container">
    <div class="item">Taufiq</div>
  </div>

</body>
</html>
```

Hasil



Kesimpulan

Align items digunakan untuk mengatur tataletak suatu elemen secara vertikal

JUSTIFY CONTENT

Penjelasan

`justify-content` adalah properti CSS yang digunakan untuk mengatur cara penempatan dan penyebaran ruang ekstra dari sebuah flex container secara horizontal.

Beberapa contoh property value nya :

- `flex-start` : Item-item akan diletakkan di awal container.
- `flex-end` : Item-item akan diletakkan di akhir container.
- `center` : Item-item akan diletakkan di tengah container.
- `space-between` : Item-item akan ditempatkan dengan jarak yang sama di antara mereka, tetapi tidak di sisi kanan dan kiri container.
- `space-around` : Item-item akan ditempatkan dengan jarak yang sama di antara mereka, termasuk di sisi kanan dan kiri container.
- `space-evenly` : Item-item akan ditempatkan dengan jarak yang sama di antara mereka, termasuk di sisi kanan dan kiri container, serta jarak yang sama di sekeliling mereka.

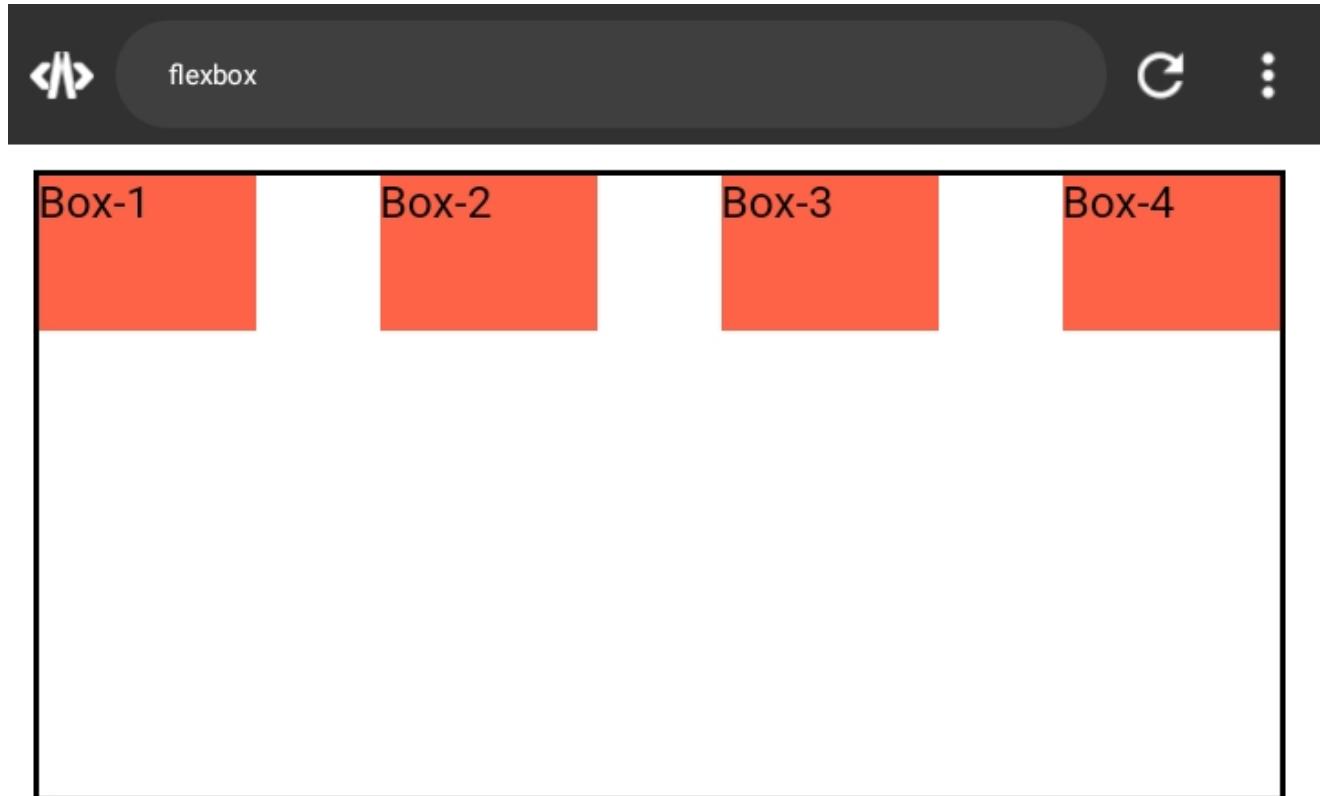
Kode Program

```
<html>
  <head>
    <title>flexbox</title>
  </head>
  <style>
    .container {
      display: flex;
      height: 200px;
      width: 400px;
      border: 2px solid black;
      justify-content: space-between;
    }

