

Anatomi CSS

Kode Program

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

Penjelasan

- Tag `<p>` adalah selector yang ingin di modifikasi.
- property adalah `color` pada komponen textnya
- property values adalah `red` kita mau modifikasi seperti apa warnanya

Kesimpulan

warna teks (text color) dari elemen HTML yang memiliki tag P akan diubah menjadi merah (red).

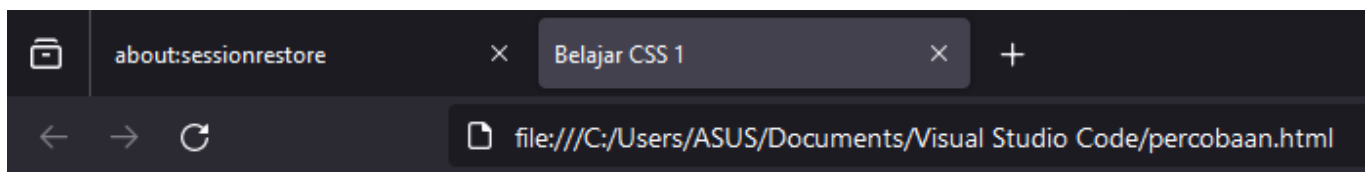
Percobaan pertama

Kode Program

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Belajar CSS 1</title>  
    <style>  
      p {
```

```
        color: red;
    }
</style>
</head>
<body>
    <p>Welcome CSS</p>
</body>
</html>
```

Hasil



welcome CSS

Penjelasan

1. `<!DOCTYPE html>` : Mendefinisikan jenis dokumen HTML yang digunakan, dalam hal ini HTML5.
2. `<html>` : Elemen utama yang memuat seluruh konten dokumen.
3. `<head>` : Bagian yang berisi informasi tambahan tentang dokumen, seperti judul dan link ke stylesheet eksternal.
4. `<title>` : Menentukan judul halaman web yang akan ditampilkan di tab browser.
5. `<style>` : Bagian di mana Anda dapat menambahkan aturan CSS untuk mengubah tampilan elemen HTML di halaman.

6. `p { color: red; }` : Aturan CSS yang mengubah warna teks pada semua elemen `<p>` menjadi merah.
7. `<body>` : Bagian yang berisi konten aktual halaman web, seperti teks, gambar, atau elemen lainnya.
8. `<p>Welcome CSS</p>` : Elemen paragraf dengan teks "Welcome CSS", yang akan ditampilkan dengan warna merah karena aturan CSS yang telah ditentukan sebelumnya.

Percobaan Kedua

Kode CSS

```
button {  
    font-size: 20px;  
    border: none;  
    width: 150px;  
    height: 50px;  
    font-family: 'sans-serif';  
}
```

Font-size

Before

"CSS/aset css/Screenshot(7).png" could not be found.

After

"CSS/aset css/Screenshot(6).png" could not be found.

Font-size adalah property CSS yang berfungsi untuk mengatur ukuran font.

Border

Before

“CSS/aset css/Screenshot(7).png” could not be found.

After

“CSS/aset css/Screenshot(6).png” could not be found.

🔗 [Apa itu Border? >](#)

Border merupakan property CSS yang berguna untuk mengatur garis pinggir dari tombol.

Font-family

Before

“CSS/aset css/Screenshot(7).png” could not be found.

After

“CSS/aset css/Screenshot(6).png” could not be found.

🔗 [Apa itu font-family? >](#)

font-family adalah property CSS yang berfungsi untuk mengubah jenis font.

Cara pemanggilan CSS

IN-LINE

Penggunaan Inline CSS: 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: 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: 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

Elemen Selector

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.

Class Selector

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.

Id Selector

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.

TEXT

Text decoration

Penjelasan

- `text-decoration: overline;` digunakan untuk menambahkan garis atas pada teks.
- `text-decoration: underline;` digunakan untuk menambahkan garis bawah pada teks.
- `text-decoration: line-through;` digunakan untuk menambahkan garis melalui pada teks.
- `text-decoration: none;` digunakan untuk menghapus dekorasi pada teks.

Kode Program

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

Hasil

“CSS/aset css/Screenshot(8).png” could not be found.

Kesimpulan

Text-decoration adalah fitur CSS yang digunakan untuk menambahkan dekorasi pada teks, seperti garis bawah, garis atas, dan garis melalui.

Text transform

Penjelasan

- `none` (default): Menjaga kapitalisasi teks seperti pada penulisan aslinya.

- `uppercase` : Mengubah semua huruf menjadi huruf besar (uppercase).
- `lowercase` : Mengubah semua huruf menjadi huruf kecil (lowercase).
- `capitalize` : Mengubah huruf pertama setiap kata menjadi huruf besar (capitalize), mirip gaya penulisan judul.

Kode Program

```
p {  
    text-transform: uppercase;  
}
```

Hasil

“CSS/aset css/Screenshot(9).png” could not be found.

Kesimpulan

Text-transform adalah fitur CSS yang digunakan untuk mengubah tampilan teks, seperti huruf besar, huruf kecil, dan huruf kapital.

Background

Background-color

Penjelasan

background-color merupakan properti css yang mengatur warna latar belakang halaman.

Kode program

```
<!DOCTYPE html>

<html lang="en">

<head>

    <style>

        body {

            background-color: #001A54;

        }

    </style>

</head>

</html>
```

Hasil

“CSS/aset css/Screenshot(2).png” could not be found.

Kesimpulan

Merupakan properti CSS yang memberikan warna pada latar belakang web

Background-image

Penjelasan

Background-image dalam CSS adalah properti yang digunakan untuk menentukan gambar yang akan digunakan sebagai latar belakang dari suatu elemen.

Kode program

```
<head>
<style>
body {
    background-image: url("95754.jpg");
}
</style>
</head>
<body>
<p>Hello World!</p>
</body>
```

Hasil

“CSS/aset css/Screenshot(10).png” could not be found.

Kesimpulan

merupakan property CSS yang dapat mengatur gambar pada latar belakang web.

Background-repeat

Penjelasan

Background-repeat adalah properti yang digunakan untuk mengatur gambar latar belakang yang digunakan apakah ingin di ulang atau

tidak.

Kode program

```
<head>
<style>
body {
    background-image: url("mawar.png");
    background-repeat: no-repeat;
}
</style>
</head>
<body>
<p>Hello World!</p>
</body>
```

Hasil

"CSS/aset css/Screenshot(11).png" could not be found.

Kesimpulan

Background-repeat adalah properti yang digunakan untuk mengulang gambar latar belakang atau tidak.

Background-attachment

Penjelasan

Properti `background-attachment` digunakan untuk mengontrol apakah gambar latar belakang akan tetap diam (`fixed`) atau akan bergulir bersamaan dengan konten (`scroll`) saat pengguna

menggulir halaman. Dalam contoh ini, gambar latar belakang tetap diam bahkan saat halaman digulir.

Kode program

```
<!DOCTYPE html>

<html>

<head>

<style>

body {

    background-image: url("avatarlogin.png");

    background-repeat: no-repeat;

    background-attachment: fixed;

