

Pengenalan CSS

Anatomi CSS

Anatomi CSS



- **Selector.....???**

Apa yang ingin dimodifikasi

- **Property...????**

Bagian apa yang ingin dimodifikasi

- **Property Value...???**

Bentuk modifikasinya seperti apa

Contoh lain, yang ingin dimodifikasi adalah seluruh **jendela** pada komponen **kusenya** menjadi **terbuka**

Percobaan Pertama CSS

Kode program

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

Hasil



Welcome CSS!

Welcome CSS!

Penjelasan

Deklarasi gaya CSS { color: red; } menetapkan properti color ke nilai "red", sehingga teks dalam elemen paragraf akan ditampilkan dengan warna merah.

Percobaan Kedua CSS

Kode Program

```
button{width: 150px; height: 50px;
font-size: 100px;
Background-color:red;
color:blue;}
```

Font size

```
button{font-size : 100px;}
```

Hasil



Penjelasan

`button { font-size: 100px; }` adalah aturan CSS yang menentukan gaya untuk semua elemen `< button >` dalam dokumen HTML

Button: Ini adalah selector yang menargetkan semua elemen `< button >` dalam dokumen HTML.

font-size :100px; :Ini adalah deklarasi gaya yang terkait dengan selector tersebut. Di sini, kita menetapkan properti font-size dengan nilai 100px, yang berarti teks di dalam elemen `< button >` akan ditampilkan dengan ukuran font sebesar 100 piksel.

Background Color

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

Hasil



Penjelasan

Kode program `button{background-color: red;}` merupakan aturan CSS yang menetapkan warna latar belakang (`background-color`) menjadi merah untuk semua elemen `<button>` dalam dokumen HTML yang terhubung dengan stylesheet tersebut. Ini berarti semua tombol dalam dokumen HTML akan memiliki latar belakang merah ketika aturan CSS ini diterapkan

Font Color

```
button{color:blue;}
```

Hasil



Penjelasan

Kode program CSS tersebut adalah aturan CSS yang mendefinisikan styling untuk elemen-elemen yang menggunakan tag `<button>`. Dalam aturan tersebut, warna teks (`color`) diatur menjadi biru (`blue`).

Cara Pemanggilan di CSS

Pemanggilan Secara Internal

Ini melibatkan penempatan kode CSS di dalam tag `<style>` di dalam elemen `<head>` dari dokumen HTML.

Contohnya:

```
<head>
  <style>
    p {
      color: blue;
    }
  </style>
</head>
<body>
  <p>Ini adalah teks biru</p>
</body>
```

Pemanggilan Secara Eksternal

Eksternal CSS: Ini melibatkan penempatan kode CSS dalam file terpisah dengan ekstensi .css, yang kemudian dipanggil dalam elemen `link` di dalam elemen `head` dari dokumen HTML. Contohnya:

```
<head>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
```

Isi Dari File Style.Css:

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

Pemanggilan Secara Inline

Inline adalah metode untuk menambahkan gaya langsung ke elemen HTML menggunakan atribut `style`. Contohnya:

```
<p style="color: red;">Ini adalah teks merah</p>.
```

Selector

Selector ID

Selector ID: Untuk memilih elemen dengan ID tertentu, gunakan tanda pagar (#) diikuti dengan nama ID sebagai selector. Contohnya:

```
#judul {  
    font-size: 24px;  
}
```

#judul Ini akan menerapkan gaya pada elemen dengan ID judul.

Selector Class

Selector Class: Untuk memilih elemen dengan class tertentu, gunakan titik (.) diikuti dengan nama class sebagai selector. Contohnya:

```
.teks-merah {  
    color: red;  
}
```

Ini akan menerapkan gaya pada semua elemen dengan class teks-merah.

Selector Elemen

Selector Elemen: Untuk memilih semua elemen dengan tag tertentu, gunakan nama tag sebagai selector. Contohnya:

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

Ini akan menerapkan gaya pada semua elemen `<p>` dalam dokumen HTML.

Materi Text

Text-Align

Penjelasan

Kode CSS `p {text-align: right;}` digunakan untuk mengatur tata letak teks dalam elemen paragraf (p) di halaman web. Dalam konteks ini, `text-align: right;` mengarahkan teks di dalam elemen paragraf untuk diposisikan ke kanan dari kotak yang memuatnya, sehingga teks tersebut akan rata kanan.

Kode program

```
p{Text-Align:right;}
```

Hasil



Kesimpulan

text-align merupakan property untuk membuat teks pada tampilan browser menjadi kekanan (text-align:right;),kekiri (Text-Align:left;), ke tengah(Text-Align:center;), dan rata kanan kiri(Text-Align:justify;)

Text-decoration

Penjelasan

Kode CSS p {text-decoration: underline;} digunakan untuk memberikan dekorasi garis bawah pada teks dalam elemen paragraf (p) di halaman web. Dengan demikian, semua teks di dalam elemen paragraf akan memiliki garis bawah.

Kode program

```
p{Text-decoration:underline;}
```

Hasil



Welcome CSS!

Kesimpulan

Kesimpulannya, kode CSS `p {text-decoration: underline;}` digunakan untuk memberikan dekorasi garis bawah pada teks dalam elemen paragraf (`<p>`) di halaman web.

Text-Tranform

Penjelasan

Kode CSS `p {text-transform: lowercase;}` digunakan untuk mengubah semua teks dalam elemen paragraf (p) menjadi huruf kecil (lowercase). Ini berarti semua huruf dalam teks akan ditampilkan dalam bentuk huruf kecil.

Kode program

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

Hasil



welcome css!

Welcome CSS!

Kesimpulan

Kesimpulannya, kode `{text-transform:lowercase;}` ini akan membuat semua teks di dalam elemen paragraf menjadi huruf kecil.

Text-indent

Penjelasan

`text-indent`: Ini adalah properti CSS yang mengatur jarak indentasi awal dari teks dalam sebuah elemen.

`50px`: Ini adalah nilai indentasi yang diberikan dalam piksel. Dalam kasus ini, teks dalam semua elemen paragraf akan di-indentasi sejauh 50 piksel dari sisi kiri.

Kode program

```
p{Text-indent:50px;}
```

Hasil



Welcome CSS!

Welcome CSS!

Kesimpulan

Kesimpulannya, kode tersebut mengatur indentasi awal teks dalam semua elemen paragraf sejauh 50 piksel dari sisi kiri.

Letter-spacing

Penjelasan

`{Letter-spacing: 20px;}` mengatur jarak antara baris dalam elemen paragraf (p) menjadi 20 piksel. Ini berarti setiap baris teks dalam elemen paragraf akan memiliki jarak horizontal

sebesar 20 piksel.

Kode program

```
p{letter-spacing:20px}
```

Hasil



Kesimpulan

Kesimpulannya, kode CSS ini akan membuat jarak antara baris dalam elemen paragraf menjadi 20 piksel, menciptakan ruang horizontal yang lebih besar di antara setiap baris teks.

Line-height

Penjelasan

line-height: Ini adalah properti CSS yang mengatur tinggi baris di dalam elemen. Di sini, nilainya adalah 50px, yang berarti jarak antara baris dalam paragraf akan setara dengan 50 piksel.

Kode program

```
P{Line-height:50px;}
```

Hasil



Welcome CSS!

Welcome CSS!

Kesimpulan

kode Line-height:20px; mengatur tinggi baris untuk semua elemen paragraf menjadi 20 piksel.

Word-Spacing

Penjelasan

word-spacing: Ini adalah properti CSS yang mengatur jarak antara kata-kata di dalam elemen. Di sini, nilainya adalah 50px, yang berarti jarak antara kata-kata dalam paragraf akan setara dengan 50 piksel.

Kode program

```
P{Word-Spacing:50px;}
```

Hasil



Welcome CSS!

Welcome CSS!

Kesimpulan

kode Word-spacing:50px; mengatur jarak antara kata-kata di dalam semua elemen paragraf menjadi 50 piksel.

Materi Background

Background-Size

Penjelasan

background-size: 200px;; Properti ini mengatur ukuran latar belakang. Di sini, latar belakang akan memiliki lebar 200 piksel, dengan tinggi yang disesuaikan agar proporsi aslinya tetap terjaga.

Kode program

```
P{background-size:200px;}
```

Hasil



Kesimpulan

Kode property tersebut bertujuan mengatur ukuran gambar latar belakang dengan size pixel.

Background-Repeat

Penjelasan

background-repeat: no-repeat; Properti ini menentukan apakah gambar latar belakang akan diulang atau tidak. Dalam kasus ini, gambar latar belakang tidak akan diulang.

Kode program

```
P{background-repeat:no-repeat;}
```

Hasil

12.30



93.0 KB/S VoLTE 4G 51



Percobaan Pertama CSS



Welcome CSS!



Kesimpulan

Kode property tersebut berguna agar gambar tidak akan ber ulang Dengan menggunakan value (no-repeat)

Background-attachment

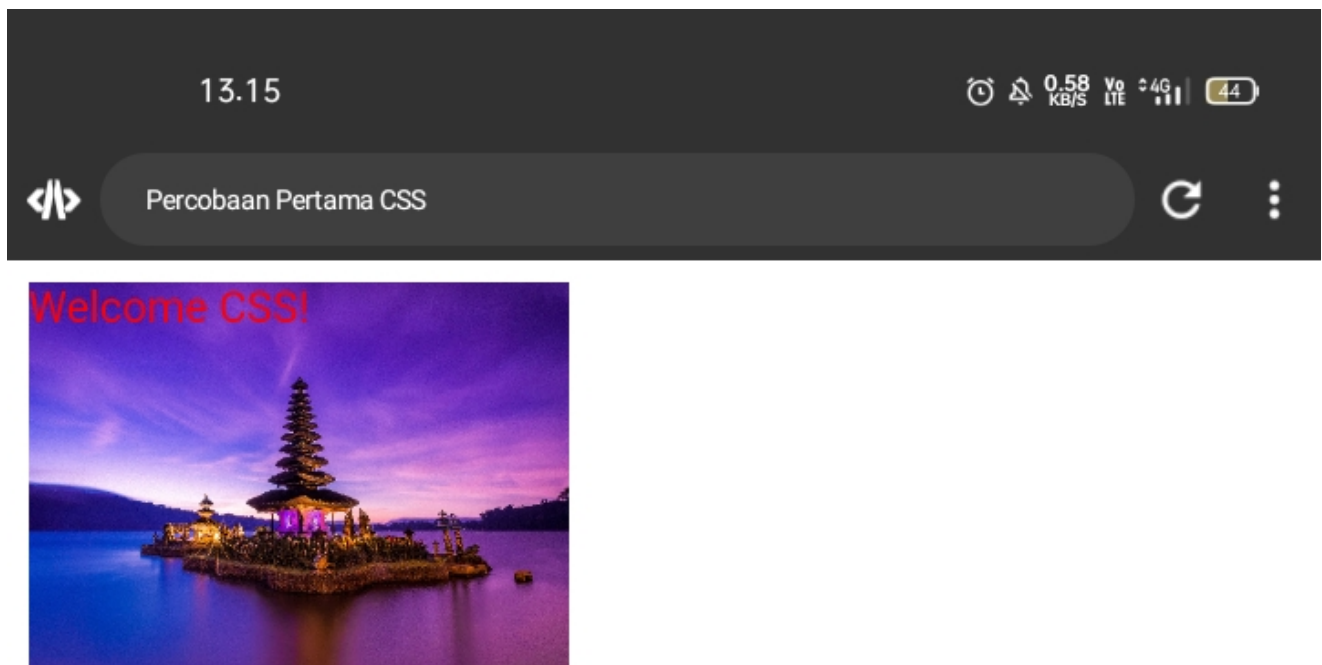
Penjelasan

background-attachment: fixed; Properti ini menentukan apakah latar belakang akan tetap diam atau akan bergulir bersama dengan isi elemen saat pengguna menggulir halaman. Dalam kasus ini, latar belakang akan tetap diam, artinya posisinya akan tetap konstan saat halaman digulir.