    .item {
      background-color: tomato;
      height: 50px;
      width: 70px;
    }
  </style>
<body>
```

```
<div class="container">
  <div class="item">Box-1</div>
  <div class="item">Box-2</div>
  <div class="item">Box-3</div>
  <div class="item">Box-4</div>
</div>
</body>
</html>
```

Hasil



Kesimpulan

`Justify content` digunakan untuk mengatur jarak antar elemen secara horizontal atau baris

ALIGN CONTENT

Penjelasan

`Align-content` adalah properti CSS yang digunakan untuk mengatur cara konten dalam sebuah flex container disusun secara vertikal ketika ada ruang ekstra di dalam kontainer flex tersebut.

Beberapa Property Value nya :

- `flex-start` : item akan berada di awal container
- `flex-end` : item akan berada di akhir container
- `center` : item akan diletakkan ditengah container
- `space-between` : item diperintahkan diletakkan dengan jarak yang sama, pertama di awal dan terakhir di akhir.
- `space-around` : item diperintahkan diletakkan dengan jarak yang sama di sekitar setiap item.
- `space-evenly` : item diperintahkan untuk diletakkan dengan jarak yang sama di antara dan sekitar setiap item.
- `stretch` : item diperintahkan untuk meregangkan untuk mengisi seluruh kontainer fleksibel.

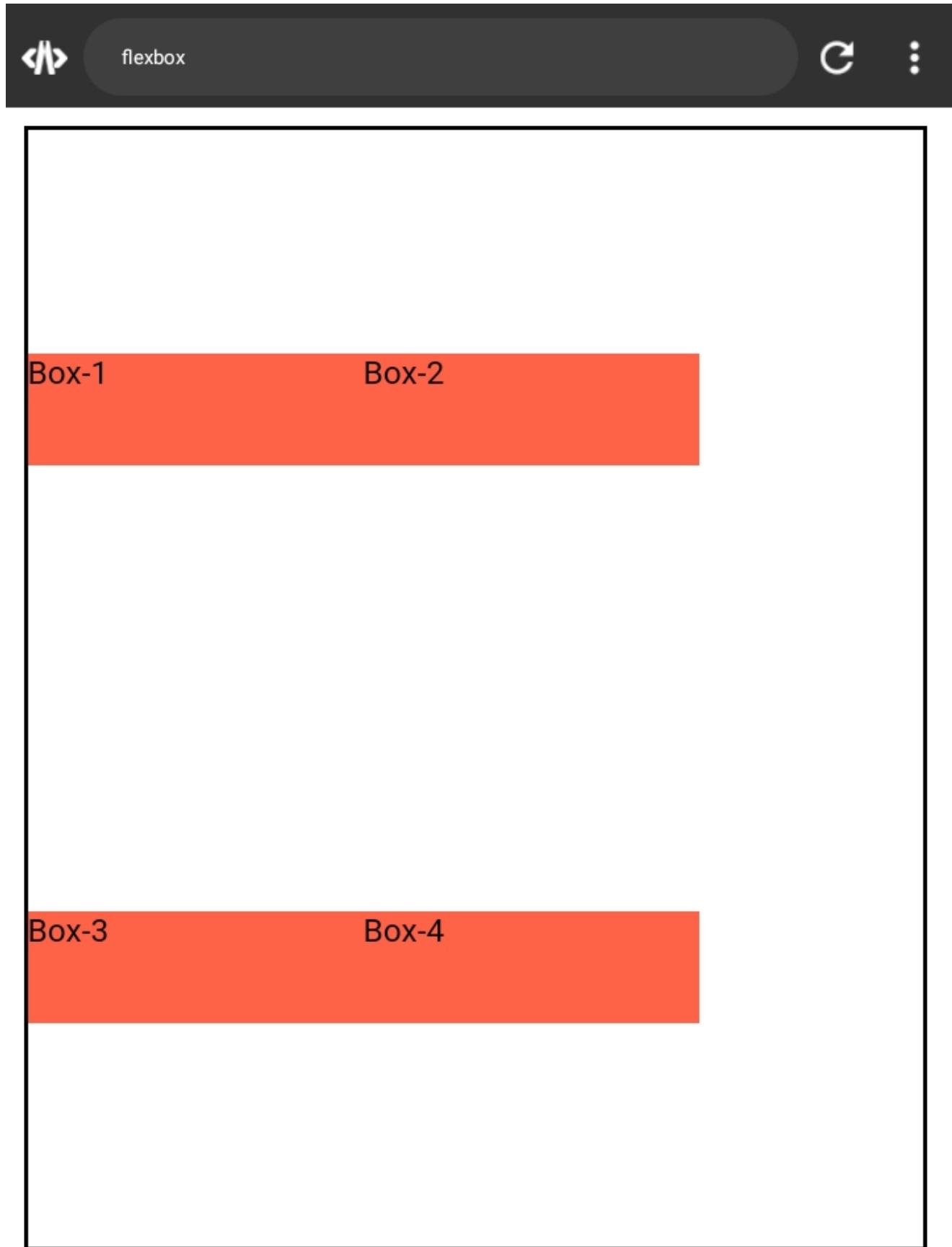
Kode Program