}

</style>

</head>

<body>

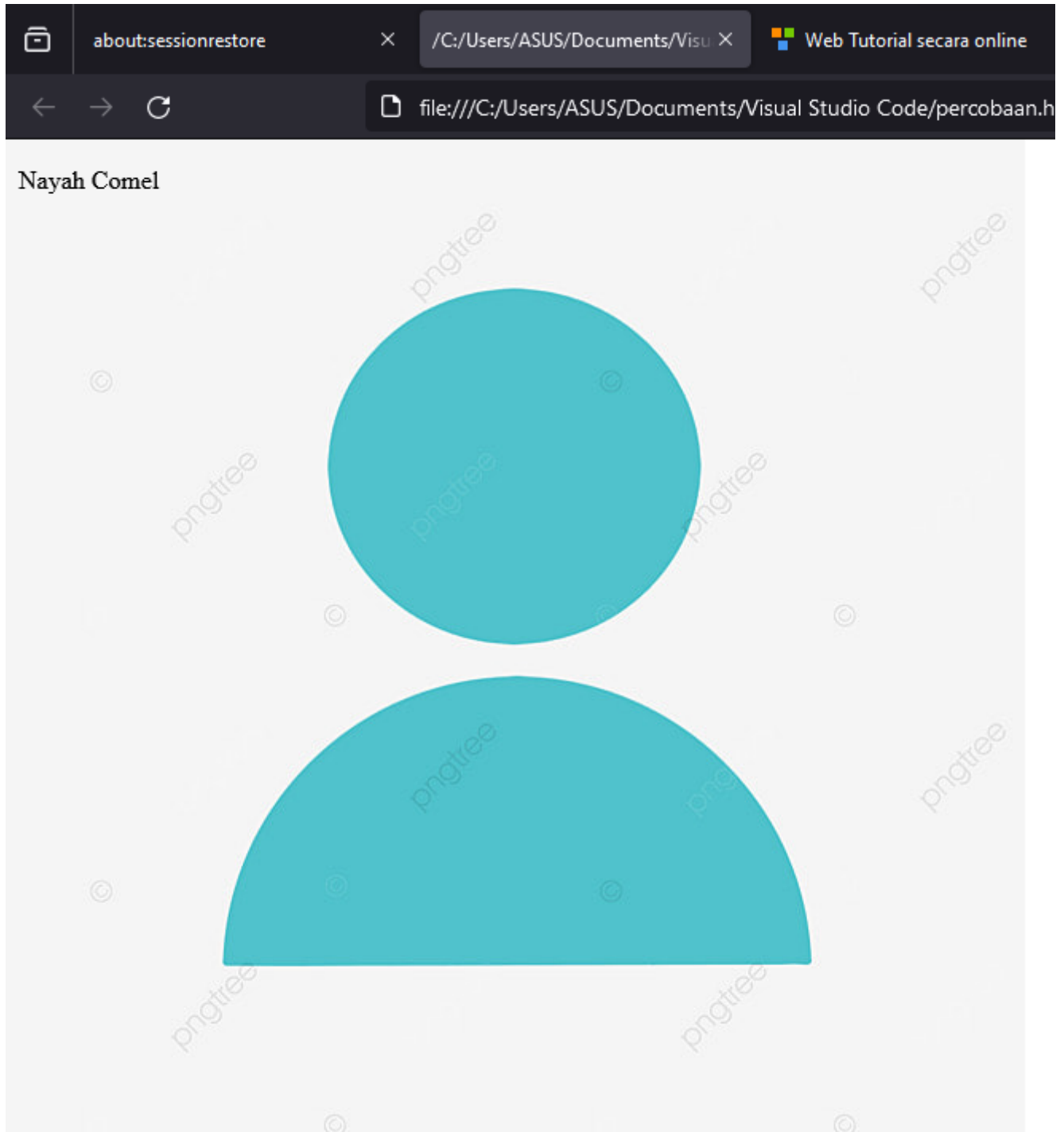
    <p>Nayah Come</p>

</body>
```



```
</html>
```

Hasil



Kesimpulan

background tersebut digunakan untuk menambahkan gambar tetapi halaman tersebut tdk dapat digulir ke halaman berikutnya

FONT

Font-Size

Penjelasan

font-size merupakan properti css yang digunakan untuk mengatur ukuran font.

Kode program

```
<head>
  <title>font</title>
  <style>
    .size {
      font-size: 100px;
    }
  </style>
</head>
<body>
  <p class="size">nayah</p>
</body>
```

Hasil

“CSS/aset css/Screenshot(3).png” could not be found.

Kesimpulan

font-size merupakan properti yang digunakan untuk mengatur seberapa besar atau kecil ukuran font yang di inginkan.

font-weight

Penjelasan

font-weight adalah properti CSS yang dapat mengubah ketebalan sebuah teks.

Kode program

```
<head>
  <title>font</title>
  <style>
    .weight {
      font-weight:bolder;
    }
  </style>
</head>
<body>
  <p class="weight">nayah</p>
</body>
```

Hasil

"CSS/aset css/Screenshot(13).png" could not be found.

Kesimpulan

font-weight adalah properti yang digunakan untuk mengatur ketebalan teks sesuai keinginan.

font-style

Penjelasan

font-style merupakan properti CSS yang mengatur gaya dari teks, seperti kemiringan.

Kode program

```
<head>
  <title>font</title>
  <style>
    .style {
      font-style:italic;
    }
  </style>
</head>
<body>
  <p class="style">nayah</p>
</body>
```

Hasil



Kesimpulan

font-style digunakan untuk mengatur gaya dari sebuah teks

font-family

Penjelasan

font-family adalah property CSS yang dapat mengubah jenis font suatu text.

Kode Program

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

Hasil

“CSS/aset css/Screenshot(12).png” could not be found.

Kesimpulan

Font-family adalah property CSS yang digunakan untuk mengatur jenis text. Font-family juga dapat menampung beberapa nama font sesuai selera, jika font pertama tidak terbaca maka font selanjutnya akan dijalankan. Nama font harus dipisah dengan tanda koma

Box-Model

border

penjelasan

Border (batas) dalam CSS adalah garis yang mengelilingi suatu elemen HTML. Border dapat digunakan untuk memberikan tampilan

visual yang jelas dan terpisah antara elemen-elemen di dalam halaman web.

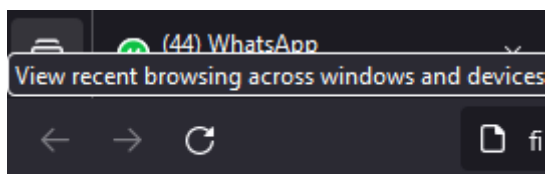
Ada tiga properti utama yang dapat digunakan untuk mengatur border suatu elemen:

1. `border-width` : Mengatur lebar border.
2. `border-style` : Mengatur jenis atau gaya border, seperti solid, dashed, dotted, double, dsb.
3. `border-color` : Mengatur warna border.
4. `boder-radius` : membuat sudut elemen HTML menjadi melengkung daripada tajam.

kode program

```
Button {  
border-color: red;  
}
```

hasil



kesimpulan

kesimpulan kode tersebut `div { border-color: red; }` adalah bahwa semua elemen `<div>` pada halaman web akan memiliki border dengan warna merah.

Dengan kode tersebut, setiap elemen `<div>` akan memiliki border dengan lebar default (biasanya 1 piksel), gaya default (biasanya solid), namun warna bordernya akan diatur menjadi merah. Ini berarti elemen `<div>` akan memiliki garis pinggiran berwarna merah mengelilingi seluruh elemennya.

Margin

penjelasan

Margin dalam CSS adalah ruang kosong di sekeliling elemen HTML yang digunakan untuk mengatur jarak antara elemen dengan elemen lainnya atau dengan batas-batas luar halaman web.

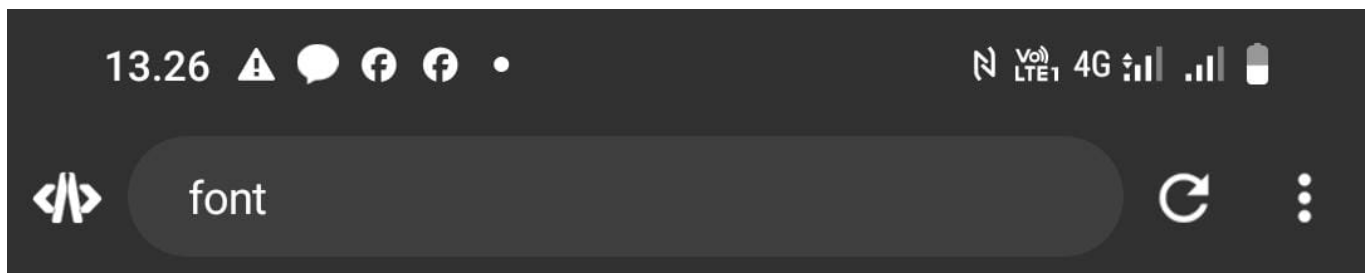
Ada empat properti margin yang dapat digunakan:

1. `margin-top` : Mengatur jarak atas elemen.
2. `margin-right` : Mengatur jarak kanan elemen.
3. `margin-bottom` : Mengatur jarak bawah elemen.
4. `margin-left` : Mengatur jarak kiri elemen.

kode program