Kode program

```
p{background-attachment:fixed;}
```

Hasil



Kesimpulan

Kode property tersebut akan mengatasi gambar yang bergulir akan diam dengan menggunakan value (fixed)

Background-position

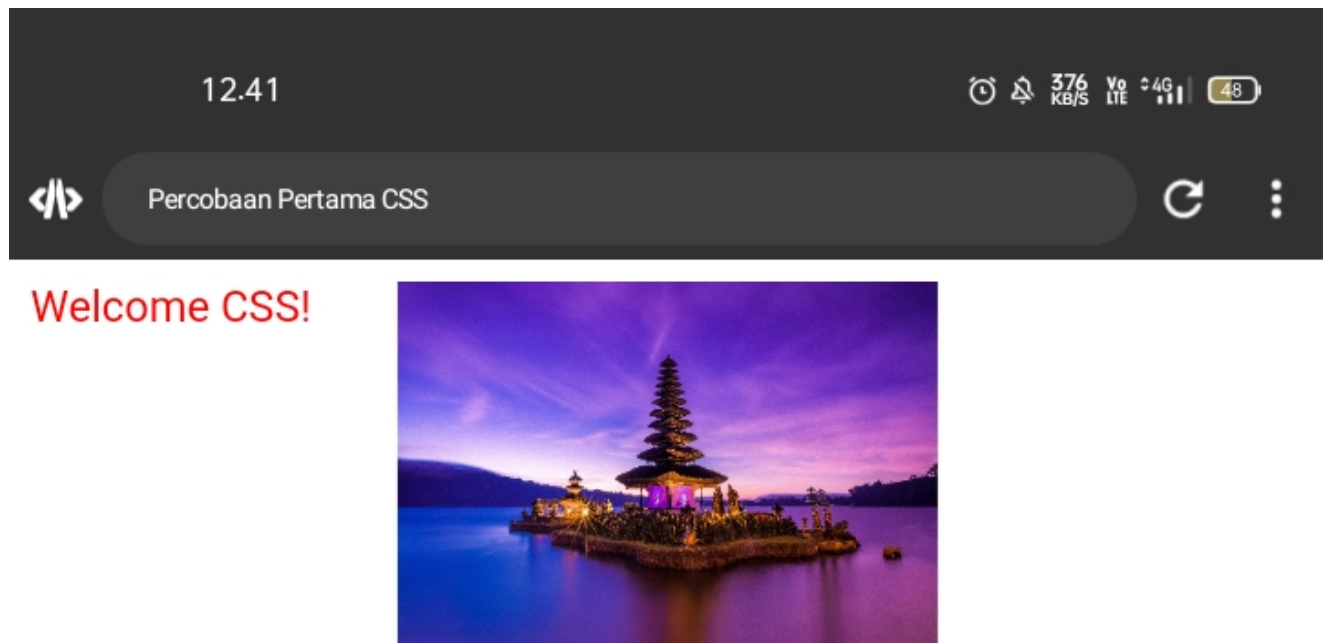
Penjelasan

background-position: top; Properti ini mengatur posisi latar belakang di dalam elemen. Di sini, latar belakang akan diposisikan di bagian atas elemen.

Kode program

```
p
{background-position:top;}
```

Hasil



Kesimpulan

Kode property tersebut akan menentukan posisi gambar latar belakang ke atas (top).

Background-image

Penjelasan

background-image: url(gambar/wallpaper.jpg); Properti ini menentukan gambar apa yang akan digunakan sebagai latar belakang. Di sini, gambar dengan nama file "walpaper.jpg" akan digunakan.

Kode program

```
p
```



```
{Background-image:url(gambar/walpaper.jpg);}
```

Hasil

12.36

77.0 KB/S VoLTE 4G 49



Percobaan Pertama CSS



Welcome CSS!

Kesimpulan

Kode property tersebut akan menentukan gambar Apa yang digunakan untuk latar-belakang.

Materi Font

Font-Weight

Penjelasan

Font-weight: bold; adalah properti CSS yang digunakan untuk mengatur ketebalan teks. Dalam hal ini, nilai yang diberikan adalah "bold", yang membuat teks yang menggunakan properti ini untuk diatur dengan ketebalan yang lebih besar dari teks biasa.

Kode program

```
p{Font-Weight:bold;}
```

Hasil



Welcome CSS!

welcome css

Kesimpulan

Properti CSS ini digunakan untuk mengatur ketebalan teks. Dalam hal ini, nilai yang diberikan adalah "bold".

Font-Size

Penjelasan

Font-size: 50px; adalah properti CSS yang digunakan untuk mengatur ukuran font menjadi 50 piksel. Ini berarti teks yang diberikan akan ditampilkan dengan ukuran 50 piksel, membuatnya lebih besar dari ukuran font standar yang biasa digunakan dalam tata letak halaman web

Kode program

```
P{Font-Size:50px;}
```

Hasil



Kesimpulan

Kode property bertujuan agar teks membuatnya lebih besar dari ukuran font standar yang biasa digunakan dalam tata letak halaman web.

Font-Style

Penjelasan

Font-style: italic; adalah properti CSS yang digunakan untuk mengatur gaya teks menjadi miring (italic). Ini berarti teks yang diberikan akan miring, seperti yang sering kita lihat dalam gaya teks untuk menekankan kata-kata atau membuat teks menonjol.

Kode program

```
p{Font-Style:italic;}
```

Hasil



Kesimpulan

Kode properti yang bertujuan untuk mengatur gaya teks yang dalam kasus ini menjadi miring (italic).

Font-family

Penjelasan

Font-family: Times New Roman; adalah properti CSS yang digunakan untuk menentukan jenis font yang akan digunakan untuk menampilkan teks. Dalam hal ini, jenis font yang dipilih adalah "Times New Roman." Ini akan mengubah teks yang menggunakan properti ini untuk diatur dengan gaya huruf yang khas dari jenis font Times New Roman, yang sering kali terlihat formal dan terstruktur.

Kode program

```
P{Font-family:Times-New-Roman;}
```

Hasil



Welcome CSS!

welcome css

Kesimpulan

Kode properti ini digunakan untuk menetapkan jenis font yang akan digunakan untuk menampilkan teks, di mana dalam kasus ini, jenis font yang dipilih adalah Times New Roman.

Box Model

Materi Border

Border-Radius

Penjelasan

border-radius adalah properti CSS yang digunakan untuk menentukan sudut lengkung dari sudut-sudut sebuah elemen kotak. Dengan menggunakan nilai 10px, Anda menentukan radius lengkung untuk setiap sudut elemen secara terpisah, yaitu: atas kiri, atas kanan, bawah kanan, dan bawah kiri.

Kode program

```
Button{border-radius:10px;}
```

Hasil



Kesimpulan

border-radius dengan nilai 10px 10px 10px 10px mengatur radius lengkung untuk setiap sudut dari sebuah elemen kotak menjadi 10 piksel secara terpisah: atas kiri, atas kanan, bawah kanan, dan bawah kiri.

Border-color

Penjelasan

Properti border-color dalam CSS digunakan untuk menentukan warna dari tepi (border) sebuah elemen HTML. Properti ini adalah salah satu dari beberapa properti yang terkait dengan border dalam model kotak (box model).

Kode program

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

Hasil



Kesimpulan

Properti border-color digunakan untuk menentukan warna dari tepi (border) sebuah elemen HTML dalam model kotak (box model). Anda dapat menggunakan nilai warna seperti nama

warna, nilai RGB, nilai HEX, atau nilai yang relatif seperti "transparent" untuk membuat tepi transparan.

Border-style

Penjelasan

Properti border-style dalam CSS digunakan untuk menentukan jenis garis tepi (border) dari sebuah elemen HTML. Properti ini memiliki beberapa nilai yang dapat digunakan, antara lain:

none : Tidak ada garis tepi. solid : Garis tepi berupa garis lurus dan solid. dashed : Garis tepi untuk garis putus-putus.

Kode program

```
Button{border-style:dashed;}
```

Hasil



Kesimpulan

border-style digunakan untuk menentukan jenis garis tepi (border) dari sebuah elemen HTML. Properti ini mengontrol penampilan visual dari garis tepi dengan nilai seperti solid, dashed, dan lainnya.

Border-width

Penjelasan

Properti border-width digunakan untuk mengatur ketebalan (lebar) dari batas (border) pada sebuah elemen HTML. Properti ini merupakan bagian dari model kotak (box model) yang digunakan untuk menentukan tata letak dan dimensi elemen.

Kode program

```
Button{border-width:20px;}
```

Hasil



Kesimpulan

border-width digunakan untuk mengatur ketebalan (lebar) dari batas (border) pada sebuah elemen HTML.

Anda dapat menentukan nilai untuk border-width dalam piksel, persentase, atau menggunakan kata kunci seperti thin, medium, atau thick. Nilai ini mengontrol tebalnya batas yang mengelilingi elemen.

Study Kasus

Deskripsi

1. Terdapat elemen `<html>` yang mengapit seluruh isi halaman.
2. Elemen `<head>` berisi informasi tentang halaman seperti judul dan stylesheet.
3. Elemen `<body>` berisi konten yang akan ditampilkan pada halaman web.
4. Terdapat sebuah gambar dengan tag `` yang memiliki atribut-atribut seperti src, margin-left, margin-bottom, border, border-radius, height, width, dan position. Ini membuat gambar ditampilkan dengan ukuran tertentu, dengan batasan tepi putih, dan ditempatkan secara tetap di layar.
5. Terdapat paragraf dengan tag `<p>` yang berisi teks "Selamat Datang Di Web Taufik" dengan beberapa pengaturan gaya seperti font, ukuran, dan jarak margin.
6. Terdapat sebuah tombol `<button>` dengan teks "Klik Disini", dengan beberapa gaya seperti tinggi, lebar, warna latar belakang, dan efek hover yang membuatnya berubah warna dan memperbesar saat digerakkan mouse di atasnya.