```
<html>
  <head>
    <title>flexbox</title>
  </head>
  <style>
    .container {
      display: flex;
      height: 500px;
      width: 400px;
      border: 2px solid black;
      align-content: space-around;
      flex-wrap: wrap;
    }

    .item {
      background-color: tomato;
      height: 50px;
      width: 150px;
    }
  </style>
  <body>
    <div class="container">
      <div class="item">Box-1</div>
      <div class="item">Box-2</div>
      <div class="item">Box-3</div>
      <div class="item">Box-4</div>
    </div>

  </body>
</html>
```

Hasil



Kesimpulan

Align content digunakan untuk mengatur jarak antar elemen secara vertikal

FLEX ITEM

ORDER

Penjelasan

Order adalah Property CSS dalam flexbox yang digunakan untuk mengatur tata letak flex item dalam sebuah flex container, value yang di berikan kepada order adalah bilangan bulat.

Kode Program

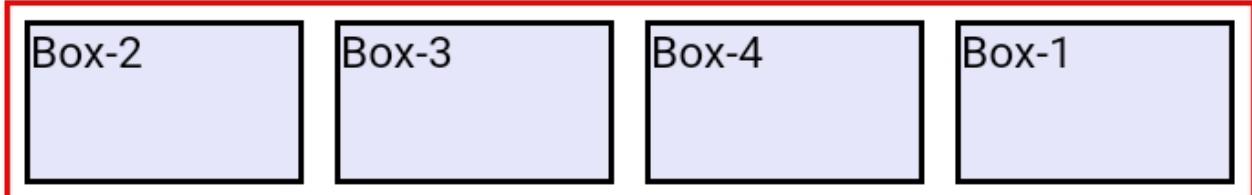
```
<html>
  <head>
    <title>flexbox</title>
  </head>
  <style>
    .container {
      display: flex;
      flex-direction: row;
      border: 2px solid red;
      height: auto;
      width: 400px;
    }

    .item {
      background-color: lavender;
      border: 2px solid black;
      height: 50px;
      width: 150px;
      margin: 5px;
    }

    .satu {
      order: 4;
    }

  </style>
  <body>
    <div class="container">
      <div class="item satu">Box-1</div>
      <div class="item dua">Box-2</div>
      <div class="item tiga">Box-3</div>
      <div class="item empat">Box-4</div>
    </div>
  </body>
</html>
```

Hasil



Kesimpulan

Order digunakan ketika kita ingin mengatur tata letak flex item dalam flex container

FLEX-GROW

Penjelasan

Flex-Grow adalah Property CSS yang mengatur seberapa besar flex item akan memperluas dirinya dalam flex container relatif terhadap item-item lain di dalam container tersebut. Nilai dari `flex-grow` menentukan seberapa banyak ruang tambahan yang akan diambil oleh item flex dalam container jika ada ruang kosong yang tersedia setelah item-item lain telah menempati ruang mereka sendiri.

note

Tidak perlu memberikan width pada itemnya, jika kita berikan maka Flex-Grow tidak tereksekusi.

Kode Program

```
<html>
  <head>
    <title>flexbox</title>
  </head>
  <style>
    .container {
```

```
        display: flex;
        flex-direction: row;
        border: 2px solid red;
        height: auto;
        width: 400px;
    }

.item {
    display: flex;
    background-color: lavender;
    border: 2px solid black;
    height: 50px;
    margin: 5px;
}