```
img{  
  
    margin-left: 600px;  
}
```

Hasil



kesimpulan

Kesimpulan dari kode CSS `div { margin-left: 600px; }` adalah bahwa semua elemen `` pada halaman web akan memiliki margin sebesar 600 piksel di sisi kanan.

Dengan kode tersebut, setiap elemen `` akan memiliki ruang kosong (margin) sebesar 600 piksel di sisi kiri, yang berarti elemen-elemen tersebut akan terpisah dengan elemen lain di sebelah kanannya sejauh 600 piksel. Ini akan memengaruhi tata letak (layout) dari elemen-elemen `` tersebut dalam halaman web.

Padding

penjelasan

`padding` digunakan dalam CSS untuk menentukan ruang kosong di sekeliling konten suatu elemen HTML. Properti `padding` dapat diatur secara terpisah untuk setiap sisi elemen, yaitu atas (top), kanan (right), bawah (bottom), dan kiri (left).

Properti `padding` digunakan dalam CSS untuk menentukan ruang kosong di sekeliling konten suatu elemen HTML. Properti `padding` dapat diatur secara terpisah untuk setiap sisi elemen, yaitu atas (top), kanan (right), bawah (bottom), dan kiri (left).

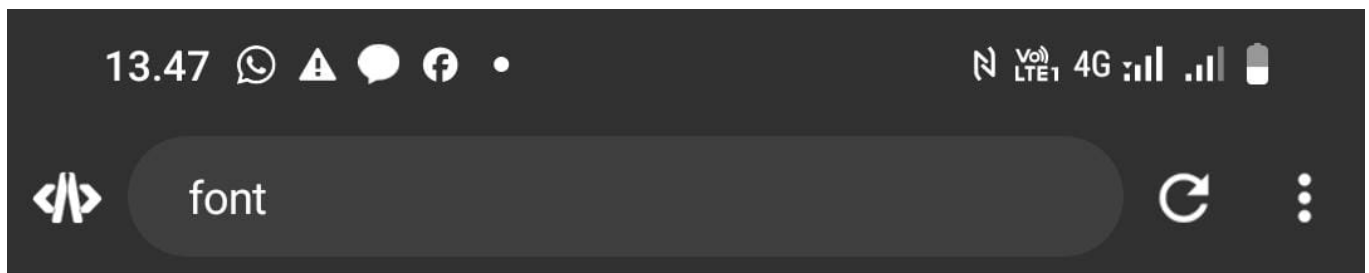
Ini berarti elemen yang diatur akan memiliki padding sebagai berikut:

- `padding-left: 10px;` : Padding sebesar 10 piksel di sisi kiri elemen.
- `padding-bottom: 10px;` : Padding sebesar 10 piksel di sisi bawah elemen.
- `padding-right: 10px;` : Padding sebesar 10 piksel di sisi kanan elemen.
- `padding-top: 10px;` : Padding sebesar 10 piksel di sisi atas elemen.

kode program

```
Img {  
padding-left: 10px;  
}
```

Hasil



Saya berada di kelas 11



kesimpulan

Kesimpulan dari kode CSS `div { padding-left: 100px; }` adalah bahwa semua elemen `` pada halaman web akan memiliki padding sebesar 100 piksel di sisi kiri.

Dengan kode tersebut, setiap elemen `` akan memiliki ruang kosong 100 piksel di sisi kiri, yang berarti konten di dalam elemen tersebut akan tergeser ke kiri sejauh 100 piksel dari batas kiri elemen.

Jadi, setiap elemen `` akan memiliki padding sebesar 100 piksel di sisi kiri elemen.

Tantangann Box model

Kode Program

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <title>CSS</title>

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

  </head>

  <body bgcolor="purple">

    <span> 

    <p>Welcome<br>

    <b>Guys</p></b>

  </span>

  <button>Klik!</button>

</body>

</html>
```

```
p{

    font-size:50px ;

    font-style: italic;

    margin-top: 100px;

    margin-left: 50px;

    color: white;

}

img{

    margin-right: 130px;

    margin-top:-1px;

    border: 5px solid white;

    border-radius: 1500% 1500%;

}

button{

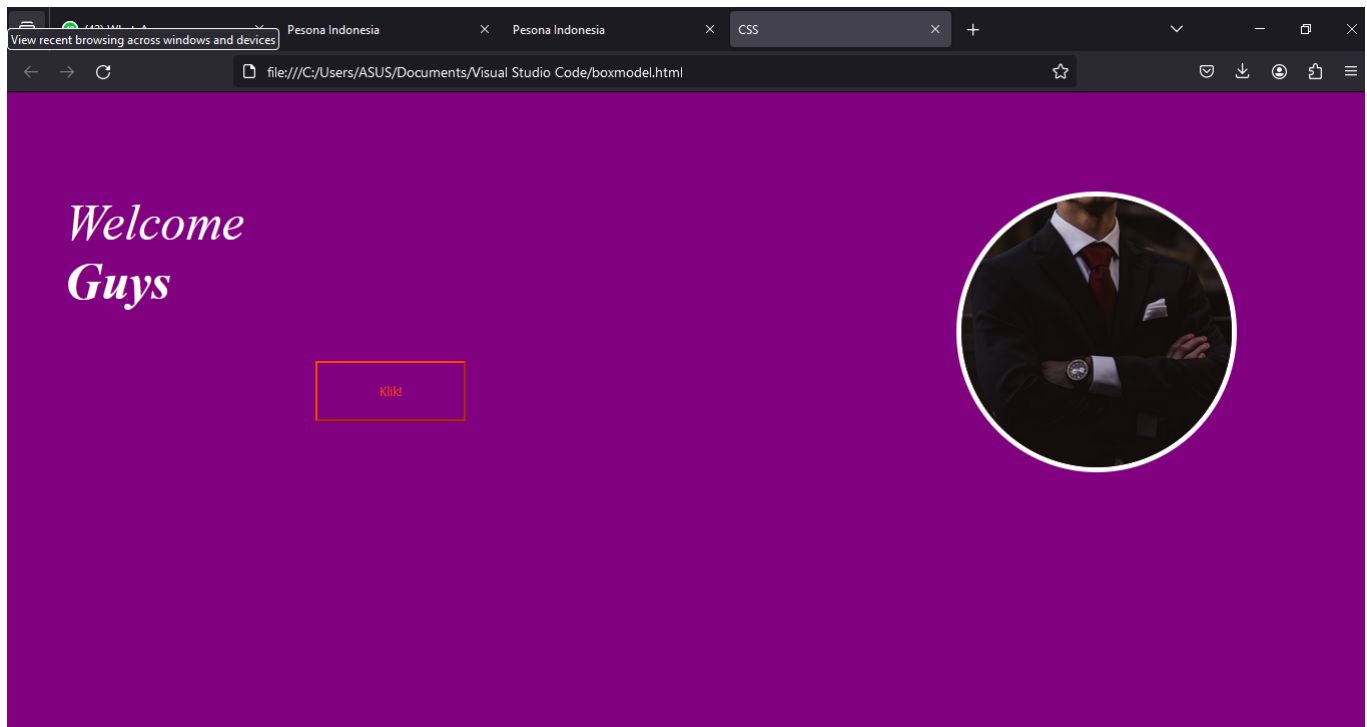
    background-color: purple;

    width:150px;

    height:60px;
```

```
border-width: 2px;  
  
border-style:solid 50px;  
  
border-color: orangered;  
  
color: orangered;  
  
margin-bottom:20px ;  
  
margin-left: 300px;  
  
}
```

Hasil



Kesimpulan

Paragraf :

- **Teks besar dan miring** : Ukuran font 75px dan gaya font *italic* membuat teks paragraf menonjol dan terlihat formal.
- **Font klasik** : Penggunaan font 'times new roman' memberikan kesan klasik dan elegan.
- **Margin besar** : Margin atas 150px, bawah 100px, kiri 50px, dan kanan 100px memberikan jarak yang besar antara paragraf dan elemen lainnya.
- **Warna biru pastel** : Warna *aliceblue* memberikan kesan lembut dan dingin pada teks.

image :

- **Margin kanan besar** : Margin kanan 200px memberikan spasi yang lebar antara gambar dan elemen lain di sebelah kanan.
- **Margin atas negatif** : Margin atas -50px kemungkinan digunakan untuk menaikkan posisi gambar agar sejajar dengan bagian atas paragraf.
- **Garis tepi tebal** : Border 10px membuat garis tepi gambar tebal dan terlihat jelas.
- **Sudut membulat ekstrem** : Border radius 1500px membuat sudut gambar membulat hampir menjadi lingkaran penuh.

button :

- **Warna ungu** : Background color *purple* memberikan kesan profesional dan modern pada tombol.
- **Ukuran sedang** : Lebar 100px dan tinggi 50px membuat tombol berukuran sedang dan mudah diklik.

- `Garis tepi oranye` : Border width 2px dan color `orangered` membuat garis tepi tombol berwarna oranye dan terlihat jelas.