Kode Program

```
<!Doctype html>
<html>
<title> belajar css</title>
<head>
<style>
body{background-color: purple;}
p{font-family: Times New Roman;
font-size: 25px;
margin-top: 30px;}
img{
margin-left:230px;
margin-bottom: 100px;
border: 5px solid white;
border-radius: 50% 50%;
height: 20vh;
width: 20vh;
position: fixed;
}
button{height: 50px;width: 100px;
background-color: red;
border: solid ;
margin-left: 50px;
}
</style>
</head>
<body>
<span>

<p>Selamat Datang <br>Di <b>Web Taufik</b></p> </span>
<span>
</span>
<button>Klik Disini</button>
</body>
</html>
```

Hasil



belajar css



Selamat Datang Di Web Taufik

Klik Disini



Materi Height & width

Height

Penjelasan

height: Properti ini digunakan untuk menentukan tinggi (vertikal) dari sebuah elemen. Anda bisa memberikan nilai dalam piksel, persentase, atau nilai lainnya seperti auto yang membiarkan browser menentukan tinggi elemen berdasarkan kontennya.

Kode program

```
Button{height:100px;}
```

Hasil



Kesimpulan

Properti height digunakan untuk mengatur tinggi (vertikal) dari sebuah elemen.

Width

Penjelasan

width: Properti ini digunakan untuk menentukan lebar (horizontal) dari sebuah elemen.

Sama seperti height, nilai dapat diberikan dalam piksel, persentase, atau menggunakan nilai seperti auto yang mengizinkan browser menentukan lebar elemen berdasarkan kontennya.

Kode program

```
Button{width:100px;}
```

Hasil



Kesimpulan

Properti width digunakan untuk mengatur lebar (horizontal) dari sebuah elemen.

Materi Padding

Padding-top

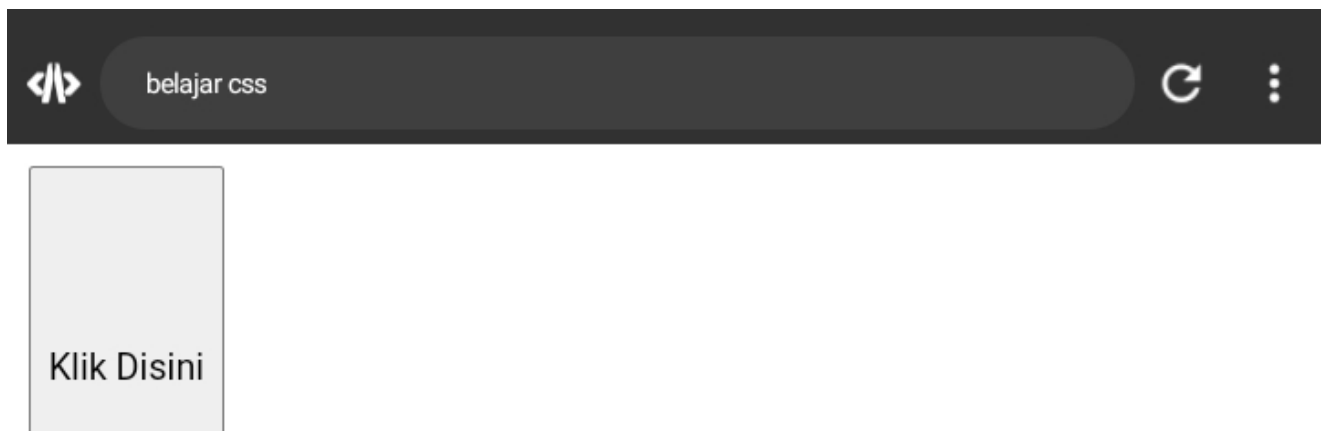
Penjelasan

padding-top adalah properti CSS yang digunakan untuk menentukan jumlah ruang polos yang diberikan di bagian atas dari konten sebuah elemen HTML. Ini memungkinkan Anda untuk menambahkan ruang tambahan di bagian atas elemen, antara konten dan batas atasnya

Kode program

```
Button{padding-top:100px;}
```

Hasil



Kesimpulan

padding-top dalam CSS digunakan untuk menambahkan ruang di bagian atas konten dalam sebuah elemen HTML, antara konten dan batas atas elemen tersebut.

Padding-bottom

Penjelasan

padding-bottom adalah properti CSS yang digunakan untuk menambahkan ruang polos di bagian bawah dari sebuah elemen HTML. Ini memberikan jarak antara konten dalam

elemen dan batas bawah dari elemen tersebut dalam model kotak CSS.

Kode program

```
Button{padding-bottom:100px;}
```

Hasil



Kesimpulan

Dengan menggunakan property padding-bottom, Anda dapat mengontrol seberapa jauh konten di dalam elemen tersebut berjarak dari batas bawahnya.

Padding-left

Penjelasan

Padding-left adalah bagian dari model kotak (box model) di CSS yang mengacu pada jarak antara tepi kiri dari sebuah elemen dan konten di dalamnya. Ini memungkinkan untuk memberikan ruang tambahan di sebelah kiri elemen tanpa mempengaruhi lebar total elemen tersebut.

Kode program

```
Button{padding-left:100px;}
```

Hasil



belajar css



Klik Disini

Kesimpulan

Padding-left adalah jarak antara tepi kiri dari sebuah elemen dan konten di dalamnya dalam model kotak CSS. Ini memungkinkan penambahan ruang di sebelah kiri elemen tanpa memengaruhi lebar total elemen tersebut.

Padding-right

Penjelasan

padding-right adalah properti CSS yang digunakan untuk menentukan jumlah ruang kosong (padding) di sebelah kanan elemen HTML. Padding adalah jarak antara tepi dalam elemen dan kontennya.

Kode program

```
Button{padding-right:100px;}
```

Hasil



Kesimpulan

padding-right adalah properti CSS yang digunakan untuk menentukan jumlah ruang kosong di sebelah kanan elemen HTML, yang berguna untuk menciptakan tata letak yang lebih baik dan meningkatkan tampilan desain yang bagus di halaman web.

Materi Margin

Margin-top

Penjelasan

margin-top adalah properti CSS yang digunakan untuk mengatur jarak antara tepi atas (atas) dari suatu elemen dan elemen sekitarnya dalam model kotak (box model). Dengan properti ini, Anda dapat menentukan seberapa jauh elemen tersebut akan diletakkan dari elemen di atasnya.

Kode program

```
Button{Margin-top:100px;}
```

Hasil



Klik Disini

Kesimpulan

margin-top adalah properti yang digunakan untuk mengatur jarak antara tepi atas (atas) dari suatu elemen dan elemen sekitarnya dalam model kotak (box model).

Margin-bottom

Penjelasan

margin-bottom adalah properti CSS yang digunakan untuk menentukan jarak antara bagian bawah dari sebuah elemen dengan elemen-elemen di sekitarnya. Properti ini adalah bagian dari model kotak (box model) CSS dan digunakan untuk mengatur tata letak dan ruang di antara elemen-elemen pada halaman web.

Kode program

```
Button{Margin-bottom:100px;}
```

Hasil



Klik Disini

Klik Disini

Kesimpulan

margin-bottom adalah properti CSS yang digunakan untuk menentukan jarak antara bagian bawah dari sebuah elemen dengan elemen-elemen di sekitarnya dalam model kotak (box model).

Margin-left

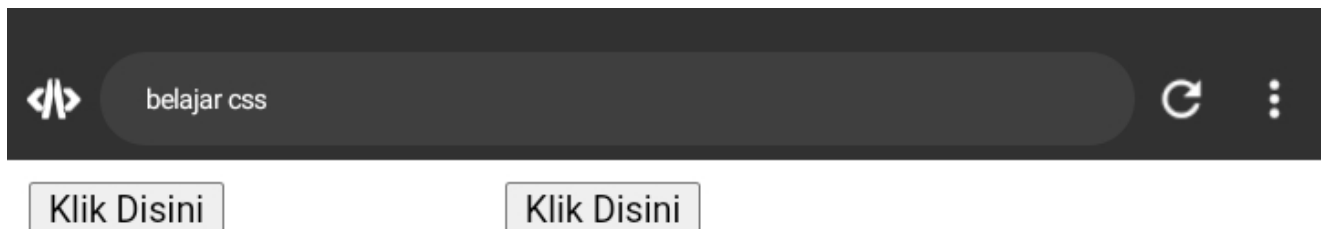
Penjelasan

margin-left adalah properti CSS yang digunakan untuk menentukan jarak (spasi) dari sisi kiri sebuah elemen terhadap elemen sebelumnya dalam tata letak halaman web. Ini memungkinkan Anda untuk mengatur ruang kosong di sebelah kiri elemen, sehingga memengaruhi tata letak keseluruhan halaman web

Kode program

```
Button{Margin-left:100px;}
```

Hasil



Kesimpulan

margin-left adalah properti CSS yang digunakan untuk menentukan jarak dari sisi kiri sebuah elemen terhadap elemen sebelumnya dalam tata letak halaman web. Semakin besar nilai margin-left, semakin besar pula jaraknya dari sisi kiri elemen sebelumnya.

Margin-right

Penjelasan

margin-right adalah properti CSS yang digunakan untuk menentukan jarak (margin) antara tepi kanan sebuah elemen dengan elemen lain di sekitarnya. Ini memengaruhi tata letak horizontal dari elemen tersebut di dalam model kotak (box model).

Kode program

```
Button{Margin-right:100px;}
```

Hasil



Kesimpulan

margin-right adalah properti CSS yang digunakan untuk mengatur jarak antara tepi kanan suatu elemen dengan elemen lain di sekitarnya dalam model kotak CSS.

