

Pengenalan CSS

Anatomi CSS

Kode Program

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

Hasil



Welcome CSS

Penjelasan

1. p merupakan selector yang dimana selector adalah sebuah penanda yang digunakan untuk memberikan tanda terhadap tag html yang ingin di modifikasi
2. color merupakan property yang dimana property digunakan untuk menambahkan atau mengatur ukuran teks, jenis font, warna teks, warna background, dan sebagainya
3. red adalah nilai dari property

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. : 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 stylesheets 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.

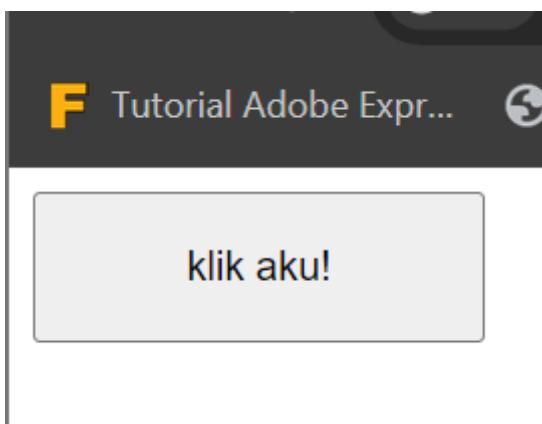
Percobaan Kedua

Kode CSS

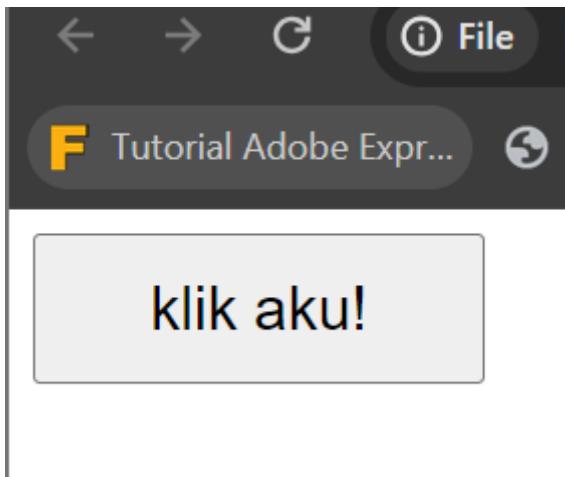
```
button {
    font-size: 20px;
    border:none;
    width: 150px;
    height: 50px;
    font-family:'sans-serif';
}
```

Font-size

Before



After

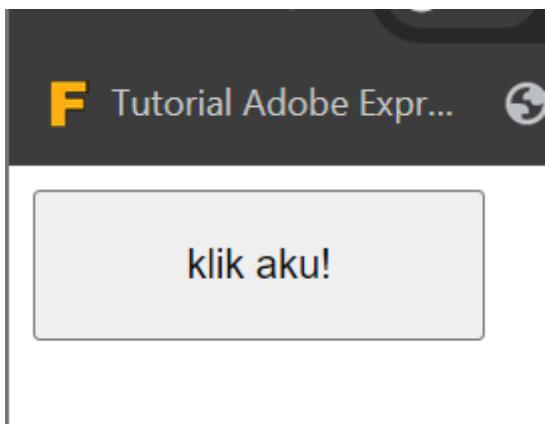


② Apa itu font-size? >

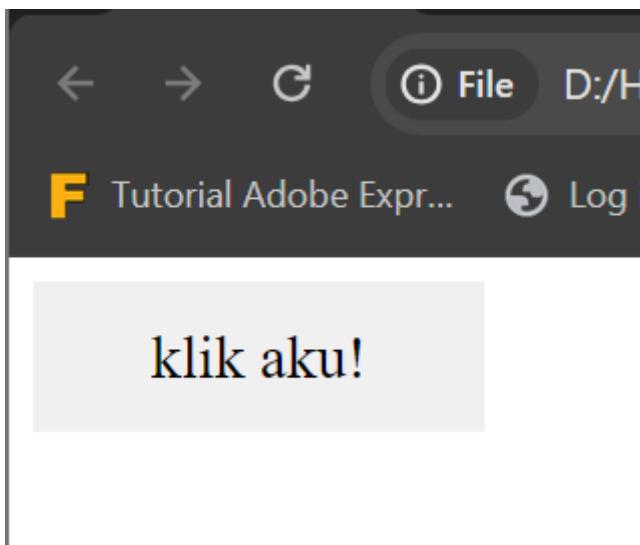
Font-size adalah property CSS yang berfungsi untuk mengatur ukuran font.

Border

Before



After

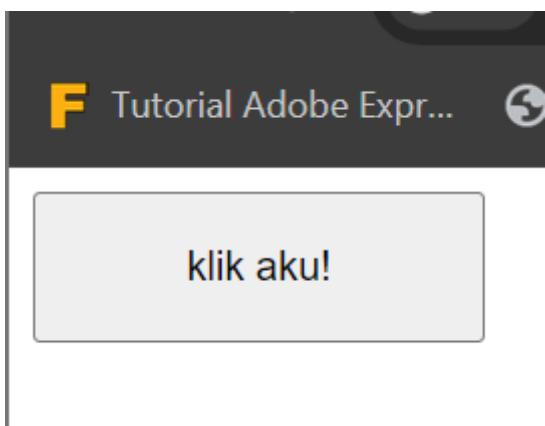


[② Apa itu Border? >](#)

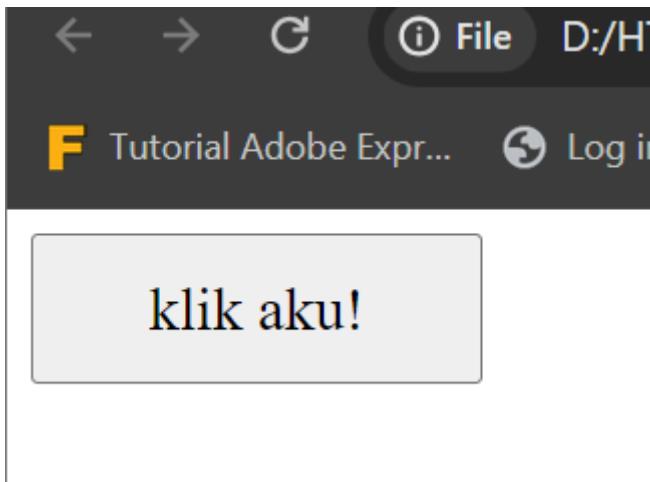
Border merupakan property CSS yang berguna untuk mengatur garis pinggir dari tombol.

Font-family

Before



After



② Apa itu font-family? >

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

Cara pemanggilan CSS

In-line

In-line merupakan salah satu cara untuk memanggil css yaitu dengan cara memanggilnya kedalam baris yang sama dengan tag yang ingin di modifikasi contoh

Kode Program

```
<p style="color: red;">Welcome CSS</p>
```

Hasil

Welcome CSS

Penjelasan

Penjelasan

In-line CSS: Dalam metode ini, CSS ditulis langsung di dalam tag HTML menggunakan atribut style. Ini berguna ketika Anda ingin menerapkan gaya yang spesifik hanya untuk satu elemen.

Internal

Kode Program

```
<html>
  <head>
    <title>Belajar CSS 1</title>
    <style>
      p {
        color: red;
      }
    </style>
  </head>
  <body>
    <p>Welcome CSS</p>
  </body>
</html>
```

Hasil

Welcome CSS

Penjelasan

Penjelasan

Internal CSS: Dalam metode ini, CSS ditulis di dalam tag `<style>` di dalam elemen `<head>` dokumen HTML. Ini berguna ketika Anda memiliki beberapa halaman yang memerlukan gaya yang sama. Dengan menggunakan internal CSS, Anda dapat menetapkan gaya sekali dan akan berlaku untuk seluruh halaman tersebut.

External

Kode Program

Kode Program HTML yang dimana ini adalah tempat css di panggil dengan menggunakan metode external

```
<html>
  <head>
    <title>Belajar CSS2</title>
    <link rel="stylesheet" href="belajar css1.css">
  </head>
  <body>
    <p>Welcom CSS</p>
  </body>
</html>
```

Kode Program CSS

CSS

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

Hasil

Welcome CSS

Penjelasan

 Penjelasan

External CSS: Dalam metode ini, CSS ditulis dalam file terpisah dengan ekstensi .css, dan kemudian dihubungkan dengan dokumen HTML menggunakan tag . Hal ini memisahkan struktur konten (HTML) dari presentasi (CSS), membuatnya lebih mudah untuk memelihara dan memperbarui gaya di seluruh situs web Anda

SELECTOR CSS

Elemen Selector

Penjelasan

`h1` : Ini adalah elemen selector yang menargetkan semua elemen `<h1>` dalam dokumen HTML untuk menerapkan aturan gaya tertentu, seperti warna teks biru dan margin atas sebesar 10000 piksel.

Code

```
<h1>Hallo RPL 1</h1>
```

```
h1 {  
    color: blue;  
}
```

Hasil

Haloo RPL

Class Selector

Penjelasan

- `<!DOCTYPE html>` : Mendefinisikan tipe dokumen sebagai HTML versi 5.
- `<html>` : Menandakan awal dan akhir dari dokumen HTML.

- <head> : Berisi informasi tambahan tentang dokumen HTML, seperti judul dan tautan ke file CSS.
- <title>CSS</title> : Menentukan judul dokumen yang akan ditampilkan di tab browser.
- <link rel="stylesheet" href="Style.CSS"> : Menautkan dokumen HTML dengan file CSS eksternal bernama "Style.CSS", sehingga dokumen HTML dapat menerapkan gaya yang didefinisikan dalam file CSS tersebut.
- <body> : Berisi konten yang akan ditampilkan kepada pengguna.
- <p class="biru"> ini warna biru </p> : Elemen ini hanya memiliki satu kelas CSS, yaitu "biru". Oleh karena itu, elemen tersebut akan menerapkan gaya yang telah ditentukan untuk kelas "biru" dalam file CSS atau dalam bagian gaya internal/inline yang berkaitan dengan kelas tersebut. Dan memberikan warna biru.

Code

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <link rel="stylesheet" href="Style.CSS">
    </head>
    <body>
        <p class="biru">ini warna biru </p>
    </body>
</html>
```

```
.merah {
    color:red;
}
.biru {
    color:blue;
}
.kuning {
    color:yellow;
}
```

Hasil

Ini Warna Merah

Ini Warna Biru

Ini Warna Kuning

ID Selector

Penjelasan

- <!DOCTYPE html> : Mendefinisikan tipe dokumen sebagai HTML versi 5.
- <html> : Menandakan awal dan akhir dari dokumen HTML.
- <head> : Berisi informasi tambahan tentang dokumen HTML, seperti judul dan tautan ke file CSS.
- <title>CSS</title> : Menentukan judul dokumen yang akan ditampilkan di tab browser.
- <link rel="stylesheet" href="Style.CSS"> : Menautkan dokumen HTML dengan file CSS eksternal bernama "Style.CSS", sehingga dokumen HTML dapat menerapkan gaya yang didefinisikan dalam file CSS tersebut.
- <body> : Berisi konten yang akan ditampilkan kepada pengguna.
- Elemen <p id="hijau"> ini warna hijau </p> memiliki atribut id yang menentukan identitas uniknya dalam dokumen HTML. Dengan demikian, fungsi dari atribut id ini adalah untuk memberikan cara yang unik untuk merujuk atau menargetkan elemen tersebut dalam CSS. Dan memberikan warna hijau pada teks.

Code

```
<!DOCTYPE html>
<html>
  <head>
    <title>CSS</title>

    <link rel="stylesheet" href="Style.CSS">

  </head>
  <body>
    <p id="hijau" > ini warna hijau </p>
  </body>
```

```
</html>

#hijau {
    color:green;
}
```

Hasil

Ini Warna Hijau

TEXT

Text-decoration

Overline

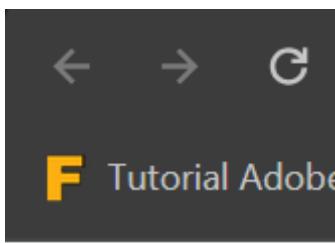
Penjelasan

text-decoration overline merupakan properti CSS yang digunakan untuk memberi garis di atas teks.

Kode program

