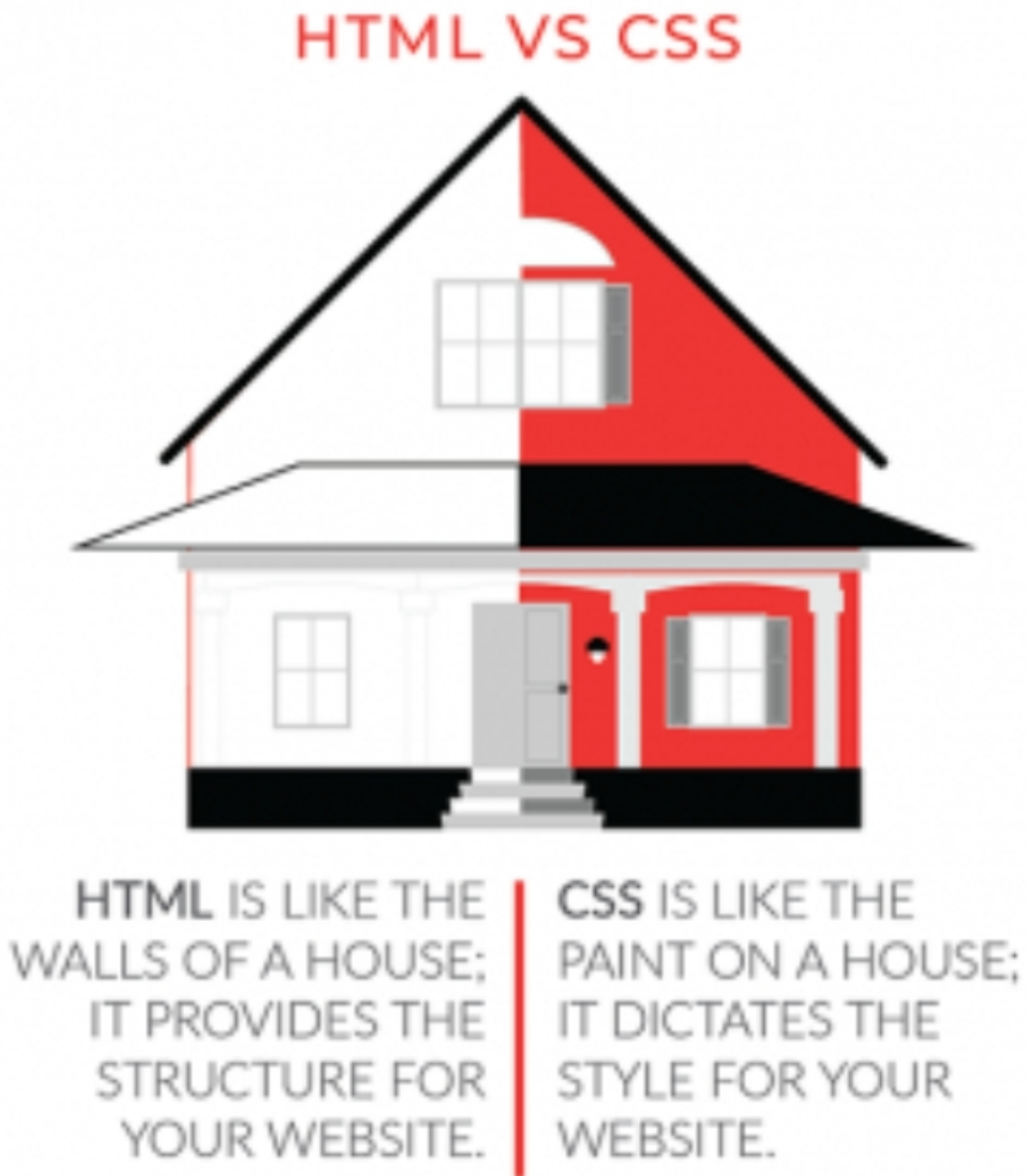


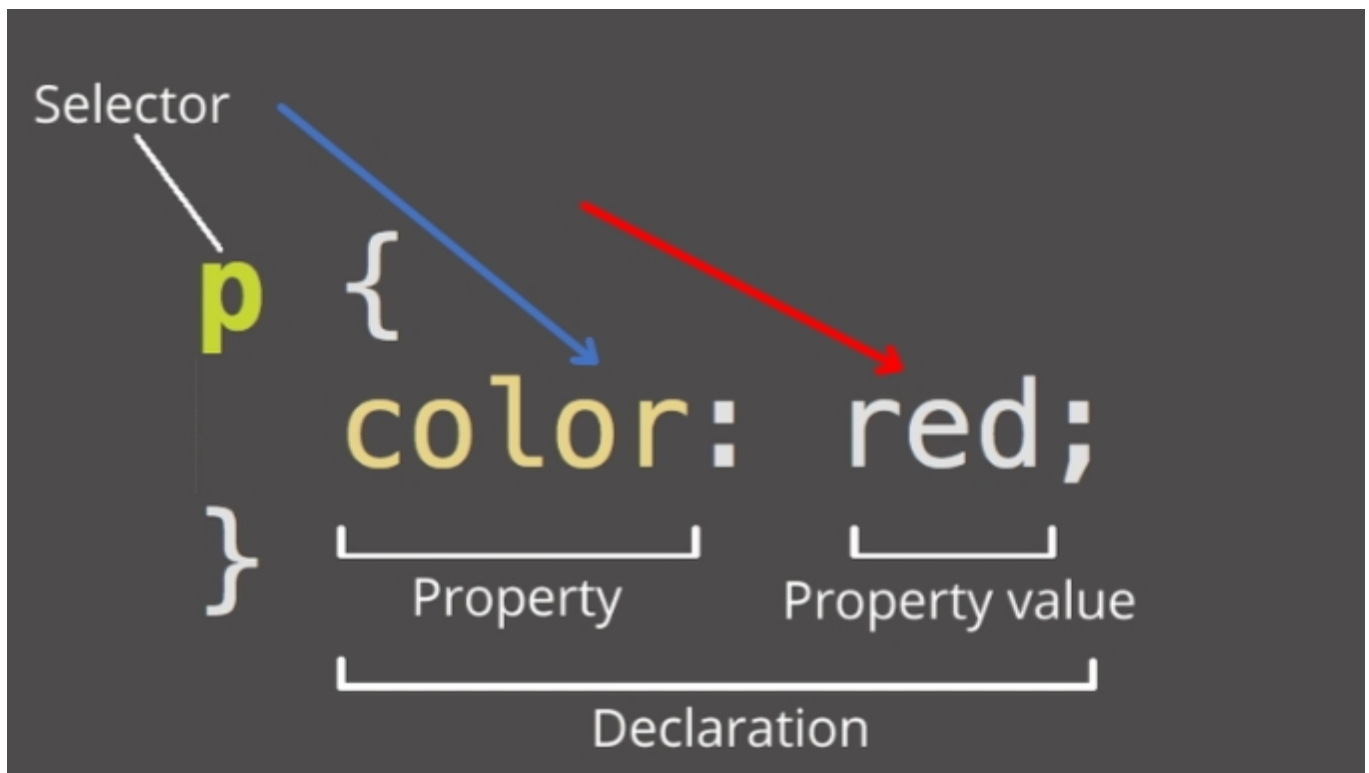
Anatomi CSS



Selector : pinggiran jendela,dinding,atap(seng),tangga,tiang penahan,lampu (dekat pintu)

property: color

property value:warna pada pinggiran jendela,dinding,atap(seng),tangga,tiang penahan di ubah menjadi warna merah,hitam,abu-abu,abu-abu gelap



Contoh diatas,yang ingin dimodifikasi adalah seluruh tag `<p>` pada komponen warna teksnya menjadi warna merah

Percobaan Pertama

Kode Program

```
<!DOCTYPE html>

<html>

  <head>

    <style>

      p{

        color:red;

      }

    </style>

    <title>DIV-SPAN</title>

  </head>
```

```
<body>

    <p>welcome css</p>

</body>

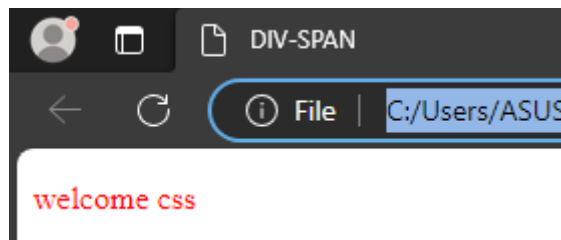
</html>
```

Penjelasan

Baris pertama (`p`) adalah selector CSS, yang menargetkan semua elemen `<p>`. Di dalam blok kurung tersebut, kita memiliki properti `color` yang ditetapkan nilainya menjadi `red`. Artinya, teks dalam semua elemen `<p>` akan ditampilkan dengan warna merah. `<style>`: Bagian di mana Anda dapat menambahkan aturan CSS untuk mengubah tampilan elemen HTML di halaman.

`p { color: red; }`: Aturan CSS yang mengubah warna teks pada semua elemen `<p>` menjadi merah.

Hasil



Percobaan Kedua

Kode CSS

```
button{

    width: 150px;

    height: 50px;

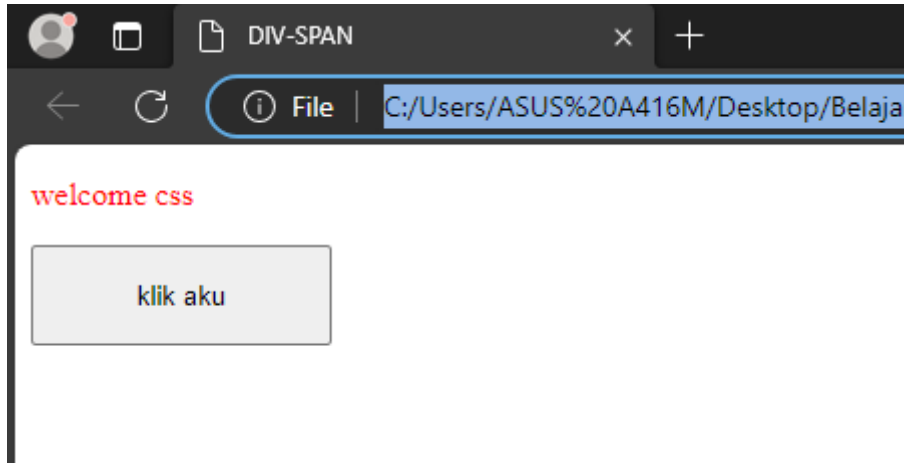
    color: aqua;

    background-color: #7949ff;
```

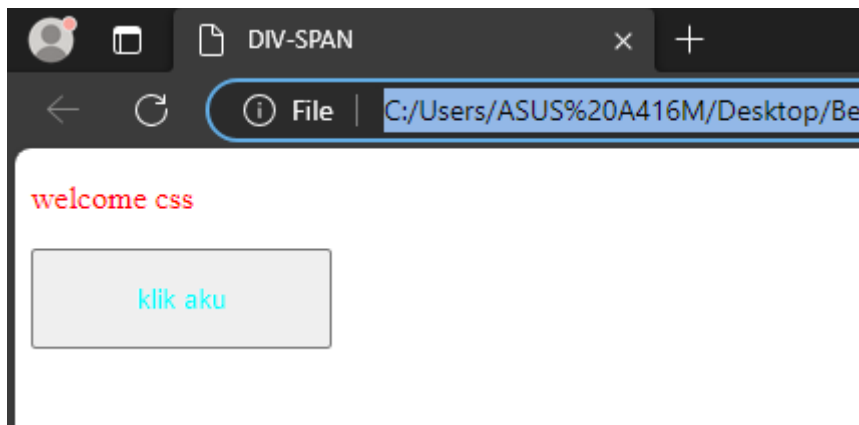
text-align:

Color

Before



After

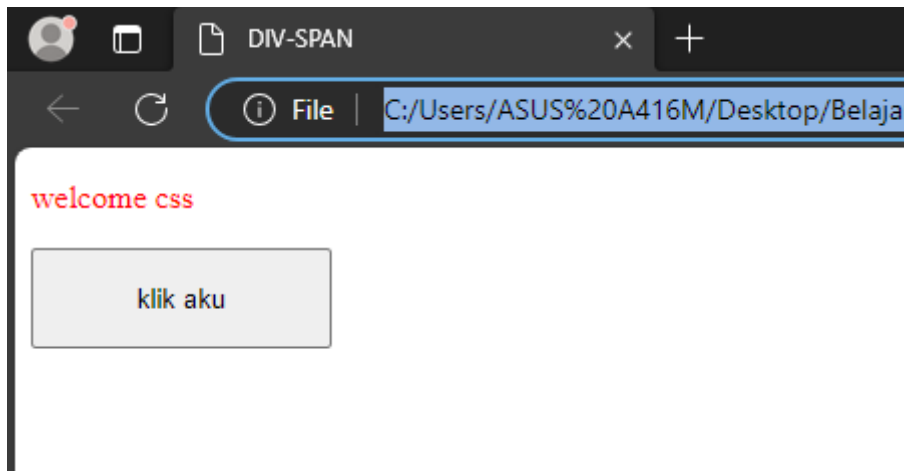


Penjelasan

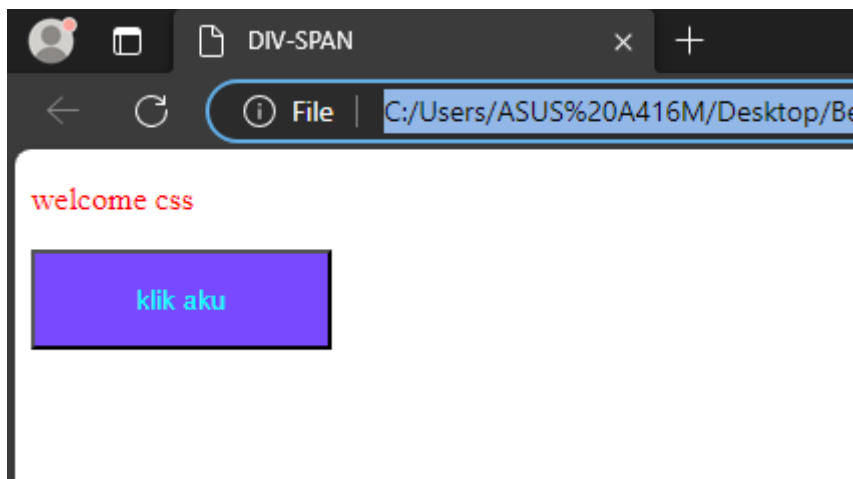
Memberikan warna pada teks `klik aku` dengan warna `aqua`

Background-color

Before



After

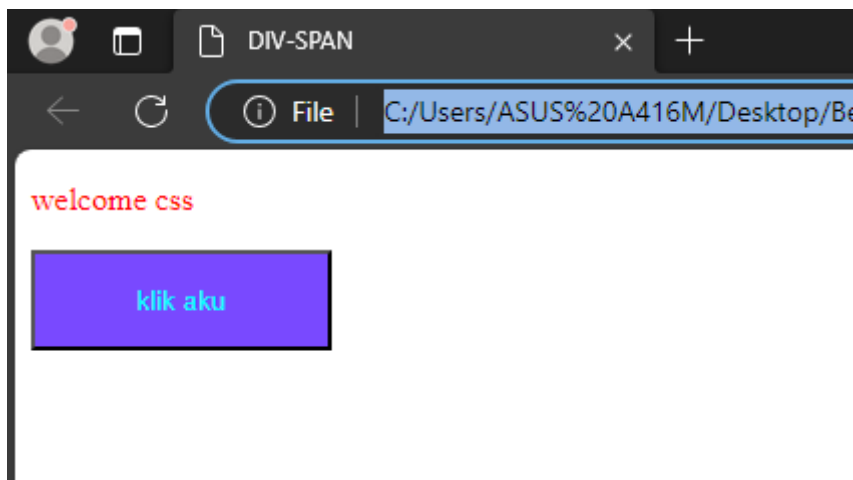


Penjelasan

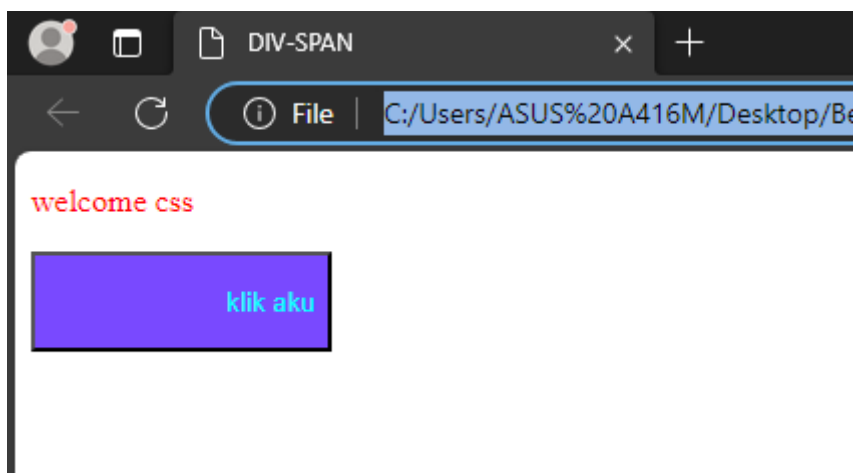
Memberikan warna pada latar belakang (`background-color`) dengan warna `ungu`

Text-align

Before



After



Penejelasan

Teks `klik aku` berpindah posisi karena memakai perintah `text-align: right`

DEKLARASI CSS

IN-LINE

Penggunaan Inline CSS: dideklarasikan langsung pada tag yang ingin di modifikasi.

CSS dapat dimasukkan langsung ke dalam atribut style pada elemen HTML. Ini adalah cara paling langsung untuk menerapkan gaya pada suatu elemen.

Contoh:

```
<p style="font-size: 48px;">welcome css</p>
```

INTERNAL

Penggunaan Internal CSS: dideklarasikan pada file yang sama.

CSS dapat ditulis di dalam tag `<style>` di bagian `<head>` dari dokumen HTML. Ini memungkinkan Anda untuk mendefinisikan gaya untuk seluruh halaman web.

Contoh

```
<!DOCTYPE html>

<html>

  <head>

    <style>

      p{

        color:red

      }

      button{

        width: 150px;

        height: 50px;

        color: aqua;

        background-color: #7949ff;

        text-align: right;

      }

    </style>

    <title>DIV-SPAN</title>

  </head>
<title>DIV-SPAN</title>

</head>
```

```
<body>

  <p>welcome css</p>

  <button>klik aku</button>

</body>
```

EXTERNAL

Penggunaan External CSS: dideklarasikan pada file yang berbeda dan dipanggil menggunakan tag `<link>` .

CSS dapat disimpan di file terpisah dengan ekstensi `.css` dan kemudian dihubungkan ke dalam dokumen HTML menggunakan tag `<link>` . Ini memungkinkan Anda untuk memisahkan struktur dan gaya dalam pengembangan web.

Contoh

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <title>Document</title>

    <link rel="stylesheet" href="p.css">

  </head>

  <body>

    <p style="font-size: 48px;">welcome css</p>

    <h1>AND THE GANK</h1>

    <P class="MERAH"> INI WARNA MERAH</P>
```



```
<P class="BIRU teks-besar"> INI WARNA BIRU</P>
```

```
<P class="kuning poke"> INI WARNA kuning</P>
```

```
<p id="HIJAU">ini warna hijau</p>
```

```
</body>
```

```
</html>
```

```
h1{  
  
    color: darkseagreen;  
  
    font-size: 100px;  
  
    font-weight: bold;  
  
}
```

```
.MERAH {  
  
    color: red;  
  
}
```

```
.BIRU {  
  
    color: blue;  
  
}
```

```
.kuning {  
  
    color: yellow;  
  
}
```

```
#HIJAU{  
  
    color: green;
```

```
}

.teks-besar{

    color: gold;

}

.poke {

    font-size: 40px;

}
```

SELECTOR CSS

Elemen

```
p{

    color: red;

}
```

- *Penjelasan:* Dalam contoh ini, kita memiliki sebuah selektor elemen p. Ini berarti kita sedang memilih semua elemen paragraf dalam dokumen. Gaya CSS yang didefinisikan akan diterapkan pada semua elemen tersebut. Dalam kasus ini, elemen-elemen tersebut akan memiliki ukuran font 16 piksel dan jarak antar baris (line-height) sebesar 1.5.

ID

```
#css {

    color: red;

}
```

- *Penjelasan:* Dalam contoh ini, kita memiliki sebuah selektor ID `#css`. Ini berarti kita sedang memilih elemen dengan ID "header". Gaya CSS yang didefinisikan akan diterapkan pada elemen tersebut. Dalam kasus ini, elemen tersebut akan memiliki latar belakang berwarna biru muda, teks berwarna putih, dan padding sebesar 10 piksel.

Class

```
.css {  
    color: white;  
}
```

- *Penjelasan:* Dalam contoh ini, kita memiliki sebuah selektor class `.button`. Ini berarti kita sedang memilih semua elemen yang memiliki kelas "button". Gaya CSS yang didefinisikan akan diterapkan pada semua elemen tersebut. Dalam kasus ini, elemen-elemen tersebut akan memiliki latar belakang hijau, teks berwarna putih, padding sebesar 8 piksel di atas dan bawah serta 16 piksel di kiri dan kanan, tidak ada border, dan sudut-sudut yang diperbulatkan
- Elemen Selector : https://www.w3schools.com/cssref/sel_element.php
- Class Selector : https://www.w3schools.com/cssref/sel_class.php
- ID Selector : https://www.w3schools.com/cssref/sel_id.php

Property Dasar

Materi TEXT

Kode HTML & CSS

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
    <title>text</title>  
  
    <link rel="stylesheet" href="text.css">  
  
</head>  
  
<body>  
  
    <p class="oi">INSTAGRAM</p>  
  
    <p class="p">INSTAGRAM</p>  
  
    <p class="oo">INSTAGRAM</p>  
  
</body>
```

```
</html>
```

```
.oi {  
  
    text-decoration: overline;  
  
}  
  
.p {  
  
    text-decoration: underline;  
  
}  
  
.oo {  
  
    text-decoration: line-through;  
  
}
```

Text-Decoration

Overline

Penjelasan

Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "oi" dan mengatur efek dekorasi teks menjadi garis di atas teks (overline).

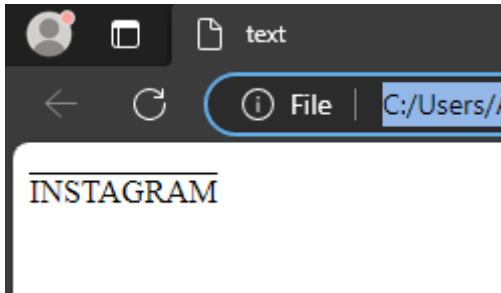
Dalam contoh ini, kita menggunakan CSS untuk menambahkan efek overline pada teks paragraf dengan kelas "oi". Dengan demikian, teks "INSTAGRAM" akan ditampilkan dengan garis di atasnya.