.satu {
    flex-grow: 2;
}

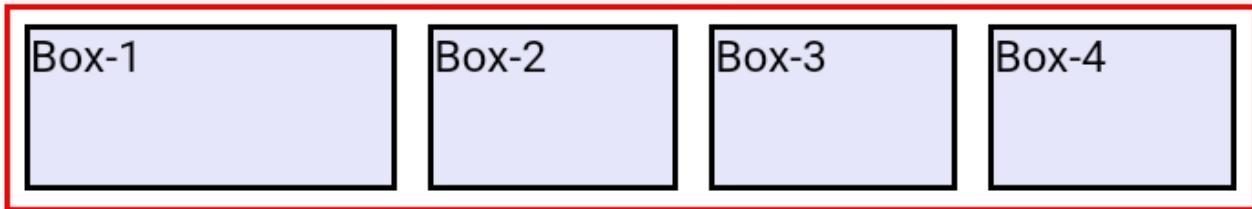
.dua {
    flex-grow: 1;
}

.tiga {
    flex-grow: 1;
}

.empat {
    flex-grow: 1;
}

</style>
<body>
<div class="container">
    <div class="item satu">Box-1</div>
    <div class="item dua">Box-2</div>
    <div class="item tiga">Box-3</div>
    <div class="item empat">Box-4</div>
</div>
</body>
</html>
```

Hasil



Kesimpulan

`Flex-Grow` digunakan ketika kita ingin memperluas salah satu flex item dalam sebuah flex container.

FLEX-SHRINK

Penjelasan

`flex-shrink` adalah properti dalam CSS yang digunakan dalam konteks Flexbox untuk mengontrol seberapa besar sebuah elemen fleksibel dapat menyusut jika tidak ada cukup ruang di dalam kontainernya. Properti ini mengatur tingkat penyusutan relatif dari elemen-elemen fleksibel dalam sebuah flex container.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Contoh Penggunaan flex-shrink</title>
<style>
    .flex-container {
        display: flex;
    }

    .flex-item {
```

```

        flex: 1;
        padding: 10px;
        border: 1px solid black;
    }

.item1 {
    flex-shrink: 1;
}

.item2, .item3 {
    flex-shrink: 2;
}

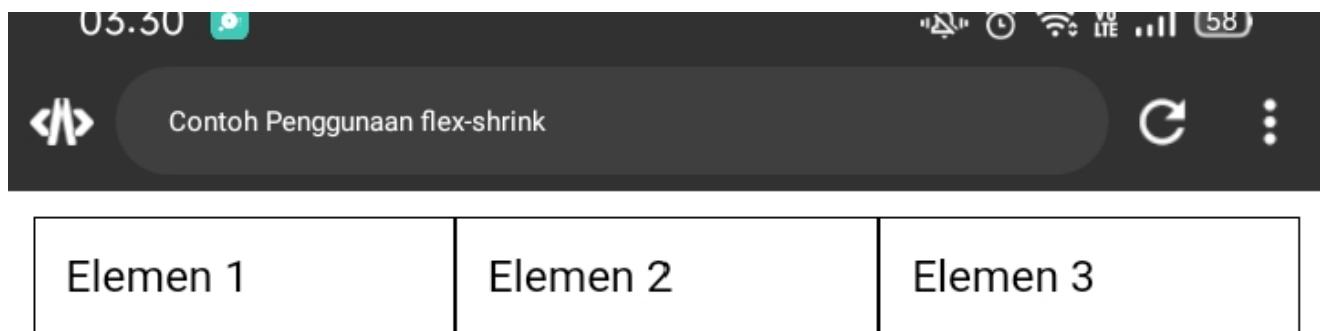
</style>
</head>
<body>

<div class="flex-container">
    <div class="flex-item item1">Elemen 1</div>
    <div class="flex-item item2">Elemen 2</div>
    <div class="flex-item item3">Elemen 3</div>
</div>

</body>
</html>

```

Hasil



Kesimpulan

Flex-shrink mengatur ukuran item ketika ruang tidak cukup.

FLEX-BASIS

Penjelasan

`flex-basis` adalah properti CSS dalam desain tata letak flexbox yang menentukan ukuran awal (basis) dari item fleksibel sebelum penyesuaian ukuran fleksibel terjadi.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Contoh Penggunaan flex-basis</title>
<style>
    .flex-container {
        display: flex;
    }

    .flex-item {
        flex-grow: 1;
        border: 1px solid black;
        margin: 5px;
    }

    .item1 {
        flex-basis: 50px;
    }

    .item2 {
        flex-basis: 70px;
    }

    .item3 {
        flex-basis: 90px;
    }
</style>
</head>
<body>

<div class="flex-container">
    <div class="flex-item item1">Elemen 1</div>
    <div class="flex-item item2">Elemen 2</div>
    <div class="flex-item item3">Elemen 3</div>
</div>

</body>
</html>
```

```
<div class="flex-item item3">Elemen 3</div>
</div>

</body>
</html>
```

Hasil



The screenshot shows a browser window with a dark header bar. On the left of the header is a code icon. In the center, the text "Contoh Penggunaan flex-basis" is displayed. On the right side of the header are a refresh icon and a three-dot menu icon. Below the header, there are three rectangular boxes labeled "Elemen 1", "Elemen 2", and "Elemen 3" from left to right, separated by thin vertical lines. All three boxes have a thin black border.

Kesimpulan

`flex-basis` menentukan ukuran awal elemen sebelum fleksbox membagikan ruang yang tersisa.

ALIGN-SELF

Penjelasan

`align-self` adalah properti CSS yang digunakan dalam konteks Flexbox untuk mengontrol penempatan vertikal individu dari elemen flex dalam container. Properti ini mengatur penempatan vertikal elemen tunggal dalam halaman, mengesampingkan nilai `align-items` yang diterapkan pada container.

Nilai yang umum digunakan untuk `align-self` adalah:

- `flex-start` : Elemen diletakkan di bagian atas container.
- `flex-end` : Elemen diletakkan di bagian bawah container.
- `center` : Elemen diletakkan di tengah container.
- `baseline` : Elemen diletakkan pada garis dasar container.
- `stretch` : Elemen diregangkan untuk mencakup tinggi container.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Contoh Penggunaan align-self</title>
<style>
    .flex-container {
        display: flex;
        height: 200px; /* Tinggi kontainer */
        border: 1px solid black;
    }

    .flex-item {
        width: 100px;
        margin: 5px;
        background-color: lightblue;
    }

    .item1 {
        align-self: flex-start;
    }

    .item2 {
        align-self: center;
    }

    .item3 {
        align-self: flex-end;
    }
</style>
</head>
<body>
<div class="flex-container">
    <div class="flex-item item1">Elemen 1</div>
    <div class="flex-item item2">Elemen 2</div>
    <div class="flex-item item3">Elemen 3</div>
</div>
</body>
</html>
```