```
<head>
    <title>Text</title>
    <style>
        .over {
            text-decoration: overline;
        }
    </style>
</head>
<body>
    <p class="over">ALYA</p>
</body>
```

Hasil



ALYA

Kesimpulan

overline merupakan salah satu jenis garis pada teks dekorasi yang berguna untuk memberi garis atas pada teks.

Underline

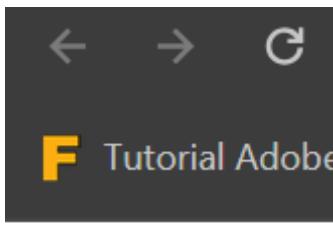
Penjelasan

Text-decoration underline merupakan properti CSS yang memberi garis bawah pada teks.

Kode program

```
<head>
    <title>Text</title>
    <style>
        .under {
            text-decoration: underline;
        }
    </style>
</head>
<body>
    <p class="under">ALYA</p>
</body>
```

Hasil



ALYA

Kesimpulan

underline merupakan salah satu jenis garis yang memberikan garis bawah pada teks.

Line-through

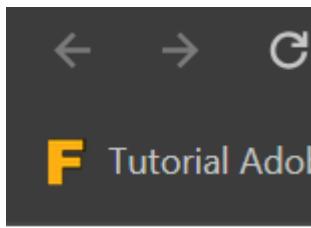
Penjelasan

text-decoration line-through merupakan dekorasi garis di tengah atau teks tercoret.

Kode program

```
<head>
    <title>Text</title>
    <style>
        .through {
            text-decoration: line-through;
        }
    </style>
</head>
<body>
    <p class="through">ALYA</p>
</body>
```

Hasil



~~ALYA~~

Kesimpulan

line-through adalah salah satu jenis garis di CSS yang memberi kesan tercoret pada teks atau garis tengah pada teks.

Text align

PENJELASAN

- text-align:left ; Text akan ditampilkan di sebelah kiri dokumen.
- text-align:right ; Text akan ditampilkan di sebelah kanan dokumen.
- text-align:center ; Text akan ditampilkan di tengah-tengah dokumen.
- text-align:justify ; Text akan ditampilkan dengan mengikuti spasi yang ada di antara kata-kata.

KODE PROGRAM

```
p {  
text-align: center;  
}
```

HASIL

BEFORE

AFTER

ALYA

KESIMPULAN

dari kode program yang diberikan `p { text-align: center; }` adalah bahwa kode ini mengatur properti CSS `text-align` untuk elemen `<p>` (paragraf) dalam sebuah dokumen HTML. Dengan nilai `center`, semua teks dalam elemen `<p>` akan diatur ke tengah dari elemen tersebut.

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: lowercase;  
}
```

HASIL

BEFORE

ALYA

AFTER

alya

KESIMPULAN

Kesimpulan dari text-transform: uppercase; adalah bahwa teks yang diberi gaya dengan properti ini akan diubah menjadi huruf kapital semua.

Text indent

PENJELASAN

Text indent atau indentasi teks adalah proses membuat jarak antara teks dengan margin atas dari sebuah paragraf. Ini dapat membuat teks terlihat lebih rapi dan terstruktur. Teks dapat diatur dengan menggunakan tanda indentasi seperti tab atau spasi.

KODE PROGRAM

```
p {  
text-indent: 100px;  
}
```

HASIL

BEFORE

ALYA

AFTER

KESIMPULAN

Kesimpulan dari penggunaan properti CSS text-indent: 100px; adalah bahwa paragraf (elemen `<p>`) akan memiliki indentasi teks sebesar 100 piksel dari sisi kiri. ini digunakan untuk mengatur jarak atau spasi antara tepi kiri elemen paragraf dan teks di dalamnya.

BACKGROUND

Background-color

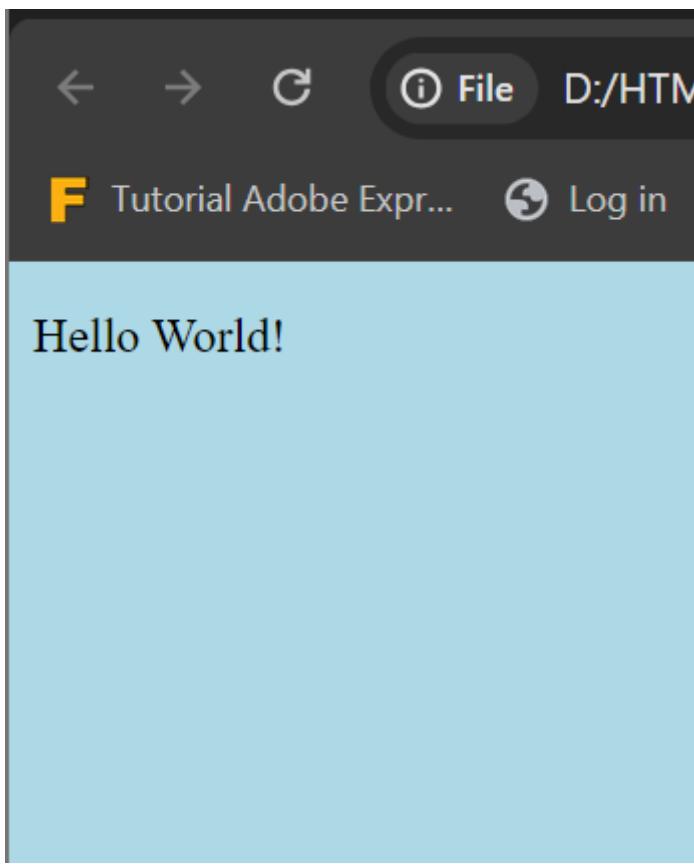
Penjelasan

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

Kode program

```
<head>
<style>
body {
    background-color: lightblue;
}
</style>
</head>
<body>
<p>Hello World!</p>
</body>
```

Hasil



Kesimpulan

merupakan properti CSS yang memberikan warna pada latar belakang web.

Background-image

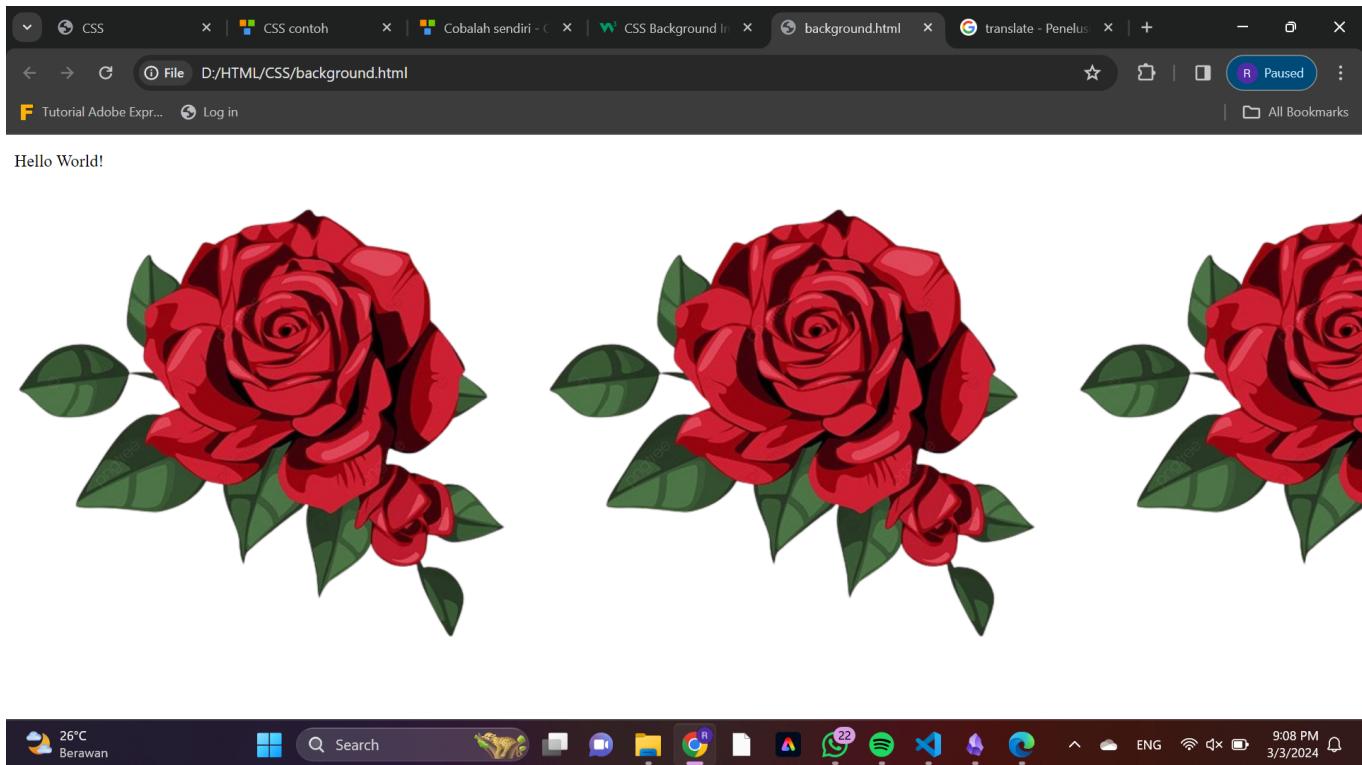
Penjelasan

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("mawar.png");
}
</style>
</head>
<body>
<p>Hello World!</p>
</body>
```

Hasil



Kesimpulan

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

Background-repeat

Penjelasan

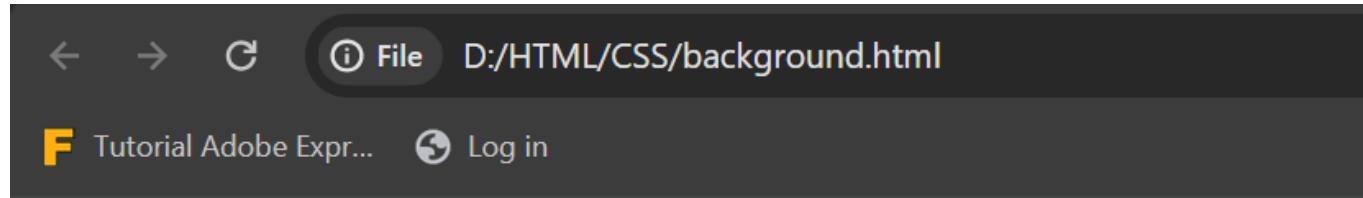
Background-repeat adalah properti yang digunakan untuk mengatur gambar latar belakang yang digunakan apakah ingin di ulang atau tidak.

Kode program

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