Kode Program

```
<p class="oi">INSTAGRAM</p>
```

```
.oi {  
  
    text-decoration: overline;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa CSS memungkinkan pengembang web untuk memberikan tampilan yang lebih menarik dan kreatif pada teks di situs web mereka. Dengan menggunakan properti seperti `text-decoration`, efek visual seperti garis di atas teks dapat dengan mudah dicapai. Hal ini memungkinkan pengembang untuk meningkatkan estetika situs web mereka dan membuatnya lebih menarik bagi pengguna.

Underline

Penjelasan

Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "p" dan mengatur efek dekorasi teks menjadi garis di bawah teks (underline).

Dalam contoh ini, kita menggunakan CSS untuk menambahkan efek underline pada teks paragraf dengan kelas "p". Sehingga teks "INSTAGRAM" akan ditampilkan dengan garis di bawahnya.

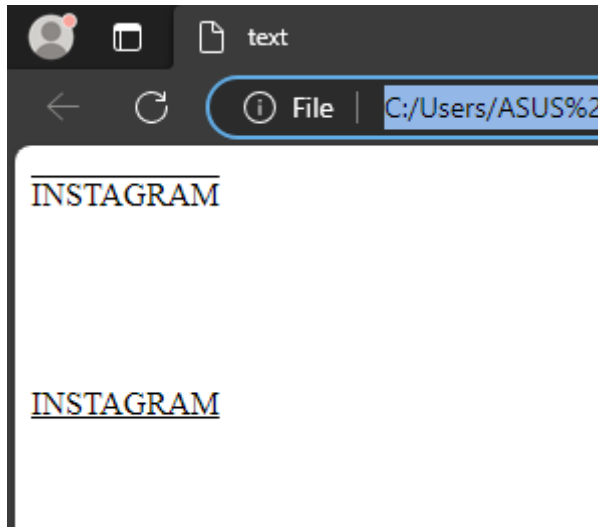
Kode program

```
<p class="p">INSTAGRAM</p>
```

```
.p {  
  
    text-decoration: underline;  
  
}
```

```
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa CSS memungkinkan pengembang web untuk memberikan tampilan yang lebih menarik dan kreatif pada teks di situs web mereka. Dengan menggunakan properti seperti text-decoration, efek visual seperti garis di atas teks dapat dengan mudah dicapai. Hal ini memungkinkan pengembang untuk meningkatkan estetika situs web mereka dan membuatnya lebih menarik bagi pengguna.

line-through

Penjelasan

Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "oo" dan mengatur efek dekorasi teks menjadi garis melintang (line-through). Ini adalah contoh sederhana dari penggunaan CSS untuk mengubah tampilan teks pada elemen HTML.

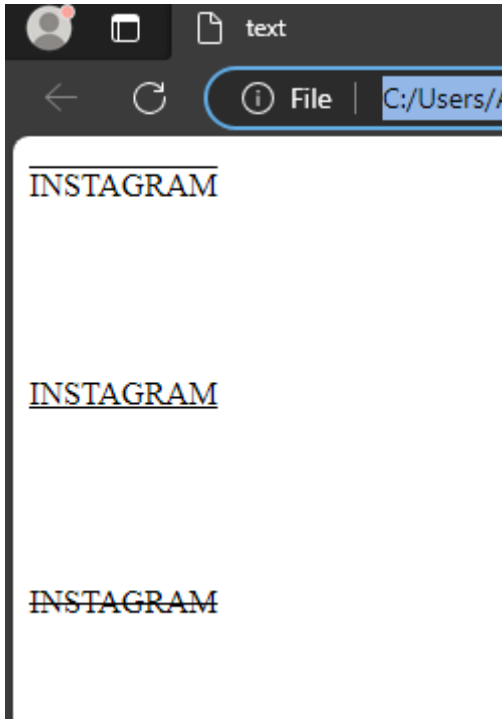
Dalam contoh ini, kita menggunakan CSS untuk menambahkan efek line-through pada teks paragraf dengan kelas "oo". Sehingga teks "INSTAGRAM" akan ditampilkan dengan garis melintang di atasnya.

Kode program

```
<p class="oo">INSTAGRAM</p>
```

```
.oo {  
  
    text-decoration: line-through;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa CSS memungkinkan pengembang web untuk memberikan tampilan yang lebih menarik dan kreatif pada teks di situs web mereka. Dengan menggunakan properti seperti `text-decoration`, efek visual seperti garis melintang dapat dengan mudah dicapai. Ini membuka peluang untuk meningkatkan estetika situs web dan membuatnya lebih menarik bagi pengguna.

Text-Align

Center

Penjelasan

1. `text-align: center;` adalah properti CSS yang digunakan untuk mengatur penempatan horizontal teks di dalam sebuah elemen. Ketika Anda menerapkan `text-align: center;`

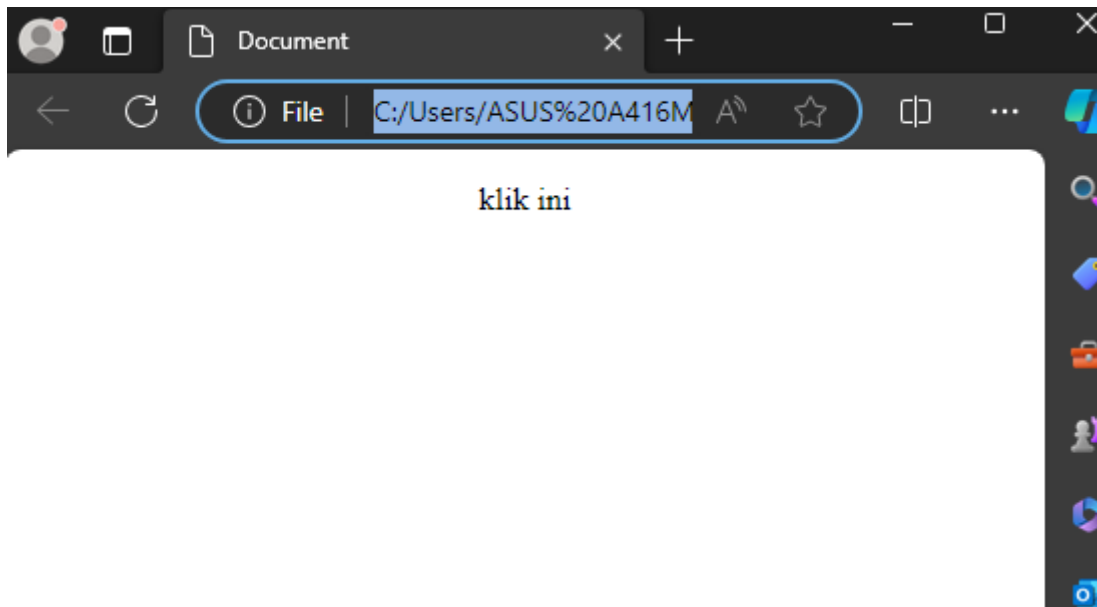
pada sebuah elemen, teks di dalamnya akan ditampilkan dengan poros pusat horizontal yang berada di tengah dari elemen tersebut

2. `text-align: right;` : Ini adalah properti CSS yang mengatur penataan teks di dalam elemen HTML agar rata kanan. Artinya, teks dalam elemen tersebut akan disusun mulai dari sisi kanan elemen ke arah kiri.
3. `text-align: left;` : Properti ini mengatur penataan teks agar rata kiri di dalam elemen HTML. Dengan demikian, teks akan disusun dari sisi kiri elemen ke arah kanan.
4. `text-align: justify;` : Properti ini mengatur penataan teks agar rata kanan dan kiri di dalam elemen HTML, sehingga teks akan berada di kedua sisi dengan rapi, dan jarak antar kata akan disesuaikan untuk mengisi seluruh lebar elemen..

Kode program

```
.oi {  
  
    text-align: center;  
  
}
```

Hasil



Kesimpulan

Menetapkan properti `text-align` dengan nilai `center` untuk semua elemen yang memiliki kelas `oi`. Dengan demikian, semua teks di dalam elemen dengan kelas `oi` akan diatur menjadi rata tengah secara horizontal.

Text-Transform

Penjelasan

Pernyataan tersebut menggunakan properti `text-transform` untuk mengatur transformasi teks pada elemen-elemen yang memiliki kelas ".oi". Properti `text-transform` digunakan untuk mengubah tampilan teks, dan dalam hal ini, nilai properti tersebut adalah "capitalize". Ini berarti teks dalam elemen-elemen dengan kelas ".oi" akan diubah sehingga setiap kata diawali dengan huruf kapital.

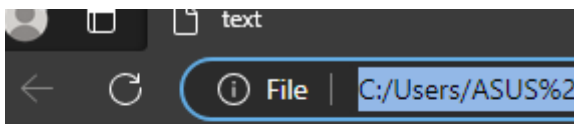
contoh lainnya :

1. `text-transform: lowercase;` : Properti ini mengatur teks di dalam elemen HTML agar diubah menjadi huruf kecil semua. Artinya, semua huruf dalam teks akan ditampilkan dalam bentuk huruf kecil.
2. `text-transform: uppercase;` : Properti ini mengatur teks di dalam elemen HTML agar diubah menjadi huruf besar semua. Artinya, semua huruf dalam teks akan ditampilkan dalam bentuk huruf besar.
3. `text-transform: none;` : Properti ini mengatur teks di dalam elemen HTML agar tidak mengalami transformasi apapun. Ini berarti teks akan ditampilkan sesuai dengan apa yang telah dituliskan, tanpa perubahan bentuk huruf.

Kode Program

```
.oi {  
  
    text-transform: capitalize;  
  
}
```

Hasil



Instagram

Kesimpulan

Kesimpulan: Program tersebut akan mengubah tampilan teks pada elemen-elemen yang memiliki kelas ".oi" agar setiap kata di dalamnya diawali dengan huruf kapital. Misalnya, jika Anda memiliki elemen HTML seperti `<p class="oi">instagram</p>`, maka setelah penerapan CSS tersebut, teks akan berubah menjadi "Instagram".

Text-Indent

Penjelasan

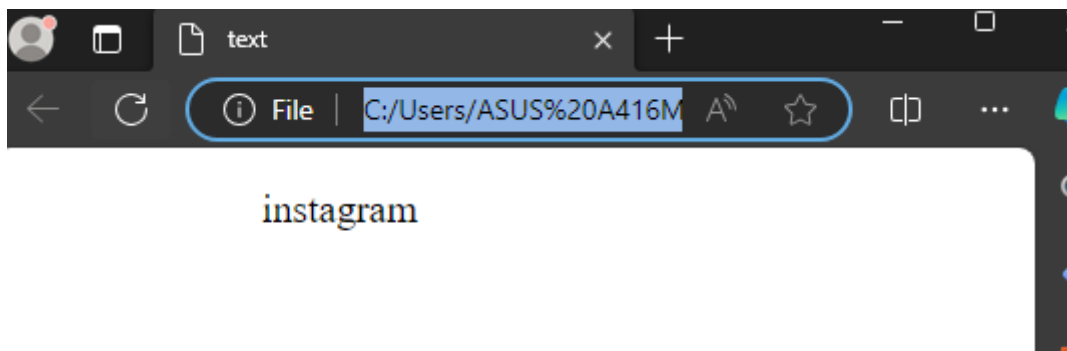
Pernyataan tersebut menggunakan properti `text-indent` untuk mengatur indentasi (pergeseran ke kanan) pada teks dalam elemen-elemen yang memiliki kelas ".oi". Properti `text-indent` biasanya digunakan untuk menentukan jarak awal paragraf atau baris pertama dari suatu teks.

Dalam hal ini, nilai properti `text-indent` adalah "100px", yang berarti teks dalam elemen-elemen dengan kelas ".oi" akan diindentasi sejauh 100 piksel dari tepi kiri halaman atau kontainer yang mengandungnya.

Kode Program

```
.oi {  
  
    text-indent: 100px;  
  
}
```

Hasil



Kesimpulan

Program tersebut akan mengatur indentasi teks pada elemen-elemen yang memiliki kelas ".oi" sehingga teks di dalamnya akan dimulai dari jarak 100 piksel dari tepi kiri halaman atau

kontainer yang mengandungnya. Misalnya, jika Anda memiliki elemen HTML seperti `<div class="oi">Lorem ipsum dolor sit amet</div>`, maka teks di dalamnya akan dimulai 100 piksel dari tepi kiri.

Letter Spacing

Penjelasan

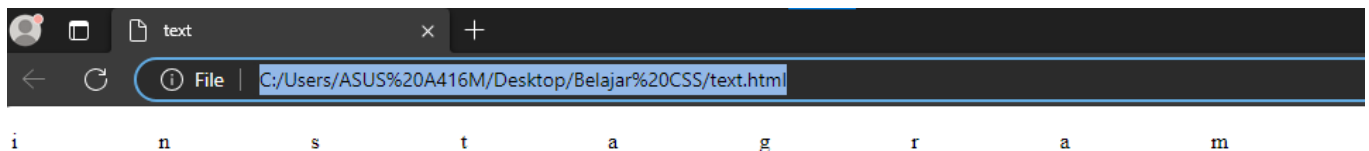
Pernyataan tersebut menggunakan properti `letter-spacing` untuk mengatur jarak antar huruf dalam teks di dalam elemen-elemen yang memiliki kelas ".oi". Properti `letter-spacing` digunakan untuk menentukan jarak horizontal antara huruf-huruf dalam teks.

Dalam hal ini, nilai properti `letter-spacing` adalah "100px", yang berarti jarak antar huruf dalam teks di dalam elemen-elemen dengan kelas ".oi" akan diperbesar menjadi 100 piksel.

Kode Program

```
.oi {  
  
    letter-spacing: 100px;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengatur jarak antar huruf dalam teks pada elemen-elemen yang memiliki kelas ".oi" sehingga jarak antar hurufnya menjadi lebih lebar sebesar 100 piksel. Misalnya, jika Anda memiliki elemen HTML seperti `<div class="oi">Hello</div>`, maka teks di dalamnya akan ditampilkan dengan jarak antar huruf yang lebih lebar dari biasanya.

Line-Height

Penjelasan

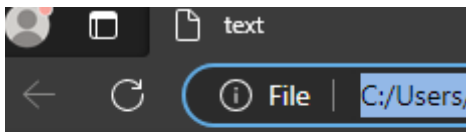
Pernyataan tersebut menggunakan properti `line-height` untuk mengatur tinggi baris dalam teks di dalam elemen-elemen yang memiliki kelas ".oi". Properti `line-height` digunakan untuk menentukan tinggi baris dalam sebuah elemen teks.

Dalam hal ini, nilai properti `line-height` adalah "100px", yang berarti setiap baris teks di dalam elemen-elemen dengan kelas ".oi" akan memiliki tinggi 100 piksel.

Kode Program

```
.oi {  
  
    line-height: 100px;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengatur tinggi baris dalam teks pada elemen-elemen yang memiliki kelas ".oi" sehingga setiap baris teks memiliki tinggi sebesar 100 piksel.

Misalnya, jika Anda memiliki elemen HTML seperti `<div class="oi">Lorem ipsum dolor sit amet</div>`, maka setiap baris teks di dalamnya akan memiliki tinggi 100 piksel.

Word-Spacing

Penjelasan

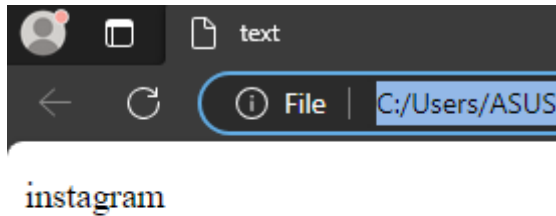
Pernyataan tersebut menggunakan properti `word-spacing` untuk mengatur jarak antar kata dalam teks di dalam elemen-elemen yang memiliki kelas ".oi". Properti `word-spacing` digunakan untuk menentukan jarak horizontal antara kata-kata dalam teks.

Dalam hal ini, nilai properti `word-spacing` adalah "100px", yang berarti jarak antar kata dalam teks di dalam elemen-elemen dengan kelas ".oi" akan diperbesar menjadi 100 piksel.

Kode Program

```
.oi {  
  
    word-spacing: 100px;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengatur jarak antar kata dalam teks pada elemen-elemen yang memiliki kelas ".oi" sehingga jarak antar kata menjadi lebih lebar sebesar 100 piksel. Misalnya, jika Anda memiliki elemen HTML seperti `<div class="oi">Hello World</div>`, maka teks di dalamnya akan ditampilkan dengan jarak antar kata yang lebih lebar dari biasanya.

MATERI FONT

Kode HTML & CSS

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
    <title>Font</title>  
  
    <link rel="stylesheet" href="Font.css">
```

```
</head>

<body>

  <p class="po">tes</p>

  <p class="poo">ftes</p>

  <p class="pooo">ftes</p>

  <p class="poooo">ftes</p>


</body>

</html>
```

```
.po {

  font-size: 100px;

}


.poo {

  font-weight: bolder;

}


.pooo {

  font-style: italic;

}


.poooo {

  font-family: 'Courier New', Courier, monospace;
```

```
}
```

Font-Size

Penjelasan

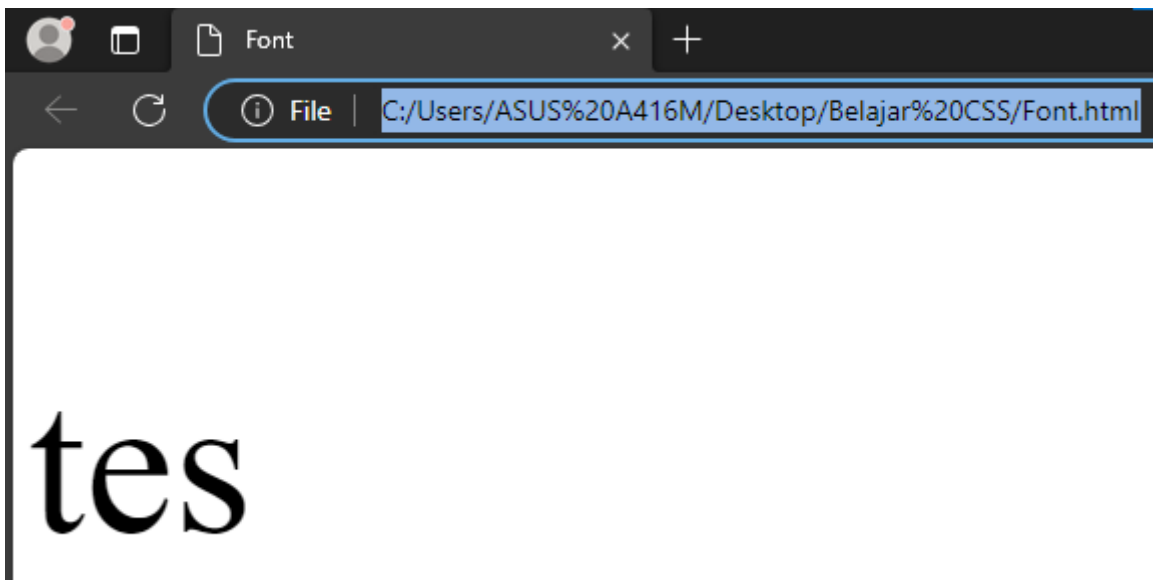
Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "po" dan mengatur ukuran fontnya menjadi 100 piksel. Ini adalah contoh sederhana dari penggunaan CSS untuk mengatur tampilan elemen HTML.

Kode Program

```
.po {  
  
    font-size: 100px;  
  
}
```

```
<p class="po">tes</p>
```

Hasil



Kesimpulan

Kesimpulan dari kode di atas adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tampilan elemen HTML. Dalam contoh ini, kita menggunakan kelas CSS

untuk mengubah ukuran font dari paragraf dengan kelas "po". Dengan menggunakan CSS, desain dan tata letak dari elemen-elemen HTML dapat disesuaikan dengan lebih fleksibel, memungkinkan pengembang web untuk menciptakan tampilan yang menarik dan responsif.

Font-weight

Penjelasan

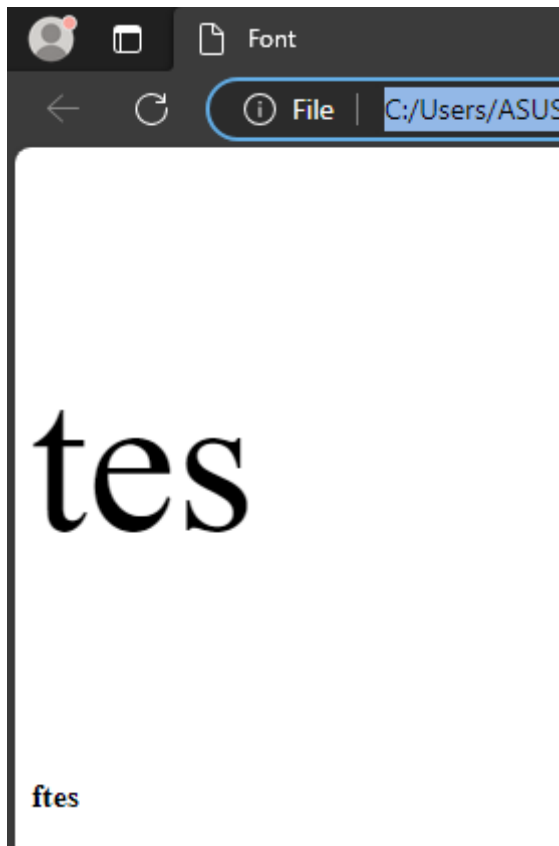
Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "poo" dan mengatur berat fontnya menjadi "bolder" atau lebih tebal. Ini adalah contoh sederhana dari penggunaan CSS untuk mengubah berat font sebuah elemen HTML.

Kode Program

```
<p class="poo">ftes</p>
```

```
.poo {  
  
    font-weight:bolder;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode di atas adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol berbagai properti gaya dari elemen-elemen HTML. Dalam contoh ini, kita menggunakan CSS untuk mengubah berat font dari paragraf dengan kelas "poo" menjadi lebih tebal. Dengan menggunakan CSS, pengembang web dapat menciptakan tampilan yang lebih menarik dan konsisten di seluruh situs web mereka.

Font-Style

Penjelasan

Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "pooo" dan mengatur gaya fontnya menjadi miring (italic). Ini adalah contoh sederhana dari penggunaan CSS untuk mengubah gaya font sebuah elemen HTML.