Hasil



Elemen 1

Elemen 2

Elemen 3

Kesimpulan

Align self mengatur penempatan item secara vertikal

FLEX

Penjelasan

Properti flex memungkinkan Anda untuk secara singkat menentukan bagaimana elemen flex akan mengisi ruang dalam flex container. Dengan menggunakan flex, Anda dapat mengatur elemen flex untuk: flex-grow, flex-shrink, dan flex-basis.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Contoh Penggunaan flex</title>
<style>
    .flex-container {
        display: flex;
    }
```

```
.flex-item {  
    flex: 1;  
    border: 1px solid black;  
    margin: 5px;  
}  
</style>  
</head>  
<body>  
  
<div class="flex-container">  
    <div class="flex-item">Elemen 1</div>  
    <div class="flex-item">Elemen 2</div>  
    <div class="flex-item">Elemen 3</div>  
</div>  
  
</body>  
</html>
```

Hasil



Kesimpulan

Flex merupakan gabungan dari flex grow, flex shrink, flex basis

STUDI KASUS FLEXBOX

Deskripsi

- **container** : Mengatur sebuah kontainer (div) sebagai flex container dengan arah row (horizontal). Kontainer ini memiliki latar belakang warna lavender (background-color: lavender;), lebar 800px, dan tinggi 400px.

- `.box-1` : Mengatur kotak pertama di dalam kontainer. Kotak ini juga merupakan flex container dengan arah column (vertikal) dan memiliki beberapa properti:
 - `align-items: end;`: Mengatur item-item di dalamnya untuk diatur ke ujung kanan (vertikal).
 - `Margin` untuk mengatur jarak dari tepi kotak dan elemen-elemen di dalamnya
- `.desk` : Mengatur teks pada paragraf dengan font size 50px, menggunakan font family 'Courier New', Courier, atau monospace. Teks ini juga memiliki padding atas 5px.
- `.btn` : Mengatur tombol dengan tinggi 80px dan lebar 120px, serta memiliki border dengan gaya dashed (putus-putus) berwarna oranye (`border: 3px dashed orangered;`). Pada hover (`btn:hover`), tombol akan mengalami transformasi posisi menggunakan translate dan perubahan warna border dengan efek transisi.
- `img`: Mengatur gambar dengan tinggi 300px, lebar 300px, dan sudut melengkung sehingga tampak seperti lingkaran (`border-radius: 100%;`). Gambar ini juga memiliki border solid berwarna crimson (`border: 3px solid crimson;`) dan margin atas 40px.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <link rel="stylesheet" href="style.css">
  <style>
.container {
  display: flex;
  flex-direction: row;
  background-color:lavender;
  width: 800px;
  height: 400px;
}

.box-1 {
  display: flex;
  flex-direction: column;
  align-items: end;
  margin-right: 50px;
  margin-left: 10px;
  margin-top: 15px;
}

.desk {
  padding-top: 5px;
  font-size: 50px;
  font-family: 'Courier New', Courier, monospace;
}
```

```

 {
    font-weight: bold;
}

.btn {
    height: 80px;
    width: 120px;
    border: 3px dashed orangered;

}

.btn:hover {
    transform: translate(-170px, -60px);
    transition: all 0.5s ease-in-out 0.3s;
    border: 3px solid orangered;
}

img {
    height: 300px;
    width: 300px;
    border-radius: 100%;
    border: 3px solid crimson;
    margin-top: 40px;
}

```

</style>

</head>

<body>

```

<div class="container">
    <div class="box-1">
        <p class="desk">Selamat datang<br>di<span> Web Adiguna</span>
    </p>
        <button class="btn">Klik disini</button>
    </div>
        
    </div>

```

</body>

</html>

Hasil

Selamat datang di **Web Adiguna**



Position

Position Relative

Penjelasan

Position Relative adalah posisi mirip seperti static position, dimana element akan ditempatkan sesuai normal flow, ketika elemen tersebut mau di pindahkan dia akan pindah dari posisi awalnya.