```
<p>Hello World!</p>
</body>
```

Hasil



Hello World!



Kesimpulan

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

Background-attachment

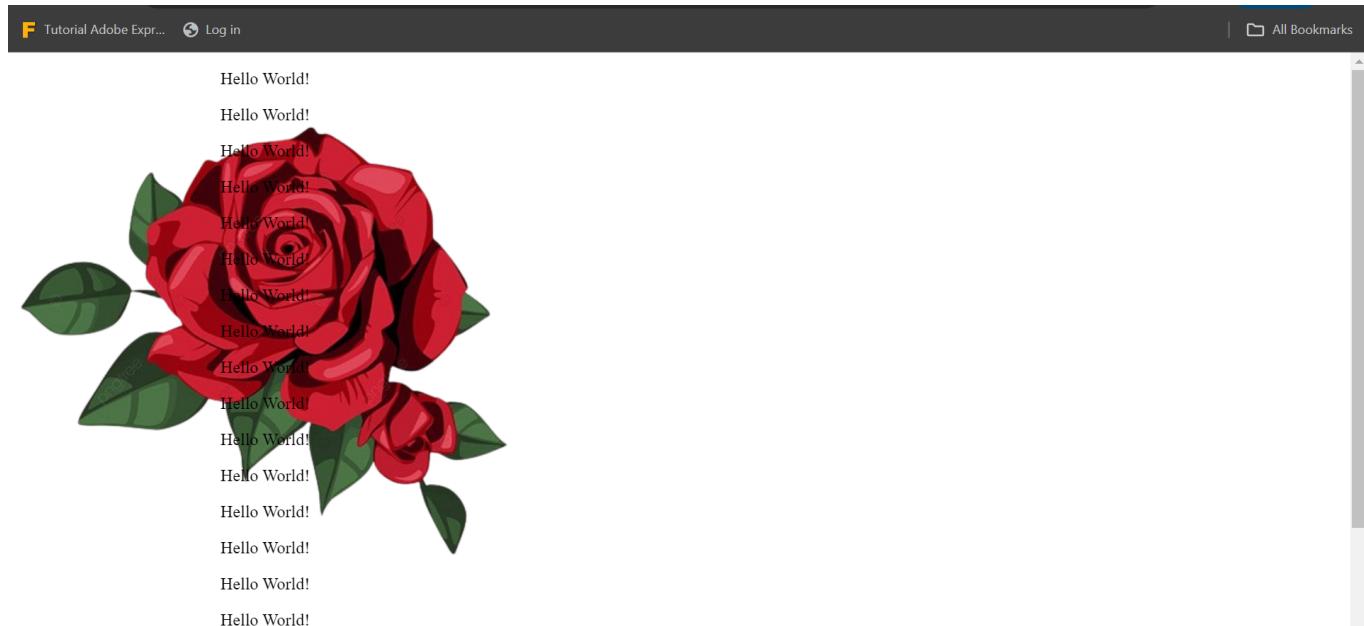
Penjelasan

Background-attachment digunakan untuk mengatur bagaimana sebuah gambar background akan bergerak atau tidak bergerak dalam halaman.

Kode program

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

Hasil





Kesimpulan

Kode CSS di atas, body { background-attachment: fixed; }, menetapkan bahwa gambar latar belakang pada halaman HTML akan memiliki sifat tetap (fixed). Ini berarti gambar latar belakang akan tetap pada posisinya relatif terhadap jendela browser, dan tidak akan bergerak saat halaman digulir.

Jadi, kesimpulan dari kode tersebut adalah bahwa gambar latar belakang pada halaman HTML akan memiliki sifat tetap dan tidak akan bergerak saat halaman digulir.

BOX MODEL

PADDING

Penjelasan

padding adalah sebuah properti CSS yang digunakan untuk menentukan jarak antara konten elemen dan garis batas elemen tersebut. Properti padding dapat diatur pada semua direksi, yaitu atas, bawah, kiri, dan kanan. Untuk mengatur padding pada direksi tertentu, gunakan properti padding-top, padding-bottom, padding-left, dan padding-right.

1. `padding-top` : Properti ini digunakan untuk menentukan jarak antara konten elemen dan garis atas elemen tersebut.
2. `padding-bottom` : Properti ini digunakan untuk menentukan jarak antara konten elemen dan garis bawah elemen tersebut.
3. `padding-left` : Properti ini digunakan untuk menentukan jarak antara konten elemen dan garis kiri elemen tersebut.
4. `padding-right` : Properti ini digunakan untuk menentukan jarak antara konten elemen dan garis kanan elemen tersebut.

Kode Program

```
.button1 {  
background-color: pink;  
width: 100px;  
height: 50px;  
border: 5px solid palevioletred;  
border-radius: 10px 15px;  
padding-left: 0px;  
padding-bottom: 50px;  
padding-right: 50px;  
padding-top: 50px;
```

Hasil



Kesimpulan

Properti padding adalah properti CSS yang digunakan untuk menentukan jarak antara konten elemen dan garis batas elemen tersebut.

MARGIN

Penjelasan

margin adalah properti CSS yang digunakan untuk menentukan jarak antara elemen dan elemen lainnya atau antara elemen dan area konten halaman. Properti margin dapat diatur pada semua direksi, yaitu atas, bawah, kiri, dan kanan. Untuk mengatur padding pada direksi tertentu, gunakan properti margin-top, margin-bottom, margin-left, dan margin-right.

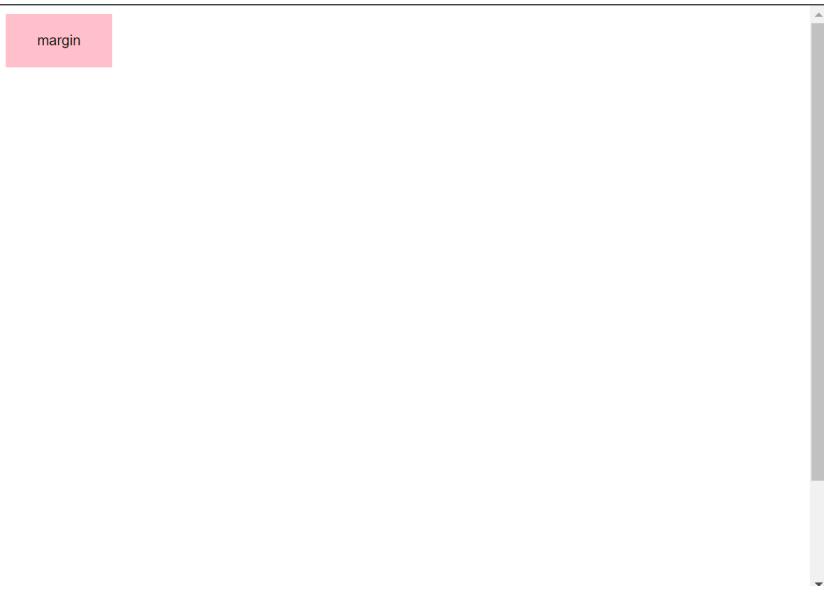
1. margin-top : menetapkan margin atas elemen, yang merupakan spasi di atas elemen.

2. `margin-bottom`: menetapkan margin bawah elemen, yang merupakan spasi di bawah elemen.
3. `margin-left`: mengatur margin kiri elemen, yang merupakan spasi di sebelah kiri elemen.
4. `margin-right`: menetapkan margin kanan elemen, yang merupakan spasi di sebelah kanan elemen.

Kode Program

```
.button2 {  
background-color: pink;  
width: 100px;  
height: 50px;  
border: 0px;  
margin-top: 600px;  
margin-bottom: 600px;  
margin-left: 500px;  
margin-right: 100px;  
}
```

Hasil



margin

Kesimpulan

margin adalah properti singkatan untuk mengatur margin elemen di empat arah atas, kanan, bawah, dan kiri.

BORDER

Penjelasan

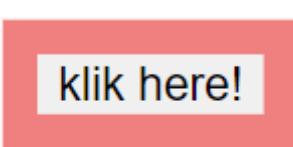
Properti dalam CSS digunakan untuk mengatur batas elemen.

1. `border-width` : Properti ini digunakan untuk mengatur lebar perbatasan, dapat diatur ke nilai panjang, seperti piksel
2. `border-style` : Properti ini digunakan untuk mengatur gaya perbatasan, dapat diatur ke nilai seperti solid, putus-putus, garis lurus, atau tidak ada.
3. `border-color` : Properti ini digunakan untuk mengatur warna perbatasan, dapat diatur ke nama warna, kode warna heksadesimal.

Kode Program

```
button {  
    border-color: lightcoral;  
    border-width: 10px;  
    border-style: solid;  
}
```

Hasil



klik here!

Kesimpulan

1. `border-width` mengatur lebar garis. Contoh nilai yang dapat digunakan adalah `thin`, `medium`, `thick`, atau nilai numerik dalam piksel (`px`), sentimeter (`cm`), atau unit lainnya.
2. `border-style` mengatur gaya garis. Beberapa nilai yang umum digunakan adalah `solid` (garis solid), `dashed` (garis putus-putus), `dotted` (garis titik-titik), `double` (garis ganda), dll.
3. `border-color` mengatur warna garis. Nilai yang dapat digunakan adalah nama warna seperti `red`, `blue`, `green`, kode hex seperti `#FF0000`, `#00FF00`, `#0000FF`, atau nilai lain seperti `rgb(255, 0, 0)`.

Latihan box model

Penjelasan

1. `p { ... }`: Ini adalah pemilih elemen yang mengatur tata letak dan gaya teks pada halaman web. Dalam kode ini, properti dan nilai yang digunakan adalah :
 - `font-size: 75px;` -> Mengatur ukuran font menjadi 75 piksel.
 - `font-family: 'arial' ;` -> Mengatur jenis font menjadi Arial.
 - `margin-top: 150px ;` -> Mengatur jarak antara teks dan elemen di atasnya menjadi 150 piksel.
 - `margin-bottom: 100px ;` -> Mengatur jarak antara teks dan elemen di bawahnya menjadi 100 piksel.
 - `margin-left: 50px;` -> Mengatur jarak antara teks dan elemen di sebelah kiri menjadi 50 piksel.
 - `margin-right: 100px;` -> Mengatur jarak antara teks dan elemen di sebelah kanan menjadi 100 piksel.
 - `color: aliceblue;` -> Mengatur warna teks menjadi aliceblue.
2. `img { ... }`: Ini adalah pemilih elemen gambar pada halaman web. Dalam kode ini, properti dan nilai yang digunakan adalah:
 - `margin-right: 200px;` -> Mengatur jarak antara gambar dan elemen di sebelah kanan menjadi 200 piksel.
 - `margin-top: -30px;` -> Mengatur jarak antara gambar dan elemen di atasnya menjadi -30 piksel (agar gambar terlihat lebih bawah).
 - `border: 10px solid white;` -> Mengatur border gambar menjadi 10 piksel dengan warna putih.
 - `border-radius: 1500px 1500px;` -> Mengatur sudut pada border gambar menjadi bulat.
3. `button { ... }` : Ini adalah pemilih elemen tombol pada halaman web. Dalam kode ini, properti dan nilai yang digunakan adalah:
 - `background-color: purple;` -> Mengatur warna latar belakang tombol menjadi ungu.
 - `width: 150px;` -> Mengatur lebar tombol menjadi 150 piksel.
 - `height: 50px;` -> Mengatur tinggi tombol menjadi 50 piksel.
 - `border-width: 2px;` -> Mengatur lebar border tombol menjadi 2 piksel.
 - `color: orange;` -> Mengatur warna teks tombol menjadi orange.
 - `border-color: orange;` -> Mengatur warna border tombol menjadi orange.
 - `margin-bottom: 20px;` -> Mengatur jarak antara tombol dan elemen di bawahnya menjadi 20 piksel.

- margin-left: 390px; -> Mengatur jarak antara tombol dan elemen di sebelah kiri menjadi 390 piksel

Kode Program

```
<!DOCTYPE html>
<html>
<head>
    <title>Box Model</title>
    <link rel="stylesheet" href="tantangan_boxmodel.css">
</head>
<body>
    <span>
        
        <p> Selamat Datang <br><b>
            di web Alya !</b>
        <p>
    </span>
    <button> klik saya </button>
</body>
</html>
```

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

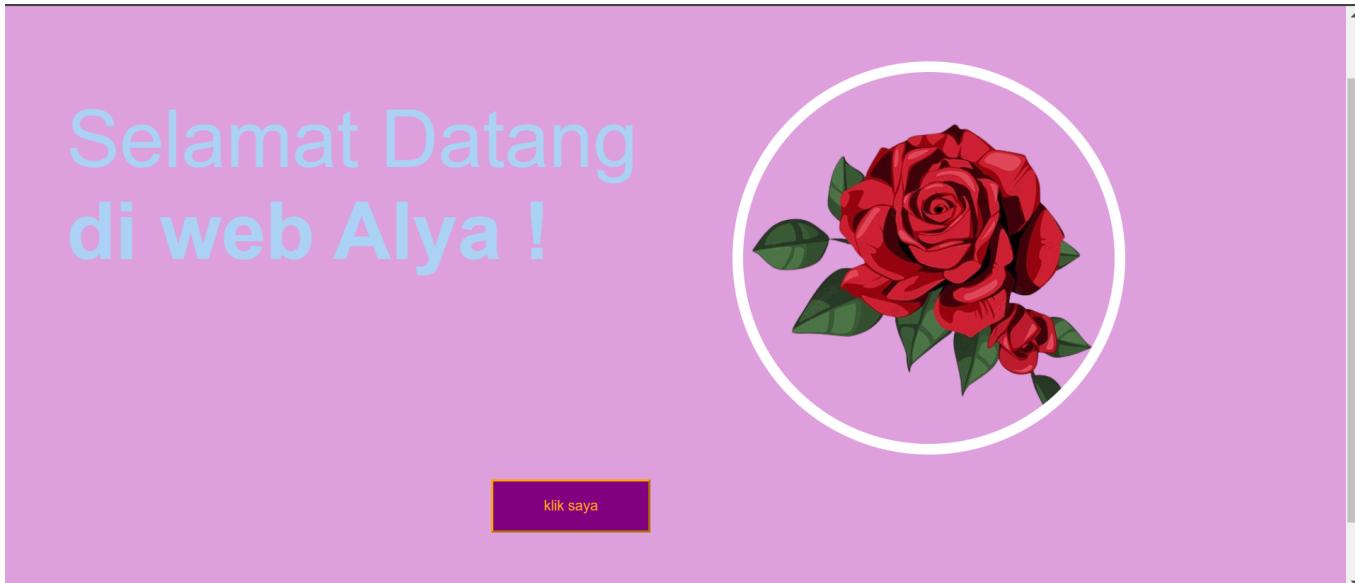
p {
    font-size: 75px;
    font-family: 'arial';
    margin-top: 150px;
    margin-bottom: 100px;
    margin-left: 50px;
    margin-right: 100px;
    color: rgb(171, 210, 244);
}

img {
    margin-right: 200px;
    margin-top: -30px;
    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: 400px;  
}
```

Hasil



PSEUDO-CLASS

Penjelasan

- `button:hover` : adalah pseudo-class selector yang menargetkan tombol (button) ketika pengguna mengarahkan kursor ke atasnya (hover) atau diklik.
- `background-color: red;` : Properti ini menetapkan warna latar belakang tombol menjadi merah ketika tombol sedang di-hover.
- `color: white;` : Properti ini menetapkan warna teks pada tombol menjadi putih ketika tombol di-hover.
- `height: 100px;` : Properti yang digunakan untuk membuat tinggi (height) tombol menjadi 100 piksel ketika tombol di-hover.
- `width: 100px;` : Properti yang digunakan untuk membuat lebar (width) tombol menjadi 100 piksel ketika tombol di-hover.
- `button:active` : adalah pseudo-class selector yang menargetkan tombol (button) ketika tombol tersebut sedang dalam keadaan "active", atau tombol sedang ditekan.
- `color: yellow;` : Properti memberikan warna kuning pada teks ketika tombol sedang active atau tombol sedang ditekan.