Pseudo-classes

Hover

Penjelasan

Pernyataan kode program 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

```
background-color: #45a049;  
  
color: #FF0000;
```

```
}
```

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 `Klik!`, 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: #FF0000;  
  
}
```

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.

Penjelasan

Pernyataan tersebut menggunakan pseudo-class `:link` untuk menargetkan perilaku saat tombol tersebut berfungsi sebagai link (google). Saat tombol tersebut di klik otomatis akan di arahkan ke google , properti `color` digunakan untuk mengatur warna teks tombol, di mana nilai yang ditetapkan adalah `"red"`, yang merupakan kode warna untuk warna merah.

Kode Program

```
.btn-link:active {  
  
    background-color: red;  
  
}
```

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 `"Color:ungu"`, yang merupakan kode warna untuk warna ungu.

Kode Program

```
.btn-visited:visited {  
  
    color: purple;  
  
}
```

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.

Materi Transition

Transition-delay

Penjelasan

1. Transition delay 10s;: Ini adalah properti CSS yang mengatur penundaan transisi saat perubahan terjadi pada elemen. Dalam hal ini, tombol memiliki penundaan transisi selama 10 detik sebelum transisi dimulai ketika tombol mengalami perubahan.
2. width: 50vh;: Properti ini mengatur lebar tombol menjadi 50% dari tinggi viewport (viewport height). Hal ini memastikan tombol memiliki lebar yang proporsional terhadap ukuran layar pengguna, sehingga dapat lebih responsif terhadap perangkat yang berbeda.
3. margin-left: 50px;: Properti ini mengatur jarak margin kiri tombol sebesar 50 piksel, memberikan ruang di sebelah kiri tombol.
4. button:hover {background-color: blue;}: Ini adalah aturan yang mengubah warna latar belakang tombol menjadi biru saat tombol ditemukan saat mouse mengarah ke atasnya.

Kode program

```
button{  
  transition-delay:10s;  
  width: 50vh;  
  margin-left: 50px;  
}  
button:hover{background-color:blue;  
}
```

Hasil Before



localhost:8158/Transition.html



Klikk Dsini

Hasil After



localhost:8158/Transition.html



Klik Dsini

Kesimpulan

Kode program tersebut menambahkan efek transisi yang ditunda, mengatur lebar tombol relatif terhadap tinggi layar, memberikan margin kiri, dan mengubah warna latar belakang tombol saat digerakkan. Ini meningkatkan interaktivitas dan estetika tombol pada halaman web.

Transition-Duration

Penjelasan

`transition-duration: 5.5s;` Mengatur durasi transisi efek saat mengubah properti CSS (dalam hal ini, efek saat mengarahkan kursor ke tombol).

`button:hover:` Ini adalah pseudo-class CSS yang menargetkan tombol ketika kursor berada di atasnya.

`background-color: yellow;` Mengatur warna latar belakang tombol menjadi kuning saat tombol diarahkan dengan kursor.

Kode program


```

button {
color: blue;
border-color: green;
border-width: 3px;
background-color: white;
font-family: Arial;
margin-top: 50px;
margin-left: 50px;
width: 100px;
height: 50px;
border-radius: 100px 100px;
transition-duration: 5.5s;
}
button:hover {
background-color: yellow;
}
...

```

hasil before

![IMG](IMG_20240427_221338.jpg)

hasil after

![IMG](IMG_20240427_221351.jpg)

Kesimpulan

kode tersebut mengatur tampilan dan perilaku sebuah tombol di halaman web, dengan mengontrol warna, ukuran, jarak, jenis font, dan efek transisi saat interaksi pengguna.

Transition-Property

Penjelasan

`transition-duration: 5.5s;` dan `transition-property: background-color, width;`

Durasi peralihan adalah 5,5 detik dan properti yang berubah saat peralihan adalah warna latar belakang dan lebar tombol.

`button:hover { ... }`: Gaya yang akan diterapkan saat mouse mengarah ke atas tombol (hover), di mana warna latar belakang berubah menjadi kuning dan lebar tombol menjadi 50 piksel.

Kode program

```CSS

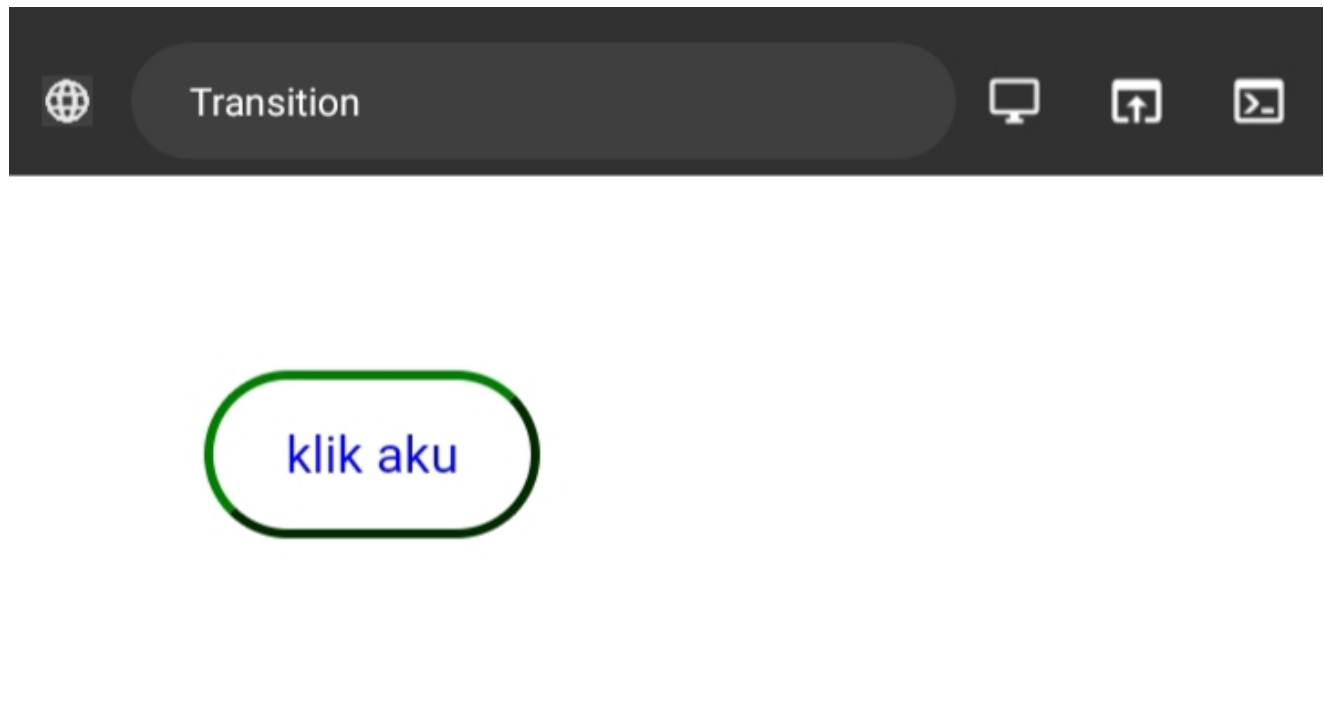
```

 button {
color: blue;
border-color: green;
border-width: 3px;
background-color: white;
font-family: Arial;
margin-top: 50px;
margin-left: 50px;
width: 100px;
height: 50px;
border-radius: 100px 100px;

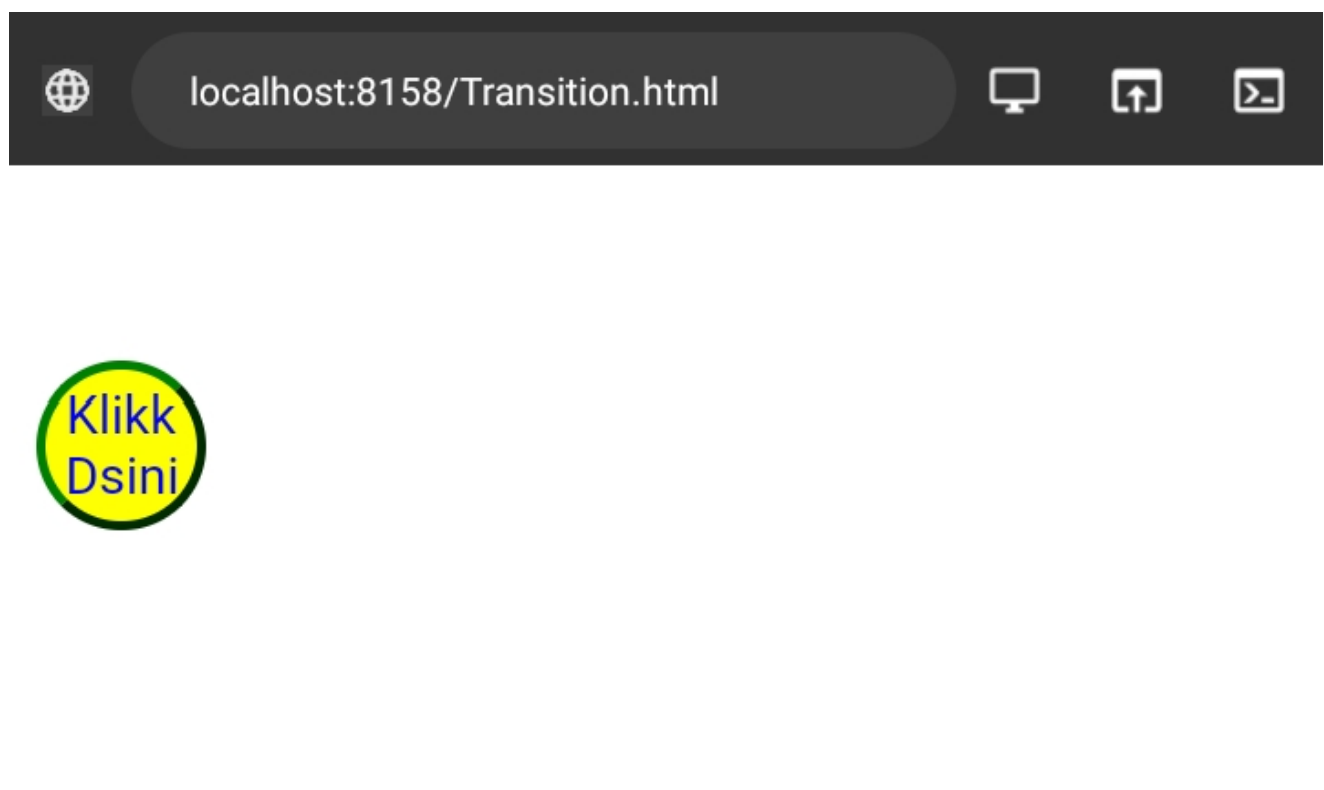
```

```
transition-duration: 5.5s;
transition-property: background-color, width;
}
button:hover {
 background-color: yellow;
 width: 50px;
}
```

## hasil before



## Hasil after



## Kesimpulan

Kode tersebut mendefinisikan gaya untuk tombol dengan menggunakan berbagai properti CSS untuk mengatur tampilan dan perilaku saat interaksi pengguna. Saat tombol dihover, warna latar belakangnya akan berubah menjadi kuning dan lebarnya akan menyusut menjadi 50 piksel.

---

## Transition-Timing-function

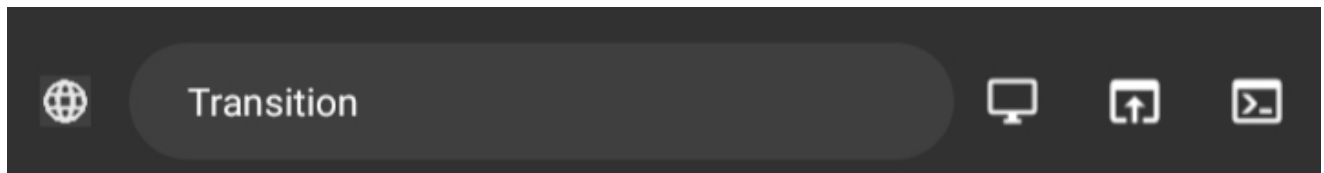
### Penjelasan

transition-timing-function: ease-in-out; Mengatur fungsi timing peralihan menjadi 'ease-in-out', yang membuat perubahan dimulai perlahan, mempercepat di tengah, dan melambat lagi saat mendekati akhir perubahan.