Kode Program

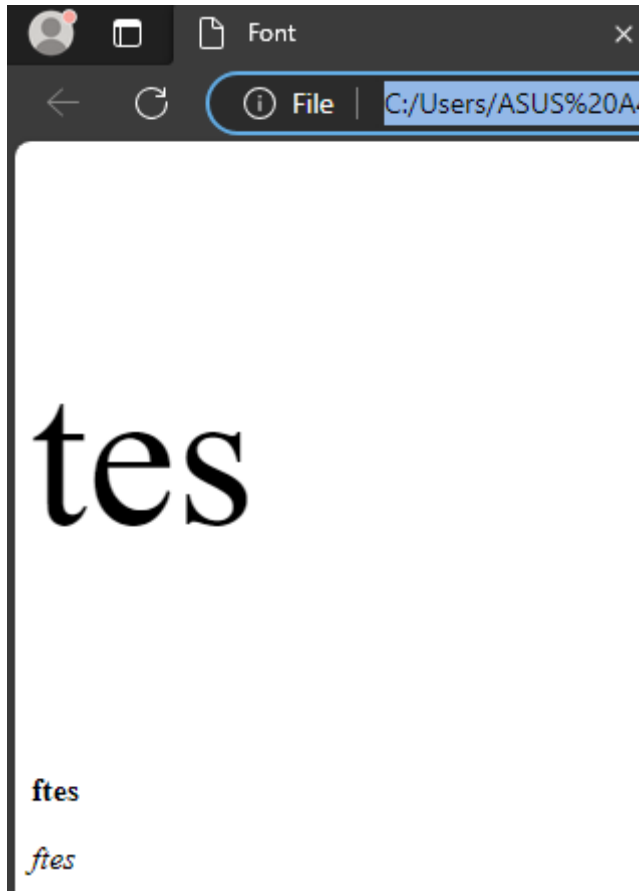
```
<p class="pooo">ftes</p>
```

```
.pooo {
```

```
font-style: italic;
```

```
}
```

Hasil



Kesimpulan

Kesimpulan dari kode di atas adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol berbagai properti gaya dari elemen-elemen HTML. Dalam contoh ini, kita menggunakan CSS untuk mengubah gaya font dari paragraf dengan kelas "pooo" menjadi miring. Dengan menggunakan CSS, pengembang web dapat menciptakan tampilan yang lebih variatif dan menarik di seluruh situs web mereka.

Font-Family

Penjelasan

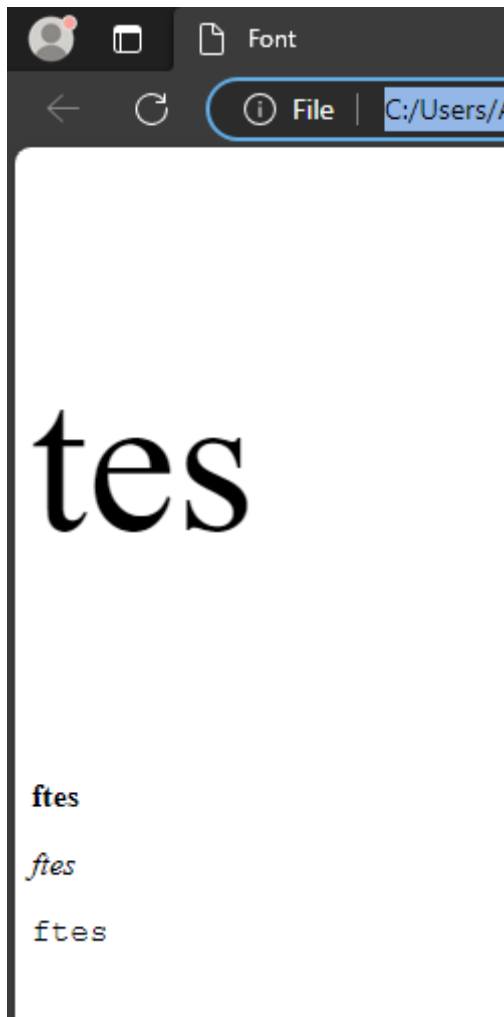
Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah paragraf dengan kelas "poooo" dan mengatur jenis fontnya menjadi 'Courier New', Courier, atau jenis font monospace lainnya. Ini adalah contoh sederhana dari penggunaan CSS untuk mengubah jenis font sebuah elemen HTML.

Kode Program

```
<p class="poooo">ftes</p>
```

```
.poooo {  
  
    font-family: 'Courier New', Courier, monospace;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengubah jenis font teks di dalam elemen HTML. Pemilihan jenis font tertentu, seperti 'Courier New' atau 'Courier', dapat memberikan tampilan teks yang konsisten dan mudah dibaca, terutama dalam konteks seperti menampilkan kode program. Penggunaan jenis font monospace juga berguna karena karakter memiliki lebar yang sama, membuatnya lebih mudah

untuk dibaca dan memelihara struktur format kode. Dengan demikian, dengan menggunakan CSS, pengembang web dapat mengontrol tampilan teks di situs mereka untuk memenuhi kebutuhan desain dan meningkatkan pengalaman pengguna.

MATERI BACKGROUND

Kode HTML & CSS

```
<!DOCTYPE html>

<html>

  <head>

    <style>

      .halaman-1 {

        background-image: url("mobil.jpg");

        height: 100vh;

        background-size: 100%;

        background-repeat: no-repeat;

        background-attachment: scroll;

        background-position: -50px -50px;

      }

    </style>

    <title></title>

  </head>

  <body>

    <div class="halaman-1">

      <h1>Selamat Datang</h1>
```

```
</div>

<br><br>

<br><br>

<br><br>

<br><br>

<br><br>

</body>

</html>
```

Background image

Penjelasan

Background-image adalah salah satu properti dalam CSS yang digunakan untuk menentukan gambar latar belakang dari sebuah elemen HTML.

Kode Program

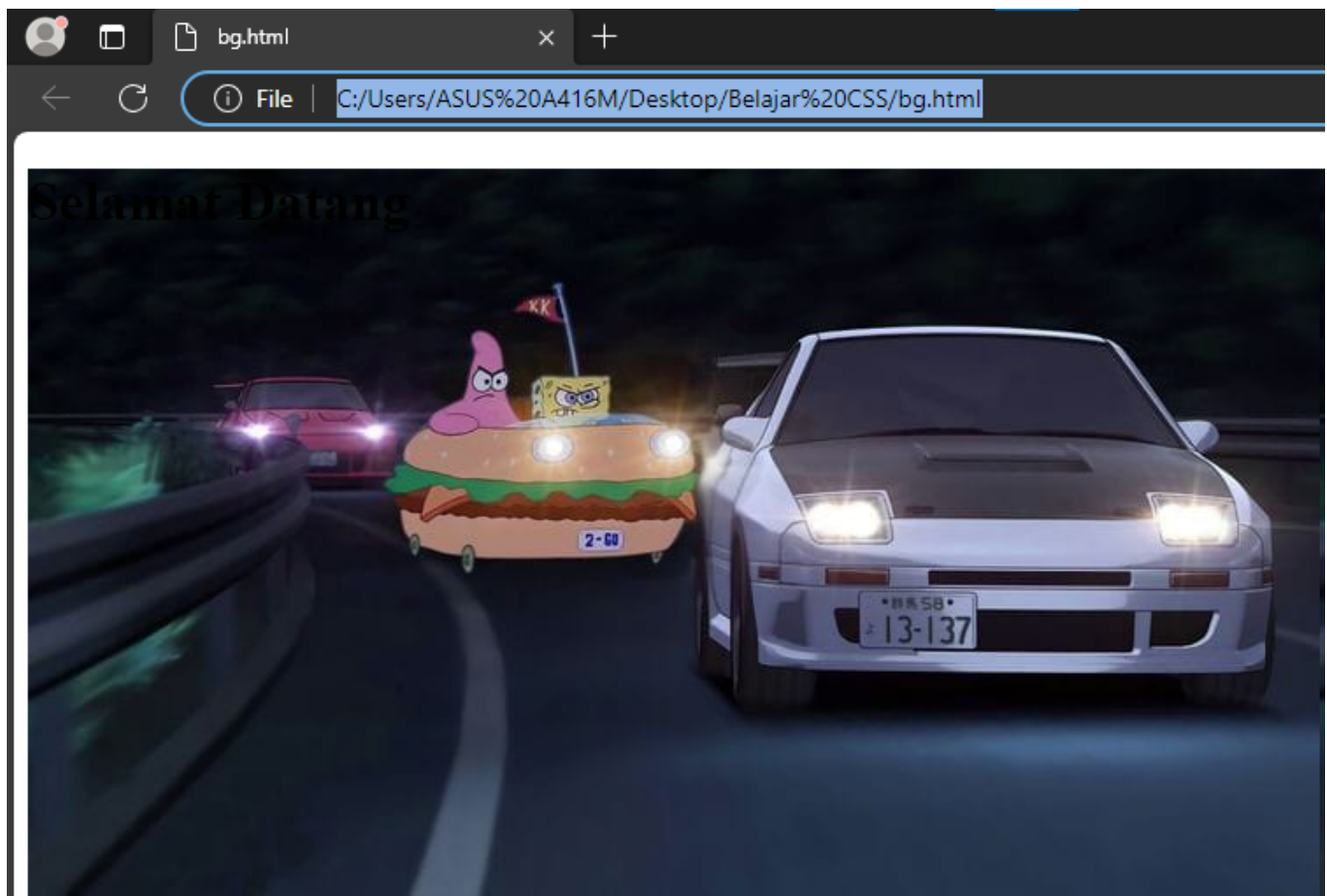
```
.halaman-1 {

    background-image: url("mobil.jpg");

    height: 100vh;

}
```

Hasil



Kesimpulan

kesimpulan dari kode tersebut adalah bahwa gambar "jkk.jpg" akan digunakan sebagai latar belakang

Background Zize

Penjelasan

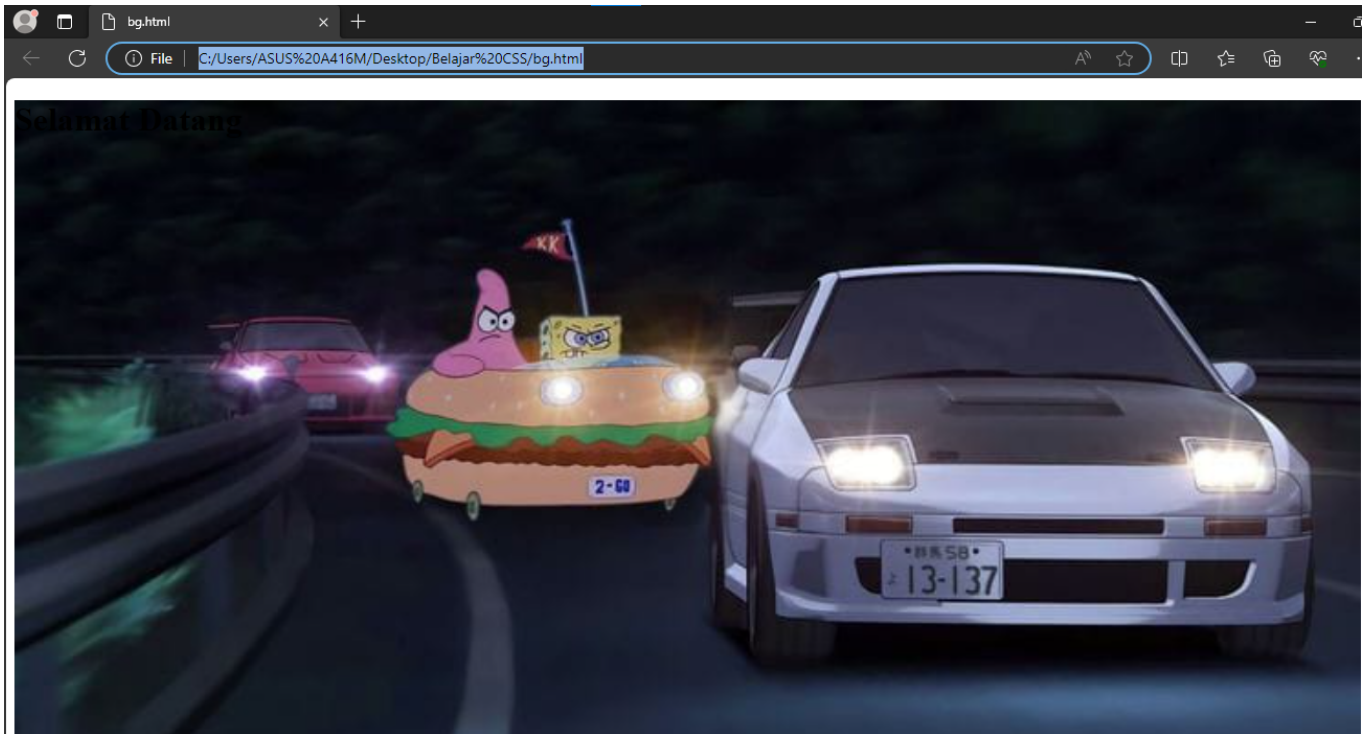
background-size adalah properti dalam CSS yang digunakan untuk mengatur ukuran dari gambar latar belakang

Kode Program

```
.halaman-1 {  
  
    background-image: url("mobil.jpg");  
  
    height: 100vh;  
  
    background-size: 100%;  
}
```

```
}
```

Hasil



Kesimpulan

Kesimpulan dari properti `background-size` dalam CSS adalah bahwa properti ini memungkinkan pengaturan ukuran gambar latar belakang pada sebuah elemen HTML. Dengan menggunakan `background-size`, Anda dapat mengontrol bagaimana gambar tersebut diperbesar atau diperkecil untuk menyesuaikan elemen yang terkait.

Background repeat

Penjelasan

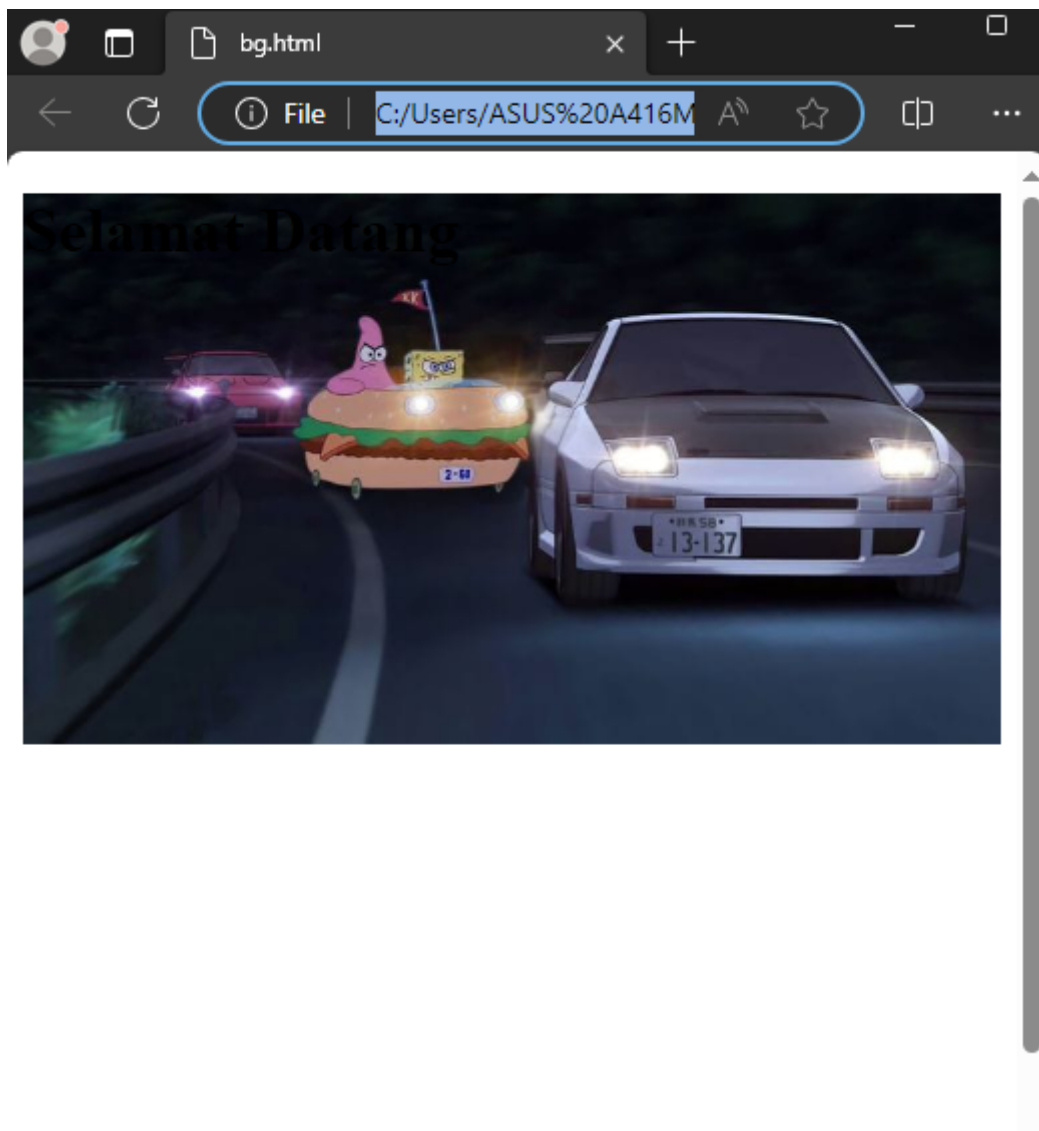
Properti `background-repeat: no-repeat;` adalah properti dalam CSS yang digunakan untuk mengontrol cara gambar latar belakang diulang atau diulang di dalam sebuah elemen HTML.

Ketika Anda menetapkan nilai `no-repeat` untuk properti `background-repeat`, ini berarti gambar latar belakang tidak akan diulang (tidak diulang baik secara horizontal maupun vertikal) di dalam elemen tersebut. Dengan kata lain, gambar latar belakang hanya akan ditampilkan sekali di dalam elemen tersebut, tanpa perulangan.

Kode Program

```
.halaman-1 {  
  
    background-image: url("mobil.jpg");  
  
    height: 100vh;  
  
    background-size: 100%;  
  
    background-repeat: no-repeat;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari penggunaan `background-repeat: no-repeat;` adalah bahwa properti tersebut menginstruksikan browser untuk tidak mengulang gambar latar belakang di dalam elemen HTML. Dengan menggunakan nilai `no-repeat`, gambar latar belakang hanya akan ditampilkan sekali di dalam elemen tersebut, tanpa diulang baik secara horizontal maupun vertikal.

Background attachment

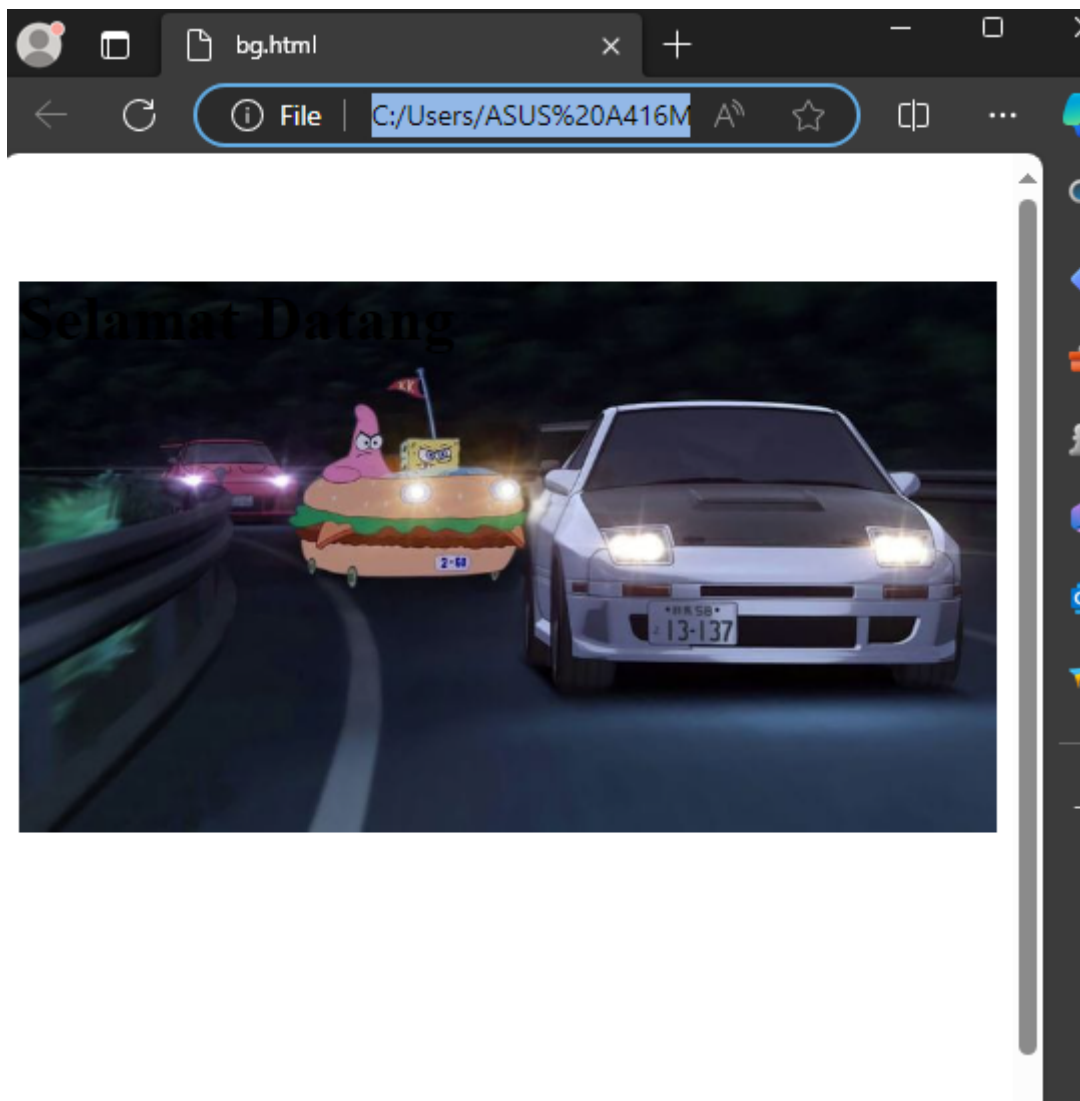
Penjelasan

Properti `background-attachment` digunakan dalam CSS untuk mengontrol apakah gambar latar belakang akan tetap diam (fix) atau akan mengikuti pergerakan konten saat pengguna melakukan scrolling. Nilai `scroll` pada properti `background-attachment` menentukan bahwa gambar latar belakang akan mengikuti pergerakan konten saat scrolling halaman.