Kode Program

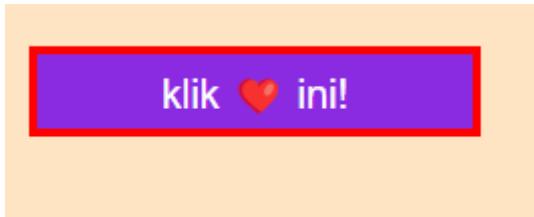
```
<!DOCTYPE html>

<html>

<head>
    <title>Latihan</title>
    <link rel="stylesheet" href="latihan.css">
</head>
<body>
    <p class="text">Klik Untuk Mendapatkannya!</p>
    <button class="button">klik ❤ ini!</button>
</body>
</html>
```

```
button:hover {
    background-color: blueviolet;
    color: white;
    width: 150px;
    transition: all 0.3s ease-in;
}
button:active {
    transform: scale(0.5);
}
```

Hasil



Kesimpulan

Setiap property memberikan perubahan yang berbeda, dan hover akan memberikan perubahan saat di klik, width untuk mengatur lebar, background-color purple memberikan warna ungu ketika diklik, dan color white digunakan untuk memberikan warna putih pada teks. Teks klik akan berganti warna menjadi putih jika ditekan dan akan bertambah lebar.

TRANSITION

Penjelasan

- **Transition** : untuk mengatur pertunjukan perubahan, mengubah penampilan elemen dari satu keadaan ke keadaan lain, memberikan dinamika visual yang menarik dalam desain web Anda.
- **Transition-delay** : properti ini memberikan momen kejutan atau antisipasi sebelum perubahan yang dijanjikan terjadi, menambah dramatisasi pada tampilan elemen.
- **Transition-duration** : menentukan lamanya waktu ketika elemen berubah, memungkinkan Anda untuk mengatur seberapa lama penonton (pengguna) menikmati perubahan tersebut.
- **Transition-property** : menentukan apa yang akan berubah, seperti warna, ukuran, atau posisi, menciptakan efek transisi yang beragam dan menarik.
- **Transition-timing-function** : properti ini mengatur ritme perubahan, menentukan apakah transisi akan bergerak dengan lembut dan halus seperti aliran air, atau tiba-tiba dan dramatis seperti petir yang menyambar.

Kode program

```
button:hover{  
    transition:all 0.9s ease-in;  
}
```

Hasil

Klik saya!

Klik saya!

Kesimpulan

Properti transition digunakan dalam CSS untuk mengatur perubahan transisi atau animasi pada elemen saat terjadi perubahan pada properti-properti CSS lainnya, seperti perubahan warna, posisi, ukuran, atau opacity.

TRANSFORM

Penjelasan

- `transform: scale(0.5);` = Mengubah ukuran elemen menjadi setengah dari ukuran aslinya.
- `transform: scalex(0.5);` = Hanya mengubah skala elemen secara horizontal menjadi setengah dari ukuran aslinya.
- `transform: rotate(45deg);` = Memutar elemen sebesar 45 derajat searah jarum jam.
- `transform: skewX(-25deg);` = Membengkokkan elemen secara horizontal sebesar -25 derajat (ke arah kiri).
- `transform: skew(25deg, 5deg);` = Membengkokkan elemen sebesar 25 derajat secara horizontal (ke kanan) dan 5 derajat secara vertikal (ke atas).
- `transform: translate(50px, 52px);` = Menggeser elemen sebesar 50 piksel ke kanan dan 52 piksel ke bawah.
- `transform: matrix(0.7, -0.5, 0.5, 0.4, 0.5, 0.7);` = Menggambarkan transformasi menggunakan matriks 2D den

Kode Program

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

Hasil

Before



After



klik ❤ ini!

Kesimpulan

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

FLEXBOX

Flex Container

Display Flex

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

Kode Program

```
.container {  
    display: flex;  
}
```

Kesimpulan

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

Flex-Direction

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

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

Kode Program

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

Kesimpulan

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

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

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

Align-Items

- `align-items: center;` : Flex items akan diatur di tengah-tengah sumbu silang dari flex container.
- `align-items: flex-start;` : Flex items akan diatur pada awal sumbu silang dari flex container.
- `align-items: flex-end;` : Flex items akan diatur pada akhir sumbu silang dari flex container.
- `align-items: baseline;` : Flex items akan diatur sedemikian rupa sehingga garis dasar dari teks pada masing-masing item berada pada level yang sama.
- `align-items: stretch;` : Flex items akan diperpanjang untuk mencapai tinggi maksimal flex container, mengisi ruang kosong di sepanjang sumbu silang.

Kode Program

```
.container{
  align-items:flex-end;
}
```

Kesimpulan

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

Contoh `center` untuk menempatkan flex items di tengah-tengah sumbu silang. `flex-start`

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

Justify-Content

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

Kode Program

```
.container{  
  justify-content: space-between;  
}
```

Kesimpulan

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

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

Flex-item

Penjelasan HTML

- Tag `<DOCTYPE html>` memberitahu web bahwa dokumen HTML adalah versi 5
- Tag pembuka `<html>` menandai awal sebuah dokumen HTML sampai dengan tag penutup `</html>`
- Tag pembuka `<head>` berisi informasi tentang halaman HTML sampai dengan tag penutup `</head>`, biasanya dalam tag head terdapat tag `<title>` untuk memberikan informasi judul halaman HTML
- Apapun tag yang berada di antara tag pembuka `<body>` sampai dengan tag penutup `</body>` akan tampil di web browser.
- `<title>` : Menentukan judul halaman.
- `<link rel="stylesheet" href="Tugassbox.css">` : Menghubungkan dokumen dengan stylesheet eksternal.
- `<body class="main-container">` : Memulai konten utama halaman dengan kelas "main-container".
- `<div class="hero-container">` : Kontainer utama untuk semua konten.
- `<div class="box-container">` : Kontainer untuk setiap kotak konten.
- `<h1 class="item box-item-1">Selamat Datang</h1>` : Teks "Selamat Datang" dengan kelas "item" dan "box-item-1".
- `<h1 class="item box-item-2">di Web Fadhil</h1>` : Teks "di Web Fadhil" dengan kelas "item" dan "box-item-2", kata "Web Fadhil" ditebalkan.
- `<div class="item box-item-3">` : Kontainer untuk tombol "Klik disini" dengan kelas "item" dan "box-item-3".
- `<button class="F">Klik disini</button>` : Tombol dengan teks "Klik disini" dan kelas "F".
- `<div class="item-box">` : Kontainer untuk gambar.
- `` : Gambar "F.png" dari direktori "camera".

Penjelasan CSS

- `body` : Mengatur latar belakang seluruh halaman menjadi warna ungu (purple).
- `.main-container` : Memberikan latar belakang warna ungu (purple) pada elemen dengan kelas "main-container". Ini mungkin digunakan sebagai wadah utama untuk konten.
- `.box-item-1` dan `.box-item-2` : Mengatur tampilan teks pada elemen-elemen dengan kelas "box-item-1" dan "box-item-2". Ini termasuk mengatur ukuran font menjadi 50px, warna putih, dan memberikan jarak atas sejauh 10px.
- `.F:hover` : Mengatur tampilan tombol dengan kelas "F" saat kursor berada di atasnya (hover). Saat tombol di-hover, ukurannya diperbesar, warna dan border berubah, dan posisinya digeser ke kiri dengan transisi animasi selama 0.9 detik.
- `button:active` : Memberikan efek transformasi rotasi sebesar 45 derajat pada tombol ketika tombol sedang aktif (ditekan).

- `img` : Mengatur tampilan gambar dengan lebar dan tinggi masing-masing 250px, dengan sudut melingkar sebesar 200px, serta menetapkan margin kiri sejauh 400px dan margin atas sejauh -400px. Hal ini dapat mempengaruhi posisi dan tata letak gambar dalam halaman.

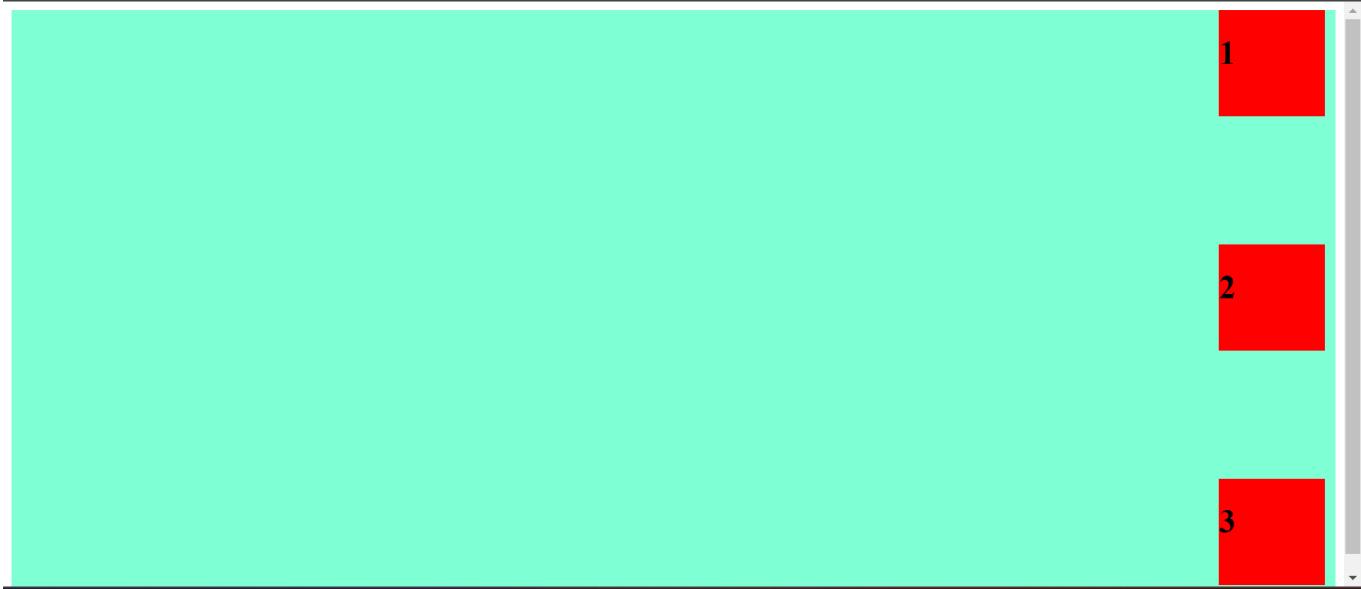
Kode Program

```
<!DOCTYPE html>
<html>
<head>
    <title>Flexbox</title>
    <link rel="stylesheet" href="flexbox.css">
</head>
<body>
    <div class="container">
        <div class="item"><h1>1</h1></div>
        <div class="item"><h1>2</h1></div>
        <div class="item"><h1>3</h1></div>
    </body>
</html>
```

```
.container {
    display: flex;
    flex-direction: column;
    height: 100vh;
    align-items: flex-end;
    justify-content: space-between;
    background-color: aquamarine;
}

/* row ke samping
column ke bawah*/
.item {
    width: 100px;
    margin-right: 10px;
    margin-bottom: 10px;
    height: 100px;
    background-color: red;
}
```