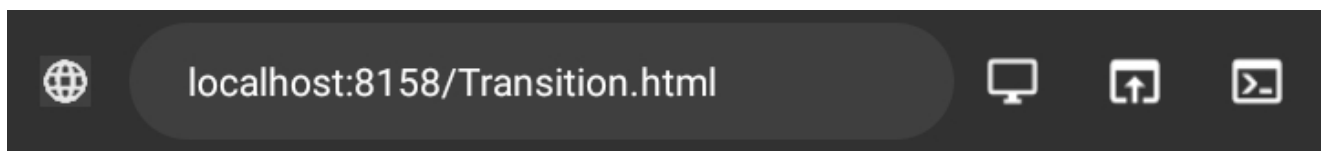
### Kode program

```
button {
color: blue;
border-color: green;
border-width: 3px;
background-color: white;
font-family: Arial;
margin-top: 50px;
margin-left: 50px;
width: 100px;
height: 50px;
border-radius: 100px 100px;
transition-duration: 5.5s;
transition-delay: 1.5s;
transition-timing-function: ease-in-out;
}
button:hover {
width: 200px;
}
```

### hasil before



## hasil after



## Kesimpulan

Kode tersebut mendefinisikan gaya untuk tombol dengan menggunakan berbagai properti CSS untuk mengatur tampilan dan perilaku saat interaksi pengguna. Saat tombol dihover, lebarnya akan diperbesar menjadi 200 piksel tanpa mengubah properti lainnya. Peralihan perubahan lebar tombol memiliki durasi 5,5 detik, dimulai setelah jeda 1,5 detik, dengan pola percepatan dan perlambatan yang halus

---

# Studi Kasus

## Deskripsi

1. Terdapat elemen `<html>` yang mengapit seluruh isi halaman.
2. Elemen `<head>` berisi informasi tentang halaman seperti judul dan stylesheet.
3. Elemen `<body>` berisi konten yang akan ditampilkan pada halaman web.
4. Terdapat pengaturan gaya CSS di dalam elemen `<style>`:  
Latar belakang halaman diatur menjadi warna ungu.
5. Paragraf(`<p>`) memiliki gaya teks yang ditetapkan menjadi Times New Roman dengan ukuran 25px dan margin atas sebesar 30px.
  - Gambar (img) memiliki beberapa gaya seperti margin kiri 230px, margin bawah 100px, border 5px solid putih, border-radius 50% untuk membuatnya bundar, dan ukuran tetap 20% tinggi dan lebar dari viewport dengan posisi tetap di layar.
  - Tombol (button) memiliki gaya seperti tinggi 50px, lebar 100px, latar belakang merah, dan margin kiri 50px.Konten yang ditampilkan pada halaman web meliputi gambar dengan teks "Selamat Datang" diikuti oleh teks "Di Web Taufik" dalam sebuah paragraf, dan sebuah tombol dengan teks "Klik Disini".

## Kode program

```
<!Doctype html>
<html>
<title> belajar css</title>
<head>
<style>
body{background-color: purple;}
p{font-family: Times New Roman;
font-size: 25px;
margin-top: 30px;
}
img{
margin-left:230px;
margin-bottom: 100px;
border: 5px solid white;
border-radius: 50% 50%;
height: 20vh;
width: 20vh;
position: fixed;
}
button{height: 50px;width: 100px;
background-color: red;
border: solid ;
margin-left: 50px;
```

```

}
button:hover{background-color:blue;
 transition:all in 1.5 ease-in-out;
 transform:scaleX(1.5) ;
}
</style>
</head>
<body>

 <p>Selamat Datang
Di Web Taufik</p>

 <button>Klik Disini</button>
 </body>
</html>

```

## Hasil before



## hasil after



## Materi Transfrom

### Transform-Scale

#### Penjelasan

Ketika tombol ditekan (button:active):

width: 150px; dan height: 150px;; Mengatur lebar dan tinggi tombol menjadi 150 piksel, membuatnya membesar saat ditekan.

transform: scale(0.70);: Mengubah skala (ukuran) tombol menjadi 0,70 kali ukuran aslinya saat ditekan, memberikan efek tampilan seperti tombol tertekan.

#### Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
```

```
margin-top: 50px;
margin-left: 50px;
width: 100px;
height: 50px;
border-radius: 50px 50px;
}
button:active {
width: 150px;
height: 150px;
transform: scale(0.70);
}
```

## hasil before



## Hasil after





## Kesimpulan

Kode tersebut mendefinisikan gaya untuk tombol dengan menggunakan berbagai properti CSS untuk mengatur tampilan dan perilaku saat interaksi pengguna. Saat tombol ditekan, ukuran tombol akan membesar dan memiliki efek visual penekanan yang menarik.

## Transform-ScaleX

### Penjelasan

Ketika tombol ditekan (`button:active`):

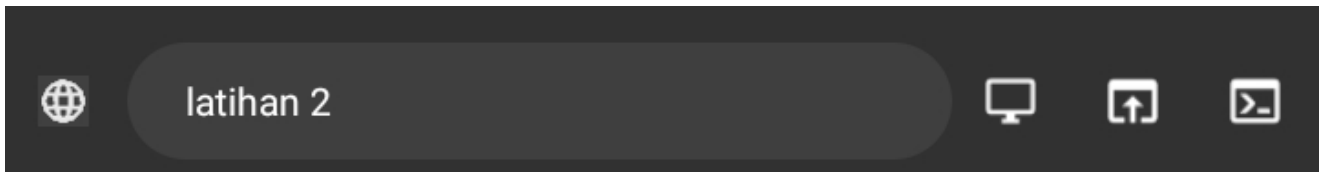
`width: 150px; dan height: 150px;` Mengatur lebar dan tinggi tombol menjadi 150 piksel, membuatnya membesar saat ditekan.

`transform: scaleX(0.70);` Mengubah skala lebar tombol menjadi 0,70 kali lebar aslinya saat ditekan, tetapi tingginya tetap sama, memberikan efek tampilan seperti tombol tertekan secara horizontal.

## Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
 margin-top: 50px;
 margin-left: 50px;
 width: 100px;
 height: 50px;
 border-radius: 50px 50px;
}
button:active {
 width: 150px;
 height: 150px;
 transform: scaleX(0.70);
}
```

## hasil before



## Hasil After



**klik sini**

## Kesimpulan

Kode tersebut mendefinisikan gaya untuk tombol dengan menggunakan berbagai properti CSS untuk mengatur tampilan dan perilaku saat interaksi pengguna. Saat tombol ditekan, ukuran tombol akan membesar dan lebarnya akan menyusut, menciptakan efek visual penekanan yang menarik.

## Transform-Rotate

### Penjelasan

Ketika tombol dihover (button: hover):

width: 150px; dan height: 50px;: Mengatur lebar tombol menjadi 150 piksel dan tetap mempertahankan tinggi 50 piksel saat tombol dihover.

transform: rotate(250deg);: Mengatur rotasi tombol sebesar 250 derajat saat tombol dihover, memberikan efek visual putaran pada tombol.

### Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
 margin-top: 50px;
 margin-left: 50px;
 width: 100px;
 height: 50px;
 border-radius: 50px 50px;
}
button:hover {
 width: 150px;
 height: 50px;
 transform: rotate(250deg)
}
```

## hasil



## Kesimpulan

Kode tersebut mendefinisikan gaya untuk tombol dengan menggunakan berbagai properti CSS untuk mengatur tampilan dan perilaku saat interaksi pengguna. Saat tombol dihover,

tombol akan memperbesar lebarnya dan melakukan rotasi sebesar 250 derajat, menciptakan efek visual menarik pada tombol.

---

## Transform-Skew

### Penjelasan

Ketika tombol ditekan (button:active):

width: 150px; dan height: 150px; Mengatur lebar dan tinggi tombol menjadi 150 piksel saat tombol aktif, membuatnya membesar saat ditekan.

transform: skewX(-25deg); Mengubah bentuk tombol dengan mencondongkan sumbu X sebesar -25 derajat saat tombol ditekan, memberikan efek visual miring ke samping.

### Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
 margin-top: 50px;
 margin-left: 50px;
 width: 100px;
 height: 50px;
 border-radius: 50px 50px;
}
button:active {
 width: 150px;
 height: 150px;
 transform: skewX(-25deg);
}
```

### hasil before



klik sni

## hasil after



*klik sni*

## Kesimpulan

Kode tersebut mendefinisikan gaya untuk tombol dengan menggunakan berbagai properti CSS untuk mengatur tampilan dan perilaku saat interaksi pengguna. Saat tombol ditekan, tombol akan membesar dan miring ke samping, menciptakan efek visual menarik pada tombol.

---

## Transform-SkewX

### Penjelasan

Ketika tombol sedang aktif (button:active):

width: 150px; dan height: 150px;: Mengatur lebar dan tinggi tombol menjadi masing-masing 150 piksel saat tombol sedang aktif, membuatnya membesar saat ditekan.

transform: skew(25deg);: Menerapkan efek transformasi skew pada tombol sebesar 25 derajat, memberikan efek visual miring pada tombol saat ditekan.

## Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
 margin-top: 50px;
 margin-left: 50px;
 width: 100px;
 height: 50px;
 border-radius: 50px 50px;
}
button:active {
 width: 150px;
 height: 150px;
 transform: skew(25deg);
}
```

## hasil before



## hasil after



latihan 2



## Kesimpulan

Kode tersebut menghasilkan tampilan tombol dengan properti-properti yang ditetapkan, dan memberikan efek membesar dan miring saat tombol ditekan

---

## Transform-Translate

### Penjelasan

Ketika tombol dihover (button: hover):

width: 150px; dan height: 50px;; Mengatur lebar tombol menjadi 150 piksel dan tetap mempertahankan tinggi 50 piksel saat tombol dihover.

transform: translate(5px, 15px); Menggeser posisi tombol sebesar 5 piksel ke kanan dan 15 piksel ke bawah saat tombol dihover, memberikan efek visual pergeseran.

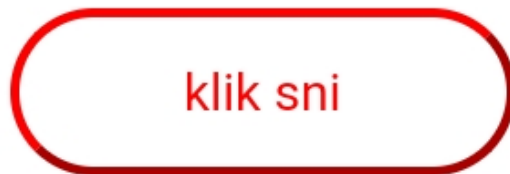
### Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
 margin-top: 50px;
 margin-left: 50px;
 width: 100px;
 height: 50px;
```



```
border-radius: 50px 50px;
}
button:hover {
 width: 150px;
 height: 50px;
 transform: translate(5px, 15px)
}
```

## hasil before



## hasil after



## Kesimpulan

Kode tersebut menghasilkan tampilan tombol dengan properti-properti yang ditetapkan, dan memberikan efek perubahan lebar serta pergeseran saat tombol dihover.