Kode Program

```
.halaman-1 {  
  
    background-image: url("mobil.jpg");  
  
    height: 100vh;  
  
    background-size: 100%;  
  
    background-repeat: no-repeat;  
  
    background-attachment: scroll;  
  
}
```

Hasil



Kesimpulan

Kesimpulannya, dengan menggunakan `background-attachment: scroll;`, Anda memastikan bahwa gambar latar belakang akan mengikuti pergerakan konten saat pengguna melakukan scrolling, menciptakan efek visual di mana gambar tersebut tetap berada di belakang konten dan bergerak bersama dengan halaman.

Background position

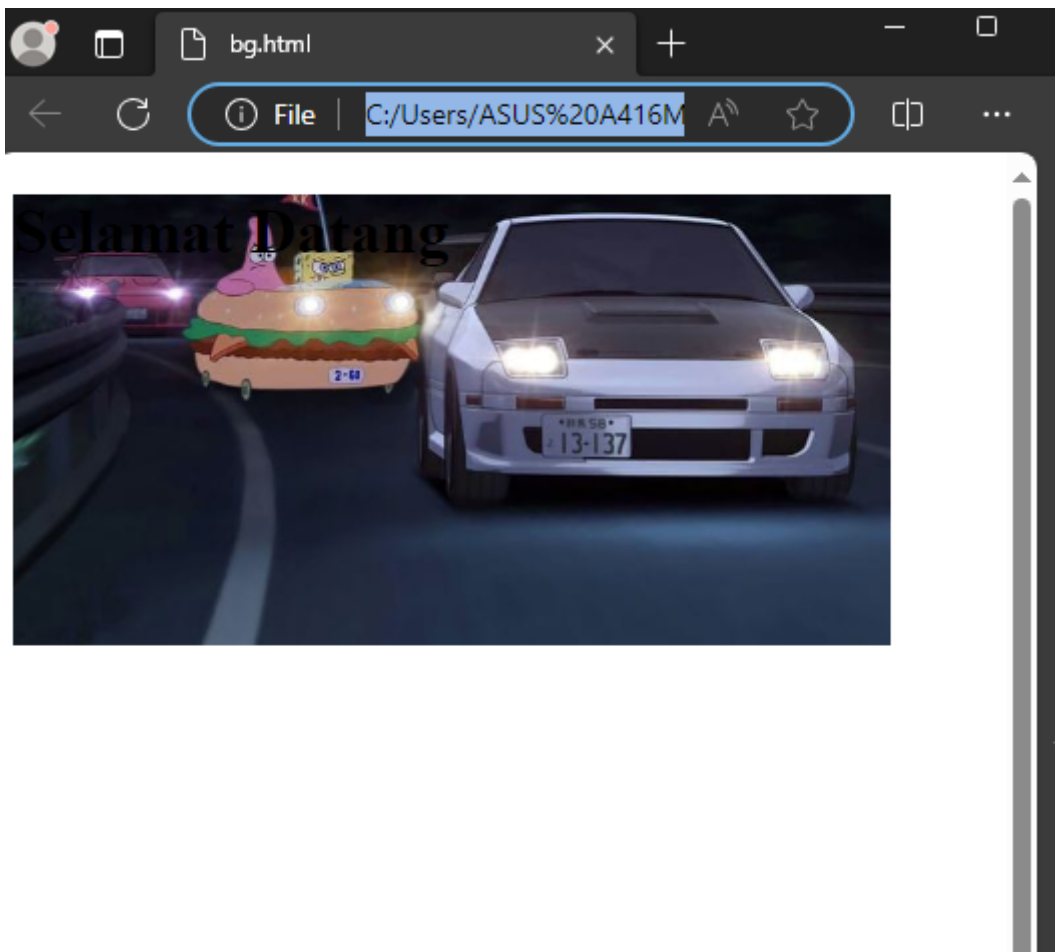
Penjelasan

Properti `background-position` digunakan dalam CSS untuk menentukan posisi awal dari gambar latar belakang di dalam sebuah elemen HTML. Properti ini memungkinkan Anda untuk mengatur di mana gambar latar belakang tersebut akan mulai ditampilkan di dalam elemen terkait.

Kode Program

```
.halaman-1 {  
  
    background-image: url("mobil.jpg");  
  
    height: 100vh;  
  
    background-size: 100%;  
  
    background-repeat: no-repeat;  
  
    background-attachment: scroll;  
  
    background-position: -50px -50px;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Dengan menggunakan `background-position`, Anda dapat mengontrol di mana gambar latar belakang akan mulai ditampilkan di dalam elemen HTML. Properti ini memungkinkan penempatan yang tepat dan fleksibel dari gambar latar belakang, baik itu berdasarkan koordinat absolut, nilai relatif, atau persentase dari ukuran elemen.

BOX MODEL

Border

Kode HTML & CSS

```
<!DOCTYPE html>

<html>

<head>

  <title>BORDER</title>

  <link rel="stylesheet" href="border.css">

</head>

<body>

  <button class="py">klik aku</button>

</body>

</html>
```

```
.py {

  color: white;

  background-color: purple;

  border-width: 5px;

  border-style: solid;

  border-color: rgb(170, 255, 0);
```

```
border-radius: 25px 10px ;  
  
}
```

Border-width

Penjelasan

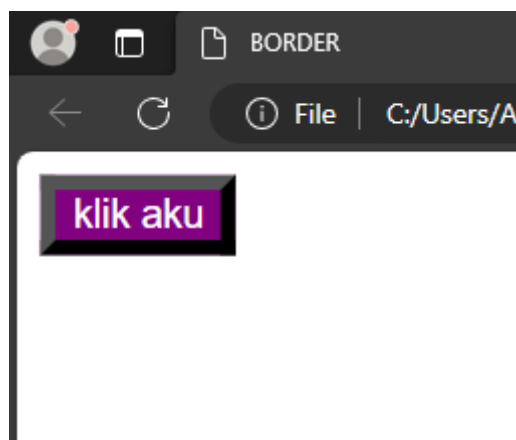
Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah tombol dengan kelas "py" dan mengatur beberapa properti gaya untuk tombol tersebut. Tombol ini memiliki teks "klik aku", menggunakan warna putih untuk teks, latar belakang ungu (purple), dan border (garis pinggir) dengan lebar 5 piksel.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa CSS memberikan fleksibilitas dalam mendesain dan menyesuaikan tampilan elemen-elemen HTML. Dalam contoh ini, tombol dibuat dengan gaya yang menarik dengan menggunakan kombinasi warna dan border.

Border-style

Penjelasan

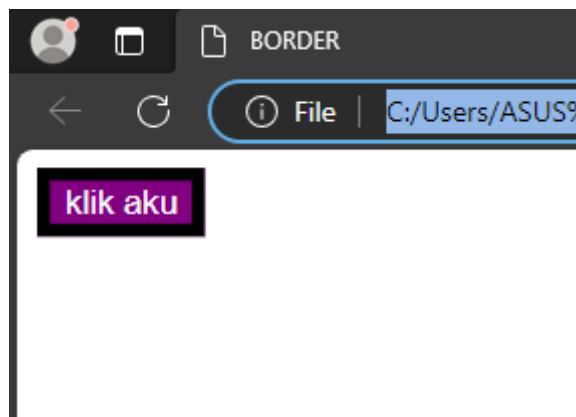
Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah tombol dengan kelas "py" dan mengatur beberapa properti gaya untuk tombol tersebut. Tombol ini memiliki teks "klik aku", menggunakan warna putih untuk teks, latar belakang ungu (purple), dan memiliki border (garis pinggir) dengan lebar 5 piksel dan gaya solid.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa CSS memberikan fleksibilitas dalam mendesain dan menyesuaikan tampilan elemen-elemen HTML. Dalam contoh ini, tombol dibuat dengan gaya yang menarik dengan menggunakan kombinasi warna, border, dan style.

Border-color

Penjelasan

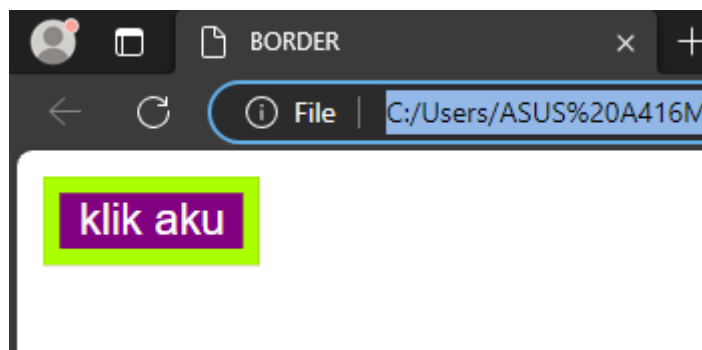
Program di atas adalah kode HTML dan CSS yang mendefinisikan sebuah tombol dengan kelas "py" dan mengatur beberapa properti gaya untuk tombol tersebut. Tombol ini memiliki teks "klik aku", menggunakan warna putih untuk teks, latar belakang ungu (purple), dan memiliki border (garis pinggir) dengan lebar 5 piksel, gaya solid, dan warna border ditentukan dalam format RGB.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa CSS memberikan fleksibilitas dalam mendesain dan menyesuaikan tampilan elemen-elemen HTML. Dalam contoh ini, tombol dibuat dengan gaya yang menarik dengan menggunakan kombinasi warna, border, dan style.

Border-radius

Penjelasan

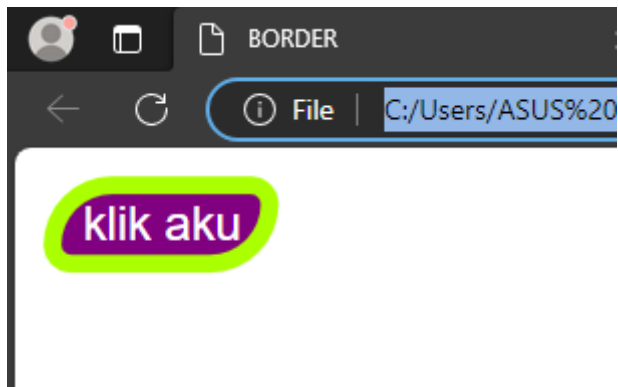
1. `color: white;` -> Mengatur warna teks tombol menjadi putih.
2. `background-color: purple;` -> Mengatur warna latar belakang tombol menjadi ungu.
3. `border-width: 5px;` -> Mengatur lebar border (garis pinggir) tombol menjadi 5 piksel.
4. `border-style: solid;` -> Mengatur gaya border tombol menjadi solid (garis lurus).
5. `border-color: rgb(170, 255, 0);` -> Mengatur warna border tombol menjadi warna dengan nilai RGB (170, 255, 0), yang mungkin menghasilkan warna yang tidak biasa, antara hijau dan kuning.
6. `border-radius: 25px 10px ;` -> Mengatur radius sudut tombol, dengan nilai 25px untuk sudut kiri atas dan sudut kanan bawah, dan 10px untuk sudut kanan atas dan sudut kiri bawah. Ini akan memberikan efek sudut bulat pada tombol.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
  
    border-radius: 25px 10px ;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tampilan elemen HTML seperti tombol. Properti-properti CSS digunakan untuk menyesuaikan warna, latar belakang, border, dan bahkan bentuk atau sudut tombol sesuai dengan preferensi desain

Padding

Kode HTML & CSS

```
<!DOCTYPE html>

<html>

<head>

    <title>Document</title>

    <link rel="stylesheet" href="padding.css">

</head>

<body>

    <button class="py">klik aku</button>

</body>

</html>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
  
    border-radius: 25px 10px ;  
  
  
    padding-left: 50px;  
  
    padding-right: 50px;  
  
    padding-top: 50px;  
  
    padding-bottom: 50px;  
  
}
```

Padding-left

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti-properti CSS yang digunakan adalah:

1. color: white; -> Mengatur warna teks tombol menjadi putih.
2. background-color: purple; -> Mengatur warna latar belakang tombol menjadi ungu.
3. border-width: 5px; -> Mengatur lebar border (garis pinggir) tombol menjadi 5 piksel.
4. border-style: solid; -> Mengatur gaya border tombol menjadi solid (garis lurus).
5. border-color: rgb(170, 255, 0); -> Mengatur warna border tombol menjadi warna dengan nilai RGB (170, 255, 0), yang mungkin menghasilkan warna yang tidak biasa, antara hijau dan kuning.
6. border-radius: 25px 10px; -> Mengatur radius sudut tombol, dengan nilai 25px untuk sudut kiri atas dan sudut kanan bawah, dan 10px untuk sudut kanan atas dan sudut kiri bawah. Ini akan memberikan efek sudut bulat pada tombol.

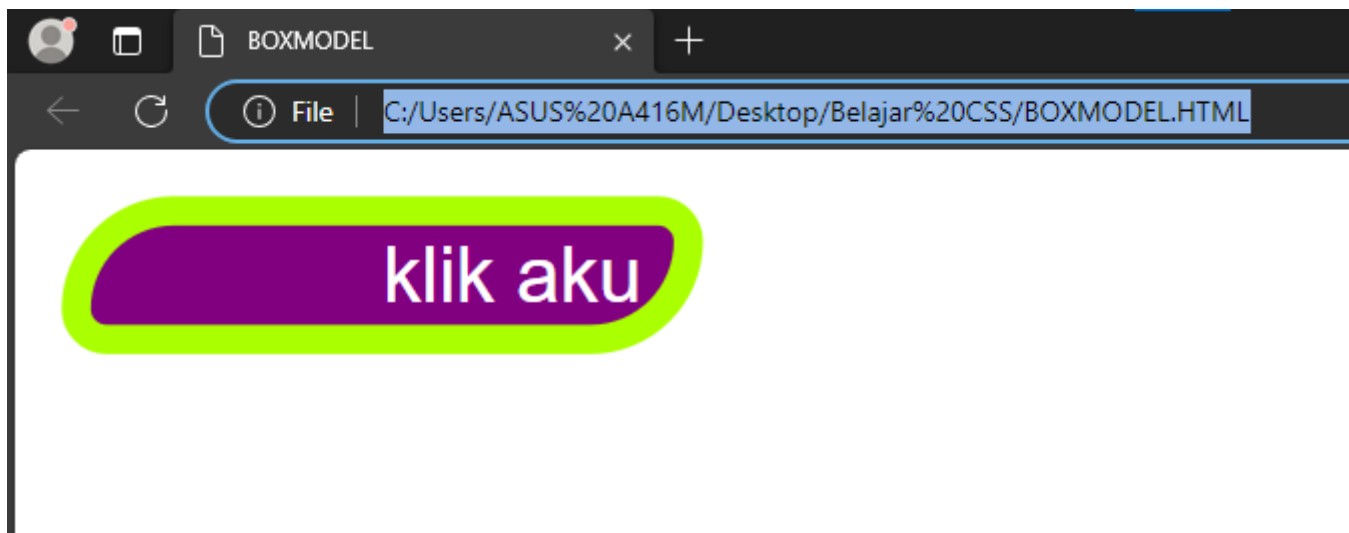
7. padding-left: 50px; -> Memberikan padding (ruang tambahan) pada sisi kiri tombol sebesar 50 piksel. Padding adalah ruang antara teks dalam elemen dan batasnya.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
  
    border-radius: 25px 10px ;  
  
  
    padding-left: 50px;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tampilan elemen HTML seperti tombol. Properti-properti CSS digunakan untuk menyesuaikan warna, latar belakang, border, padding, dan bahkan bentuk atau sudut tombol sesuai dengan preferensi desain.

Padding-right

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti-properti CSS yang digunakan adalah:

1. color: white; -> Mengatur warna teks tombol menjadi putih.
2. background-color: purple; -> Mengatur warna latar belakang tombol menjadi ungu.
3. border-width: 5px; -> Mengatur lebar border (garis pinggir) tombol menjadi 5 piksel.
4. border-style: solid; -> Mengatur gaya border tombol menjadi solid (garis lurus).
5. border-color: rgb(170, 255, 0); -> Mengatur warna border tombol menjadi warna dengan nilai RGB (170, 255, 0), yang mungkin menghasilkan warna yang tidak biasa, antara hijau dan kuning.
6. border-radius: 25px 10px; -> Mengatur radius sudut tombol, dengan nilai 25px untuk sudut kiri atas dan sudut kanan bawah, dan 10px untuk sudut kanan atas dan sudut kiri bawah. Ini akan memberikan efek sudut bulat pada tombol.
7. padding-left: 50px; -> Memberikan padding (ruang tambahan) pada sisi kanan tombol sebesar 50 piksel. Padding adalah ruang antara teks dalam elemen dan batasnya.

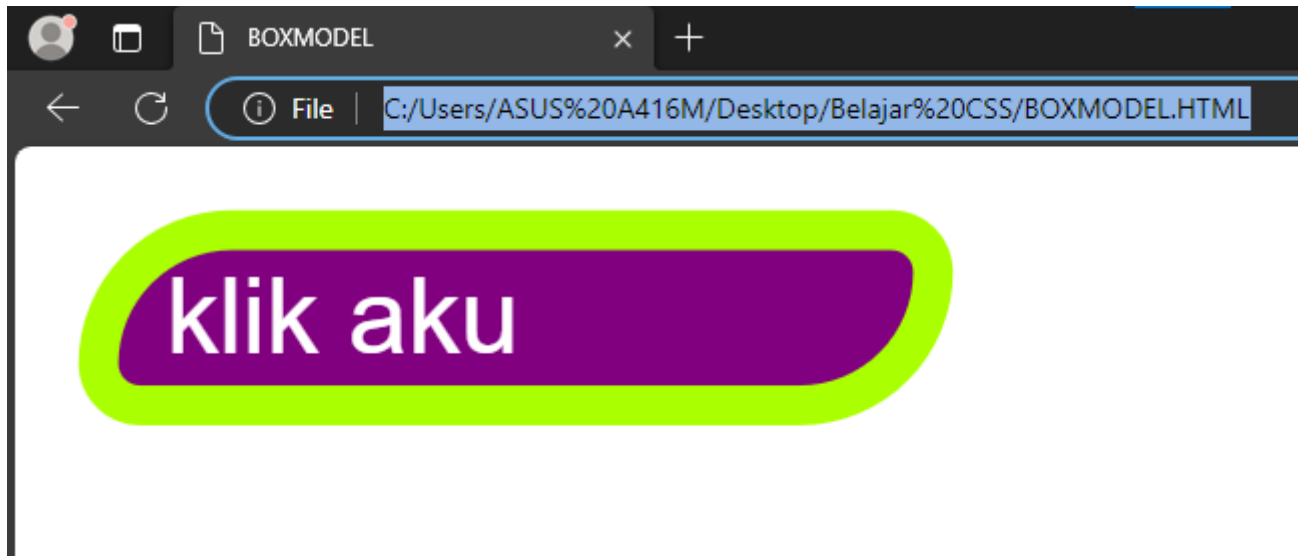
Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
  
    border-radius: 25px 10px ;  
}
```

```
padding-right: 50px;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tampilan elemen HTML seperti tombol. Properti-properti CSS digunakan untuk menyesuaikan warna, latar belakang, border, padding, dan bahkan bentuk atau sudut tombol sesuai dengan preferensi desain.

Padding-top

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti-properti CSS yang digunakan adalah:

1. color: white; -> Mengatur warna teks tombol menjadi putih.
2. background-color: purple; -> Mengatur warna latar belakang tombol menjadi ungu.
3. border-width: 5px; -> Mengatur lebar border (garis pinggir) tombol menjadi 5 piksel.
4. border-style: solid; -> Mengatur gaya border tombol menjadi solid (garis lurus).
5. border-color: rgb(170, 255, 0); -> Mengatur warna border tombol menjadi warna dengan nilai RGB (170, 255, 0), yang mungkin menghasilkan warna yang tidak biasa, antara hijau dan kuning.

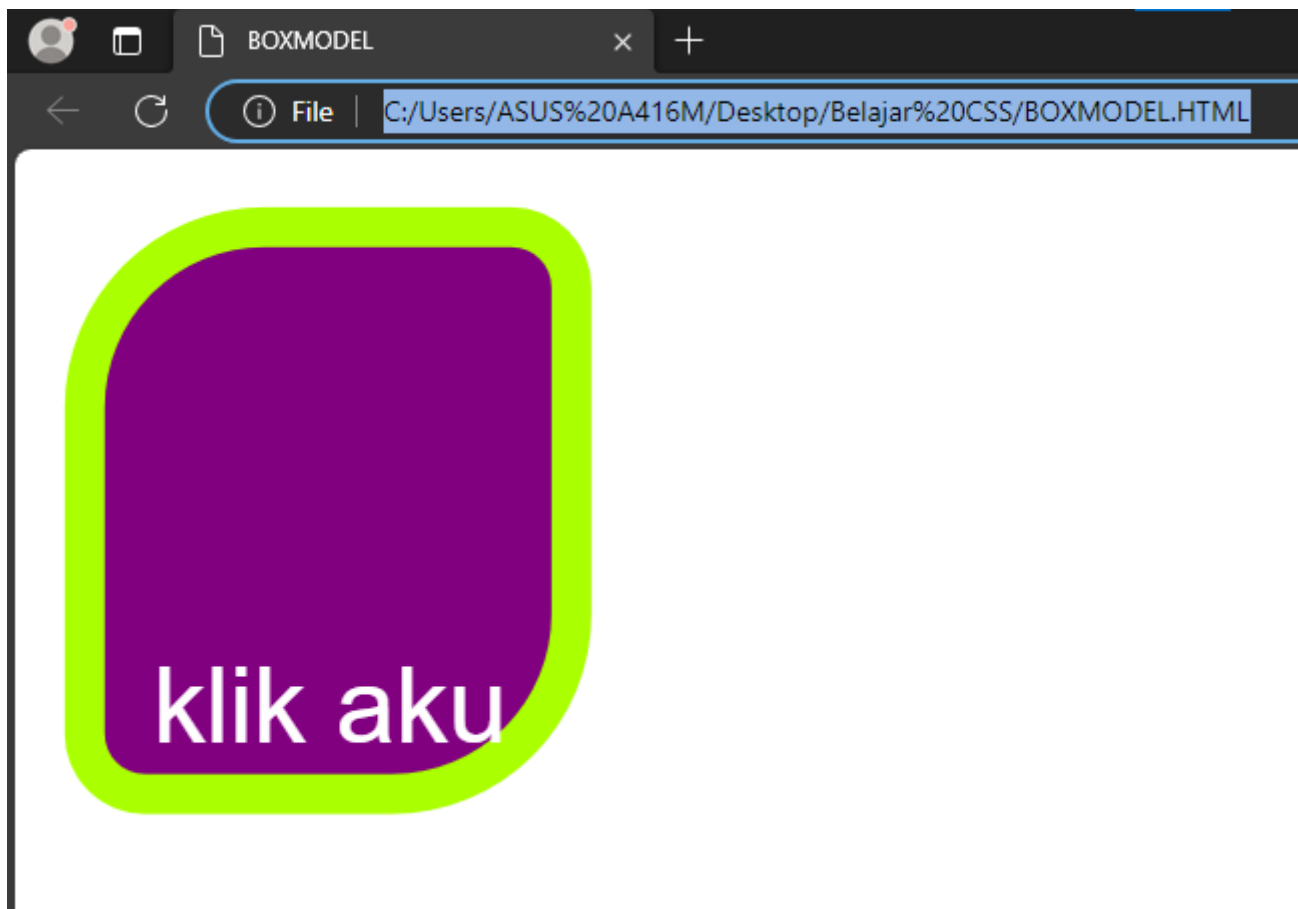
6. `border-radius: 25px 10px;` -> Mengatur radius sudut tombol, dengan nilai 25px untuk sudut kiri atas dan sudut kanan bawah, dan 10px untuk sudut kanan atas dan sudut kiri bawah. Ini akan memberikan efek sudut bulat pada tombol.
7. `padding-left: 50px;` -> Memberikan padding (ruang tambahan) pada sisi atas tombol sebesar 50 piksel. Padding adalah ruang antara teks dalam elemen dan batasnya.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
  
    border-radius: 25px 10px ;  
  
  
    padding-top: 50px;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tampilan elemen HTML seperti tombol. Properti-properti CSS digunakan untuk menyesuaikan warna, latar belakang, border, padding, dan bahkan bentuk atau sudut tombol sesuai dengan preferensi desain.

Padding-bottom

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti-properti CSS yang digunakan adalah:

1. `color: white;` -> Mengatur warna teks tombol menjadi putih.
2. `background-color: purple;` -> Mengatur warna latar belakang tombol menjadi ungu.
3. `border-width: 5px;` -> Mengatur lebar border (garis pinggir) tombol menjadi 5 piksel.
4. `border-style: solid;` -> Mengatur gaya border tombol menjadi solid (garis lurus).
5. `border-color: rgb(170, 255, 0);` -> Mengatur warna border tombol menjadi warna dengan nilai RGB (170, 255, 0), yang mungkin menghasilkan warna yang tidak biasa, antara hijau dan kuning.

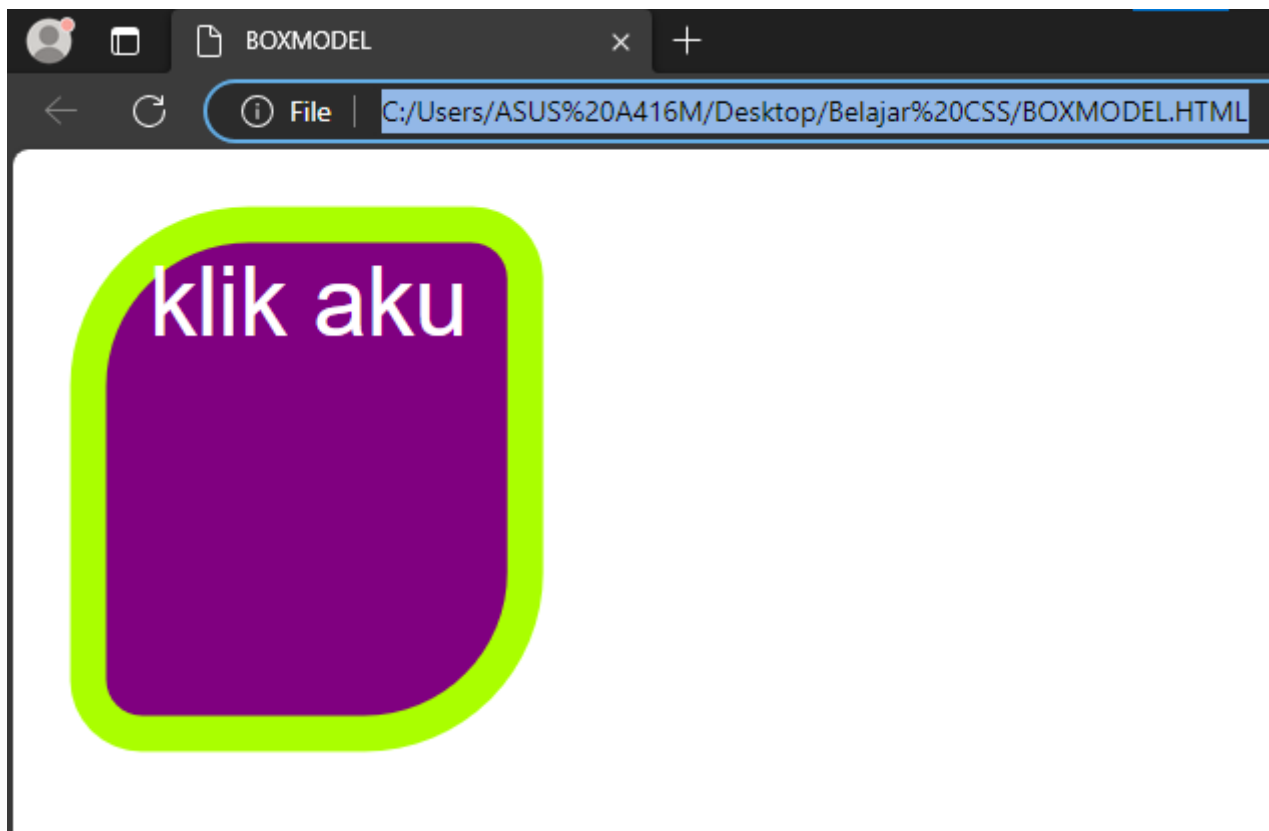
6. `border-radius: 25px 10px;` -> Mengatur radius sudut tombol, dengan nilai 25px untuk sudut kiri atas dan sudut kanan bawah, dan 10px untuk sudut kanan atas dan sudut kiri bawah. Ini akan memberikan efek sudut bulat pada tombol.
7. `padding-left: 50px;` -> Memberikan padding (ruang tambahan) pada sisi bawah tombol sebesar 50 piksel. Padding adalah ruang antara teks dalam elemen dan batasnya.

Kode Program