Hasil



POSITION

Position Relative

Penjelasan

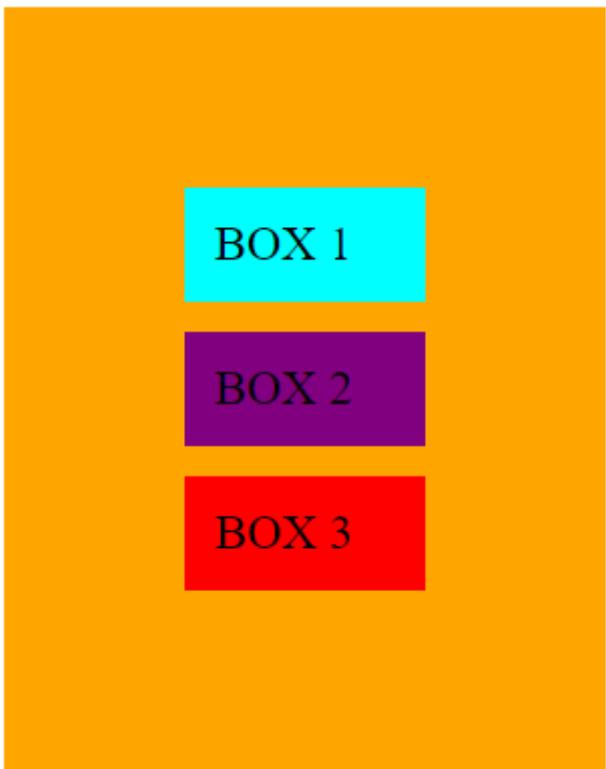
- `background-color: aqua;` : Memberikan warna latar belakang elemen menjadi aqua.
- `position: relative;` : Menetapkan posisi elemen relatif terhadap posisi defaultnya.
- `top: 20px;` : Menggeser elemen 20 piksel ke bawah dari posisi defaultnya.
- `left: 10px;` : Menggeser elemen 30 piksel ke kanan dari posisi defaultnya.

Kode Program

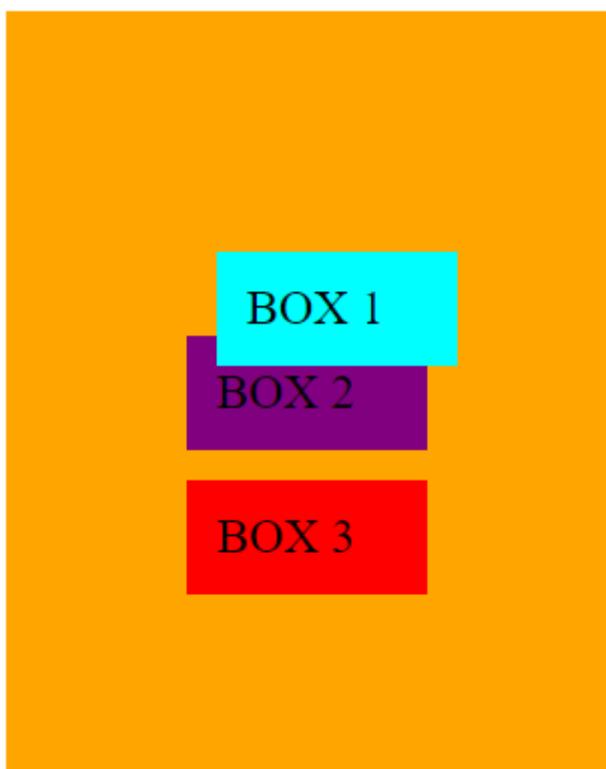
```
.box-1{  
    background-color: aqua;  
    position: relative;  
    left: 10px;  
    top: 20px;  
}
```

Hasil

BEFORE



AFTER



Kesimpulan

elemen dengan kelas "box-1" mengatur latar belakang menjadi warna aqua dan menggeser elemen 20 piksel ke bawah dan 10 piksel ke kanan dari posisi defaultnya.

Position Absolute

Penjelasan

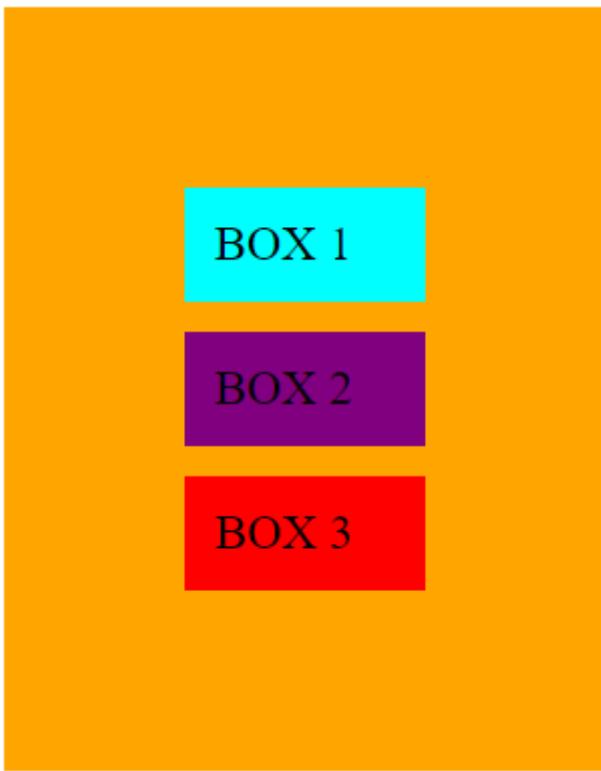
- `background-color: purple;` : Mengatur latar belakang elemen menjadi warna ungu (purple).
- `position: absolute;` : Menetapkan elemen ke dalam konteks posisi yang dihitung relatif terhadap elemen induk yang memiliki posisi non-static.
- `top: 15px;` : . elemen 15 piksel dari atas relatif terhadap posisi elemen induk yang ditetapkan dalam konteks posisi absolut.
- `left: 150px;` : Menggeser elemen 10 piksel dari kiri relatif terhadap posisi elemen induk yang ditetapkan dalam konteks posisi absolut.

Kode Program

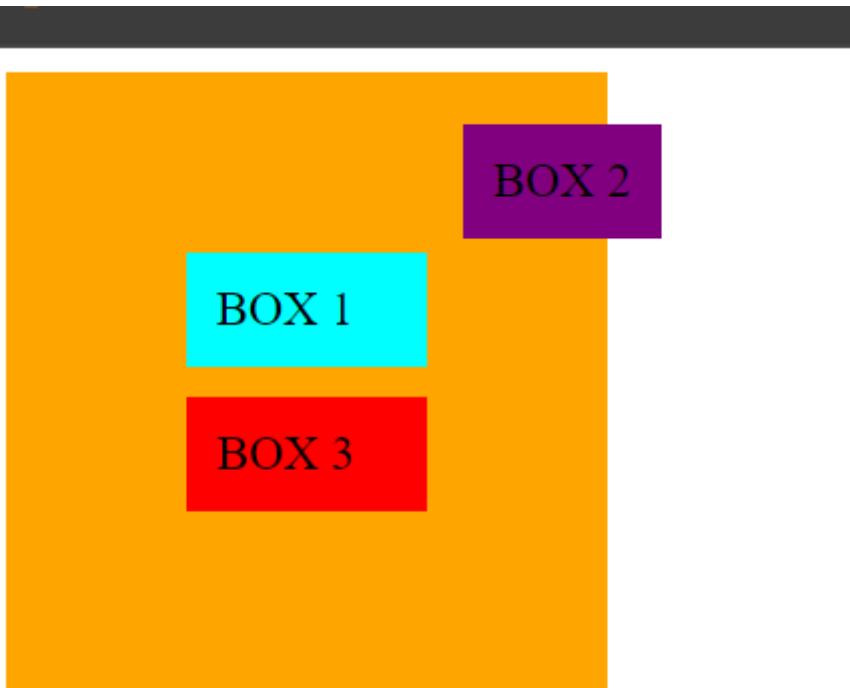
```
.box-2{  
    background-color: purple;  
    position: absolute;  
    top: 15px;  
    left: 150px;  
}
```

Hasil

BEFORE



AFTER



Kesimpulan

kelas "box-2" menetapkan latar belakangnya menjadi warna ungu (purple) dan mengatur posisinya secara absolut,dimana elemen digeser 15 piksel dari atas dan 150 piksel dari kiri.

Position Fixed

Penjelasan

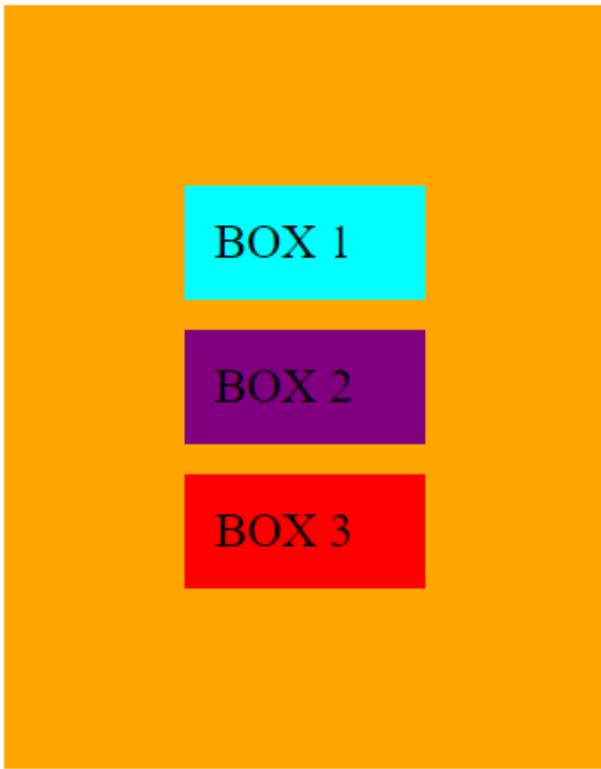
- `top: 20px;` : Menggeser elemen 50 piksel ke bawah dari posisi defaultnya.
- `left: 10px;` : Menggeser elemen 55 piksel ke kanan dari posisi defaultnya.
- `background-color: aqua;` : Mengatur latar belakang elemen menjadi warna aqua.
- `position: fixed;` : Mengatur elemen dalam posisi tetap, yang berarti elemen akan tetap berada pada lokasi yang sama saat kita scroll.