Kode Program

```
<html>
  <head>
    <title>position</title>
    <style>
      .container {
        display: flex;
        flex-direction: row;
        width: 150px;
        height: 150px;
        background-color: red;
      }

      .item1 {
        width: 50px;
        height: 50px;
        background-color: aqua;
      }
    </style>
  </head>
  <body>
    <div class="container">
      <div class="item1"></div>
    </div>
  </body>
</html>
```

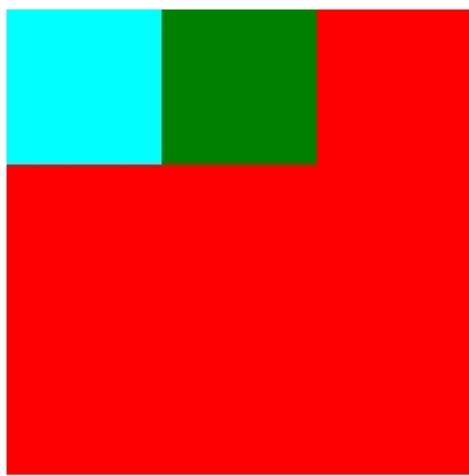
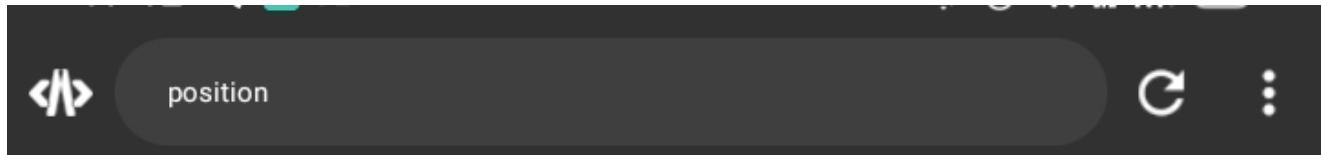
```
}

.item2 {
    width: 50px;
    height: 50px;
    background-color: green;
    position: relative;
    left: 30px;
}

</style>
</head>
<body>
    <div class="container">
        <div class="item1"></div>
        <div class="item2"></div>
    </div>
</body>
</html>
```

Hasil

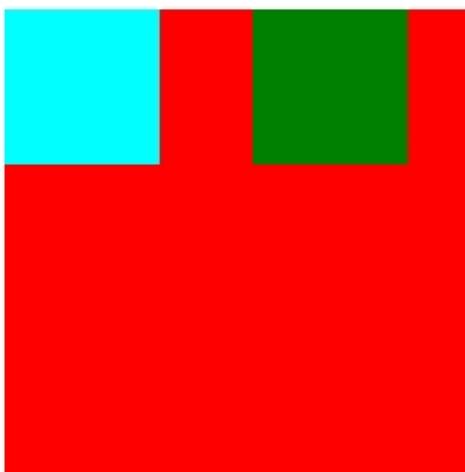
Before



After



position



Kesimpulan

Position relative berpindah dari tempat awalnya yang akan mengganggu posisi elemen lain.

Position Absolute

Penjelasan

Posisi absolute dalam CSS mengarahkan sebuah elemen untuk ditempatkan secara independen di dalam elemen tertentu, baik itu elemen yang berposisi relatif atau seluruh halaman (viewport) jika tidak ada elemen yang berposisi relatif.

Kode Program

```
<html>
  <head>
    <title>position</title>
    <style>
      .container {
        display: flex;
        flex-direction: row;
        width: 150px;
        height: 150px;
        background-color: red;
      }
```