```
<button class="py">klik aku</button>
```

```
.py {  
  
    color: white;  
  
    background-color: purple;  
  
    border-width: 5px;  
  
    border-style: solid;  
  
    border-color: rgb(170, 255, 0);  
  
    border-radius: 25px 10px ;  
  
  
    padding-bottom: 50px;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tampilan elemen HTML seperti tombol. Properti-properti CSS digunakan untuk menyesuaikan warna, latar belakang, border, padding, dan bahkan bentuk atau sudut tombol sesuai dengan preferensi desain.

Margin

Kode HTML & CSS

```
<!DOCTYPE html>

<html>

<head>

    <title>Document</title>

    <link rel="stylesheet" href="margin.css">

</head>

<body>
```

```
<button class="py">klik aku disini</button>

</body>

</html>
```

```
.py {

    margin-left: 300px ;

    margin-right: 300px ;

    margin-top: 300px ;

    margin-bottom: 300px;

}
```

Margin left

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti CSS yang digunakan adalah:

1. margin-left: 300px; -> Properti ini memberikan jarak (margin) pada sisi kiri tombol sebesar 300 piksel. Ini akan membuat tombol dipindahkan 300 piksel dari sisi kiri dari posisi normalnya.

Kode Program

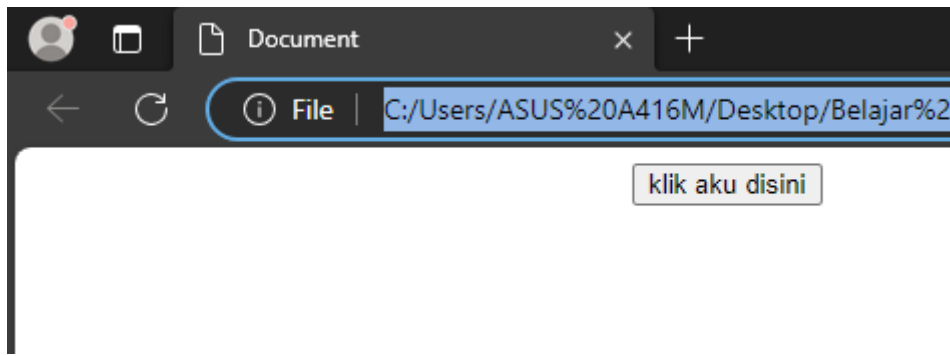
```
<button class="py">klik aku disini</button>
```

```
.py {

    margin-left: 300px ;

}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tata letak elemen HTML seperti tombol. Properti margin memungkinkan kita untuk mengatur jarak antara elemen tersebut dengan elemen lain di sekitarnya. Dalam hal ini, tombol dipindahkan ke kanan sebesar 300 piksel dari posisi normalnya.

Margin right

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti CSS yang digunakan adalah:

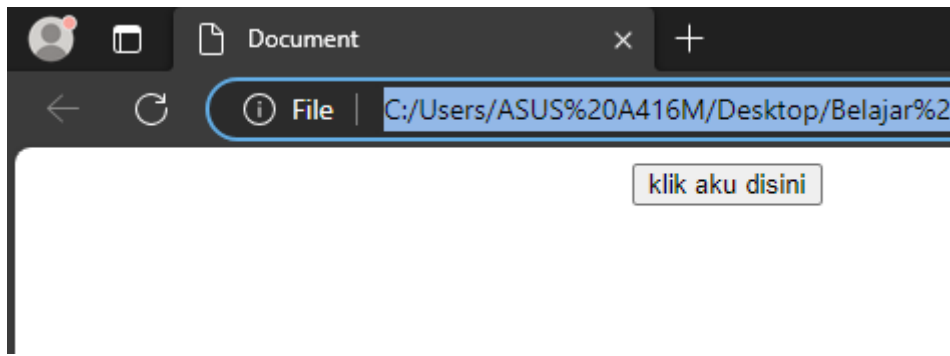
1. margin-right: 300px; -> Properti ini memberikan jarak (margin) pada sisi kanan tombol sebesar 300 piksel. Ini akan membuat tombol dipindahkan 300 piksel dari sisi kanan dari posisi normalnya.

Kode Program

```
<button class="py">klik aku disini</button>
```

```
.py {  
    margin-right: 300px ;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tata letak elemen HTML seperti tombol. Properti margin memungkinkan kita untuk mengatur jarak antara elemen tersebut dengan elemen lain di sekitarnya. Dalam hal ini, tombol dipindahkan ke kiri sebesar 300 piksel dari posisi normalnya.

Margin top

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti CSS yang digunakan adalah:

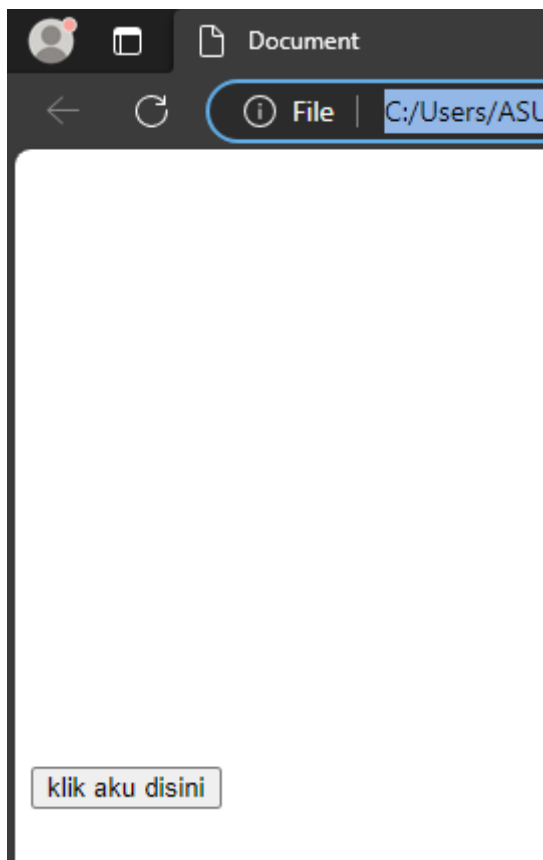
1. margin-top: 300px; -> Properti ini memberikan jarak (margin) pada sisi atas tombol sebesar 300 piksel. Ini akan membuat tombol dipindahkan 300 piksel dari sisi atas dari posisi normalnya.

Kode Program

```
<button class="py">klik aku disini</button>
```

```
.py {  
    margin-top: 300px ;  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tata letak elemen HTML seperti tombol. Properti margin memungkinkan kita untuk mengatur jarak antara elemen tersebut dengan elemen lain di sekitarnya. Dalam hal ini, tombol dipindahkan ke bawah sebesar 300 piksel dari posisi normalnya.

Margin bottom

Penjelasan

Program di atas adalah kode CSS yang mengatur tampilan sebuah tombol dengan kelas "py". Properti CSS yang digunakan adalah:

1. margin-bottom: 300px; -> Properti ini memberikan jarak (margin) pada sisi bawah tombol sebesar 300 piksel. Ini akan membuat tombol dipindahkan 300 piksel dari sisi bawah dari posisi normalnya.

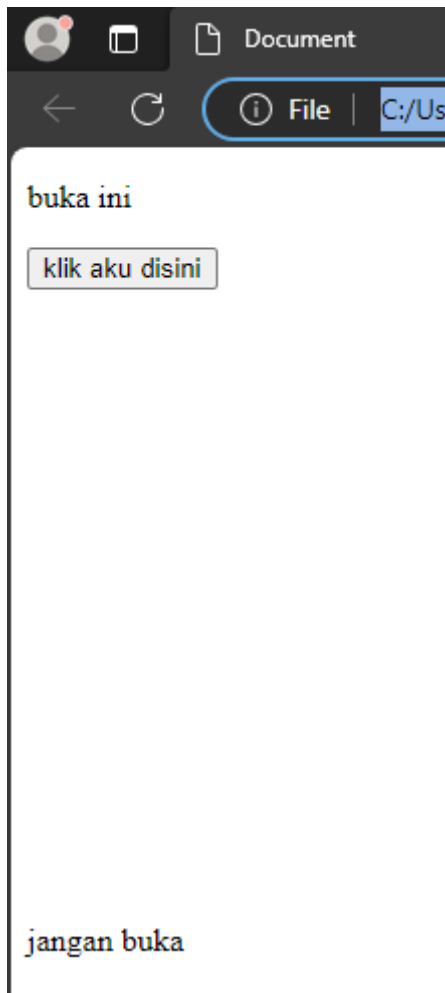
Kode Program

```
<p>buka ini</p>  
  
<button class="py">klik aku disini</button>
```

```
<p>jangan buka</p>
```

```
.py {  
  
    margin-bottom: 300px ;  
  
}
```

Hasil



Kesimpulan

Kesimpulan dari kode tersebut adalah bahwa dengan menggunakan CSS, kita dapat dengan mudah mengontrol tata letak elemen HTML seperti tombol. Properti margin memungkinkan kita untuk mengatur jarak antara elemen tersebut dengan elemen lain di sekitarnya. Dalam hal ini, tombol dipindahkan ke atas sebesar 300 piksel dari posisi normalnya.

Studi Kasus

Deskripsi Kasus

Program di atas adalah halaman HTML sederhana yang mencakup beberapa elemen HTML seperti gambar, teks, dan tombol. Selain itu, terdapat juga sebuah file CSS eksternal yang digunakan untuk memperindah tampilan halaman.

Berikut deskripsi dari setiap bagian program tersebut:

1. **Elemen HTML:** Terdapat elemen-elemen HTML seperti ``, ``, `<p>`, dan `<button>`. Elemen `` digunakan untuk mengelompokkan beberapa elemen bersama-sama, sedangkan `` digunakan untuk menampilkan gambar. Elemen `<p>` digunakan untuk menampilkan teks paragraf, dan `<button>` digunakan untuk membuat tombol.
2. **CSS Eksternal:** Terdapat juga file CSS eksternal yang disematkan menggunakan tag `<link>` di dalam elemen `<head>`. File CSS ini disebut "latian.css" dan digunakan untuk mengatur tampilan elemen-elemen HTML di halaman.
3. **Pengaturan CSS:** Di dalam file CSS "latian.css", terdapat beberapa aturan CSS yang mengatur tampilan elemen-elemen HTML. Beberapa properti yang diatur antara lain warna teks, ukuran font, margin, border, dan warna latar belakang.
4. **Tampilan Halaman:** Halaman HTML ini memiliki latar belakang warna ungu ("purple") yang didefinisikan melalui atribut `bgcolor` pada elemen `<body>`. Selain itu, terdapat teks berwarna putih ("white") dengan ukuran font besar yang ditampilkan di tengah halaman. Gambar samurai juga ditampilkan di sebelah kanan halaman dengan sedikit efek margin dan border.
5. **Tombol:** Terdapat tombol dengan teks "klik aku" yang memiliki warna teks oranye ("orange") dan latar belakang ungu ("purple").

Langkah Pengerjaan

1. cara pertama yaitu ganti latar belakang menjadi ungu dengan cara menambahkan `bgcolor` di dalam tag `<body>`
2. kaitkan link html ke css dengan cara `<link rel="stylesheet" href="latian.css">` agar bisa memodifikasi atau desain program melalui css
3. buatlah tag `` dan masukkan gambar dengan menggunakan tag `` juga masukkan lebar dan tinggi dari gambar tersebut
4. setelah memasukkan gambar buatlah teks selamat datang di web Farhan menggunakan tag `<p>` dan di dalam tag `<p>` juga ada tag `` dan `
` contoh

```
`<span> 
```

```
<p>selamat datang <br>  
  
di<b> web Farhan</b></p>  
  
</span>
```

5. sesudah itu buatlah tombol menggunakan tag `<button>` yang isinya klik aku
6. lanjut ke css,kita akan memodifikasi tag `<p>` dan mengubah nya seperti warna teks nya menjadi putih,ukuran font nya menjadi 75px dan menggunakan margin untuk mengubah posisinya . contoh program

```
p {  
  color: white;  
  font-size: 75px;  
  margin-top: 150px;  
  margin-bottom: 100px;  
  margin-left: 50px;  
  margin-right: 100px;  
}
```

7. selanjutnya kita akan memodifikasi tag `` atau gambarnya dan mengubahnya menggunakan margin,border,dan border radius.ukuran margin-right nya adalah 200px,margin top nya adalah -30px dan didalam border ada ukuran style dan warna nya,dan gunanya border radius adalah untuk membulatkan gambar dengan memberikan ukuran 150px .contoh program:

```
img {  
  
  margin-right: 200px;  
  
  margin-top: -30px;  
  
  border: 5px solid white;  
  
  border-radius: 150px 150px;
```

8. terakhir kita akan memodifikasi tag `<button>` dengan mengubah warna teksnya menjadi orange,latar belakangnya menjadi warna ungu,lebar nya 100px tingginya 50px,juga warna dari bordernya menjadi warna orange,dan mengubah posisinya dengan menggunakan margin dan ukuran margin nya adalah margin-top -100px margin-left 400px.contoh program:


```
button {  
  
    color: orange;  
  
    background-color: purple;  
  
    width: 100px;  
  
    height: 50px;  
  
    border-color: orangered;  
  
    margin-top: -100px;  
  
    margin-left: 400px;  
  
}
```

9. selesai

Box Model(Tantangan)

Penjelasan HTML

- `<!DOCTYPE html>` : Mendefinisikan jenis dokumen HTML.
- `<html lang="en">` : Mengawali dokumen HTML dengan bahasa Inggris.
- `<head>` : Bagian kepala dokumen yang berisi informasi meta dan judul.
- `<meta charset="UTF-8">` : Mendefinisikan karakter set dokumen sebagai UTF-8.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">` : Mengatur tampilan responsif di perangkat dengan lebar layar yang berbeda.
- `<title>Document</title>` : Judul halaman web.
- `<link rel="stylesheet" href="tampilanweb.css">` : Menghubungkan halaman HTML dengan file CSS eksternal.
- `<body bgcolor="purple">` : Mengatur warna latar belakang halaman menjadi ungu (purple).
- `` : Membuat bagian teks menjadi bagian inline.
- `<div class="text">` : Membuat sebuah div dengan kelas "text" yang berisi teks "Selamat datang".
- `` : Membuat sebuah span dengan kelas "text2" yang berisi teks "di web Rahmat".

- `` : Menampilkan gambar dengan nama "naon.jpg" dengan kelas "foto".
- `<button class="tombol">Klik ini</button>` : Membuat sebuah tombol dengan teks "Klik ini" dan kelas "tombol".

Penjelasan CSS

- `.text` dan `.text2` : Mengatur gaya teks untuk kelas "text" dan "text2". Menggunakan font Times New Roman, ukuran font 30mm, dan warna putih.
- `.foto` : Mengatur tampilan gambar dengan kelas "foto". Menerapkan display block, membuat sudut gambar menjadi bulat dengan border-radius, mengatur lebar dan tinggi gambar, mengatur posisi gambar dengan margin-left dan margin-top, serta memberikan border putih dengan lebar 10px dan gaya solid.
- `.tombol` : Mengatur tampilan tombol dengan kelas "tombol". Memberikan warna teks dan border oranye, mengatur lebar dan tinggi tombol, mengatur tampilan tombol dengan display block, memberikan warna latar belakang transparan, mengatur border dengan lebar 3px dan gaya solid, mengatur posisi tombol dengan margin-top dan margin-left, serta mengatur ukuran font.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="tampilanweb.css">
</head>
<body bgcolor="purple">
  <span>
    <div class="text">Selamat datang
  </div>
</span><br>
  <span class="text2">di web Rahmat</span><br>
  
  <button class="tombol">Klik ini</button>
</body>
</html>
```

```
.text{
  font-family: 'Times New Roman';
```