Kode Program

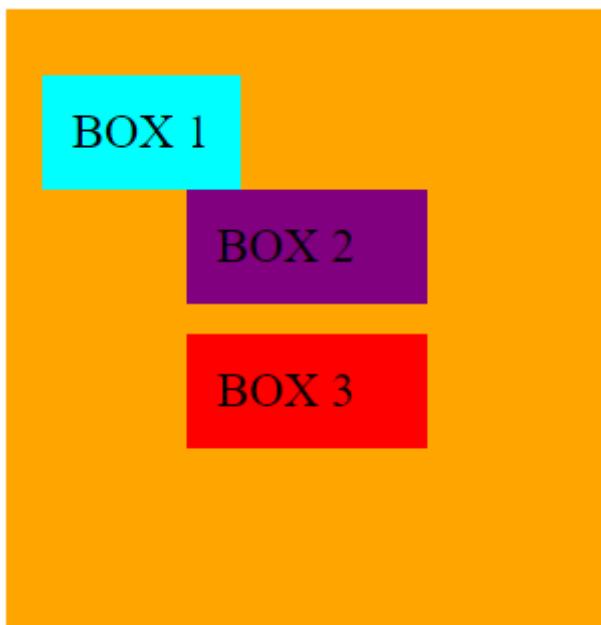
```
.box-1{  
    background-color: aqua;  
    position: fixed;  
    left: 10px;  
    top: 20px;  
}
```

Hasil

BEFORE



AFTER



Kesimpulan

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

Position Sticky

Penjelasan

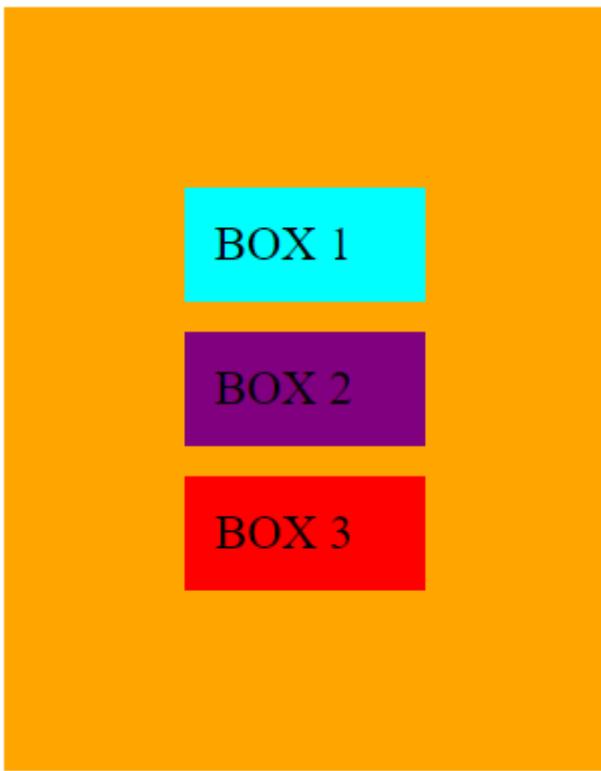
- `background-color: aqua;` : Mengatur latar belakang elemen menjadi warna aqua.
- `Position: sticky;` : Mengatur elemen dalam posisi "sticky", yang berarti elemen akan berperilaku seperti posisi relatif sampai mencapai posisi tertentu saat scroll kemudian akan tetap berada di tempat.elemen akan "melekat" pada posisi yang telah ditentukan saat jarak scroll mencapai nilai tertentu.
- `top: 20px;` : Menggeser elemen sejauh 50 piksel dari atas elemen induk yang memiliki posisi tetap.
- `left: 10px;` : Menggeser elemen sejauh 55 piksel dari sisi kiri elemen induk yang memiliki posisi tetap.

Kode Program

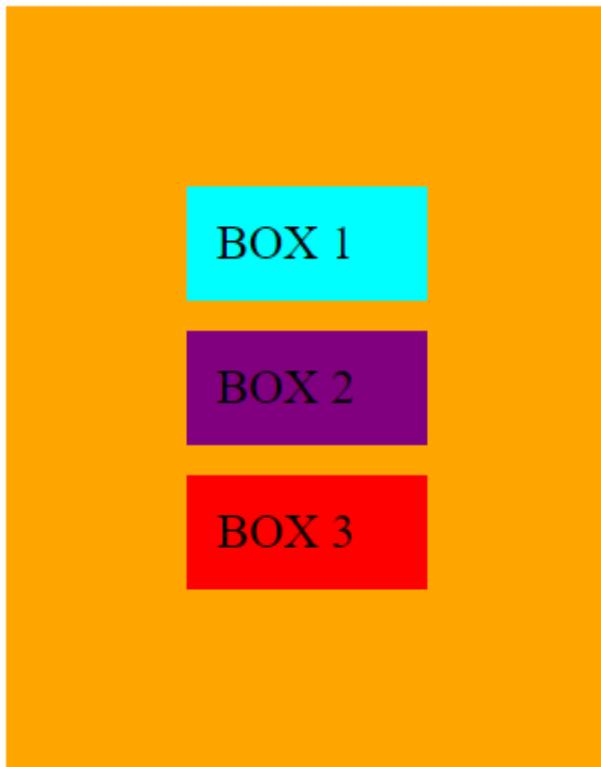
```
.box-1{  
    background-color: aqua;  
    position: sticky;  
    left: 10px;  
    top: 20px;  
}
```

Hasil

BEFORE



AFTER



Kesimpulan

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

Tantangan Box-model

Penjelasan

1. body { ... }

- background-color: plum; : Mengatur warna latar belakang halaman web menjadi warna plum.

2. p { ... }

- font-size: 75px; : Mengatur ukuran font untuk paragraf (<p>) menjadi 75px.
- font-family: 'arial'; : Mengatur jenis font untuk paragraf menjadi "arial".
- margin-top: 150px; : Mengatur margin atas paragraf menjadi 150px.
- margin-bottom: 100px; : Mengatur margin bawah paragraf menjadi 100px.
- margin-left: 50px; : Mengatur margin kiri paragraf menjadi 50px.
- margin-right: 100px; : Mengatur margin kanan paragraf menjadi 100px.
- color: rgb(171, 210, 244); : Mengatur warna font paragraf menjadi biru muda (rgb(171, 210, 244)).

3. img { ... }

- margin-right: 200px; : Mengatur margin kanan gambar () menjadi 200px.
- margin-top: -30px; : Mengatur margin atas gambar menjadi -30px (negatif, artinya margin ini akan "menarik" gambar ke atas).
- border: 10px solid white; : Mengatur border gambar menjadi 10px tebal, berwarna putih, dan dengan jenis garis solid.
- border-radius: 1500px 1500px; : Mengatur bentuk border gambar menjadi bulat dengan radius 1500px (menghasilkan bentuk lingkaran).

4. button { ... }

- background-color: purple; : Mengatur warna latar belakang tombol menjadi ungu (purple).
- width: 150px; : Mengatur lebar tombol menjadi 150px.
- height: 50px; : Mengatur tinggi tombol menjadi 50px.
- border-width: 2px; : Mengatur lebar border tombol menjadi 2px.
- color: orange; : Mengatur warna font tombol menjadi orange.
- border-color: orange; : Mengatur warna border tombol menjadi orange.

- `margin-bottom: 20px;` : Mengatur margin bawah tombol menjadi 20px.
 - `margin-left: 400px;` : Mengatur margin kiri tombol menjadi 400px.
5. `button:hover { ... }`
- `background-color: blueviolet;` : Mengatur warna latar belakang tombol menjadi ungu kebiruan (`blueviolet`) ketika mouse di atas tombol.

Kode Program

```
<!DOCTYPE html>
<html>
<head>
    <title>Box Model</title>
    <link rel="stylesheet" href="tantangan_boxmodel.css">
</head>
<body>
    <span>
        
        <p> Selamat Datang <br><b>
            di web Alya !</b>
        <p>
    </span>
    <button> klik saya </button>
</body>
</html>
```

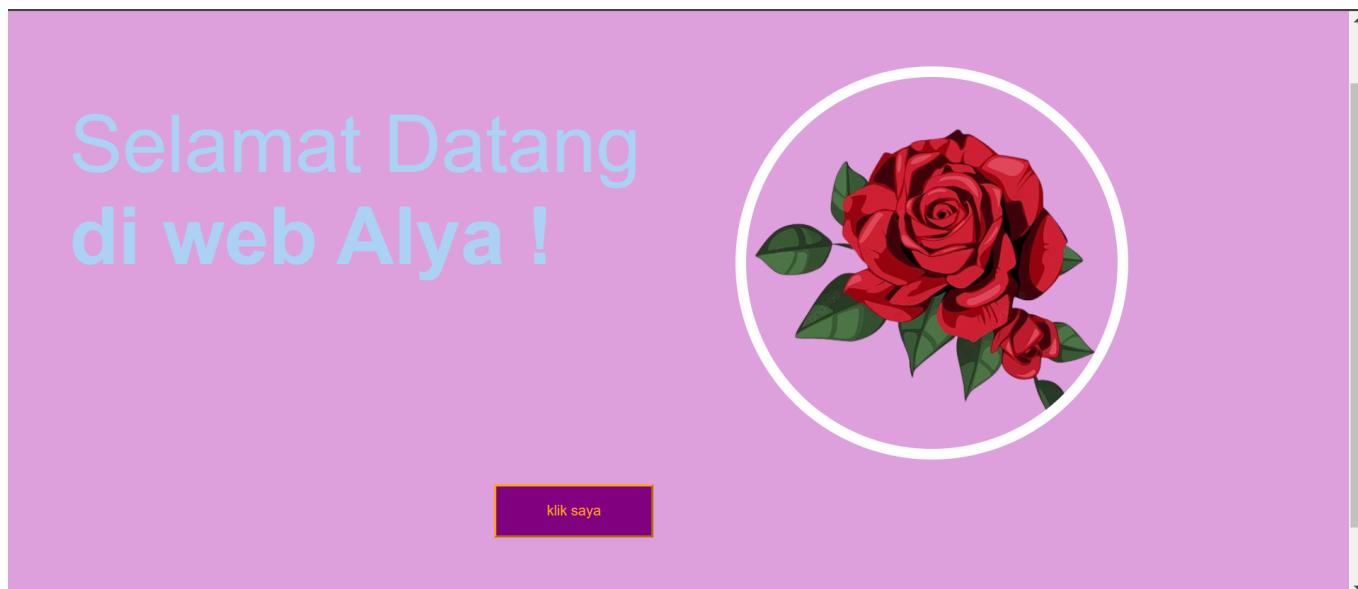
```
body{
    background-color: plum ;
}

p {
    font-size: 75px;
    font-family: 'arial';
    margin-top: 150px;
    margin-bottom: 100px;
    margin-left: 50px;
    margin-right: 100px;
    color: rgb(171, 210, 244);
}

img {
    margin-right: 200px;
    margin-top: -30px;
    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: 400px;  
}  
  
button:hover {  
    background-color: blueviolet;  
}
```

Hasil



Kesimpulan

Kode di atas menghasilkan halaman web sederhana dengan latar belakang plum, sebuah gambar bulat dengan border putih di sebelah kanan, dan sebuah tombol ungu dengan teks "klik saya" di sebelah kiri. Teks "Selamat Datang" dan "di web Alya !" ditampilkan dalam paragraf dengan font besar dan warna biru muda. Ketika mouse di atas tombol, tombol akan berubah menjadi ungu kebiruan.

Tantangan Transition

Penjelasan

1. body { ... }

- `background-color: rgb(248, 210, 163);` : Mengatur warna latar belakang halaman web menjadi warna peach (`rgb(248, 210, 163)`).
- `width: 100%;` : Mengatur lebar halaman web menjadi 100% dari lebar jendela browser.

2. `.container { ... }`

- `display: contents;` : Mengatur agar elemen `container` tidak menampilkan sendiri sebagai kotak, tetapi hanya menampilkan konten di dalamnya.
- `align-items: flex-end;` : Mengatur agar item-item di dalam `container` sejajar ke bagian bawah.
- `flex-direction: row;` : Mengatur agar item-item di dalam `container` disusun secara horizontal (dari kiri ke kanan).
- `justify-content: space-around;` : Mengatur agar item-item di dalam `container` memiliki jarak yang sama di antara mereka, dengan jarak yang lebih besar di tepi kiri dan kanan.

3. `.box-2 { ... }`

- `font-size: 75px;` : Mengatur ukuran font untuk elemen dengan kelas `box-2` menjadi 75px.
- `font-family: 'arial';` : Mengatur jenis font untuk elemen dengan kelas `box-2` menjadi "arial".
- `margin-top: 150px;` : Mengatur margin atas elemen dengan kelas `box-2` menjadi 150px.
- `margin-bottom: 100px;` : Mengatur margin bawah elemen dengan kelas `box-2` menjadi 100px.
- `margin-left: 50px;` : Mengatur margin kiri elemen dengan kelas `box-2` menjadi 50px.
- `margin-right: 100px;` : Mengatur margin kanan elemen dengan kelas `box-2` menjadi 100px.
- `color: rgb(104, 104, 104);` : Mengatur warna font untuk elemen dengan kelas `box-2` menjadi abu-abu (`rgb(104, 104, 104)`).

4. `.box-1 { ... }`

- `margin-right: 200px;` : Mengatur margin kanan elemen dengan kelas `box-1` menjadi 200px.
- `margin-top: -30px;` : Mengatur margin atas elemen dengan kelas `box-1` menjadi -30px (negatif, artinya margin ini akan "menarik" elemen ke atas).
- `border: 10px solid white;` : Mengatur border elemen dengan kelas `box-1` menjadi 10px tebal, berwarna putih, dan dengan jenis garis solid.
- `border-radius: 1500px 1500px;` : Mengatur bentuk border elemen dengan kelas `box-1` menjadi bulat dengan radius 1500px (menghasilkan bentuk lingkaran).