Pseudo-classes

Haver

Penjelasan

Digunakan untuk mengganti tampilan elemen saat kursor mouse berada di atasnya. Ini sering digunakan untuk efek hover seperti perubahan warna latar belakang atau tampilan teks yang berubah.

Kode Program

```
<!DOCTYPE html>

<html>

<head>

<style>

a:hover {

    background-color: yellow;

}

</style>

</head>
```

```
<body>
```

```
<h1>Contoh hover</h1>
```

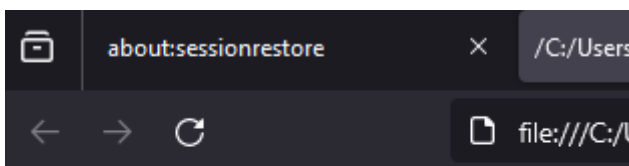
```
<p>Nayah comel</p>
```

```
<a href="https://www.wikipedia.org">wikipedia.org</a>
```

```
</body>
```

```
</html>
```

Hasil



Contoh hover

Nayah comel

[wikipedia.org](https://www.wikipedia.org)

Active

Penjelasan

Digunakan untuk mengganti tampilan elemen saat elemen tersebut sedang aktif atau ditekan, seperti tombol yang sedang ditekan.

Kode Program

```
<!DOCTYPE html>

<html>

<head>

<style>

a:active {

    background-color: yellow;

}

</style>

</head>

<body>

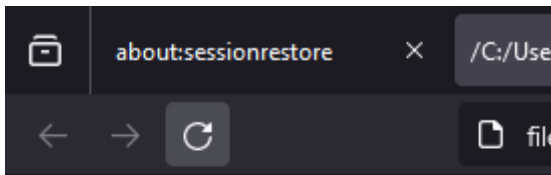
<h1>Contoh Active</h1>
```

```
<a href="https://www.smk7mks.com">SMK 7</a><br>

</body>

</html>
```

Hasil



Contoh Active

[SMK 7](#)

Visited

Penjelasan

digunakan untuk memilih tautan yang dikunjungi

Kode Program

```
<!DOCTYPE html>

<html>
```

```
<head>

<style>

a:visited {

    color: pink;

}

</style>

</head>

<body>

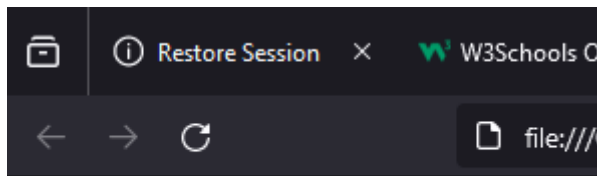

<h1>Contoh Visited</h1>


<a href="https://www.smk7mks.com">smk7</a><br>


</body>

</html>
```

Hasil



Contoh Visited

[smk7](#)

Transition

Transition-property

Penjelasan

adalah atribut yang digunakan untuk transisi yang apabila kursor didekatkan maka transisi akan berfungsi.

Kode Program

```
<html>

<head>

<style>

div {

    width: 100px;

    height: 100px;

    background: red;
```

```
    transition-property: width;

    transition-duration: 2s;
}

div:hover {

    width: 300px;
}

</style>

</head>

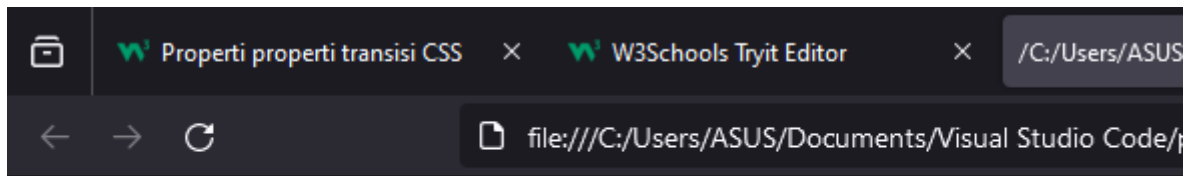
<body>

<div></div>

</body>

</html>
```

Hasil



Transition-duration

Penjelasan

Properti ini `transition-duration` menentukan berapa detik (s) atau milidetik (ms) yang diperlukan untuk menyelesaikan efek transisi.

Kode Program

```
<html>

<head>

<style>

div {

    width: 100px;

    height: 100px;

    background: red;
```

```
    transition-property: width;

    transition-duration: 5s;
}

div:hover {

    width: 300px;
}

</style>

</head>

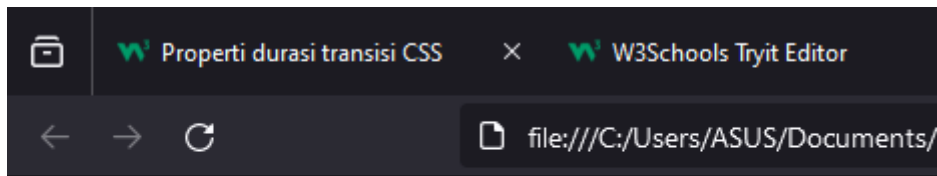
<body>

<div></div>

</body>

</html>
```

Hasil



Transition-timing function

Penjelasan

Properti `transition-timing-function` menentukan kurva kecepatan efek transisi.

Properti ini memungkinkan efek transisi untuk mengubah kecepatan sepanjang durasinya

Kode program

```
<!DOCTYPE html>

<html>

<head>

<style>

div {

    width: 100px;
```



```
height: 100px;

background: blueviolet;

transition: width 2s;

transition-timing-function: linear;
}


div:hover {

    width: 300px;

}

</style>

</head>

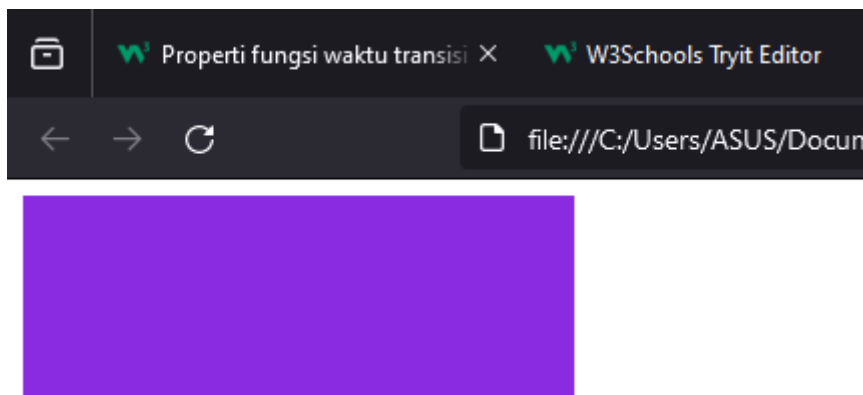
<body>

<div></div>

</body>

</html>
```

Hasil



Tantangan Transition

Penjelasan

1. `body` :

- `background-color: rgb(248, 210, 163);` : Menetapkan warna latar belakang body menjadi warna oranye (dinyatakan dalam format RGB).
- `width: 100%;` : Membuat body mengisi lebar penuh layar.

2. `.container` :

- `display: contents;` : Membuat kontainer tersebut memiliki perilaku seperti tidak ada kontainer. Artinya, anak-anak langsung dari kontainer akan ditata sesuai dengan tata letak induknya.
- `align-items: flex-end;` : Menetapkan item dalam kontainer untuk diatur di bagian bawah kontainer.
- `flex-direction: row;` : Menjadikan anak-anak kontainer diatur dalam satu baris horizontal.

- `justify-content: space-around;` : Membuat ruang sekitar item dalam kontainer secara merata.

3. `.box-2:`

- `font-size: 75px;` : Menetapkan ukuran font sebesar 75 piksel.
- `font-family: 'arial';` : Menggunakan jenis font Arial.
- `margin-top: 150px; margin-bottom: 100px; margin-left: 50px; margin-right: 100px;` : Menetapkan jarak antara box dan elemen lain di sekitarnya.
- `color: rgb(104, 104, 104);` : Menetapkan warna teks menjadi abu-abu tua.

4. `.box-1:`

- `margin-right: 200px; margin-top: -30px;` : Menetapkan jarak dari sisi kanan dan atas box-1.
- `border: 10px solid white; border-radius: 1500px 1500px;` : Menambahkan border putih dengan ketebalan 10 piksel dan radius sudut sebesar 1500 piksel.