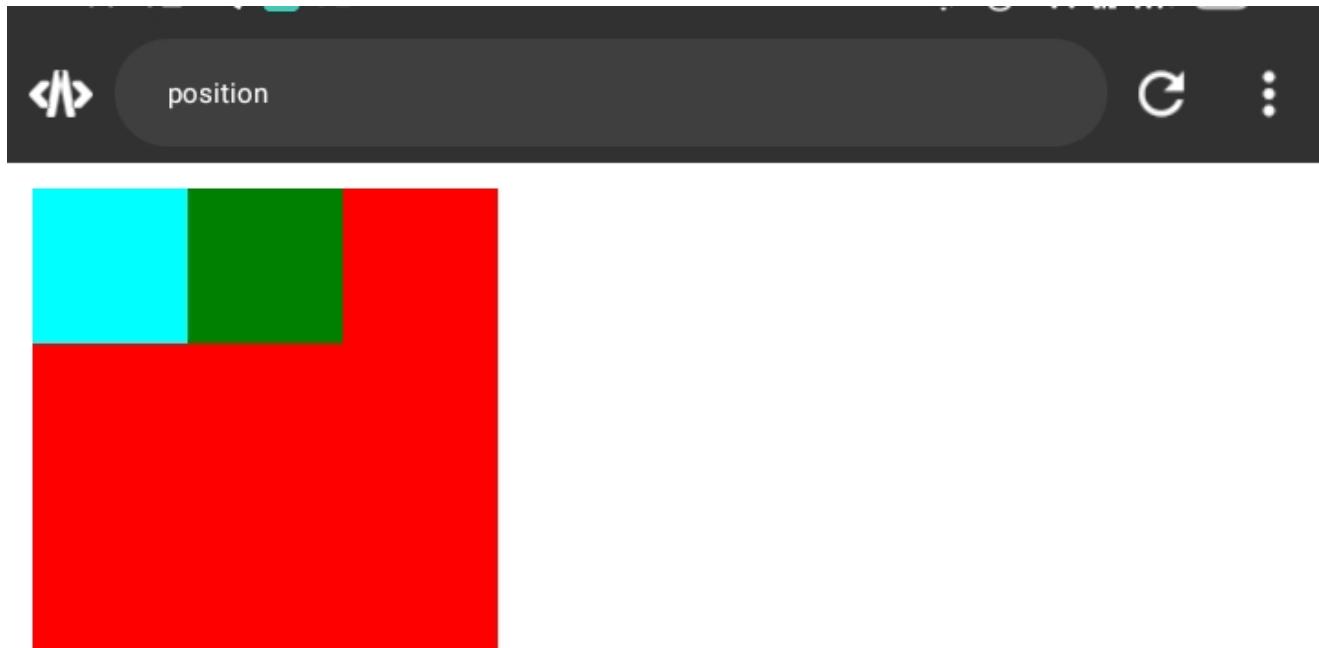
```
.item1 {
    width: 50px;
    height: 50px;
    background-color: aqua;
}

.item2 {
    width: 50px;
    height: 50px;
    background-color: green;
    position: absolute;
    left: 30px;
}

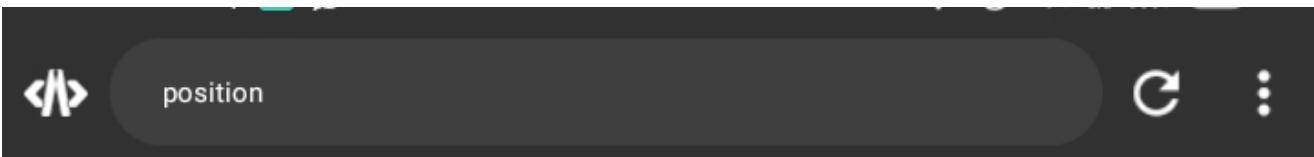
</style>
</head>
<body>
    <div class="container">
        <div class="item1"></div>
        <div class="item2"></div>
    </div>
</body>
</html>
```

Hasil

Before



After



Kesimpulan

Position absolute dia berpindah mengikuti parentnya, tanpa mempengaruhi tata letak elemen-elemen lain.

Position Fixed

Penjelasan

Position Fixed ketika suatu elemen memiliki properti position yang di atur menjadi fixed maka elemen tersebut akan terus berada dalam viewport yang sudah ditentukan, bahkan saat halaman di gulir.

Kode Program

```
<html>
  <head>
    <title>position</title>
    <style>
      .container {
        width: 100vw;
        height: 150px;
        background-color: red;
        border: 4px solid black;
        position: fixed;
      }

      .item {
        width: 50px;
        height: 50px;
        background-color: cyan;
        position: absolute;
        top: 10px;
        left: 10px;
      }
    </style>
  </head>
  <body>
    <div class="container">
      <div class="item"></div>
    </div>
  </body>
</html>
```

```
width:100vw;
height:300px;
background-color: aqua;
}

.item2 {
width:100vw;
height:300px;
background-color: black;
}
</style>
</head>
<body>
<div class="container">FIXED</div>
<div class="item"></div>
<div class="item2"></div>
</body>
</html>
```

Hasil



position



FIXED



Kesimpulan

Position Fixed digunakan ketika ada elemen, kita ingin dia berada dalam viewport terus.

Position Sticky

Penjelasan

Position Sticky gabungan dari relative dan Fixed , elemen yang di berikan position Sticky akan berperilaku relative sampe dengan titik yang ditentukan baru berubah menjadi fixed.