5. `button { ... }`

- `background-color: lightblue;` : Mengatur warna latar belakang tombol menjadi biru muda (`lightblue`).
- `width: 150px;` : Mengatur lebar tombol menjadi 150px.
- `height: 50px;` : Mengatur tinggi tombol menjadi 50px.
- `border-width: 2px;` : Mengatur lebar border tombol menjadi 2px.
- `color: rgb(138, 138, 229);` : Mengatur warna font tombol menjadi biru muda (`rgb(138, 138, 229)`).
- `border-color: rgba(73, 134, 240, 0.29);` : Mengatur warna border tombol menjadi biru muda transparan (`rgba(73, 134, 240, 0.29)`).
- `margin-bottom: 20px;` : Mengatur margin bawah tombol menjadi 20px.
- `margin-left: 400px;` : Mengatur margin kiri tombol menjadi 400px.

6. `button:hover { ... }`
- `background-color: lightcyan;` : Mengatur warna latar belakang tombol menjadi biru muda lebih terang (`lightcyan`) ketika mouse di atas tombol.
 - `font-weight: bolder;` : Mengatur font tombol menjadi lebih tebal ketika mouse di atas tombol.
 - `transition: all 0.3s ease-in;` : Mengatur efek transisi untuk semua properti tombol ketika mouse di atas tombol, dengan durasi 0.3 detik dan efek transisi "ease-in" (percepatan di awal transisi).

Kode Program

```
<!DOCTYPE html>
<html>
<head>
    <title>tantangan transition</title>
    <link rel="stylesheet" href="tantangan_transition.css">
</head>
<body>
    <div class="container">
        <div>
            
                <p class="box-2"> Selamat Datang <br><b>
di web Alya !</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

Selamat Datang di web Alya !

klik saya



Selamat Datang di web Alya !

klik saya



Kesimpulan

Kode di atas menghasilkan halaman web sederhana dengan latar belakang peach, sebuah gambar lingkaran dengan border putih di sebelah kanan, dan sebuah tombol biru muda dengan teks "klik saya" di sebelah kiri. Teks "Selamat Datang" dan "di web Alya !" ditampilkan dalam paragraf dengan font besar dan warna abu-abu. Ketika mouse di atas tombol, tombol akan berubah menjadi biru muda lebih terang dan fontnya akan menjadi lebih tebal dengan efek transisi selama 0.3 detik.

Tantangan Flexbox

Penjelasan

1. <!DOCTYPE html> : 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 Alya" yang ditampilkan dalam beberapa baris yang dipisahkan oleh tag . Teks "Web Alya" 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, "bg.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>
```

```

<head>
    <title>tantangan flexbox</title>
    <link rel="stylesheet" href="tantangan_flexbox.css">
</head>
<body>
    <div class="container">
        <div>
            
                <p class="box-2"> Selamat Datang <br><b>
di web Alya !</b>
                <p>
            </div>
            <button> klik saya </button>
        </div>
    </body>
</html>

```

```

body {
    background-color: bisque;
    width: 100%;
}

.container {
    display: flex;
    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(82, 82, 82);
}

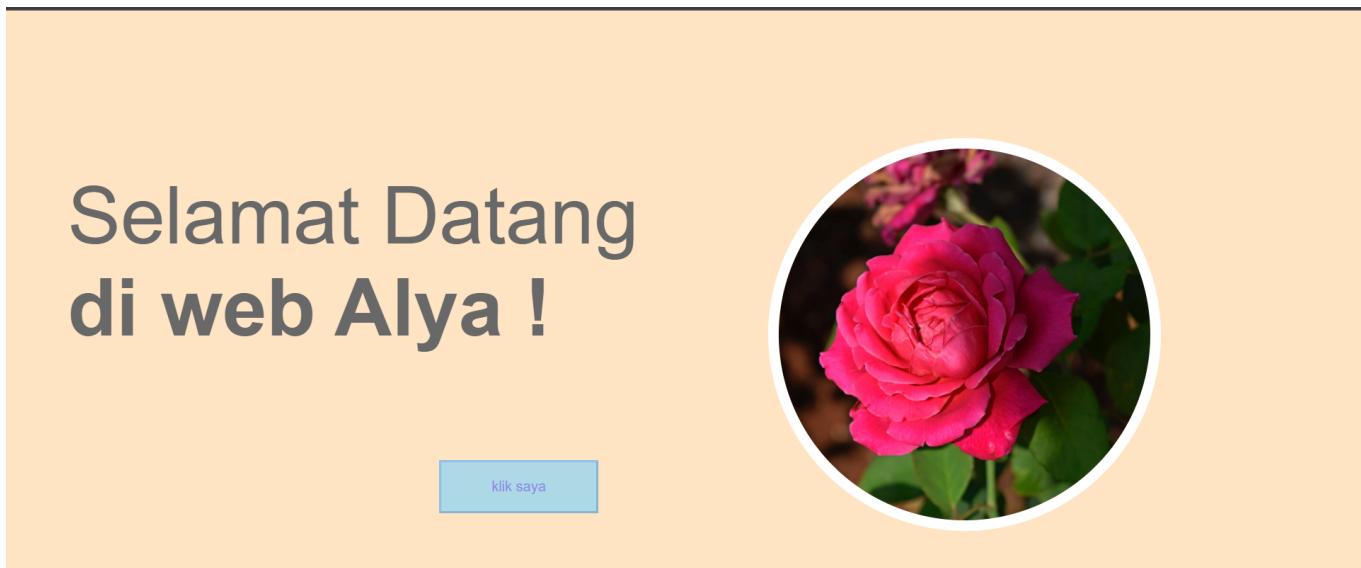
.box-1 {
    margin-right: 200px;
    margin-top: -30px;
    border: 10px solid white;
    border-radius: 1500px 1500px;
}

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;  
}
```

Hasil



Kesimpulan

Kode di atas menghasilkan halaman web sederhana dengan latar belakang bisque, sebuah gambar lingkaran dengan border putih di sebelah kanan, dan sebuah tombol biru muda dengan teks "klik saya" di sebelah kiri. Teks "Selamat Datang" dan "di web Alya !" ditampilkan dalam paragraf dengan font besar dan warna abu-abu. Ketika mouse di atas tombol, tombol akan berubah menjadi biru muda lebih terang dan fontnya akan menjadi lebih tebal.

Tantangan Position

Penjelasan

1. .container

- `position: relative;` : Mengatur posisi elemen "container" menjadi relatif. Ini berarti posisi elemen ini akan dihitung berdasarkan posisi normalnya di halaman.
- `display: flex;` : Mengatur elemen "container" menjadi container Flexbox. Ini memungkinkan untuk mengatur penempatan item-item di dalam elemen "container" dengan lebih mudah.
- `flex-direction: column;` : Mengatur agar item-item di dalam elemen "container" disusun secara vertikal (ke bawah).
- `width: 100%;` : Mengatur lebar elemen "container" menjadi 100% dari lebar jendela browser.
- `height: 100%;` : Mengatur tinggi elemen "container" menjadi 100% dari tinggi jendela browser.
- `background-color: skyblue;` : Mengatur warna latar belakang elemen "container" menjadi biru langit.

2. .box

- `width: 250px;` : Mengatur lebar elemen "box" menjadi 250px.
- `height: 400px;` : Mengatur tinggi elemen "box" menjadi 400px.
- `background-color: white;` : Mengatur warna latar belakang elemen "box" menjadi putih.
- `align-items: center;` : Mengatur agar item-item di dalam elemen "box" diposisikan di tengah secara horizontal.
- `align-content: center;` : Mengatur agar item-item di dalam elemen "box" diposisikan di tengah secara vertikal.
- `border-radius: 10px;` : Mengatur bentuk border menjadi bulat dengan radius 10px.
- `align-self: center;` : Mengatur agar elemen "box" diposisikan di tengah secara horizontal di dalam container Flexbox.
- `margin-top: 150px;` : Mengatur margin atas menjadi 150px.
- `margin-bottom: 150px;` : Mengatur margin bawah menjadi 150px.

3. .item

- `width: 100%;` : Mengatur lebar elemen "item" menjadi 100%.
- `color: black;` : Mengatur warna font elemen "item" menjadi hitam.

4. .icon

- `background-color: skyblue;` : Mengatur warna latar belakang elemen "icon" menjadi biru langit.
- `position: relative;` : Mengatur posisi elemen "icon" menjadi relatif. Ini berarti posisi elemen ini akan dihitung berdasarkan posisi normalnya di halaman.

- `margin-left: 200px;` : Mengatur margin kiri menjadi 200px.
- `bottom: 35px;` : Mengatur margin bawah menjadi 35px.
- `width: 30px;` : Mengatur lebar elemen "icon" menjadi 30px.
- `height: 30px;` : Mengatur tinggi elemen "icon" menjadi 30px.
- `border-radius: 1500px;` : Mengatur bentuk border menjadi bulat dengan radius 1500px.

5. .box-1

- `height: 225px;` : Mengatur tinggi elemen "box-1" menjadi 225px.
- `width: 100%;` : Mengatur lebar elemen "box-1" menjadi 100%.
- `border-radius: 10px 10px 0px 0px;` : Mengatur bentuk border menjadi bulat di bagian atas saja.

6. .box-2

- `font-size: 10px;` : Mengatur ukuran font menjadi 10px.
- `margin-left: 20px;` : Mengatur margin kiri menjadi 20px.
- `margin-top: -30px;` : Mengatur margin atas menjadi -30px. Ini akan membuat elemen "box-2" sedikit tumpang tindih dengan elemen di atasnya.
- `font-family: Arial, Helvetica, sans-serif;` : Mengatur jenis font menjadi Arial, Helvetica, sans-serif.

7. .box-3

- `font-size: 18px;` : Mengatur ukuran font menjadi 18px.
- `margin-left: 20px;` : Mengatur margin kiri menjadi 20px.
- `padding-top: 10px;` : Mengatur padding atas menjadi 10px.
- `font-family: Arial, Helvetica, sans-serif;` : Mengatur jenis font menjadi Arial, Helvetica, sans-serif.

8. .box-4

- `font-size: 11px;` : Mengatur ukuran font menjadi 11px.
- `font-family: Arial, Helvetica, sans-serif;` : Mengatur jenis font menjadi Arial, Helvetica, sans-serif.
- `margin-left: 20px;` : Mengatur margin kiri menjadi 20px.

9. .box-5

- `background-color: rgb(193, 193, 193);` : Mengatur warna latar belakang menjadi abu-abu muda.
- `padding-left: 20px;` : Mengatur padding kiri menjadi 20px.

- `padding-bottom: 10px;` : Mengatur padding bawah menjadi 10px.
- `padding-top: 10px;` : Mengatur padding atas menjadi 10px.
- `margin-bottom: 100px;` : Mengatur margin bawah menjadi 100px.
- `font-family: Arial, Helvetica, sans-serif;` : Mengatur jenis font menjadi Arial, Helvetica, sans-serif.
- `border-radius: 0px 0px 10px 10px;` : Mengatur bentuk border menjadi bulat di bagian bawah saja.
- `font-size: small;` : Mengatur ukuran font menjadi "small".

10. .item-1

- `width: 10px;` : Mengatur lebar elemen "item-1" menjadi 10px.
- `padding-left: 130px;` : Mengatur padding kiri menjadi 130px.