5. `button:`

- `background-color: lightblue; width: 150px; height: 50px;` : Menetapkan warna latar belakang, lebar, dan tinggi tombol.
- `border-width: 2px; color: rgb(138, 138, 229); border-color: rgba(73, 134, 240, 0.29);` : Menetapkan ketebalan border, warna teks, dan warna border (dalam format RGBA) tombol.
- `margin-bottom: 20px; margin-left: 400px;` : Menetapkan jarak bawah dan kiri tombol.

6. `button:hover:`

- `background-color: lightcyan; font-weight: bolder; transition: all 0.3s ease-in;`: Menetapkan efek saat tombol dihover, seperti perubahan warna latar belakang dan penambahan ketebalan font. Ini menggunakan transisi selama 0.3 detik untuk menciptakan perubahan yang mulus.

Kode Program

```
<!DOCTYPE html>

<html>

<head>

    <title>tantangan transition</title>

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

</head>

<body>

    <div class="container">

        <div>

            <p class="box-2"> Selamat Datang <br><b>
```

```
        di web Nayah !</b>

        <p>

    </div>

    <button> klik saya </button>

</div>

</body>

</html>
```

```
body {

    background-color: rgb(248, 210, 163);

    width: 100%;

}

.container {

    display: contents;

    align-items: flex-end;

    flex-direction: row;

    justify-content: space-around;

}
```

```
.box-2 {  
  
    font-size: 75px;  
  
    font-family: 'arial';  
  
    margin-top: 150px;  
  
    margin-bottom: 100px;  
  
    margin-left: 50px;  
  
    margin-right: 100px;  
  
    color: rgb(104, 104, 104);  
  
}  
  
.box-1 {  
  
    margin-right: 200px;  
  
    margin-top: -30px;  
  
    border: 10px solid white;  
  
    border-radius: 1500px 1500px;  
  
}  
  
button {
```

```
background-color: lightblue;

width: 150px;

height: 50px;

border-width: 2px;

color: rgb(138, 138, 229);

border-color: rgba(73, 134, 240, 0.29);

margin-bottom: 20px;

margin-left: 400px;

}

button:hover {

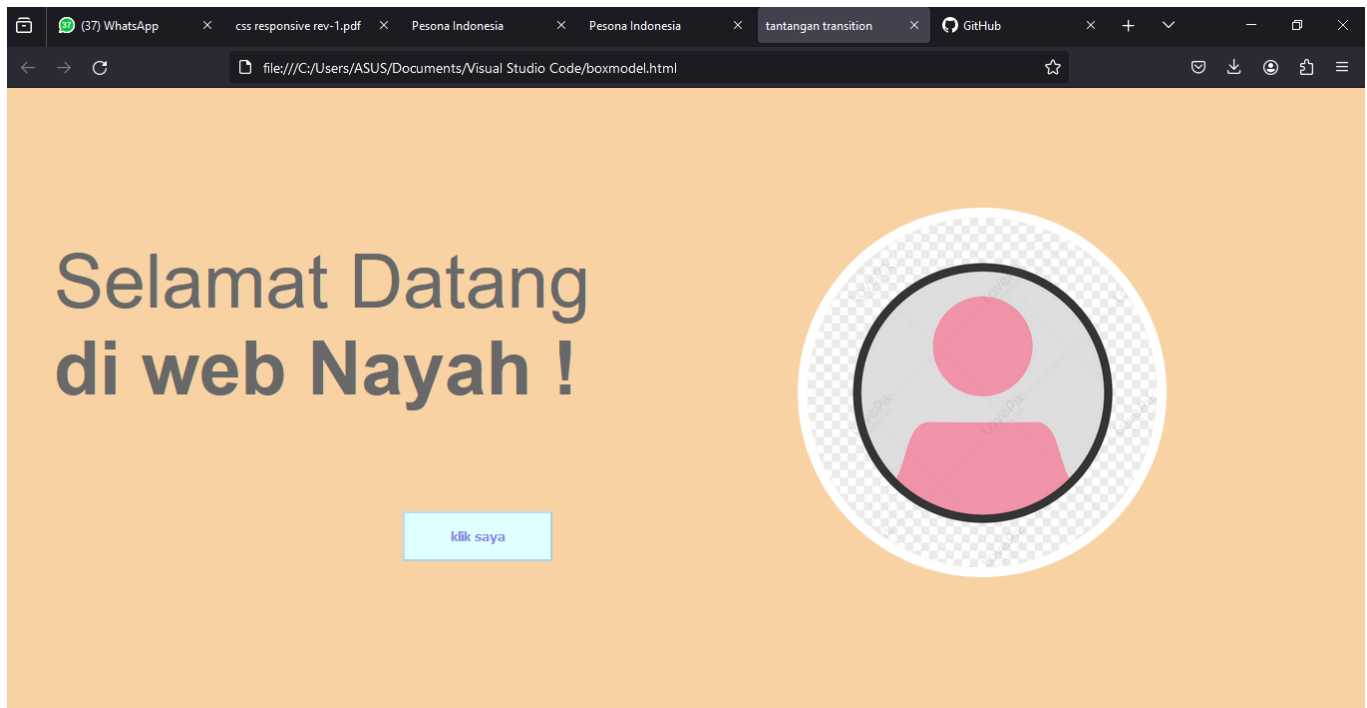
background-color: lightcyan;

font-weight: bolder;

transition: all 0.3s ease-in;

}
```

Hasil



Kesimpulan

Latar belakang body akan memiliki warna `RGB (248, 210, 163)` yang merupakan kombinasi dari merah, hijau, dan biru.

Lebar body akan setara dengan 100% lebar viewport (area tampilan browser).

Elemen dengan kelas `"container"` akan memiliki tata letak kontennya yang diatur secara fleksibel, dengan elemen-elemen yang diatur dalam baris.

Konten dalam elemen dengan kelas `"container"` akan diatur agar berada di bagian bawah (`align-items: flex-end`) dan memiliki ruang kosong merata di sekitarnya (`justify-content: space-around`).

Elemen dengan kelas `"box-2"` akan memiliki ukuran `font 75 piksel`, menggunakan jenis `font Arial`, dan memiliki `margin atas, bawah, kiri, dan kanan` yang masing-masing telah ditentukan.

Elemen dengan kelas `"box-1"` akan memiliki `margin kanan 200`

piksel, margin atas -30 piksel (mendorong elemen ke atas), border dengan ketebalan 10 piksel, dan radius lengkungan border sebesar 1500 piksel pada sudut-sudutnya.

Tombol akan memiliki latar belakang warna lightblue, lebar 150 piksel, tinggi 50 piksel, ketebalan border 2 piksel, warna teks RGB (138, 138, 229), warna border RGBA (73, 134, 240, 0.29), dan margin bawah dan kiri yang telah ditentukan.

Saat tombol dihover, latar belakangnya akan berubah menjadi lightcyan, teksnya akan menjadi lebih tebal, dan akan ada efek transisi selama 0.3 detik dengan fungsi timing ease-in.

Transform

Penjelasan

Transform adalah untuk mengubah tampilan elemen HTML, seperti menggeser, memutar, atau mengubah ukurannya ketika di klik. Ini adalah cara untuk membuat animasi sederhana atau mengatur posisi elemen dengan lebih fleksibel, atau lebih singkatnya mengubah gaya pada suatu elemen HTML ketika diklik.

Kode Program

```
<html>

  <head>

    <title>Home</title>

    <link rel="stylesheet" href="percobaan1.css">
```

```
</head>

</head>

<body>

    <p>Klik untuk membuat dirinya menyukaimu</p>

    <button class="tombol">AYO KLIKKK!!!</button>

</div>

</body>

</html>
```

```
.tombol {

    background-color: blue;

    color: white;

    border: none;

    width: 180px;

    height: 40px;

    font-size: 15px;

    border: 30px;
```

```
}
```

```
.tombol:hover {
```

```
    background-color: yellow;
```

```
    color: black;
```

```
    border: none;
```

```
    width: 160px;
```

```
    height: 40px;
```

```
    font-size: 20px;
```

```
    font-family: 'Segoe UI';
```

```
    font-weight: bold;
```

```
    border-radius: 30px;
```

```
    transition: all 0.3s ease-in;
```

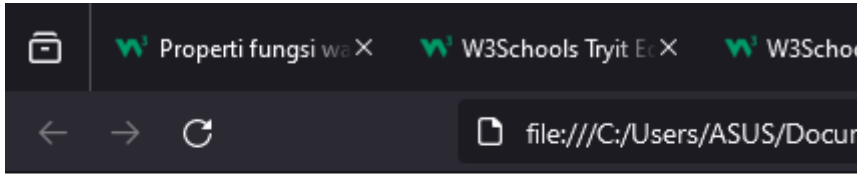
```
}
```

```
.tombol:active {
```

```
    transform: scale(0.15);
```

```
}
```

Hasil



Klik untuk membuat dirinya menyukaimu

AYO KLIKKK!!!