---

## Transform-Matrix

### Penjelasan

Ketika tombol dihover (`button:hover`):

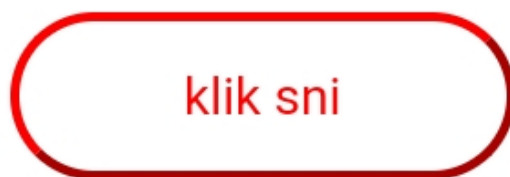
`width: 50px;` dan `height: 150px;`: Mengatur lebar tombol menjadi 50 piksel dan tingginya menjadi 150 piksel saat tombol dihover.

`transform: matrix(5px, 15px, 15px, 15px);`: Mengubah bentuk dan posisi tombol dengan matriks transformasi. Namun, properti `matrix()` tidak cocok untuk mengatur ukuran dan transformasi seperti yang dimaksud dalam kasus ini.

### Kode program

```
button {
 color: red;
 border-color: red;
 border-width: 3px;
 background-color: white;
 font-family: Arial;
 margin-top: 50px;
 margin-left: 50px;
 width: 100px;
 height: 50px;
 border-radius: 50px 50px;
}
button:hover {
 width: 50px;
 height: 150px;
 transform:matrix(5px,15px,15px,15px)
}
```

## hasil before



## hasil After



## Kesimpulan

Meskipun kode memiliki kesalahan sintaks dalam penggunaan properti `matrix()` untuk transformasi, tujuannya adalah untuk memberikan efek perubahan ukuran dan bentuk tombol saat dihover. Untuk mencapai hasil yang diinginkan, properti transformasi yang lebih tepat seperti `scale()` dan `translate()` dapat digunakan.

---

## Materi flexbox

---

### Display Flex

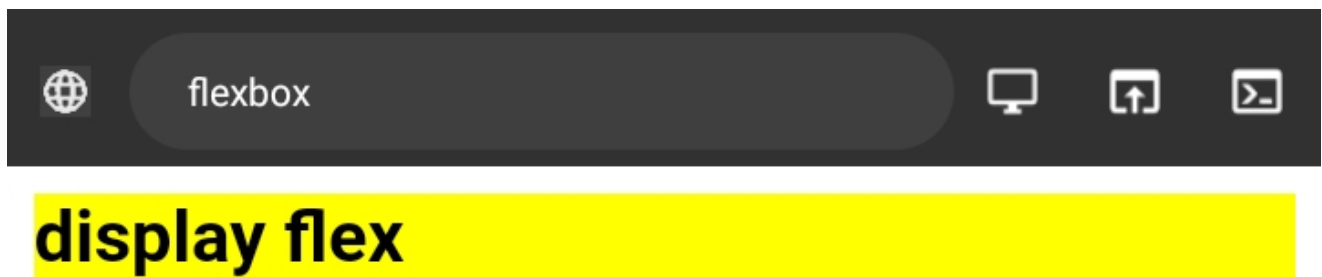
#### Penjelasan

`display: flex;` adalah sebuah properti CSS yang digunakan untuk mengatur sebuah container agar anak-anak elemennya (child elements) menjadi flex items dan mengaktifkan model layout Flexbox.

## Kode Program

```
<!DOCTYPE html>
<html>
 <head>
 <title>materi flexbox</title>
 <style>
 h3 {
 display: flex;
 background-color: yellow;
 }
 </style>
 </head>
 <body>
 <h2>display flex</h2>
 </body>
</html>
```

## Hasil



## Kesimpulan

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

## flex-direction

## Penjelasan

.container adalah kelas CSS yang diberikan kepada sebuah elemen div. Ini digunakan untuk membuat sebuah container dengan `display flex` dan `flex-direction column`, sehingga elemen-elemen di dalamnya akan ditata dalam satu kolom.

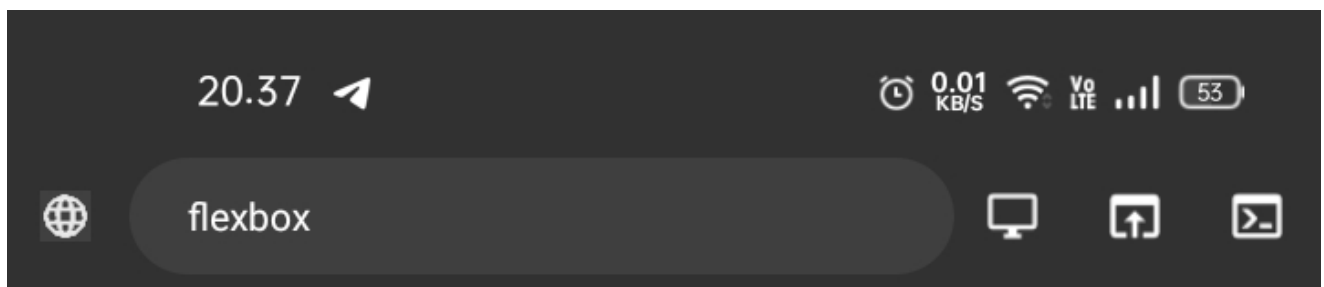
.item-1 adalah kelas CSS yang diberikan kepada sebuah elemen div di dalam container. Elemen ini memiliki warna latar merah dan ditentukan lebar dan tingginya.

## Kode program

```
<!DOCTYPE html>
<html>
 <head>
 <title>flexbox</title>
 <style>
 .container {
 display: flex;
 flex-direction: column;
 background-color: green;
 height: 250px;
 width: 500px;

 }
 .item-1 {
 display: flex;
 background-color: red;
 width: 100px;
 height: 100px;
 }
 </style>
 </head>
 <body>
 <div class="container">
 <div class="item-1"></div>
 </div>
 </body>
</html>
```

**hasil**



## Kesimpulan

Kode program tersebut membuat sebuah halaman web dengan background hijau, container dengan lebar 500px dan tinggi 250px. Di dalam container, terdapat sebuah elemen div dengan background merah, lebar 100px, dan tinggi 100px. Pengaturan tersebut dicapai menggunakan CSS Flexbox.

---

## align-items

## Penjelasan

Nilai properti align-items dalam CSS harus ditulis dengan huruf kecil, yaitu align-items, bukan Align-items.

Setelah perbaikan tersebut, kode program akan berfungsi dengan benar.

Penjelasan: Kode program tersebut hampir sama dengan sebelumnya, namun ada tambahan aturan CSS align-items: center; pada kelas .container. Aturan ini akan membuat semua elemen di dalam container, dalam hal ini hanya .item-1, akan secara vertikal terpusat di tengah container.

## Kode program

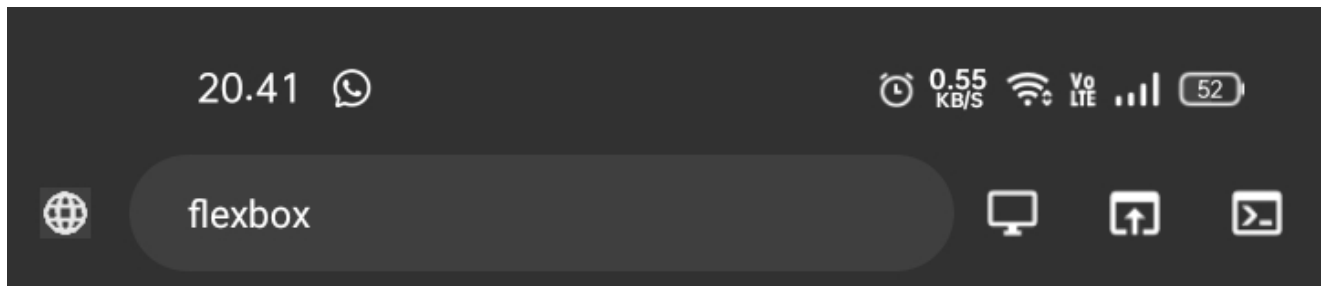
```
<!DOCTYPE html>
<html>
 <head>
 <title>flexbox</title>
 <style>
 .container {
 display: flex;
 flex-direction: column;
 background-color: green;
 height: 250px;
 width: 500px;
 align-items:center;

 }
 .item-1 {
 display: flex;
 background-color: red;
 width:100px;
 height: 100px;

 }
 </style>
 </head>
 <body>
 <div class="container">
 <div class="item-1"></div>
 </div>
 </body>
</html>
```

## hasil





## Kesimpulan

Kode program tersebut menghasilkan halaman web dengan background hijau, container dengan lebar 500px dan tinggi 250px. Di dalam container, terdapat sebuah elemen div dengan background merah, lebar 100px, dan tinggi 100px, yang terletak secara vertikal terpusat di tengah container. Ini juga menggunakan CSS Flexbox untuk pengaturan tata letaknya.

---

## justify-content

### Penjelasan

Nilai properti justify-content dalam CSS harus ditulis dengan huruf kecil, yaitu justify-content, bukan Justify-content.

Setelah perbaikan tersebut, kode program akan berfungsi dengan benar.

Penjelasan: Kode program tersebut hampir sama dengan sebelumnya, namun ada tambahan aturan CSS `justify-content: center;` pada kelas `.container`. Aturan ini akan membuat semua elemen di dalam container, dalam hal ini hanya `.item-1`, akan secara horizontal terpusat di tengah container.