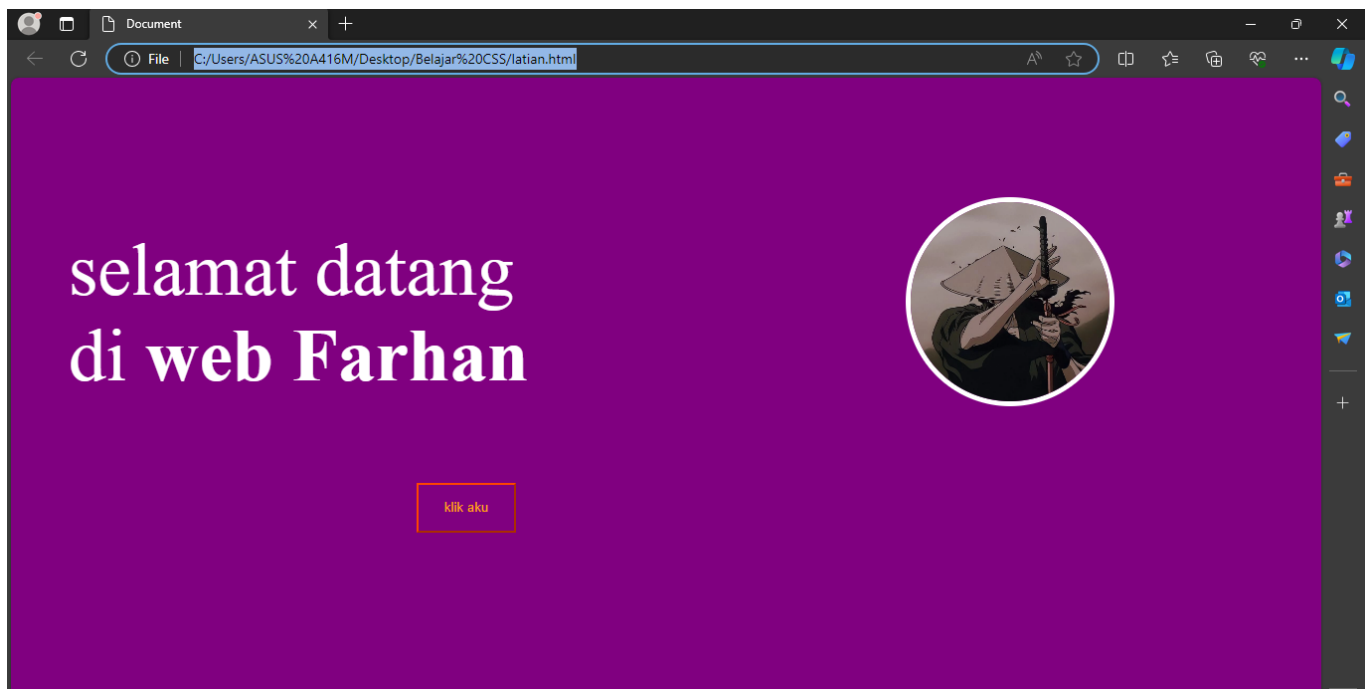
```
    font-size: 30mm;
    color: white;
}

.text2{
    font-family: 'Times New Roman';
    font-size: 30mm;
    color: white;
}

.foto{
    display: block;
    border-radius: 1000px;
    width: 400px;
    height: 400px;
    margin-left: 735px;
    margin-top: -215px;
    border-width: 10px;
    border-color: white;
    border-style: solid;
}

.tombol{
    color: orangered;
    border-color: orangered;
    width: 160px;
    height: 85px;
    display: block;
    background-color: transparent;
    border-width: 3px;
    border-style: solid;
    margin-top: -200px;
    margin-left: 13cm;
    font-size: 20px;
}
```

Hasil Pengerjaan



Pseudo-classes

Hover

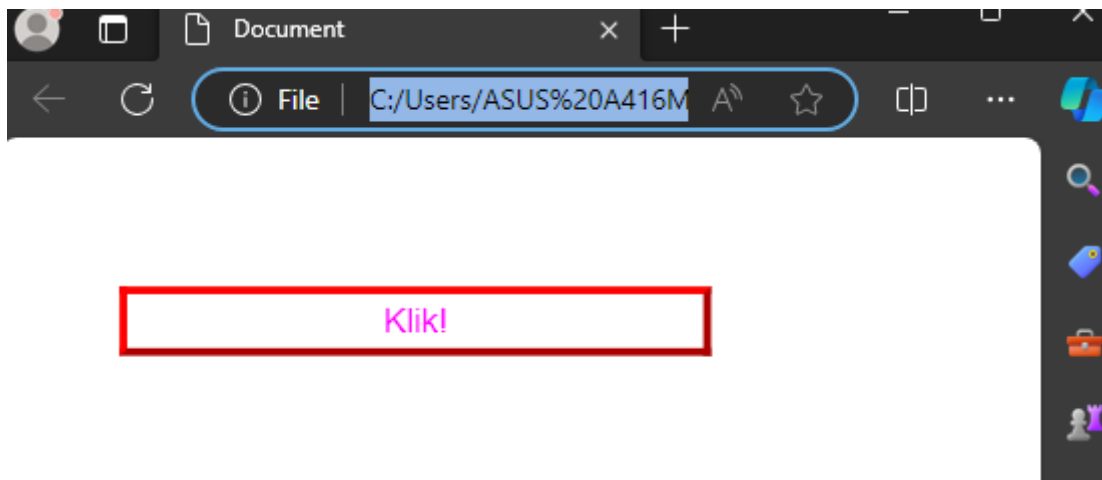
Penjelasan

Pernyataan tersebut menggunakan pseudo-class `:hover` untuk menargetkan perilaku saat kursor berada di atas elemen tombol (button). Ketika kursor berada di atas tombol tersebut, properti `color` digunakan untuk mengatur warna teks tombol, di mana nilai yang ditetapkan adalah `"#FF00FF"`, yang merupakan kode warna untuk warna magenta.

Kode program

```
button:hover {  
  
    color: #FF00FF;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengubah warna teks tombol menjadi magenta saat kursor diarahkan ke atas (hover) pada tombol tersebut. Ini memberikan umpan balik visual kepada pengguna bahwa tombol tersebut bisa diklik atau interaktif. Misalnya, jika Anda memiliki tombol HTML seperti `<button>Klik!</button>`, saat kursor diarahkan ke atas tombol tersebut, warna teksnya akan berubah menjadi magenta.

active

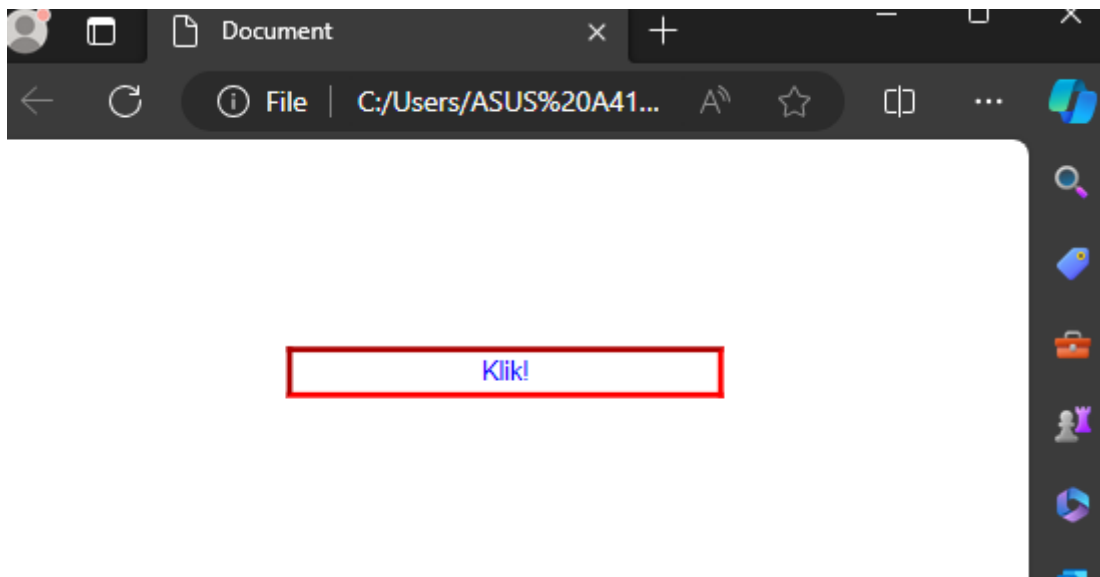
Penjelasan

Pernyataan tersebut menggunakan pseudo-class `:active` untuk menargetkan perilaku saat tombol sedang dalam keadaan aktif, yaitu saat tombol tersebut sedang ditekan oleh pengguna. Ketika tombol tersebut sedang ditekan, properti `color` digunakan untuk mengatur warna teks tombol, di mana nilai yang ditetapkan adalah `"#0000FF"`, yang merupakan kode warna untuk warna biru.

Kode program

```
button:active {  
  
    color: #0000FF;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengubah warna teks tombol menjadi biru saat tombol tersebut sedang dalam keadaan aktif, yaitu saat tombol ditekan oleh pengguna. Ini memberikan umpan balik visual kepada pengguna bahwa tindakan yang mereka lakukan sedang berlangsung. Misalnya, jika Anda memiliki tombol HTML seperti `<button>Klik!</button>`, saat tombol tersebut ditekan oleh pengguna, warna teksnya akan berubah menjadi biru.

link

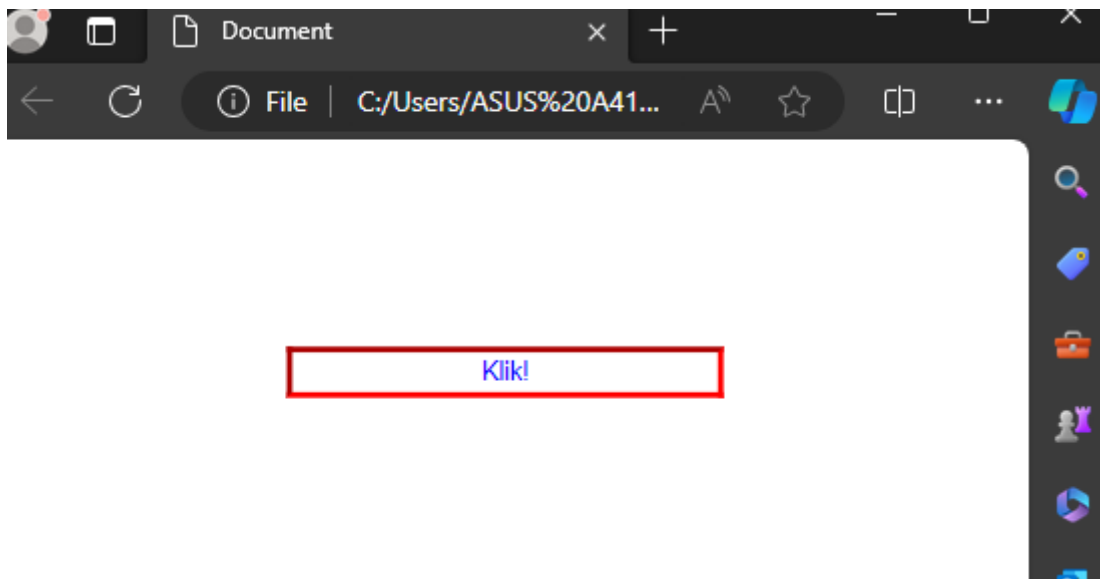
Penjelasan

Pernyataan tersebut menggunakan pseudo-class `:link` untuk menargetkan perilaku saat tombol tersebut berfungsi sebagai link (belum dikunjungi). Saat tombol tersebut dianggap sebagai link yang belum dikunjungi, properti `color` digunakan untuk mengatur warna teks tombol, di mana nilai yang ditetapkan adalah `"#FF0000"`, yang merupakan kode warna untuk warna merah.

Kode program

```
button:link {  
  
    color: #FF0000;  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengubah warna teks tombol menjadi merah saat tombol tersebut dianggap sebagai link yang belum dikunjungi. Ini memberikan umpan balik visual kepada pengguna bahwa tombol tersebut merupakan tautan yang dapat diklik. Misalnya, jika Anda memiliki tombol HTML seperti `<button>Klik!</button>` yang ditautkan ke halaman lain, saat tombol tersebut belum pernah dikunjungi oleh pengguna, warna teksnya akan berubah menjadi merah.

visited

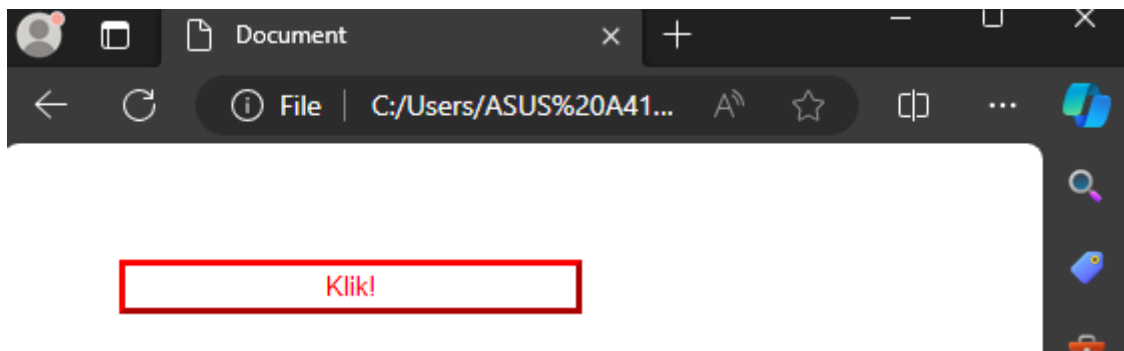
Penjelasan

Pernyataan tersebut menggunakan pseudo-class `:visited` untuk menargetkan perilaku saat tombol tersebut telah dikunjungi oleh pengguna. Saat tombol tersebut telah dikunjungi, properti `color` digunakan untuk mengatur warna teks tombol, di mana nilai yang ditetapkan adalah `"#00FF00"`, yang merupakan kode warna untuk warna hijau.

Kode program

```
button:visited {  
  
    color: #00FF00;  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengubah warna teks tombol menjadi hijau saat tombol tersebut telah dikunjungi oleh pengguna. Ini memberikan umpan balik visual kepada pengguna bahwa mereka telah mengunjungi tombol tersebut. Misalnya, jika Anda memiliki tombol HTML seperti `<button>Klik!</button>` yang telah dikunjungi oleh pengguna sebelumnya, warna teksnya akan berubah menjadi hijau.

Transitions

Transition

Penjelasan

Pernyataan tersebut menggunakan pseudo-class `:hover` untuk menargetkan perilaku saat kursor berada di atas elemen tombol (button). Ketika kursor berada di atas tombol tersebut, terdapat beberapa properti CSS yang berubah:

1. `background-color: red;` : Properti ini mengubah warna latar belakang tombol menjadi merah saat kursor diarahkan ke atas tombol.
2. `width: 200px;` : Properti ini mengubah lebar tombol menjadi 200 piksel saat kursor diarahkan ke atas tombol.
3. `color: white;` : Properti ini mengubah warna teks tombol menjadi putih saat kursor diarahkan ke atas tombol.
4. `transition: all 0.3s ease-in;` : Properti ini mengatur efek transisi untuk semua perubahan yang terjadi pada tombol saat kursor diarahkan ke atas. Transisi ini berlangsung selama 0.3 detik dan dimulai dengan kecepatan yang lambat.

Nilai yang digunakan dalam properti `transition` terdiri dari tiga bagian:

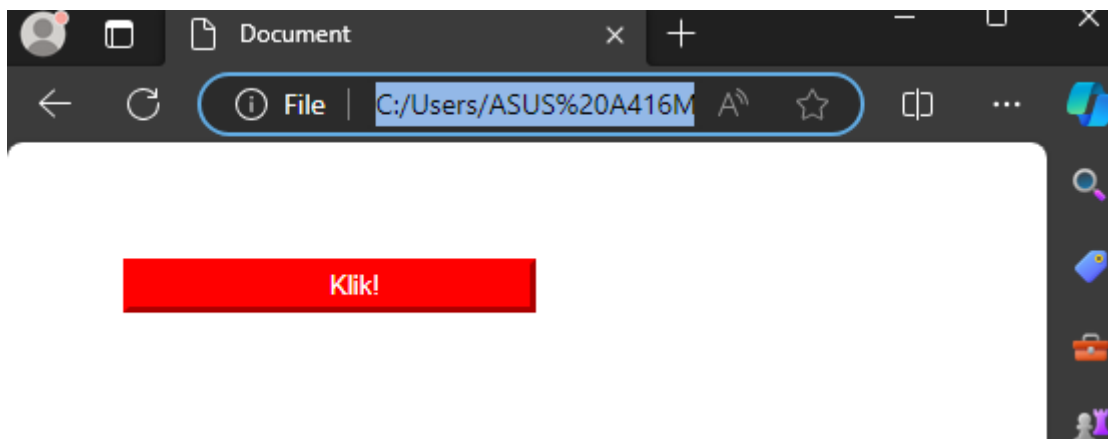
1. `all` : Ini adalah nilai yang menunjukkan bahwa semua properti CSS pada elemen akan mengalami transisi. Artinya, ketika ada perubahan pada elemen, semua properti CSS akan diterapkan transisi.

2. `0.3s` : Ini adalah durasi dari transisi, di mana "0.3s" menunjukkan bahwa transisi akan berlangsung selama 0.3 detik.
3. `ease-in` : Ini adalah fungsi waktu yang menentukan bagaimana kecepatan transisi berubah seiring waktu. Dalam kasus ini, `ease-in` menghasilkan transisi yang lambat pada awalnya, lalu semakin cepat mendekati akhir transisi.

Kode Program

```
button:hover {  
  
    background-color: red;  
  
    width: 200px;  
  
    color: white;  
  
    transition: all 0.3s ease-in;  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan mengubah tampilan tombol saat kursor diarahkan ke atasnya. Tombol akan memiliki latar belakang merah, lebar 200 piksel, dan teks berwarna putih. Perubahan-perubahan tersebut akan ditampilkan dengan efek transisi yang berlangsung selama 0.3 detik dan dimulai dengan kecepatan yang lambat. Misalnya, jika Anda memiliki tombol HTML seperti `<button>Click Me</button>`, saat kursor diarahkan ke atas tombol tersebut, tampilannya akan berubah sesuai dengan aturan CSS di atas.

Transition-delay

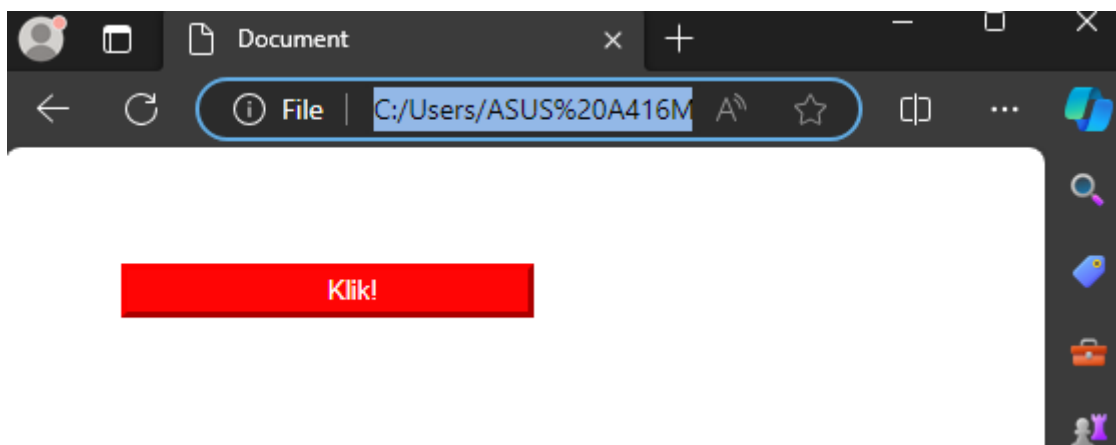
Penjelasan

`transition-delay: 1s;` : Properti ini mengatur penundaan transisi selama 1 detik sebelum perubahan dimulai.

Kode Program

```
button:hover {  
  
    background-color: red;  
  
    width: 200px;  
  
    color: white;  
  
    transition-duration: 2s;  
  
    transition-delay: 1s;
```

Hasil



Kesimpulan

Program tersebut akan membuat efek transisi pada tombol saat kursor diarahkan ke atasnya. Tombol akan memiliki latar belakang merah, lebar 200 piksel, dan teks berwarna putih. Perubahan tersebut akan ditampilkan dengan efek transisi yang dimulai setelah 1 detik penundaan dan berlangsung selama 2 detik. Misalnya, saat kursor diarahkan ke atas tombol, perubahan tampilan tombol akan dimulai setelah 1 detik dan akan berlangsung selama 2 detik dengan transisi yang halus.

Transition-duration

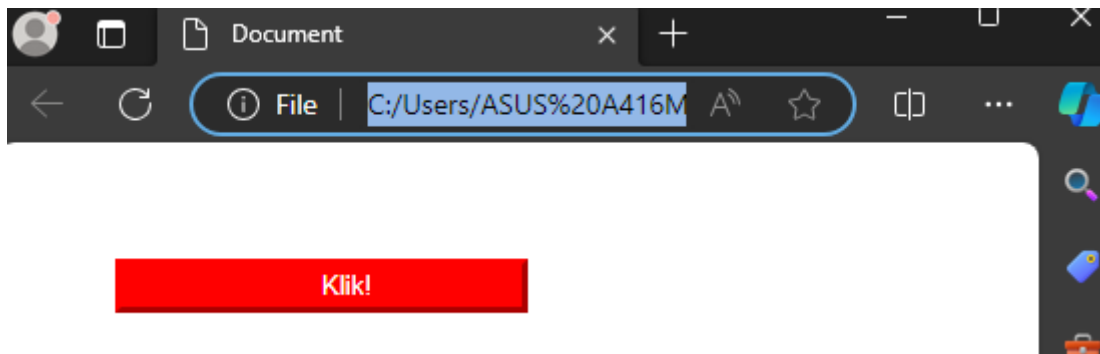
Penjelasan

Properti `transition-duration` digunakan untuk menentukan durasi dari transisi yang diterapkan pada elemen. Dalam hal ini, nilai yang ditetapkan adalah "2s", yang berarti transisi akan berlangsung selama 2 detik.

Kode Program

```
transition-duration: 2s;
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut akan membuat efek transisi pada elemen-elemen yang menerapkan transisi untuk memiliki durasi selama 2 detik. Ini berarti perubahan yang terjadi pada properti CSS yang ditentukan untuk transisi akan berlangsung secara halus selama 2 detik ketika terjadi perubahan pada elemen tersebut.

Transition-property

Penjelasan

`transition-property: width 300px;` : Menentukan properti mana yang akan mengalami transisi dan nilai dari properti tersebut selama transisi berlangsung. Dalam hal ini, saat tombol di-hover, transisi hanya diterapkan pada lebar (`width`) tombol dan lebar tombol akan berubah menjadi 300 piksel selama 2 detik.