Kode Program

```
<html>
  <head>
    <title>position</title>
    <style>
      .sticy {
        background-color: red;
        width: 100vw;
        position:sticky;
        top: 50px;
      }

    </style>
  </head>
  <body>
    <h2>SELEMAT DATANG</h2>
    <h2>SELEMAT DATANG</h2>
    <h2 class="sticy">SELEMAT DATANG</h2>
    <h2>SELEMAT DATANG</h2>
  </body>
</html>
```

Hasil

Before



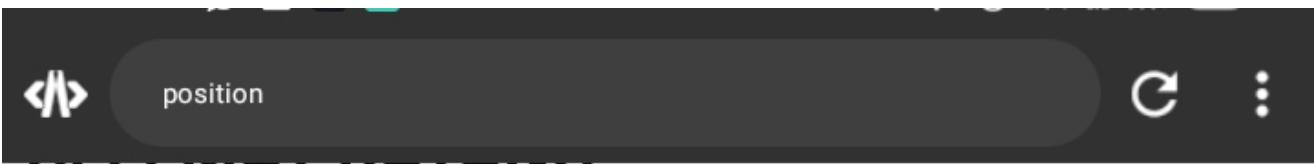
SELEMAT DATANG

SELEMAT DATANG

SELEMAT DATANG

SELEMAT DATANG

After



SELEMAT DATANG

Kesimpulan

Sticky digunakan ketika ada suatu elemen yang terdapat dalam baris, akan tetapi kita mau dia berada dalam viewport terus menerus tanpa harus ikut ke scroll.

STUDI KASUS POSITION

Deskripsi

- `body` : Mengatur tampilan body halaman dengan properti `display: flex;` untuk mengatur flexbox, `justify-content: center;` untuk memusatkan konten secara horizontal, dan `background-color: blue;` untuk memberikan warna latar belakang biru.
- `container` : Merupakan div utama yang berfungsi sebagai container di berikan properti `display: flex;` dengan `flex-direction: column;` untuk mengatur layout ke arah vertikal (kolom), `height: 500px;` untuk menetapkan tinggi, dan `width: 350px;` untuk menetapkan lebar kontainer. `margin-top: 50px;` digunakan untuk memberikan margin atas
- `item1` : Bagian pertama dari kontainer, diberikan tinggi `height: 250px;`.
- `item2` : Bagian kedua dari kontainer, berisi teks dan informasi. Diberikan latar belakang putih (`background-color: white;`), tinggi `height: 190px;` dan padding kiri `padding-left: 20px;`
- `gambar` : Mengatur gambar (ikon) di dalam `.item2` dengan `border-radius: 50%;` untuk membuat sudut melengkung, serta properti `position: relative;`, `left: 270px;`, dan `top: -17px;` untuk mengatur posisi ikon secara spesifik.
- `tanggal`, `.judul`, `.catatan` : Mengatur gaya teks pada `.item2` dengan ukuran font, warna, dan margin yang berbeda-beda.
- `item3` : Bagian ketiga dari kontainer, merupakan baris (row) yang menampilkan tombol "Read More" dan sebuah ikon. Diberikan `display: flex;` dengan `flex-direction: row;`, `justify-content: space-between;` untuk memberikan ruang kosong antara elemen di dalamnya, serta `align-items: center;` untuk memusatkan secara vertikal. Latar belakang bagian ini diatur sebagai `background-color: rgb(160, 170, 160);` dengan tinggi `height: 50px;`
- `.text`, `.icon` : Mengatur gaya teks dan ikon di dalam `.item3` dengan properti padding, ukuran font, dan bentuk ikon yang memiliki sudut melengkung (`border-radius: 50%;`).

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Document</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            background-color: blue;
        }
    </style>
</head>
<body>
    <div>
        <div>
            <h1>Hello World!</h1>
            <p>This is a simple example of a web page using CSS Positioning.</p>
            <img alt="Icon of a person" style="border-radius: 50%; width: 50px; height: 50px;"/>
            <div>
                <span>Read More</span>
                <span>Read More</span>
            </div>
        </div>
    </div>
</body>
</html>
```

```
.container {
    display: flex;
    flex-direction: column;
    height: 500px;
    width: 350px;
    margin-top: 50px;
}

.item1 {
    height: 250px;
}

.item2 {
    background-color:white;
    height: 190px;
    padding-left: 20px;
}

.gambar {
    border-radius: 50%;
    position: relative;
    left: 270px;
    top: -17px;
}

.tanggal {
    font-size: 15px;
    color: rgb(109, 111, 112);
    margin-top: -25px;
}

.judul {
    font-size: 30px;
    font-weight: bold;
    margin-top: -5px;
}

.catatan {
    font-size: 15px;
    margin-top: -15px;
}

.item3 {
    display: flex;
    flex-direction: row;
    justify-content: space-between;
    align-items: center;
    background-color:rgb(160, 170, 160);
```

```

        height: 50px;
    }

    .text {
        padding-left: 20px;
        font-size: 18px;
    }

    .icon {
        border-radius: 50%;
        padding-right: 10px;
    }


```

</style>

</head>

<body>

<div class="container">

<div class="item1">

</div>

<div class="item2">

aenean eu leo quam.</p>

</div>

<div class="item3">

</div>

</div>

</body>

</html>

Hasil



Thursday, July 16, 2015

The standard chunk of Lorem Ipsum

Sed posuere consectetur est at lobortis. aenean eu leo quam.

Read More