Flexbox

FLEX-CONTAINER

Penjelasan

Flex container adalah elemen induk yang mengatur tata letak flex item-nya. flexbox yaitu memberikan container kemampuan untuk mengatur panjang, lebar, dan posisi item-item yang berada di dalamnya agar memaksimalkan ruang yang ada. Pengaturan ini sangat penting bagi seorang frontend developer untuk membuat

sebuah website yang nyaman dilihat di berbagai device dengan berbagai macam resolusi.

1. `flex-direction` :Menentukan arah (direction) yang akan diberlakukan untuk item-item yang ada pada container flexbox.
2. `flex-wrap` :Digunakan untuk mendefinisikan bahwa elemen item di dalam container flexbox tidak harus disejajarkan dalam satu baris.
3. `justify-content` :Digunakan untuk mensejajarkan item-item diantara flexbox agar container dari flexbox tersebut bisa mendistribusikan ruang kosong yang tersisa ketika item flex dalam satu baris tersebut tidak fleksibel atau meskipun fleksibel tapi sudah mencapai batas ukuran maksimum.
4. `align-items` :Mendefinisikan bagaimana item-item pada container flex tersebut diletakkan sepanjang garis tegak lurus pada sumbu utama (cross-axis).
5. `align-content` :Digunakan untuk mensejajarkan garis flex container ketika ada ruang kosong secara garis tegak lurus pada sumbu utama (cross-axis).

Kode Program

```
.container {  
  
    display: flex;  
  
    height: 100vh;  
  
    justify-content: space-around ;  
  
    align-items: center ;  
}
```

```
background-color: blanchedalmond;

}
```

Hasil



FLEX-ITEM

Penjelasan

Dalam konteks Flexbox, elemen-elemen dianggap sebagai flex items (item fleksibel), dan mereka diatur dalam satu dimensi (baris atau kolom) menggunakan properti-properti Flexbox.

1. **flex-grow**: Properti ini menentukan sejauh mana flex item akan memperluas ruang tersedia dalam flex container.
2. **flex-shrink**: Properti ini menentukan sejauh mana flex item akan menyusut jika ruang tidak cukup dalam flex container.
3. **flex-basis**: Properti ini menentukan ukuran awal (basis) flex item sebelum flex container membagi ruang yang tersedia.
4. **flex**: Properti singkat flex digunakan untuk menggabungkan flex-grow, flex-shrink, dan flex-basis dalam satu deklarasi.

Kode Program

```
.item {  
  width: 100px;  
  margin-right: 10px;  
  margin-bottom: 10px;  
  height: 100px;  
  background-color: red;  
  display: flex;  
}
```

Hasil



Kesimpulan

Kesimpulannya, dalam CSS Flexbox, elemen-elemen dianggap sebagai flex items (item fleksibel) dan dapat diatur menggunakan properti-properti Flexbox.

Tantangan Flexbox

Penjelasan

1. : Ini adalah deklarasi tipe dokumen HTML yang menunjukkan bahwa dokumen ini adalah dokumen HTML5.
2. `<html lang="en">`: Ini adalah elemen root (akar) dari dokumen HTML. `lang="en"` menunjukkan bahwa bahasa yang digunakan dalam dokumen adalah bahasa Inggris.

3. `<head>`: Elemen `<head>` digunakan untuk menyediakan informasi tentang dokumen, seperti judul (title) dan referensi ke file eksternal seperti stylesheet (CSS).
4. `<title>`: Elemen `<title>` digunakan untuk menentukan judul dokumen yang akan ditampilkan di bilah judul browser.
5. `<link rel="stylesheet" href="tugasflex.css">`: Elemen `<link>` digunakan untuk menghubungkan dokumen HTML dengan file eksternal CSS. Dalam contoh ini, file CSS yang disebut "tugasflex.css" akan digunakan untuk mengatur tampilan halaman.
6. `<body bgcolor="purple">`: Elemen `<body>` digunakan untuk mengelilingi konten utama halaman web. `bgcolor="purple"` adalah atribut yang digunakan untuk mengatur warna latar belakang body menjadi ungu (purple).
7. `<div class="main-container">`: Elemen `<div>` adalah elemen blok yang digunakan untuk mengelompokkan dan mengatur konten. `class="main-container"` adalah atribut kelas yang memberikan nama kelas "main-container" pada elemen ini. Kelas ini nantinya dapat digunakan dalam CSS untuk mengatur tampilan elemen ini.
8. `<div class="hero-container">`: Elemen `<div>` dengan kelas "hero-container" digunakan untuk mengelompokkan konten yang terkait dengan bagian hero atau bagian utama halaman.
9. `<div class="item p">`: Elemen `<div>` dengan kelas "item p" digunakan untuk mengelompokkan konten dan memberikan atribut kelas "p".
10. `<p>`: Elemen `<p>` digunakan untuk menampilkan paragraf teks. Di dalam elemen ini, terdapat teks "Selamat Datang" dan

"di Web Adel" yang ditampilkan dalam beberapa baris yang dipisahkan oleh tag `
`. Teks "Web Adel" ditampilkan dengan teks tebal menggunakan tag ``.

11. `<button>`: Elemen `<button>` digunakan untuk membuat tombol. Di sini, tombol ditampilkan dengan teks "klik saya".
12. ``: Elemen `` digunakan untuk mengelompokkan dan memanipulasi bagian teks atau elemen lainnya dalam dokumen. Di sini, elemen span memiliki atribut kelas "img".
13. ``: Elemen `` digunakan untuk menampilkan gambar dalam halaman. Atribut src menentukan sumber gambar (dalam hal ini, "abrar.JPG"), sedangkan atribut width dan height mengatur lebar dan tinggi gambar. Atribut align="right" mengatur posisi gambar ke sebelah kanan.

Kode Program

```
<!DOCTYPE html>

<html lang="en">

<head>

    <title>Document</title>

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

</head>

<body bgcolor="purple">
```

```
<div class="main-container">

  <div class="hero-container">

    <div class="item p">

      <p> Selamat Datang <br>

        di<b>Web Nayah</b> </P>

      <button> klik saya </button>

    </div>

    <div>

      <span class="img">

      </span>

    </div>

  </div>

</div>

</body>

</html>
```

```
.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;

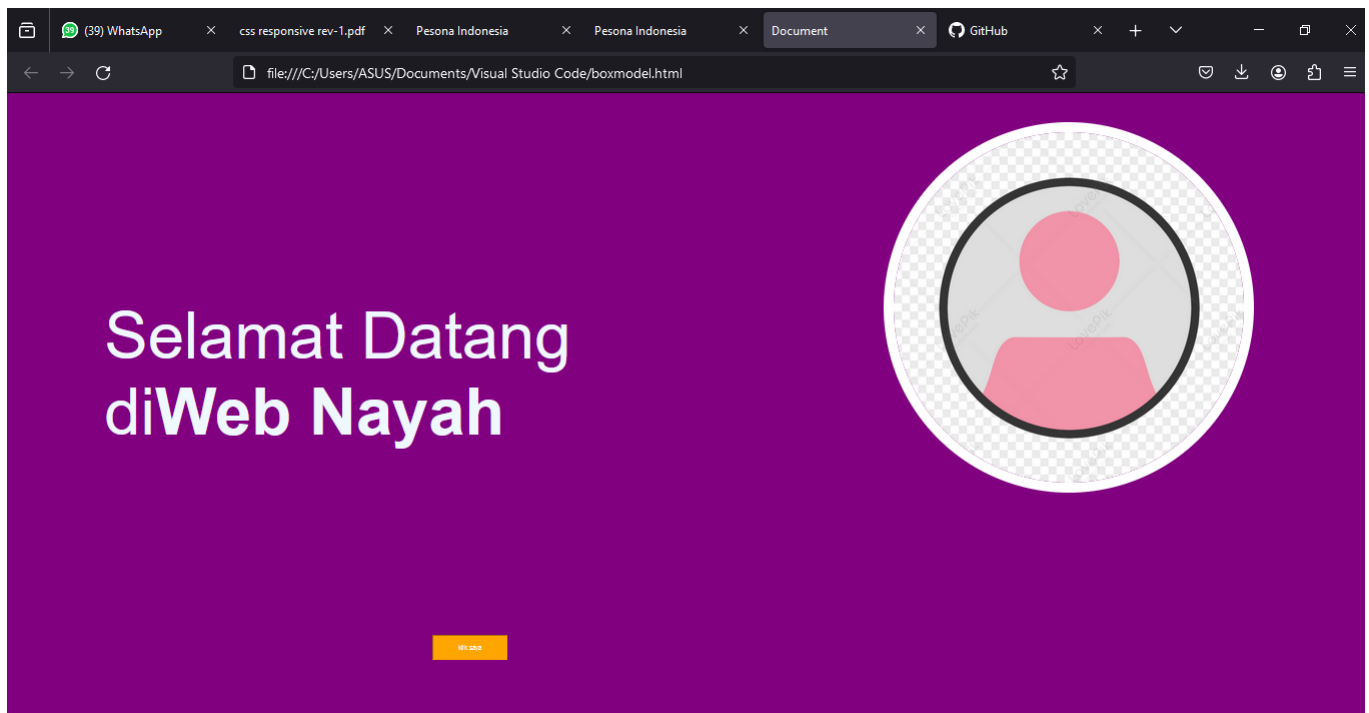
    cursor: pointer;
}

button:active {

    transform: scale(0.5);

}
```