Kode Program

```
button:hover {  
  
    background-color: red;
```

```
width: 200px;

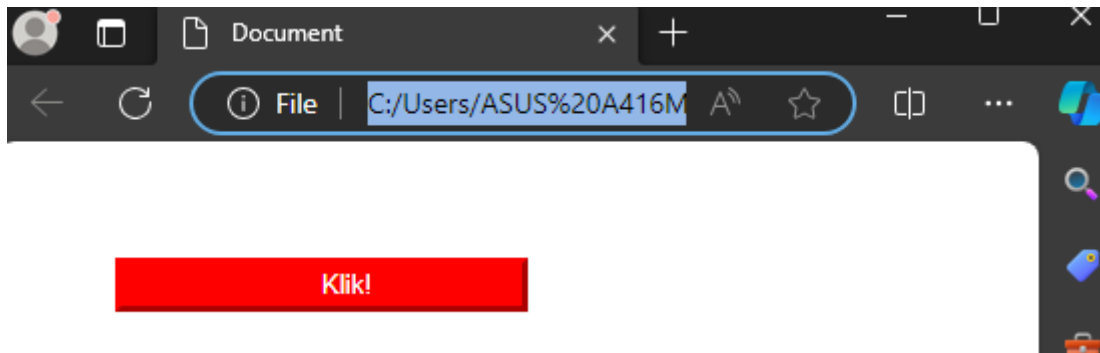
color: white;

transition-duration: 2s;

transition-property: width 300px;

}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat kursor diarahkan ke atasnya dengan efek transisi. Tombol akan berubah warna latar belakang menjadi merah, lebar tombol menjadi 200 piksel, dan warna teks tombol menjadi putih. Transisi hanya diterapkan pada perubahan lebar tombol, dengan lebar tombol yang berubah menjadi 300 piksel selama 2 detik. Ini memberikan efek visual yang halus dan menarik saat tombol di-hover oleh pengguna.

Transition-timing-function

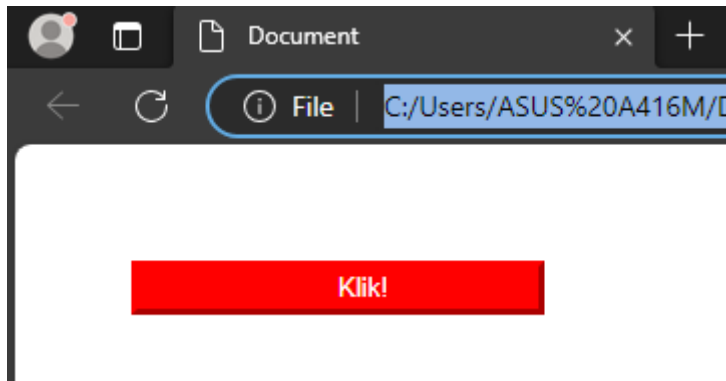
Penjelasan

`transition-timing-function: all 2s ease-in;` : Menentukan fungsi waktu transisi untuk semua properti yang mengalami transisi. Dalam hal ini, transisi akan memiliki durasi 2 detik dan efek "ease-in", yang berarti perubahan akan dimulai dengan kecepatan yang lambat dan semakin meningkat seiring waktu.

Kode Program

```
button:hover {  
  
    background-color: red;  
  
    width: 200px;  
  
    color: white;  
  
    transition-timing-function: all 2s ease-in ;  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat kursor diarahkan ke atasnya dengan efek transisi. Tombol akan berubah warna latar belakang menjadi merah, lebar tombol menjadi 200 piksel, dan warna teks tombol menjadi putih. Efek transisi akan memiliki durasi 2 detik dan akan dimulai dengan kecepatan yang lambat. Ini memberikan efek visual yang halus dan menarik saat tombol di-hover oleh pengguna.

Transition(Tantangan)

Kode Program

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <title>Document</title>  
    <link rel="stylesheet" href="tugasflex.css">  
</head>
```

```

<body bgcolor="purple">
  <div class="main-container">
    <div class="hero-container">
      <div class="item p">
        <p> Selamat Datang <br>
        di <b>Web Rahmat</b> </P>
        <button> klik saya </button>
      </div>
      <div>
        <span class="img">
          
        </span>
      </div>
    </div>
  </div>
</body>
</html>

```

Penjelasan HTML

- `<!DOCTYPE html>`, `<html lang="en">`, `<head>`, `<title>Document</title>` : Bagian awal dari dokumen HTML yang mendefinisikan tipe dokumen, bahasa, dan judul halaman.
- `<style>` : Mulai dari bagian CSS yang didefinisikan dalam dokumen HTML.
- `<body bgcolor="purple">` : Memberikan warna latar belakang halaman dengan atribut `bgcolor`.
- `<div class="main-container">`, `<div class="hero-container">` : Membuat dua div yang bertindak sebagai wadah utama dan sub-wadah.
- `<div class="item p">` : Sebuah div dengan kelas `item` dan `p` yang berisi teks dan tombol.
- `<p> Selamat Datang
 di Web Rahmat </p>` : Menampilkan teks "Selamat Datang" di web Rahmat.
- `<button> klik saya </button>` : Menampilkan tombol dengan teks "klik saya".
- `` : Menampilkan gambar dengan atribut `src`, `width`, `height`, dan `align`.

```

.main-container {
  display: flex;
  height: 100vh;
  justify-content: space-around ;
  align-items: center ;
  background-color: purple;
}

```

```
.hero-container {  
  display: flex;  
  height: 100vh;  
  justify-content: space-between;  
  align-items: center ;  
  background-color: purple;  
}
```

```
.item {  
  width: 500px;  
  height: 250px;  
  background-color: none;  
}
```

```
.p{  
  font-size: 65px;  
  font-family: 'arial';  
  margin-top: 40px;  
  margin-bottom: 100px;  
  margin-left: 50px;  
  margin-right: 100px;  
  color: aliceblue;  
}
```

```
img {  
  margin-right: 100px;  
  margin-top: -10px;  
  border: 10px solid white;  
  border-radius: 1500px 1500px;  
}
```

```
button {  
  background-color: purple;  
  width: 150px;  
  height: 50px;  
  border-width: 2px;  
  color: orange;  
  border-color: orange;  
  margin-bottom: 20px;  
  margin-left: 290px;  
  box-shadow: 20px;  
}
```

```
button:hover {  
  background-color: orange;
```

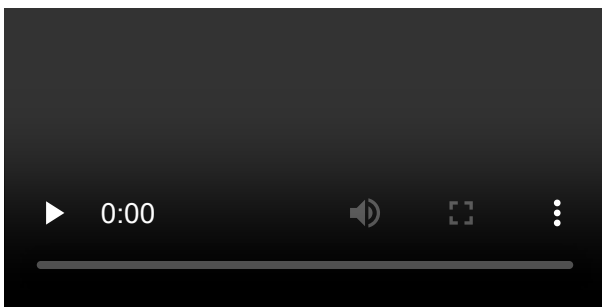
```
color: white;
width: 150px;
transition: all 0.3s ease-in;
cursor: pointer;
}

button:active {
  transform: scale(0.5);
}
```

Penjelasan CSS

- `.main-container`, `.hero-container` : Membuat tata letak menggunakan flexbox dengan menentukan dimensi, warna latar belakang, dan penempatan.
- `.item` : Mengatur dimensi dan warna latar belakang untuk item.
- `.p` : Mengatur gaya teks untuk paragraf.
- `img` : Mengatur margin dan gaya border untuk gambar.
- `button` : Mengatur gaya untuk tombol, termasuk dimensi, warna latar belakang, warna teks, dan margin.
- `button:hover` : Mengatur gaya tombol saat dihover dengan mengubah warna latar belakang, warna teks, dan ukuran tombol.
- `button:active` : Mengatur gaya tombol saat ditekan dengan melakukan transformasi scaling

Hasil



Transform

Transform scale

Penjelasan

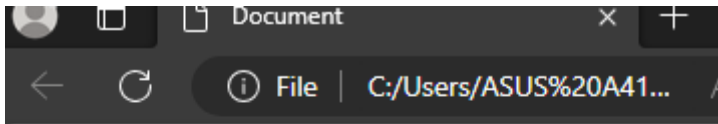
`button:active` adalah pseudo-class CSS yang menargetkan perilaku tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna.

`transform: scale(0.75);` : Properti ini menggunakan fungsi transformasi `scale()` untuk mengubah ukuran tombol. Saat tombol sedang ditekan (active), tombol akan diperkecil sebanyak 0.75 kali dari ukuran aslinya. Ini akan memberikan efek visual seperti tombol yang "ditekan" ke dalam saat ditekan oleh pengguna.

Kode Program

```
button:active {  
    transform: scale(0.75);  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi `scale()` digunakan untuk memperkecil ukuran tombol menjadi 0.75 kali ukuran aslinya saat tombol ditekan, memberikan efek visual yang menarik dan memberi umpan balik kepada pengguna bahwa tombol telah ditekan.

Transform scaleX

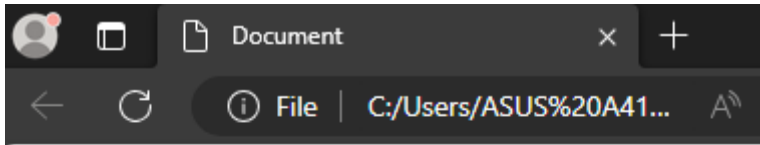
Penjelasan

`transform: scaleX(0.5);` : Properti ini menggunakan fungsi transformasi `scaleX()` untuk mengubah skala lebar tombol. Saat tombol sedang ditekan (active), tombol akan diperkecil sebanyak setengah (0.5) dari lebar aslinya. Ini akan memberikan efek visual seperti tombol yang "ditekan" ke dalam secara horizontal saat ditekan oleh pengguna.

Kode Program

```
button:active {  
  
    transform: scaleX(0.5);  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi `scaleX()` digunakan untuk memperkecil lebar tombol menjadi setengah dari lebar aslinya saat tombol ditekan, memberikan efek visual yang menarik dan memberi umpan balik kepada pengguna bahwa tombol telah ditekan.

Transform rotate

Penjelasan

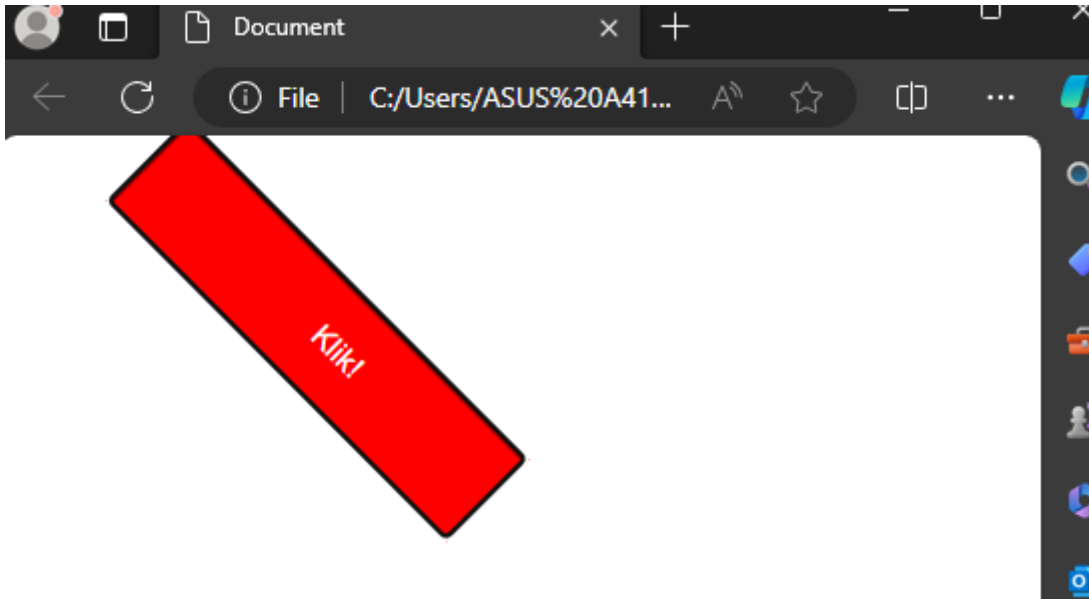
`transform: rotate(45deg);` : Properti ini menggunakan fungsi transformasi `rotate()` untuk memutar tombol sebesar 45 derajat saat tombol sedang ditekan (active). Ini akan memberikan efek visual seperti tombol yang diputar sebesar 45 derajat saat ditekan oleh pengguna.

Kode Program

```
button:active {  
  
    transform: rotate(45deg);  
  
}
```

```
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi `rotate()` digunakan untuk memutar tombol sebesar 45 derajat saat tombol ditekan, memberikan efek visual yang menarik dan memberi umpan balik kepada pengguna bahwa tombol telah ditekan.

Transform skewX

Penjelasan

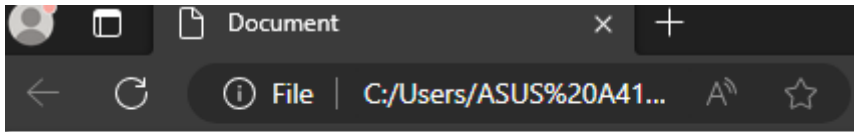
`transform: skewx(-25deg);` : Properti ini menggunakan fungsi transformasi `skewx()` untuk memiringkan tombol sebesar -25 derajat secara horizontal (ke samping) saat tombol sedang ditekan (active). Nilai negatif (-25deg) mengindikasikan bahwa miringan akan berlawanan arah jarum jam. Ini akan memberikan efek visual seperti tombol yang miring ke kiri saat ditekan oleh pengguna.

Kode Program

```
button:active {  
    transform: skewx(-25deg);
```

```
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi `skewx()` digunakan untuk memiringkan tombol sebesar -25 derajat secara horizontal saat tombol ditekan, memberikan efek visual yang menarik dan memberi umpan balik kepada pengguna bahwa tombol telah ditekan.

Transform skew

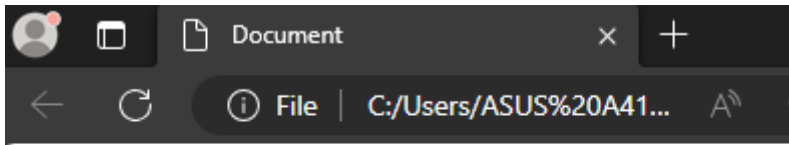
Penjelasan

`transform: skew(20deg, 5deg);` : Properti ini menggunakan fungsi transformasi `skew()` untuk memiringkan tombol sebesar 20 derajat secara horizontal (ke samping) dan 5 derajat secara vertikal (ke atas atau ke bawah) saat tombol sedang ditekan (active). Ini akan memberikan efek visual seperti tombol yang terlihat miring saat ditekan oleh pengguna.

Kode Program

```
button:active {  
  
    transform: skew(20deg, 5deg);  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi `skew()` digunakan untuk memiringkan tombol sebesar 20 derajat secara horizontal dan 5 derajat secara vertikal saat tombol ditekan, memberikan efek visual yang menarik dan memberi umpan balik kepada pengguna bahwa tombol telah ditekan.

Transform translate

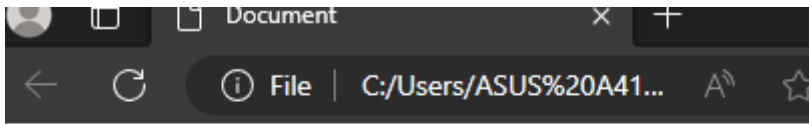
Penjelasan

`transform: translate(5px, 12px);` : Properti ini menggunakan fungsi transformasi `translate()` untuk memindahkan (translasi) posisi tombol sejauh 5 piksel ke kanan dan 12 piksel ke bawah saat tombol sedang ditekan (active). Ini akan memberikan efek visual seperti tombol yang bergeser ke arah kanan dan ke bawah saat ditekan oleh pengguna

Kode Program

```
button:active {  
  
    transform: translate(5px, 12px);  
  
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi `translate()` digunakan untuk memindahkan posisi tombol sejauh 5 piksel ke kanan dan 12 piksel ke bawah saat tombol ditekan, memberikan efek visual yang menarik dan memberi umpan balik kepada pengguna bahwa tombol telah ditekan.

Transform matrix

Penjelasan

- `transform: matrix(0.7, -0.5, 0.5, 0.4, 0.5, 0.7);` : Properti ini menggunakan fungsi transformasi `matrix()` untuk menerapkan transformasi 2D kompleks pada tombol saat tombol sedang ditekan (active). Matriks transformasi 2D yang diberikan memiliki nilai sebagai berikut:
 - Kolom pertama: `scaleX = 0.7`, `skewY = -0.5`.
 - Kolom kedua: `skewX = 0.5`, `scaleY = 0.4`.
 - Kolom ketiga: `translateX = 0.5`, `translateY = 0.7`.

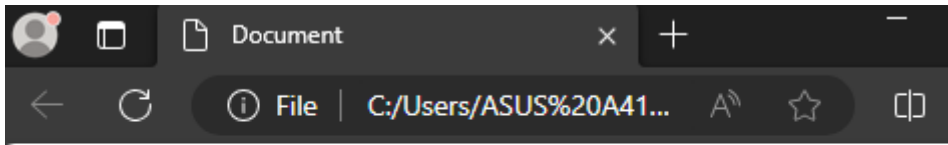
Ini akan memberikan efek visual transformasi kompleks pada tombol saat ditekan

Kode Program

```
button:active {  
  
    transform: matrix(0.7, -0.5, 0.5, 0.4, 0.5, 0.7);
```

```
}
```

Hasil



Kesimpulan

Kesimpulan: Program tersebut mengatur tampilan tombol saat tombol sedang dalam keadaan aktif, yaitu saat tombol sedang ditekan oleh pengguna. Efek transformasi matriks kompleks diberikan untuk memberikan perubahan yang lebih rumit pada tombol saat tombol ditekan. Hal ini dapat memberikan efek visual yang menarik dan unik tergantung pada nilai matriks yang diberikan.

Flexbox

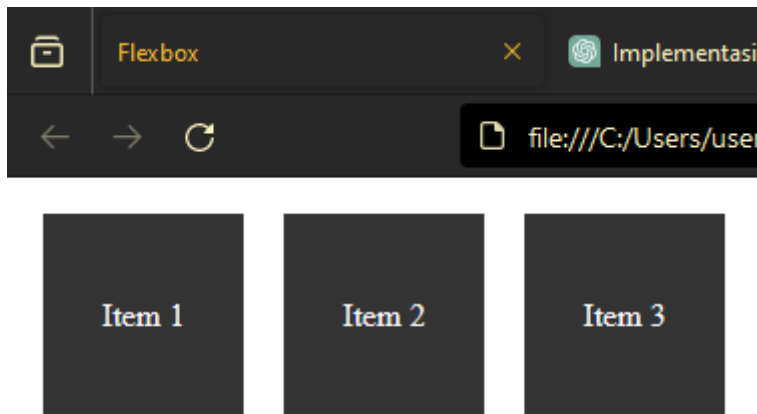
Display Flex

`display: flex`, kita dapat dengan mudah mengatur tata letak elemen-elemen di dalamnya dengan properti-properti seperti `flex-direction`, `justify-content`, `align-items`, dan lain-lain.

Kode Program

```
.box-container{  
  display: flex; /* Item-item di dalam flex container juga menjadi flex items */  
}
```

Hasil



Kesimpulan

display: flex , kita dapat dengan mudah mengatur tata letak elemen-elemen di dalamnya menggunakan properti-properti seperti flex-direction , justify-content , align-items , dan properti lainnya yang terkait dengan model tata letak flexbox.

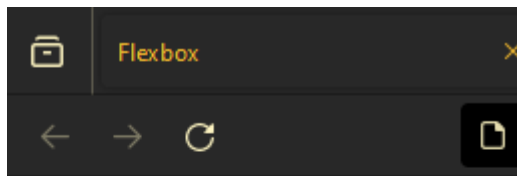
Flex-Direction

- flex-direction: column; : Mengatur tata letak dari flex container menjadi vertikal, sehingga flex items diatur dari atas ke bawah.
- flex-direction: column-reverse; : Mengatur tata letak dari flex container menjadi vertikal terbalik, sehingga flex items diatur dari bawah ke atas.
- flex-direction: row; : Mengatur tata letak dari flex container menjadi horizontal, sehingga flex items diatur dari kiri ke kanan.
- flex-direction: row-reverse; : Mengatur tata letak dari flex container menjadi horizontal terbalik, sehingga flex items diatur dari kanan ke kiri.

Kode Program

```
.container{  
  flex-direction: column;  
}
```

Hasil



Item 1

Item 2

Item 3

Kesimpulan

Setiap properti flex-direction mengubah tampilan elemen dengan cara yang berbeda-beda sesuai dengan property.

Contoh `column` untuk tata letak vertikal dari atas ke bawah.

`column-reverse` untuk tata letak vertikal terbalik dari bawah ke atas. `row` untuk tata letak horizontal dari kiri ke kanan. `row-reverse` untuk tata letak horizontal terbalik dari kanan ke kiri.

Align-Items

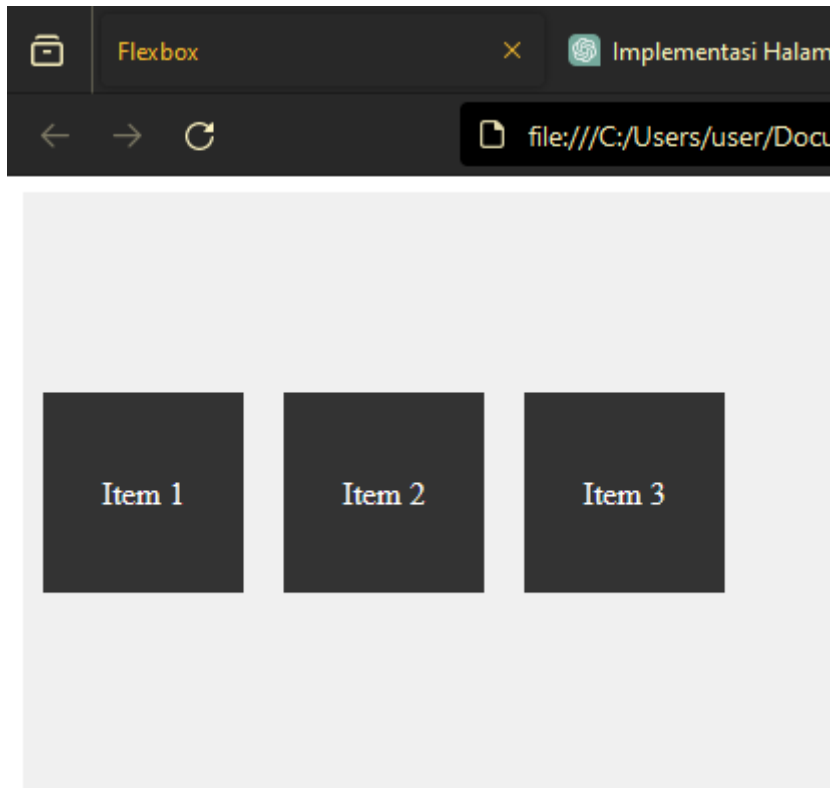
- `align-items: center;` : Flex items akan diatur di tengah-tengah sumbu silang dari flex container.
- `align-items: flex-start;` : Flex items akan diatur pada awal sumbu silang dari flex container.
- `align-items: flex-end;` : Flex items akan diatur pada akhir sumbu silang dari flex container.
- `align-items: baseline;` : Flex items akan diatur sedemikian rupa sehingga garis dasar dari teks pada masing-masing item berada pada level yang sama.

- `align-items: stretch;` : Flex items akan diperpanjang untuk mencapai tinggi maksimal flex container, mengisi ruang kosong di sepanjang sumbu silang.

Kode Program