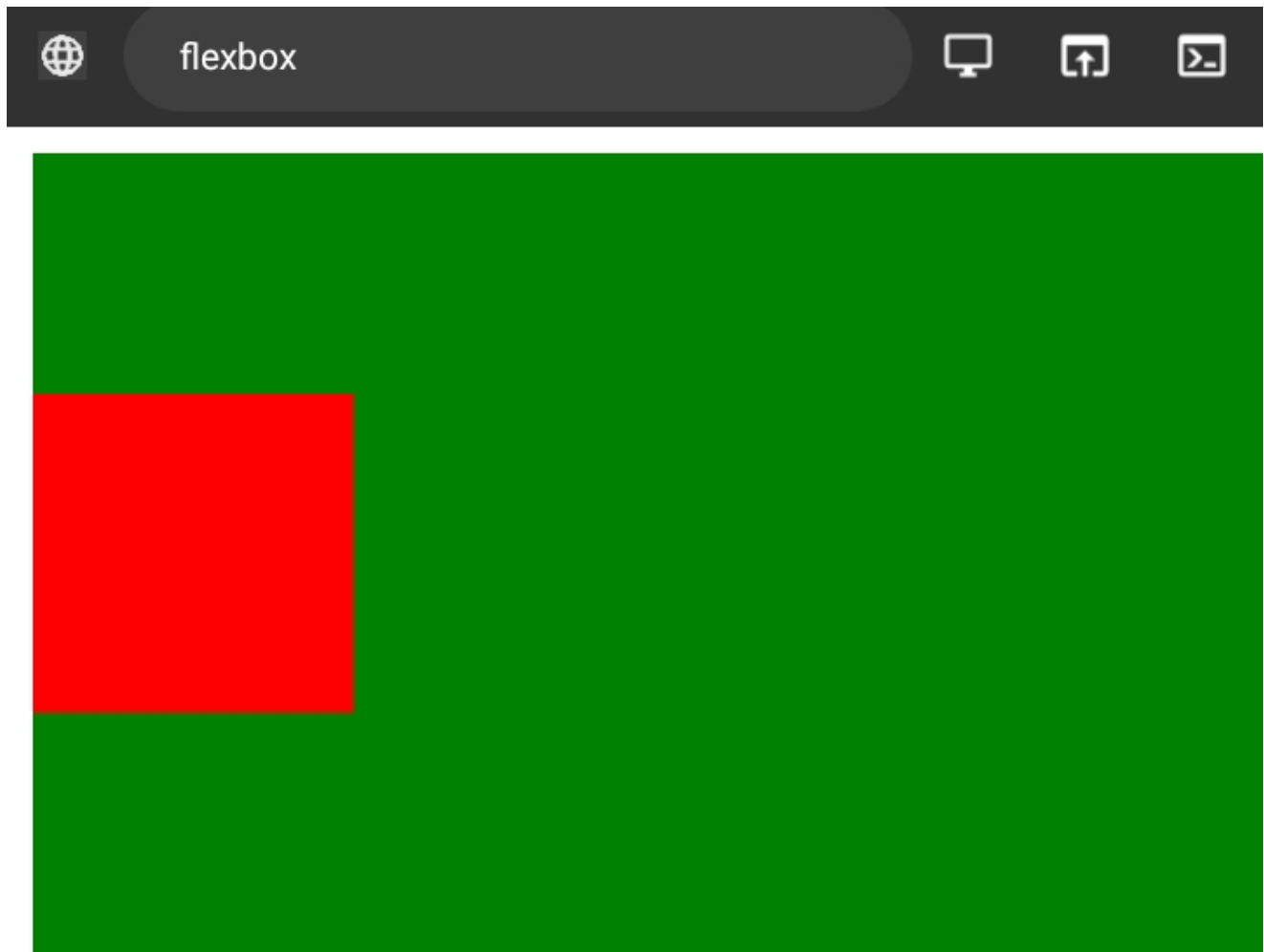
## Kode program

```
<!DOCTYPE html>
<html>
 <head>
 <title>flexbox</title>
 <style>
 .container {
 display: flex;
 flex-direction: column;
 background-color: green;
 height: 250px;
 width: 500px;
 justify-content: center;

 }
 .item-1 {
 display: flex;
 background-color: red;
 width: 100px;
 height: 100px;

 }
 </style>
 </head>
 <body>
 <div class="container">
 <div class="item-1"></div>
 </div>
 </body>
</html>
```

## hasil



## Kesimpulan

Kode program tersebut menghasilkan halaman web dengan background hijau, container dengan lebar 500px dan tinggi 250px. Di dalam container, terdapat sebuah elemen div dengan background merah, lebar 100px, dan tinggi 100px, yang terletak secara horizontal terpusat di tengah container. Ini juga menggunakan CSS Flexbox untuk pengaturan tata letaknya.

---

## STUDI KASUS

### Deskripsi

- Sebuah judul halaman dengan tulisan "Selamat Datang di Web Jordan".  
Sebuah tombol dengan teks "Klik disini".

- Sebuah gambar yang ditampilkan dalam bentuk lingkaran dengan ukuran 250x250 piksel.
- Penggunaan properti CSS untuk mengatur tampilan elemen-elemen tersebut, seperti warna latar belakang, warna teks, dan jarak antar elemen.
- Penggunaan properti CSS Flexbox untuk mengatur posisi elemen-elemen dalam container.

## Kode program

```
<!DOCTYPE html>
<html>
<head>
 <title>Tantangan flexbox</title>
</head>
<style>
 body{
 background-color:purple;
 }
 .main-container {
 background-color: purple;
 }
 .box-item-1{
 font-size: 50px;
 color: white;
 margin-top:10px;
 }
 .box-item-2{
 font-size: 50px;
 color: white;
 margin-top:10px;
 }

 .J {
 background-color: purple;
 margin-left:200px;
 padding-left:29px;
 padding-top:20px;
 padding-bottom:35px;
 padding-right:29px;
 color: orangered;
 border-color:orangered;
 margin-top: 50px;
 margin-right:60px;
 margin-bottom:90px;
 width: 130px;
 }
```

```

img{
 width: 250px;
 height: 250px;
 border-radius: 100%;
 border-color: white;
 margin-left: 350px;
 margin-top: -400px;
 position: fixed;

}
</style>
<body class="main-container">
 <div class="hero-container">
 <div class="box-container">
 <h1 class="item box-item-1">
Selamat Datang
 </h1>

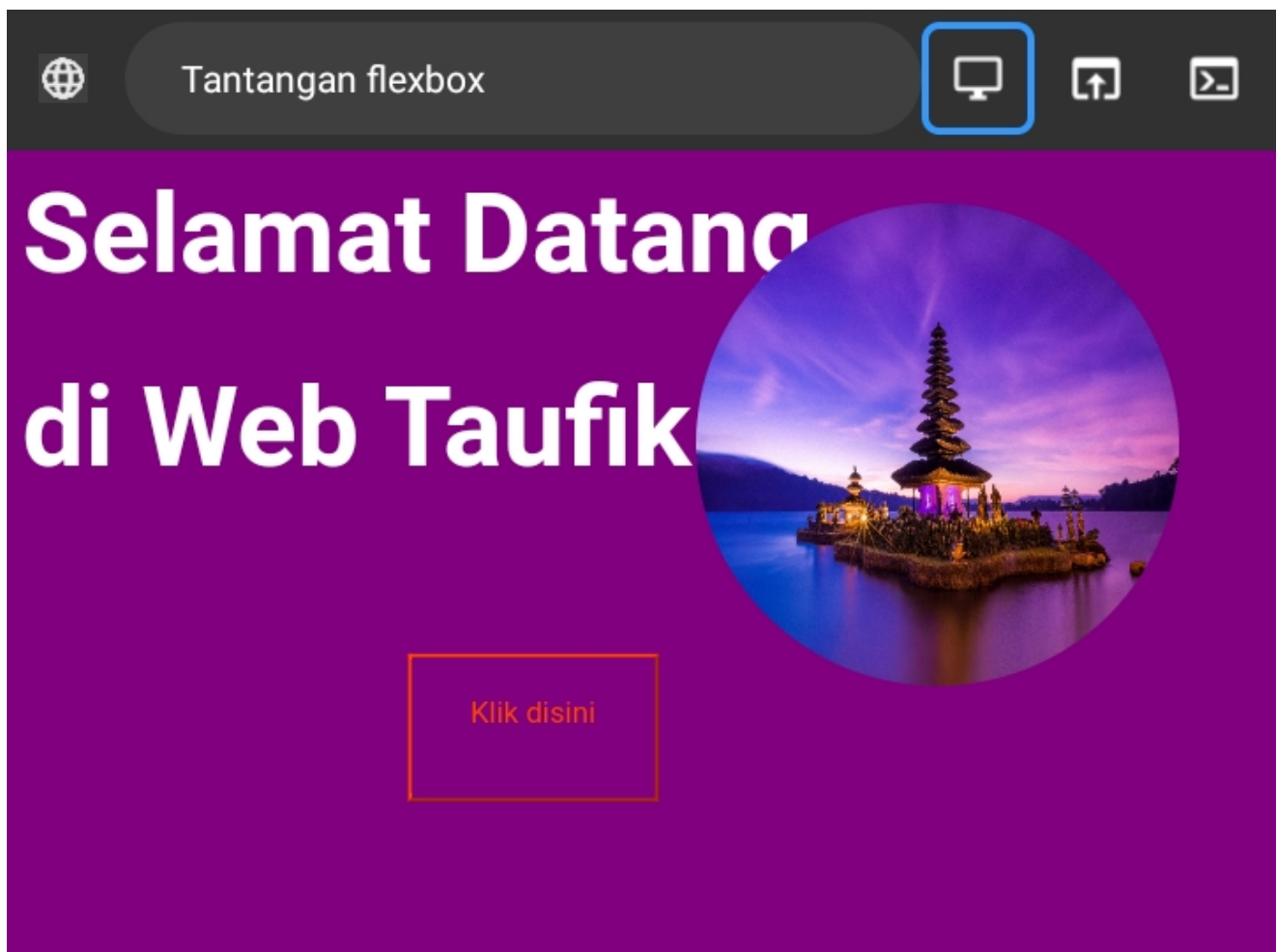
 <h1 class="item box-item-2">
di Web Jordan
 </h1>
 <div class="item box-item-3">
 <button class="J">Klik disini</button>
 </div>
 </div>

 <div class="box-container">
 <div class="item-box">

 </div>
 </div>
 </div>
</body>
</html>

```

hasil



---

## Materi Position

---

### Position Relatif

#### Penjelasan

`display: flex;` : Elemen tersebut menggunakan model kotak fleksibel (flexbox) untuk pengaturan tata letak.

`background-color: blue;` : Latar belakang elemen diwarnai biru.

`width: 95px; height: 95px;` : Lebar dan tinggi elemen ditentukan masing-masing 95 piksel.

`position: relative;` : Elemen diposisikan relatif terhadap posisi normalnya dalam aliran dokumen.

`top: 50px;` : Elemen digeser ke bawah sejauh 50 piksel dari posisi normalnya dalam aliran dokumen.