Hasil



Position

Position relative

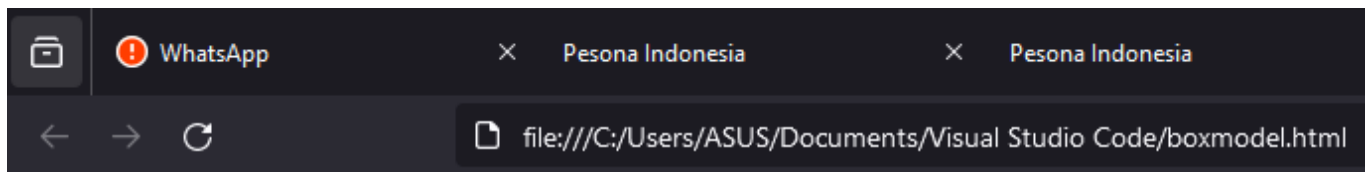
Penjelasan

`position: relative` adalah properti CSS yang digunakan untuk menetapkan posisi elemen pada halaman web relatif terhadap posisinya sendiri.

Kode program

```
.box { position: relative; top: 10px; left: 10px; width: 100px; height: 100px; background-color: red; }
```

Hasil



Kesimpulan

memungkinkan pengguna untuk mengatur posisi elemen dengan properti top, right, bottom, atau left

Position absolute

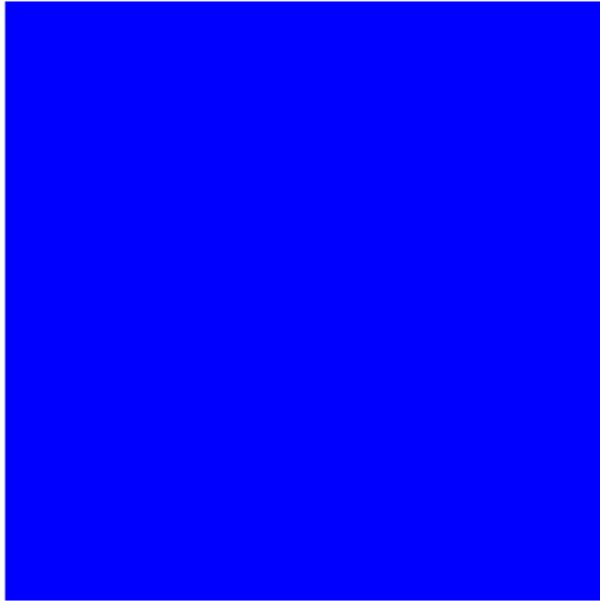
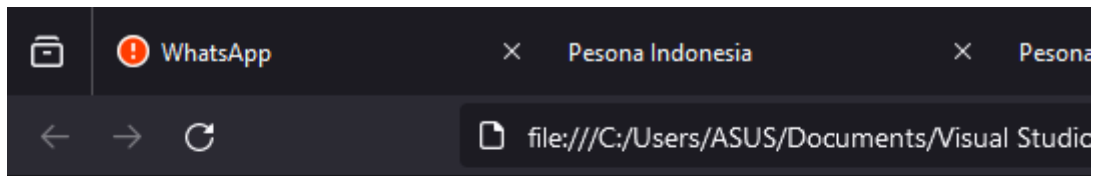
Penjelasan

`position: absolute` akan dikeluarkan dari normal flow, yang berarti elemen tersebut tidak akan memengaruhi posisi elemen lain di halaman.

Kode program

```
.box { position: absolute; top: 10px; left: 10px; width: 100px; height: 100px; background-color: red; }
```

Hasil



Kesimpulan

hanya berpengaruh pada elemen yang diatur, tidak akan berpengaruh pada posisi elemen lain.

Position fixed

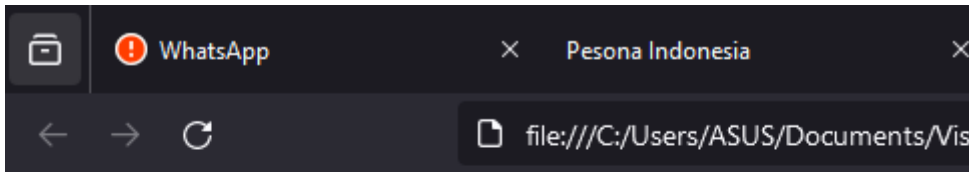
Penjelasan

`position: fixed` adalah properti CSS yang digunakan untuk menetapkan posisi elemen pada halaman web tetap dalam posisi tertentu di layar

Kode program

```
.box {  
  position: fixed;  
  width: 100px;  
  height: 100px;  
  background-color: red;  
}
```

Hasil



Kesimpulan

Untuk menetapkan posisi suatu elemen

Position sticky

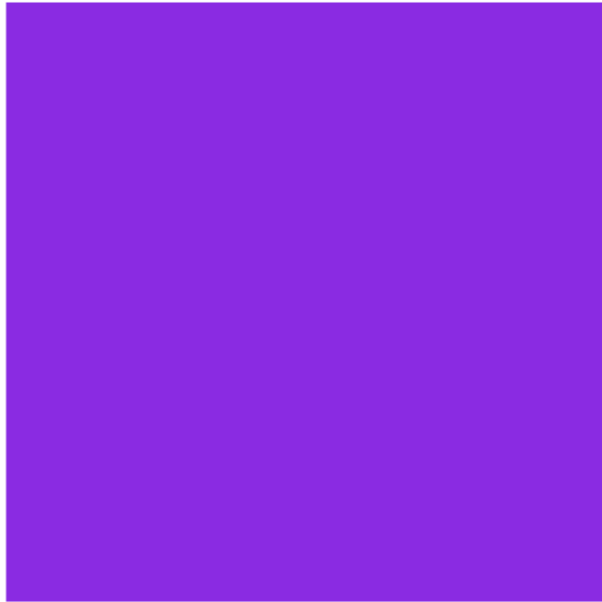
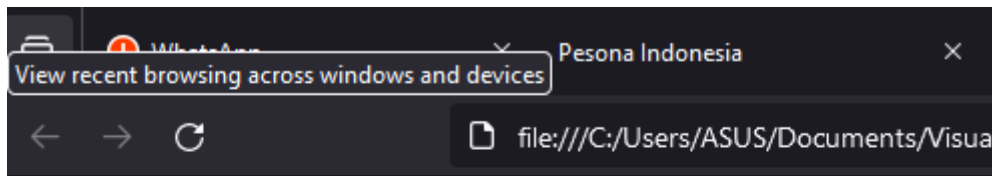
Penjelasan

Posisi sticky adalah cara efektif untuk mengubah posisi elemen dengan kecil atau sedikit, seperti menyesuaikan posisi elemen dengan kursor atau mengubah posisi elemen dalam layout.