```
.box-container{  
  align-items:center ;  
}
```

Hasil



Kesimpulan

Setiap properti `align-items` mengubah tampilan elemen dengan cara yang berbeda-beda sesuai dengan property.

Contoh `center` untuk menempatkan flex items di tengah-tengah sumbu silang. `flex-start` untuk menempatkan flex items di awal sumbu silang. `flex-end` untuk menempatkan flex items di akhir sumbu silang. `baseline` untuk menempatkan flex items sehingga garis dasar teks masing-masing item berada pada level yang sama. `stretch` untuk memperpanjang flex items sehingga mencapai tinggi maksimal flex container, mengisi ruang kosong di sepanjang sumbu silang.

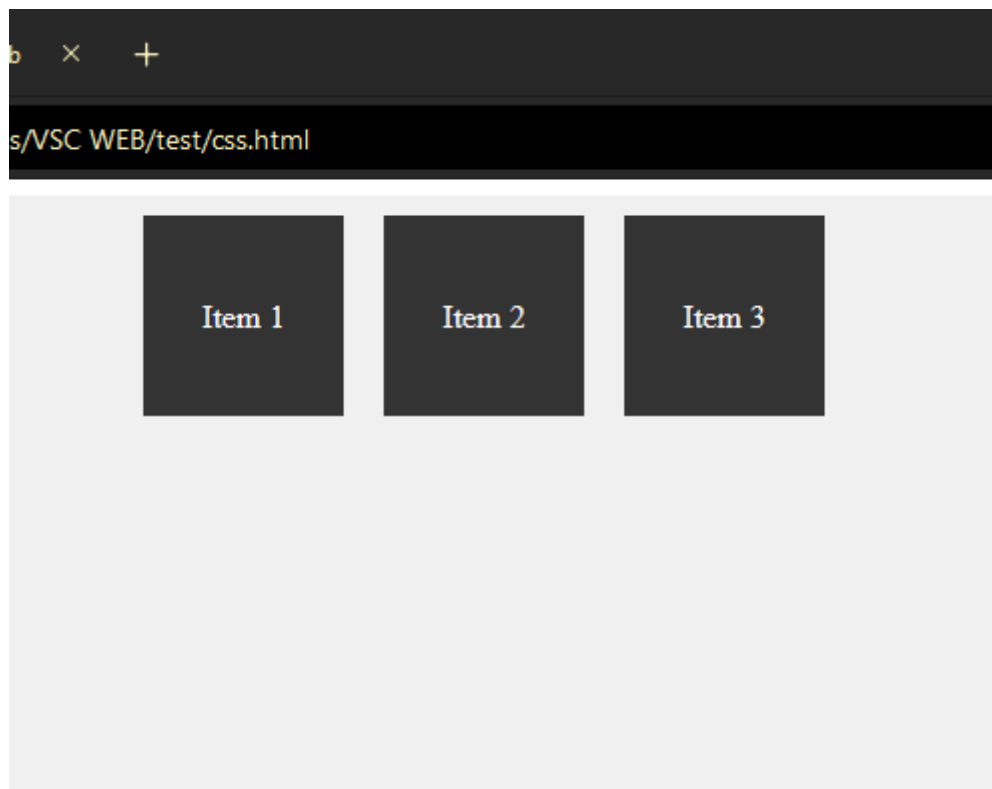
Justify-Content

- `justify-content: flex-start;` : Mengatur fleks item ditempatkan di bagian awal (mulai) dari sumbu utama flex container.
- `justify-content: flex-end;` : Mengatur fleks item ditempatkan di bagian akhir (akhir) dari sumbu utama flex container.
- `justify-content: center;` : Mengatur fleks item ditempatkan di tengah-tengah sumbu utama flex container.
- `justify-content: space-around;` : Mengatur fleks item didistribusikan secara merata di sekitar sumbu utama flex container, dengan ruang yang sama di antara mereka.
- `justify-content: space-between;` : Mengatur fleks item didistribusikan secara merata di sepanjang sumbu utama flex container, dengan ruang yang sama di antara fleks item pertama dan terakhir, tetapi tidak ada ruang di antara fleks item yang berdekatan.

Kode Program

```
.box-container{  
  justify-content: center ;  
}
```

Hasil



Kesimpulan

Setiap properti justify-Content mengubah tampilan elemen dengan cara yang berbeda-beda sesuai dengan property.

Contoh `flex-start` untuk menempatkan flex items di awal (mulai) sumbu utama. `flex-end` untuk menempatkan flex items di akhir (akhir) sumbu utama. `center` untuk menempatkan flex items di tengah-tengah sumbu utama. `space-around` untuk mendistribusikan flex items secara merata di sekitar sumbu utama dengan ruang yang sama di antara mereka. `space-between` untuk mendistribusikan flex items secara merata di sepanjang sumbu utama dengan ruang yang sama di antara flex items pertama dan terakhir, tetapi tidak ada ruang di antara flex items yang berdekatan.

Flexbox (Tantangan)

Penjelasan HTML

- `<!DOCTYPE html>` : Mendefinisikan jenis dokumen HTML.
- `<html lang="en">` : Mengawali dokumen HTML dengan bahasa Inggris.
- `<head>` : Bagian kepala dokumen yang berisi informasi meta dan judul.
- `<meta charset="UTF-8">` : Mendefinisikan karakter set dokumen sebagai UTF-8.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">` : Mengatur tampilan responsif di perangkat dengan lebar layar yang berbeda.
- `<title>Document</title>` : Judul halaman web.
- `<link rel="stylesheet" href="tampilanweb.css">` : Menghubungkan halaman HTML dengan file CSS eksternal.
- `<body bgcolor="purple">` : Mengatur warna latar belakang halaman menjadi ungu (purple).
- `` : Membuat bagian teks menjadi bagian inline.
- `<div class="text">` : Membuat sebuah div dengan kelas "text" yang berisi teks "Selamat datang".
- `` : Membuat sebuah span dengan kelas "text2" yang berisi teks "di web Rahmat".
- `` : Menampilkan gambar dengan nama "naon.jpg" dengan kelas "foto".
- `<button class="tombol">Klik ini</button>` : Membuat sebuah tombol dengan teks "Klik ini" dan kelas "tombol".

Penjelasan CSS

- `.text` dan `.text2` : Mengatur gaya teks untuk kelas "text" dan "text2". Menggunakan font Times New Roman, ukuran font 30mm, dan warna putih.

- `.foto` : Mengatur tampilan gambar dengan kelas "foto". Menerapkan display block, membuat sudut gambar menjadi bulat dengan border-radius, mengatur lebar dan tinggi gambar, mengatur posisi gambar dengan margin-left dan margin-top, serta memberikan border putih dengan lebar 10px dan gaya solid.
- `.tombol` : Mengatur tampilan tombol dengan kelas "tombol". Memberikan warna teks dan border oranye, mengatur lebar dan tinggi tombol, mengatur tampilan tombol dengan display block, memberikan warna latar belakang transparan, mengatur border dengan lebar 3px dan gaya solid, mengatur posisi tombol dengan margin-top dan margin-left, serta mengatur ukuran font.

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="tampilanweb.css">
</head>
<body bgcolor="purple">
  <span>
    <div class="text">Selamat datang
    </div>
  </span><br>
  <span class="text2">di web farhan</span><br>
  
  <button class="tombol">Klik ini</button>
</body>
</html>
```

```
.text{
  font-family: 'Times New Roman';
  font-size: 30mm;
  color: white;
}

.text2{
  font-family: 'Times New Roman';
  font-size: 30mm;
  color: white;
}
```

```
.foto{
  display: block;
  border-radius: 1000px;
  width: 400px;
  height: 400px;
  margin-left: 735px;
  margin-top: -215px;
  border-width: 10px;
  border-color: white;
  border-style: solid;
}

.tombol{
  color: orangered;
  border-color: orangered;
  width: 160px;
  height: 85px;
  display: block;
  background-color: transparent;
  border-width: 3px;
  border-style: solid;
  margin-top: -200px;
  margin-left: 13cm;
  font-size: 20px;
}
```

Hasil

Selamat datang
di web farhan

[Klik ini](#)



Position

Position Relative

Penjelasan

- Dalam HTML, kita memiliki sebuah div dengan kelas `.container` yang berisi tiga div dengan kelas `.box`.
- Dalam CSS, kita menerapkan `position: relative` pada `.container` untuk membuatnya menjadi referensi bagi posisi kotak-kotak anaknya yang memiliki `position: absolute`.
- Setiap kotak diberi properti `position: absolute` untuk memungkinkan kita menentukan posisi mereka relatif terhadap posisi normalnya.
- Properti `top` dan `left` diberikan pada masing-masing kotak untuk menentukan jarak vertikal dan horizontal dari posisi normalnya.

Kode Program

```
.container {  
  position: relative;  
  width: 400px;  
  height: 200px;  
  border: 2px solid #333;  
}  
  
.box {  
  position: relative;  
  width: 100px;  
  height: 100px;  
  background-color: #333;  
  color: #fff;  
  text-align: center;  
  line-height: 100px;  
}
```

Hasil



Kesimpulan

- Ketika sebuah elemen diberi `position: relative`, itu akan tetap berada dalam aliran dokumen normal, tetapi kita dapat menggunakan properti `top`, `right`, `bottom`, atau `left` untuk menyesuaikan posisinya relatif terhadap posisi normalnya.
- Ini berguna ketika kita ingin menyesuaikan posisi sebuah elemen tetapi tetap mempertahankan ruangnya dalam aliran dokumen normal.
- Elemen-elemen dengan `position: relative` akan mempengaruhi posisi elemen lain dalam dokumen.

Position Absolute

Penjelasan

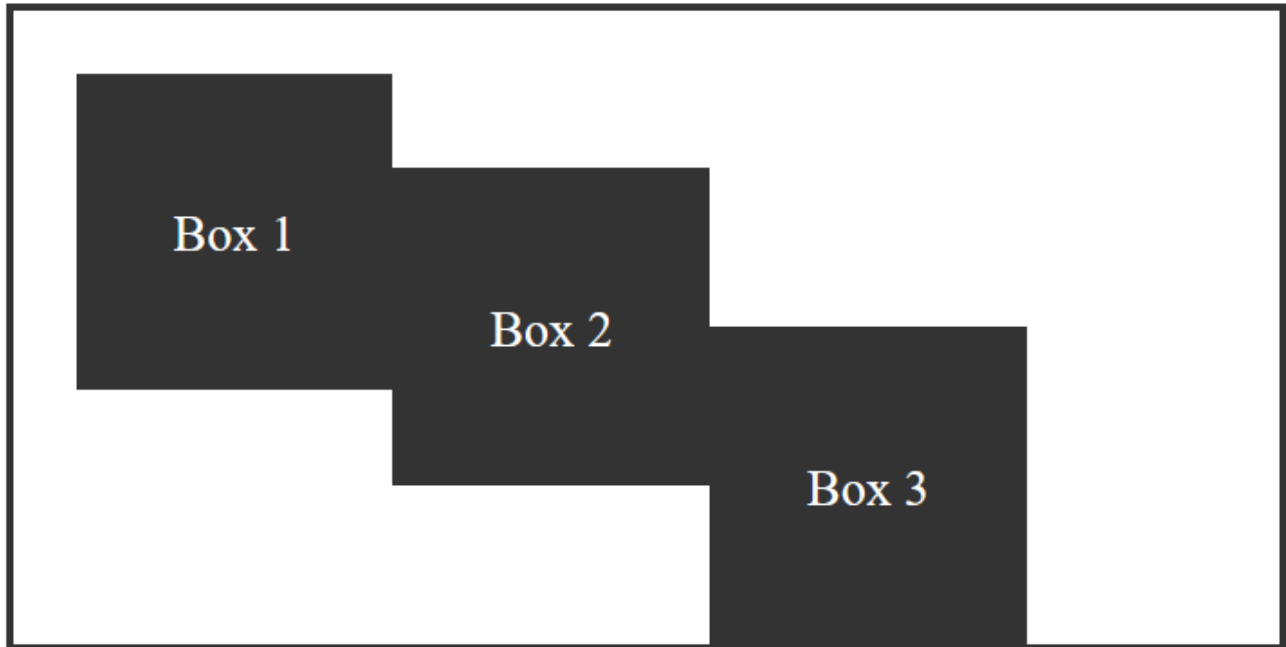
- Dalam HTML, kita memiliki sebuah div dengan kelas `.container` yang berisi tiga div dengan kelas `.box`.

- Dalam CSS, kita menerapkan `position: relative` pada `.container` untuk membuatnya menjadi referensi bagi posisi kotak-kotak yang memiliki `position: absolute`.
- Setiap `.box` diberi properti `position: absolute` untuk memungkinkan kita menentukan posisi mereka relatif terhadap `.container`.
- Properti `top` dan `left` diberikan pada masing-masing `.box` untuk menentukan jarak vertikal dan horizontal dari posisi `.container`.

Kode Program

```
.container {  
  position: absolute;  
  width: 400px;  
  height: 200px;  
  border: 2px solid #333;  
}  
  
.box {  
  position: absolute; /* Menggunakan posisi absolut */  
  width: 100px;  
  height: 100px;  
  background-color: #333;  
  color: #fff;  
  text-align: center;  
  line-height: 100px;  
}
```

Hasil



Kesimpulan

- Ketika sebuah elemen diberi `position: absolute`, itu akan dihapus dari aliran dokumen normal dan diposisikan relatif terhadap kontainer terdekat yang memiliki `position: relative`.
- Ini berguna ketika kita ingin menempatkan elemen secara spesifik di dalam kontainer atau dalam hubungan terhadap elemen lain tanpa memengaruhi layout aliran dokumen.
- Elemen-elemen dengan `position: absolute` tidak mempengaruhi posisi elemen lain dalam dokumen, kecuali elemen yang memiliki `position: relative` dan menjadi kontainernya.

Position Fixed

Penjelasan

Kode Program

```
.fixed-box {  
  position: fixed; /* Menggunakan posisi fixed */  
  top: 80px;  
  left: 50px;  
  width: 200px;  
  height: 100px;  
  background-color: #333;
```

```
color: #fff;
padding: 10px;
}
```

Hasil



Kesimpulan

position: fixed, elemen akan tetap berada pada posisinya yang ditentukan jika discroll. properti top dan left digunakan untuk menentukan jarak elemen dari tepi atas dan kiri.

Position Sticky

Penjelasan

- Kami memiliki sebuah div untuk konten (`content`) di HTML.
- Di CSS, kami memberikan padding pada `.content` untuk memberikan ruang di sekitar konten.
- Kami menambahkan sebuah div dengan kelas `.sticky-box`, yang akan dijadikan elemen sticky. Properti `position: sticky` diterapkan untuk membuatnya tetap menempel di bagian atas layar saat pengguna menggulir.

Kode Program

```

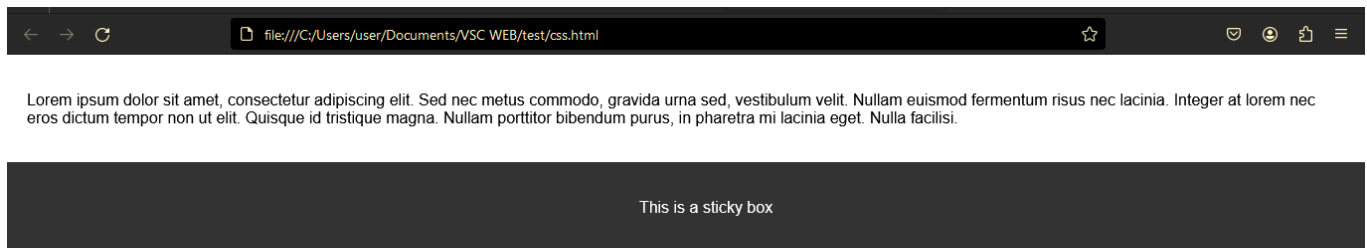
body {
    margin: 0;
    font-family: Arial, sans-serif;
}

.content {
    padding: 20px;
}

.sticky-box {
    background-color: #333;
    color: #fff;
    text-align: center;
    position: sticky;
    top: 0;
    width: 100%;
    padding: 20px;
}

```

Hasil



Kesimpulan

`position: sticky`, elemen akan seperti posisi relatif sampai jarak scroll mencapai nilai tertentu. elemen akan tetap berada pada posisi yang telah ditentukan saat jarak scroll mencapai nilai tertentu, sambil tetap mengikuti pergeseran scroll di bawahnya.

Positon(Tantangan)

Kode Program

```

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

```

```

<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Tantangan Position</title>
</head>
<body>
  <div class="container">
    <div class="item1">
      
    </div>
    <div class="item2">
      <p class="text1">Thursday, July 16, 2015</p>
      <h4 class="text2">The standard chunk of Lorem Ipsum</h4>
      <p class="text3">Sed posuere consectetur est at lobortis. Aenean eu leo
quam.</p>
    </div>
    <div class="item3">
      <p class="text4">Read more</p>
      
    </div>
    <div class="like">
      <button>
        
      </button>
    </div>
  </div>
</body>
</html>

```

Penjelasan HTML

- `<!DOCTYPE html>` : Mendefinisikan tipe dokumen HTML.
- `<html lang="en">` : Menandakan awal dari dokumen HTML dengan bahasa Inggris.
- `<head>` : Bagian kepala dokumen HTML yang berisi metadata dan referensi ke stylesheet.
- `<meta charset="UTF-8">` : Menentukan pengkodean karakter dokumen sebagai UTF-8.
- `<meta http-equiv="X-UA-Compatible" content="IE=edge">` : Menentukan versi IE yang kompatibel.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">` : Menyesuaikan tampilan halaman dengan lebar perangkat.
- `<title>Tantangan Position</title>` : Memberikan judul halaman.
- `<style>` : Mulai dari bagian CSS yang didefinisikan dalam dokumen HTML.
- `<body>` : Bagian utama dari dokumen HTML yang berisi konten yang akan ditampilkan di browser.

- `<div class="container">` : Sebuah div yang bertindak sebagai wadah untuk semua konten. Ini memiliki beberapa div anak yang akan di-styling.
- `<div class="item1">`, `<div class="item2">`, `<div class="item3">` : Div-div yang bertindak sebagai item atau bagian-bagian dari konten.
- `` : Elemen gambar untuk menampilkan gambar.
- `<p>`, `<h4>` : Elemen paragraf dan heading untuk menampilkan teks.
- `<button>` : Tombol untuk berbagai fungsi, seperti menyukai konten.
- `<div class="like">` : Sebuah div yang berisi tombol "suka" atau "like".

```
body{
    background-color: aqua;
}

.container {
    display: flex;
    flex-direction: column;
    background-color: white;
    height: 458px;
    width: 300px;
    border-radius: 10px;
    margin-top: 150px;
    margin-left: 720px;
    position: relative;
}

.item1{
    background-color: red;
    height: 250px;
    width: 300px;
    border-radius: 10px 10px 0px 0px;
}

img {
    width:100%;
    height:100%;
    border-radius: 5px 5px 0px 0px;
}

.item2{
    background-color: whitesmoke;
    height: 175px;
    width: 300px;
    justify-content: center;
    align-items: center;
}
```

```
}

.text1 {
  font-size: small;
  font-family: Arial, Helvetica, sans-serif;
  margin-left: 20px;
}

.text2 {
  font-size: 20px;
  font-family: Arial, Helvetica, sans-serif;
  margin-left: 20px;
}

.text3 {
  margin-left: 20px;
  font-family: Arial, Helvetica, sans-serif;
  font-size: medium;
  margin-bottom: 30px;
}

.item3 {
  display: flex;
  flex-direction: row;
  padding: 2px;
  background-color: gainsboro;
  border-radius: 0px 0px 5px 5px;
  width: 296px;
  justify-content: space-between;
  font-weight: bold;
}

.text4 {
  margin-left: 20px;
  font-family: Arial, Helvetica, sans-serif;
}

button {
  width: 62px;
  height: 55px;
  background-color: transparent;
  border: none;
  border-radius: 100px 100px 100px 100px;
  position: fixed;
  background-repeat: no-repeat;
  top: 350px;
```

```

        right: 520px;

    }

    .like img {
        border-radius: 100% ;
        margin-right: 110px;
        margin-left: 135px;
        position: absolute;
    }

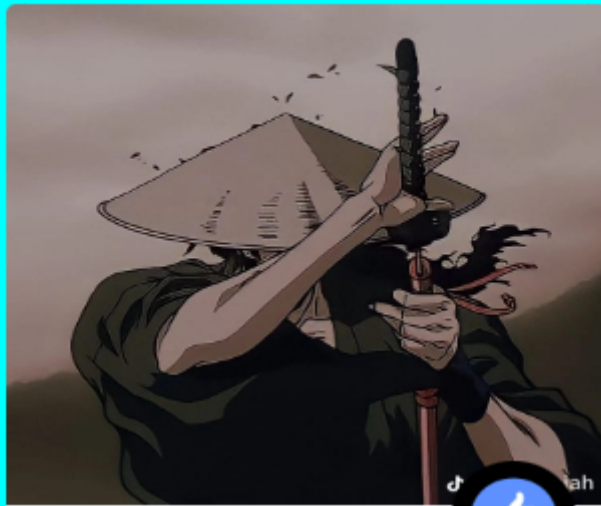
    .item3 img {
        height: 30px;
        width: 30px;
        border-radius: 100% 100% ;
        align-items: center;
        justify-items: center;
        margin-right: 25px;
        margin-top: 10px;
    }

```

Penjelasan CSS

- `body` : Memberi warna latar belakang aqua pada halaman.
- `.container` : Membuat tata letak kontainer menggunakan flexbox, memberi warna latar belakang putih, dan menentukan dimensi dan posisi.
- `.item1`, `.item2`, `.item3` : Masing-masing memberikan warna latar belakang dan menentukan dimensi dan posisi elemen.
- `img` : Mengatur dimensi gambar dan memberikan efek sudut melengkung pada beberapa gambar.
- `.text1`, `.text2`, `.text3`, `.text4` : Mengatur gaya teks untuk paragraf dan judul.
- `button` : Mengatur gaya untuk tombol, termasuk dimensi, warna latar belakang transparan, dan posisi.
- `.like img` : Mengatur gaya untuk gambar di dalam tombol "suka".
- `.item3 img` : Mengatur gaya untuk gambar di dalam item 3.

Hasil



Thursday, July 16, 2015



The standard chunk of Lorem Ipsum

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

[Read more](#)



Kesimpulan

Program ini menggunakan HTML untuk membuat struktur konten dan CSS untuk merancang tampilannya. Elemen-elemen seperti gambar, teks, dan tombol diberikan gaya untuk menciptakan tampilan yang menarik dan responsif. Penggunaan posisi relatif, absolut, dan sticky memungkinkan penempatan yang tepat dari elemen-elemen tersebut di halaman.