#### Kode program

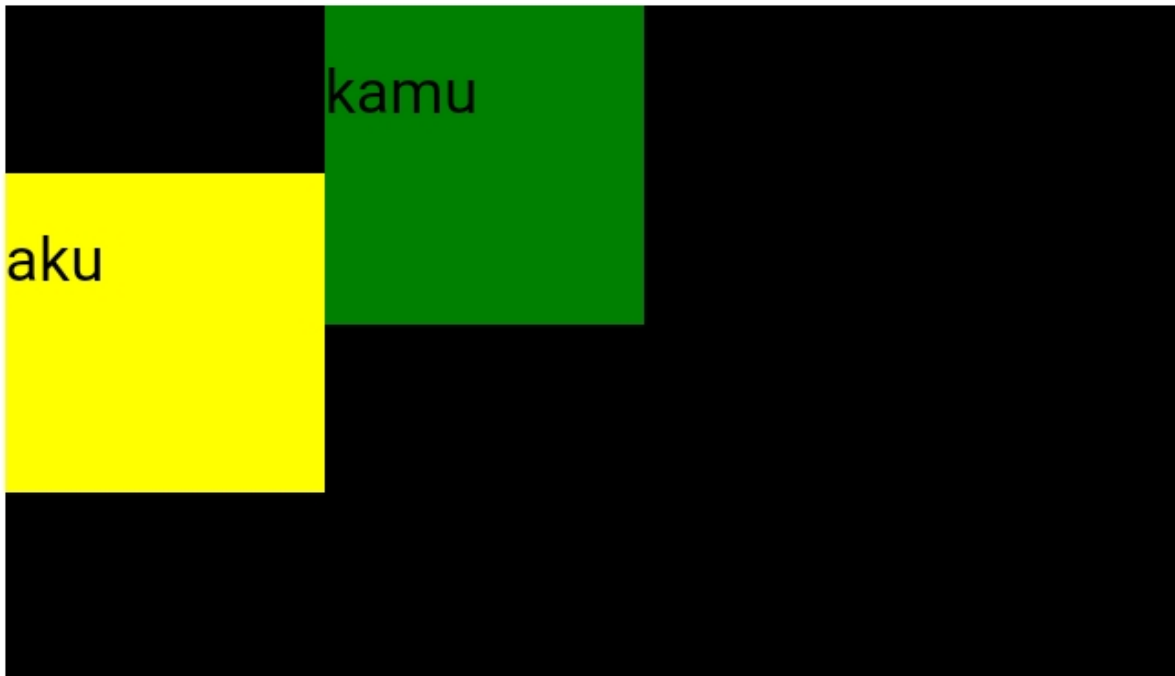
```
<!DOCTYPE html>
<html>
 <head>
 <title>tentang position</title>
 <style>
 .container {
 display: flex;
 flex-direction: row;
 background-color: black;
 height: 200px;
 width: 350px;
 }
 .item-1 {
 display: flex;
 background-color: blue;
 width: 95px;
 height: 95px;
 position: relative;
 top: 50px;
 }
 .item-2 {
 display: flex;
 background-color: red;
 width: 95px;
 height: 95px;
 }
 </style>
 </head>
 <body>
 <div class="container">
 <div class="item-1"><p>saya</p></div>
 <div class="item-2"><p>dia</p></div>

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

**hasil**



tentang position



## Kesimpulan

penggunaan CSS position untuk menggeser posisi elemen-1 ke bawah sejauh 50 piksel dari posisi normalnya. Elemen-2 tetap berada pada posisi normalnya.

## Position Absolute

### Penjelasan

`display: flex;` : Mengatur elemen untuk menggunakan model kotak fleksibel (flexbox), yang memungkinkan pengaturan tata letak dalam satu atau lebih sumbu.

`background-color: blue;` : Memberikan warna latar belakang biru pada elemen.

`width: 95px; height: 95px;` : Menentukan lebar dan tinggi elemen, masing-masing 95 piksel.

`position: absolute;` : Mengatur posisi elemen secara absolut relatif terhadap kontainer induknya. Elemen ini ditempatkan relatif terhadap kontainer terdekat yang memiliki posisi relatif atau absolut.

`top: 50px;` : Menentukan jarak antara elemen dan tepi atas kontainer terdekat yang memiliki posisi relatif atau absolut, sebesar 50 piksel.

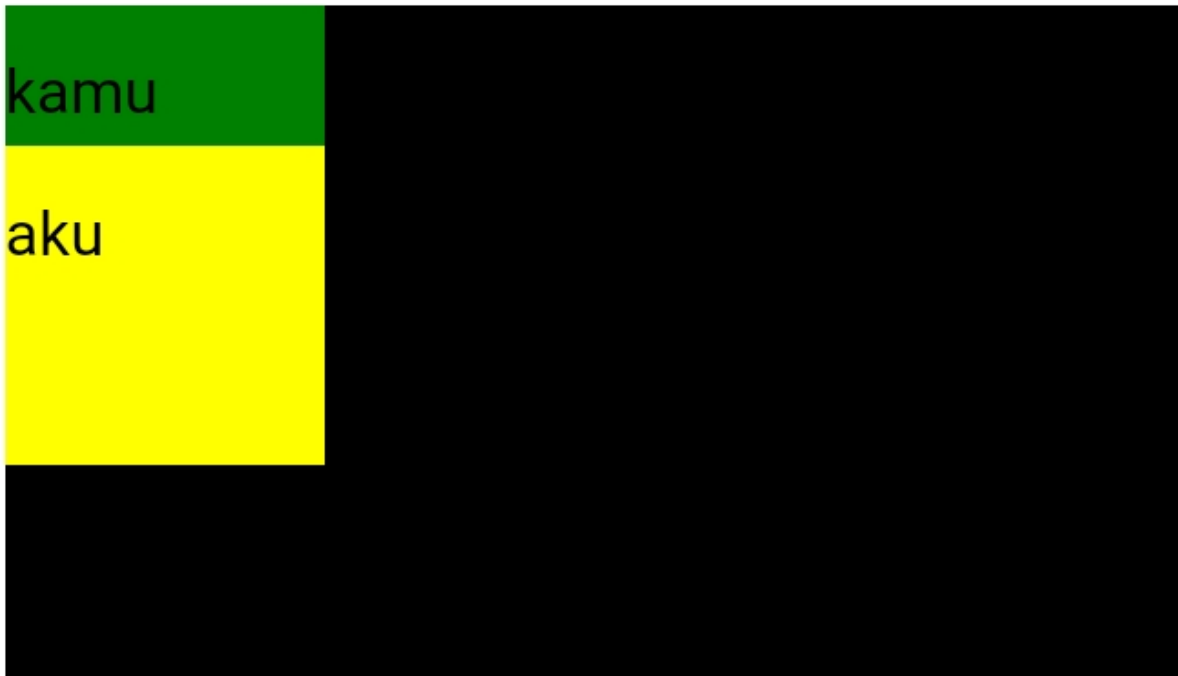


# Kode program

```
<!DOCTYPE html>
<html>
 <head>
 <title>tentang position</title>
 <style>
 .container {
 display: flex;
 flex-direction: row;
 background-color: black;
 height: 200px;
 width: 350px;
 }
 .item-1 {
 display: flex;
 background-color: yellow;
 width: 95px;
 height: 95px;
 position: absolute;
 top: 50px;
 }
 .item-2 {
 display: flex;
 background-color: green;
 width: 95px;
 height: 95px;
 }
 </style>
 </head>
 <body>
 <div class="container">
 <div class="item-1"><p>saya</p></div>
 <div class="item-2"><p>dia</p></div>

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

## hasil



## Kesimpulan

elemen dengan kelas "item-1" akan memiliki tata letak fleksibel, latar belakang biru, lebar dan tinggi masing-masing 95 piksel, dan akan ditempatkan secara absolut dengan jarak 50 piksel dari atas kontainer terdekat yang memiliki posisi relatif atau absolut

---

## Position Fixed

### Penjelasan

- `display: flex;` : Elemen tersebut menggunakan model kotak fleksibel (flexbox) untuk pengaturan tata letak.
- `background-color: blue;` : Latar belakang elemen diwarnai biru.
- `width:95px; height: 95px;` : Lebar dan tinggi elemen ditentukan masing-masing 95 piksel.
- `position: fixed;` : Elemen diposisikan secara tetap dalam tata letak dokumen, berdasarkan viewport.
- `top: 50px;` : Elemen diposisikan sejauh 50 piksel dari atas viewport.

## Kode program

```
<!DOCTYPE html>
<html>
 <head>
 <title>tentang position</title>
 <style>
 .container {
 display: flex;
 flex-direction: row;
 background-color: black;
 height: 200px;
 width: 350px;
 }
 .item-1 {
 display: flex;
 background-color: blue;
 width: 95px;
 height: 95px;
 position: relative;
 top: 50px;
 }
 .item-2 {
 display: flex;
 background-color: red;
 width: 95px;
 height: 95px;
 }
 </style>
 </head>
 <body>
 <div class="container">
 <div class="item-1"><p>saya</p></div>
 <div class="item-2"><p>dia</p></div>

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

**hasil**



## Kesimpulan

elemen dengan kelas "item-1" akan memiliki tata letak fleksibel, latar belakang biru, lebar dan tinggi masing-masing 95 piksel, dan akan diposisikan secara tetap pada posisi 50 piksel dari atas viewport.

## Tantangan Position

### Deskripsi

Halaman web tersebut terstruktur memiliki

1. Terdapat elemen `<html>` yang mengapit seluruh isi halaman.
2. Elemen `<head>` berisi informasi tentang halaman seperti judul dan stylesheet.
3. Elemen `<body>` berisi konten yang akan ditampilkan pada halaman web.
4. Terdapat sebuah container dengan kelas `.container`, yang memiliki tata letak flex dan mengatur penampilan konten di dalamnya.
5. Container memiliki tiga item:
  - .item1: Sebuah gambar dengan latar belakang merah.

- .item2: Sebuah blok teks yang berisi tanggal, judul, dan deskripsi.
  - .item3: Sebuah baris yang berisi teks "Read more" dan sebuah ikon gambar.
6. Terdapat sebuah tombol "like" di bawah kontainer.

## Kode program

```
<!DOCTYPE html>
<html>
<head>
 <title>Tantangan Position</title>
 <style>
 body{
 background-color: blue;
 }
 .container {
 display: flex;
 flex-direction: column;
 background-color: white;
 height: 458px;
 width: 300px;
 border-radius: 10px;
 margin-top: 150px;
 margin-left: 200px;
 }
 .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;
 }
 </style>
</head>
<body>
 <div class="container">
 <div class="item1">

 </div>
 <div class="item2">
 <div class="text1">
 <p>Read more</p>
 </div>
 </div>
 </div>
</body>
</html>
```

```
.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;
}
.item3 {
 display: flex;
 flex-direction: row;
 padding: 2px;
 background-color: grey;
 border-radius: 0px 0px 5px 5px;
 width: 296px;
 justify-content: space-between;
}
.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% ;
}
.item3 img {
 height: 30px;
 width: 30px;
 border-radius: 100% 100% ;
 align-items: center;
 justify-items: center;
 margin-right: 25px;
 margin-top: 10px;
}
</style>
</head>
<body>
```

```

<div class="container">
 <div class="item1">

 </div>
 <div class="item2">
 <p class="text1">Thursday, July 16, 2015</p>
 <h4 class="text2">Lorem ipsum dolor sit amet consectetur.</h4>
 <p class="text3">Lorem ipsum dolor sit amet the, consectetur
adipisicing.</p>
 </div>
 <div class="item3">
 <p class="text4">Read more</p>

 </div>
 <div class="like">
 <button>

 </button>
 </div>
</div>
</body>
</html>

```

## Hasil

16.36

0.0 KB/S VoLTE 4G 89



Tantangan Position



Thursday, July 16, 2015

**Lorem ipsum dolor sit  
amet consectetur.**

Lorem ipsum dolor sit amet the,  
consectetur adipiscing.

Read more