Kode program

```
.box {  
  position: sticky;  
  top: 10px;  
  width: 100px;  
  height: 100px;  
  background-color: red;  
}
```

Hasil



Kesimpulan

`position: sticky` akan bergulir seperti normal hingga mencapai titik tertentu

Tantangan Position

Penjelasan

- `position: static;` : Kontainer tersebut memiliki posisi statis.
- `display: flex;` : Anak-anak dari kontainer akan diatur dalam satu baris, berdasarkan sumbu utama yang secara default akan menjadi horizontal.
- `flex-direction: column;` : Anak-anak dari kontainer akan diatur dalam satu kolom.

- `width: 100%;` : Kontainer akan mengisi lebar penuh dari elemen induknya.
- `height: 580px;` : Tinggi kontainer ditetapkan pada 580 piksel.
- `background-color: rgba(122, 122, 247, 0.628);` : Warna latar belakang kontainer diatur sebagai nilai RGBA.
- `width: 250px;` : Lebar box ditetapkan pada 250 piksel.
- `height: 350px;` : Tinggi box ditetapkan pada 350 piksel.
- `background-color: white;` : Warna latar belakang box diatur sebagai putih.
- `align-items: center;` : Anak-anak dari box akan dipusatkan secara horizontal.
- `align-content: center;` : Konten di dalam box akan dipusatkan secara vertikal.
- `border-radius: 10px;` : Sudut box akan dibulatkan sebesar 10 piksel.
- `align-self: center;` : Box akan dipusatkan di dalam kontainer secara horizontal.
- `margin-top: 150px;` : Jarak antara bagian atas kontainer dan bagian atas box ditetapkan pada 150 piksel.
- `margin-bottom: 200px;` : Jarak antara bagian bawah kontainer dan bagian bawah box ditetapkan pada 200 piksel.

Kode Program

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>POSITION</title>

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

</head>

<body>

  <div class="container">

    <div class="box">

      <p> </p>

      <p class="item box-2">Saturday, April 27,
2024</p>

      <h1 class="item box-3">The standard chunk of
<br>Lorem Ipsum</h1>

      <p class="item box-4">Sed posuere consectetur
est at lobortis.<br>Aenean eu leo quam</p>

      <p class="box-5"><b>Read more </b> </p>

    </div>

  </div>
```

```
</body>
```

```
</html>
```

```
.container {
```

```
    position: static;
```

```
    display: flex;
```

```
    flex-direction: column;
```

```
    width: 100%;
```

```
    height: 580px;
```

```
    background-color: rgba(122, 122, 247, 0.628);
```

```
}
```

```
.box {
```

```
    width: 250px;
```

```
    height: 350px;
```

```
    background-color: white;
```

```
    align-items: center;
```

```
    align-content: center;
```

```
border-radius: 10px;

align-self: center;

margin-top: 150px;

margin-bottom: 200px;

}


.item {

    width: 100%;

    color: black;

}


.box-1 {

    height: 225px;

    width: 100%;

    border-radius: 10px 10px 0px 0px;

}
```



```
.box-2 {  
  
    font-size: 10px;  
  
    margin-left: 10px;  
  
    margin-top: 10px;  
  
    padding-top: 10px;  
  
    font-family: Arial, Helvetica, sans-serif;  
  
}
```

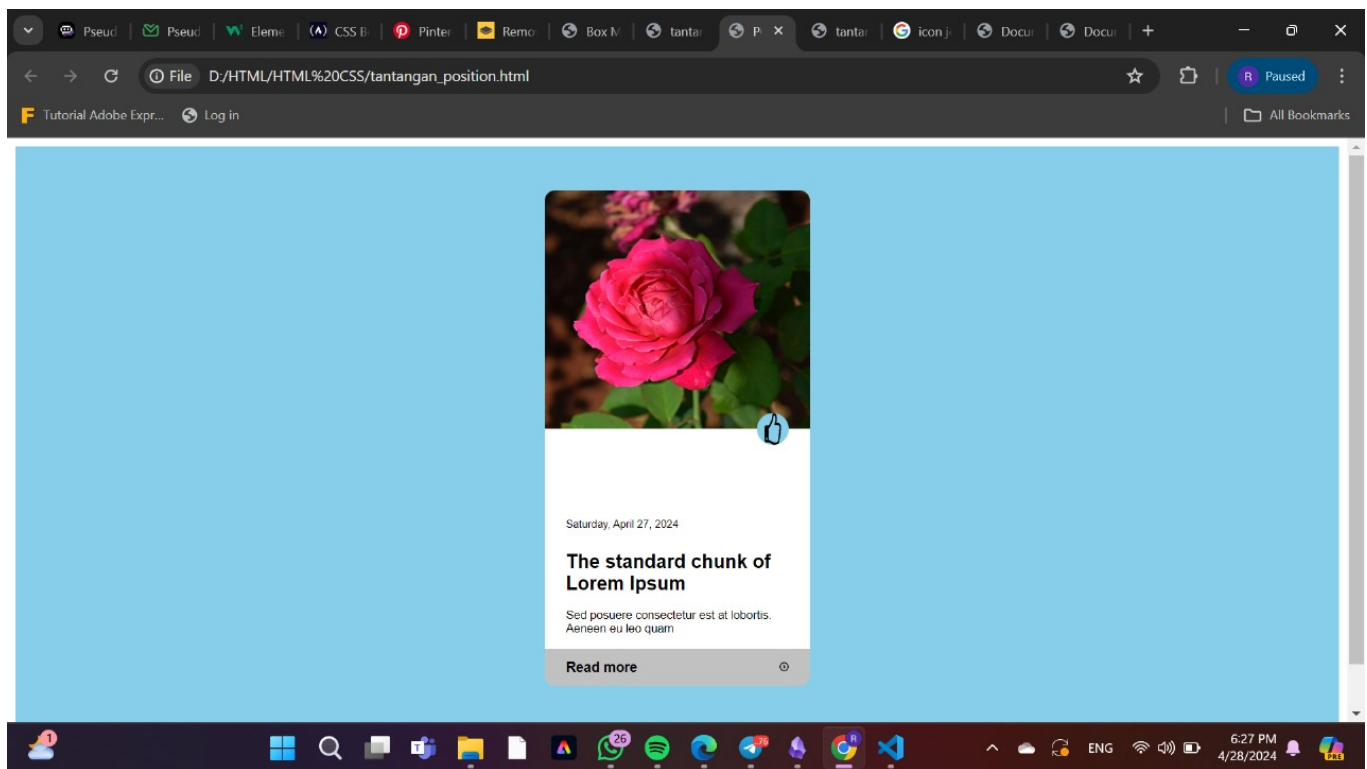
```
.box-3 {  
  
    font-size: 18px;  
  
    margin-left: 10px;  
  
    margin-top: 10px;  
  
    padding-top: 10px;  
  
    font-family: Arial, Helvetica, sans-serif;  
  
}
```

```
.box-4 {  
  
    font-size: 11px;  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    margin-left: 10px;  
  
}
```

```
.box-5 {  
  
    background-color: rgb(193, 193, 193);  
  
    padding-left: 20px;  
  
    padding-bottom: 10px;  
  
    padding-top: 10px;  
  
    margin-bottom: 90px ;  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    border-radius: 0px 0px 10px 10px;  
  
    font-size: small;  
  
}
```

```
.item-1 {  
  
    width: 10px;  
  
    padding-left: 130px;  
  
}  
  
.icon {  
  
    background-color: skyblue;  
  
    position: relative;  
  
    left: 200px;  
  
    bottom: 35px;  
  
    width: 30px;  
  
    height: 30px;  
  
    border-radius: 1500px;  
  
}
```

Hasil



Kesimpulan

- Kontainer tersebut memiliki posisi statis dan akan mengisi lebar penuh dari elemen induknya.
- Tinggi kontainer ditetapkan pada 580 piksel, dengan latar belakang berwarna RGBA (122, 122, 247, 0.628).
- Anak-anak dari kontainer akan diatur dalam satu kolom, sehingga properti `flex-direction: column;` diterapkan.
- Lebar box ditetapkan pada 250 piksel dan tinggi pada 350 piksel, dengan latar belakang berwarna putih.
- Isi dari box akan dipusatkan secara horizontal dan vertikal dengan properti `align-items: center;` dan `align-content: center;`.
- Sudut box dibulatkan sebesar 10 piksel dan box itu sendiri akan dipusatkan di dalam kontainer secara horizontal dengan menggunakan `align-self: center;`.

- Terdapat juga penyesuaian margin, di mana jarak antara bagian atas kontainer dan bagian atas box ditetapkan pada 150 piksel, dan jarak antara bagian bawah kontainer dan bagian bawah box ditetapkan pada 200 piksel.