Kode Program

```
<!DOCTYPE html>
<html>
<head>
    <title>POSITION</title>
    <link rel="stylesheet" href="tantangan_position.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: relative;
    display: flex;
    flex-direction: column;
    width: 100%;
```

```
height: 100%;  
background-color: skyblue;  
}  
  
.box {  
width: 250px;  
height: 400px;  
background-color: white;  
align-items: center;  
align-content: center;  
border-radius: 10px;  
align-self: center;  
margin-top: 150px;  
margin-bottom: 150px;  
}  
  
.item {  
width: 100%;  
color: black;  
}  
  
.icon {  
background-color: skyblue;  
position: relative;  
margin-left: 200px;  
bottom: 35px;  
width: 30px;  
height: 30px;  
border-radius: 1500px;  
}  
  
.box-1 {  
height: 225px;  
width: 100%;  
border-radius: 10px 10px 0px 0px;  
}  
  
.box-2 {  
font-size: 10px;  
margin-left: 20px;  
margin-top: -30px;  
font-family: Arial, Helvetica, sans-serif;  
}  
  
.box-3 {  
font-size: 18px;
```

```

margin-left: 20px;
padding-top: 10px;
font-family: Arial, Helvetica, sans-serif;
}

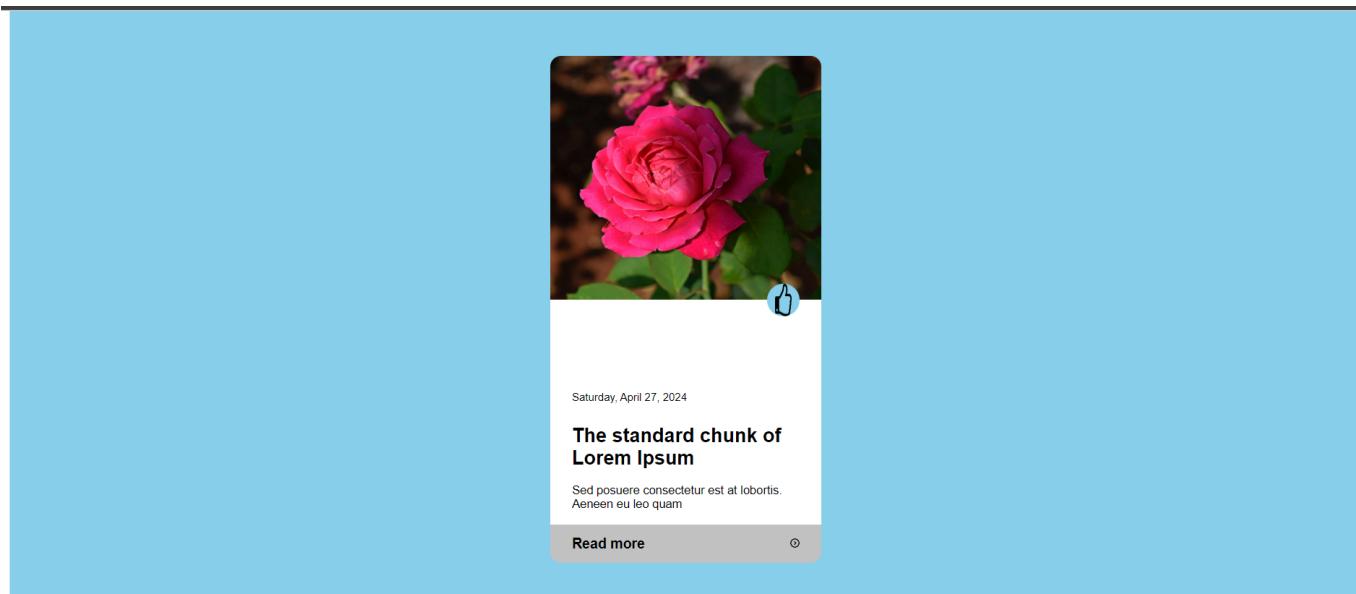
.box-4 {
  font-size: 11px;
  font-family: Arial, Helvetica, sans-serif;
  margin-left: 20px;
}

.box-5 {
  background-color: rgb(193, 193, 193);
  padding-left: 20px;
  padding-bottom: 10px;
  padding-top: 10px;
  margin-bottom: 100px ;
  font-family: Arial, Helvetica, sans-serif;
  border-radius: 0px 0px 10px 10px;
  font-size: small;
}

.item-1 {
  width: 10px;
  padding-left: 130px;
}

```

Hasil



Kesimpulan

Kode di atas menampilkan sebuah halaman web sederhana dengan konten yang berisi gambar, teks, dan tombol "Read more". Kode CSS mengatur tampilan elemen-elemen tersebut dengan pengaturan margin, padding, warna, ukuran font, dan bentuk. Elemen `container` dan `box` menggunakan `Position` untuk mengatur penempatan item-item di dalamnya.

Pengenalan Bootstrap

Apa Itu Bootstrap?

Bootstrap adalah salah satu dari banyak framework front-end yang ada di web development. Framework-front end menyediakan serangkaian alat dan gaya bawaan untuk mempercepat proses pengembangan web dengan menyediakan komponen-komponen UI siap pakai dan sistem grid yang responsif. Bootstrap khususnya, terkenal dengan kemampuannya dalam menciptakan tata letak yang responsif dan komponen-komponen UI yang seragam.

Instalasi Bootstrap

Cara instalasi Bootstrap secara online / CDN.

sekarang kita akan menjelaskan bagaimana cara menginstall bootstrap secara online. Berikut langkah-langkahnya:

1. Masuklah ke website resmi Bootstrap dengan mengunjungi situs web resmi di [https://getbootstrap.com/..](https://getbootstrap.com/).
2. Di laman utama web Bootstrap, cari dan klik menu "Introduction" yang terletak pada sisi sebelah kiri website.
3. Pada bagian "Starter Template" di halaman Introduction, kita akan melihat kode yang dapat kita gunakan untuk menghubungkan website kita yang mau diterapkan Bootstrap.
4. Klik tombol "Copy" untuk menyalin seluruh kode tersebut.
5. Buatlah file baru dengan nama index.html atau nama yang kita inginkan menggunakan teks editor atau editor HTML yang kita biasa pakai.
6. Tempelkan(paste) kode yang telah kita salin dari langkah sebelumnya ke dalam file html yang baru kita buat. dengan menempelkan kode tersebut maka bootstrap kita sudah terhubung namun harus secara online.
7. kita dapat mulai memberikan gaya pada tag-tag atau elemen dalam file HTML tersebut dengan menggunakan kelas-kelas Bootstrap. kita dapat melihatnya pada website resmi Bootstrap untuk mempelajari lebih lanjut tentang kelas-kelas yang tersedia dan cara penggunaannya.
8. Untuk melihat hasilnya, buka file html tadi menggunakan web browser kita.

9. Halaman web yang ditampilkan akan menggunakan Bootstrap untuk gaya dan fungsionalitasnya

Mengunduh dan menginstal Bootstrap secara lokal.

Untuk menginstal Bootstrap secara offline, kita perlu mengunduh file Bootstrap dan menyimpannya di folder proyek kita. Berikut adalah langkah-langkah untuk menginstal Bootstrap secara offline:

1. **Unduh File Bootstrap:** Kunjungi situs web resmi Bootstrap di <https://getbootstrap.com/> dan cari tautan unduhan untuk versi Bootstrap yang diinginkan. Klik atau ikuti instruksi untuk mengunduh file ZIP Bootstrap.
2. **Ekstrak File Bootstrap:** Setelah mengunduh file ZIP Bootstrap, temukan file tersebut di komputer kita dan ekstrak isi file ZIP ke folder proyek kita. kita dapat menggunakan aplikasi pengarsipan file atau ekstraksi bawaan (Archiver, 7zip, dll) pada sistem operasi kita untuk mengekstrak file ZIP . Setelah diekstrak, kita akan memiliki folder Bootstrap yang berisi berkas-berkas Bootstrap yang diperlukan.
3. **Hubungkan Berkas Bootstrap pada Halaman HTML:** Buka file HTML proyek kita menggunakan teks editor atau editor HTML yang biasa kita gunakan. Di dalam tag pada halaman HTML , tambahkan tautan ke berkas CSS Bootstrap dan skrip JavaScript Bootstrap. Gunakan tag link di dalam tag head untuk tautan CSS dan tag script di dalam tag body untuk javascript. Berikut contoh tautan yang umum di bootstrap:

```
<link href="path/to/bootstrap.min.css" rel="stylesheet">
// tautan bootstrap CSS
<script src="path/to/bootstrap.bundle.min.js"></script>
// tautan bootstrap Javascript
```

dengan menuliskan script diatas, maka kita telah menghubungkan antara file html kita dengan bootstrap yang kita unduh tadi.

4. **Gunakan Kelas Bootstrap:** Setelah tautan Bootstrap ditambahkan, Kita dapat menggunakan kelas-kelas Bootstrap dalam elemen HTML proyek Kita untuk menerapkan gaya dan fungsionalitas yang disediakan oleh bootstrap kita. Kita dapat masuk ke web resmi Bootstrap untuk mempelajari lebih lanjut tentang kelas-kelas yang tersedia dan cara penggunaannya.
5. **Jalankan File HTML:** Setelah Kita selesai menghubungkan tautan dan menggunakan kelas Bootstrap, Kita dapat menjalankan halaman HTML kita di web browser Kita untuk melihat hasilnya. Buka file HTML menggunakan web browser Kita dan lihat hasilnya

Komponen-Komponen Bootstrap

Bootstrap menyediakan berbagai komponen yang siap pakai untuk membangun tampilan website yang responsif dan menarik. Berikut adalah beberapa komponen utama yang disediakan oleh Bootstrap:

Grid System

Grid system Bootstrap adalah sistem layout yang responsif dan fleksibel. kita dapat membagi halaman menjadi baris(rows) dan kolom(columns) yang membentuk grid. Grid terdiri dari 12 kolom, yang dapat kita susun sesuai kebutuhan. Dengan menggunakan kelas CSS yang disediakan oleh Bootstrap, kita dapat dengan mudah menentukan berapa banyak kolom yang akan digunakan oleh setiap elemen di halaman web kita. Grid system ini sangat berguna dalam menciptakan tata letak yang responsif dan dapat menyesuaikan diri dengan berbagai ukuran layar

Typography

Komponen typography Bootstrap menyediakan gaya dan kelas CSS yang konsisten untuk tipografi di halaman web kita. kita dapat dengan mudah mengatur ukuran teks, gaya huruf, dan pengaturan lainnya menggunakan kelas-kelas yang telah ditentukan. Ini memastikan bahwa teks di halaman web kita memiliki tampilan yang konsisten dan mudah dibaca di berbagai perangkat.

Button

Komponen tombol Bootstrap memungkinkan kita dengan mudah membuat tombol dengan tampilan yang menarik dan responsif. kita dapat menggunakan kelas-kelas Bootstrap untuk mengatur berbagai gaya tombol, termasuk ukuran (large, small), warna (default, primary, secondary, dll.), dan variasi lainnya. Tombol-tombol ini dapat digunakan untuk tindakan seperti mengirim formulir, memuat ulang halaman, atau memicu tindakan lainnya di aplikasi web kita.

Forms

Bootstrap menyediakan komponen form yang mudah digunakan untuk membuat form input. Ini termasuk input teks, area teks, kotak centang (checkbox), tombol radio, dropdown, dan lain-lain. Komponen form Bootstrap telah dirancang dengan tampilan yang responsif dan mudah dikustomisasi. kita dapat dengan mudah menambahkan validasi form dan mengatur tampilan form kita dengan menggunakan kelas-kelas Bootstrap yang telah ditentukan.

Navbar

Komponen navbar Bootstrap memungkinkan kita membuat navigasi yang responsif dan mudah dikustomisasi di halaman web. kita dapat menambahkan logo, menu, tombol, dan komponen

lainnya ke navbar dengan mudah. Navbar Bootstrap juga menyediakan fitur seperti menu dropdown, navigasi yang terlipat untuk perangkat mobile, dan tata letak yang fleksibel.

Cards

Cards adalah komponen yang digunakan untuk menampilkan informasi dalam format yang terstruktur. Komponen kartu Bootstrap memungkinkan kita untuk membuat kolom dengan gambar, teks, tombol, dan komponen lainnya. cards ini dapat digunakan untuk menampilkan artikel, produk, profil pengguna, atau konten lainnya dengan tampilan yang menarik.

Modal

Komponen modal Bootstrap digunakan untuk menampilkan jendela pop-up yang tumpang tindih dengan konten utama halaman. Modal ini berfungsi untuk menyoroti konten tambahan, pesan, atau form yang membutuhkan fokus pengguna. kita dapat menyesuaikan tampilan modal, mengatur ukuran, menambahkan judul, dan mengatur perilaku saat modal ditampilkan atau ditutup.

Carousel

Carousel adalah komponen Bootstrap yang digunakan untuk membuat tampilan slide gambar atau konten lainnya. kita dapat menambahkan gambar, teks, tombol navigasi, dan indikator slide untuk membuat tampilan yang menarik dan interaktif. Carousel Bootstrap mendukung navigasi otomatis, kontrol manual, dan animasi transisi yang halus.

Icons

Bootstrap sendiri menggunakan ikon dari Font Awesome, yang merupakan kumpulan ikon vektor yang sangat populer dan kaya akan fitur. kita dapat dengan mudah menambahkan ikon ke elemen seperti tombol, tautan, dan elemen lainnya menggunakan class-class ikon Bootstrap. Ini berfungsi untuk memperindah halaman web kita.

Jumbotron

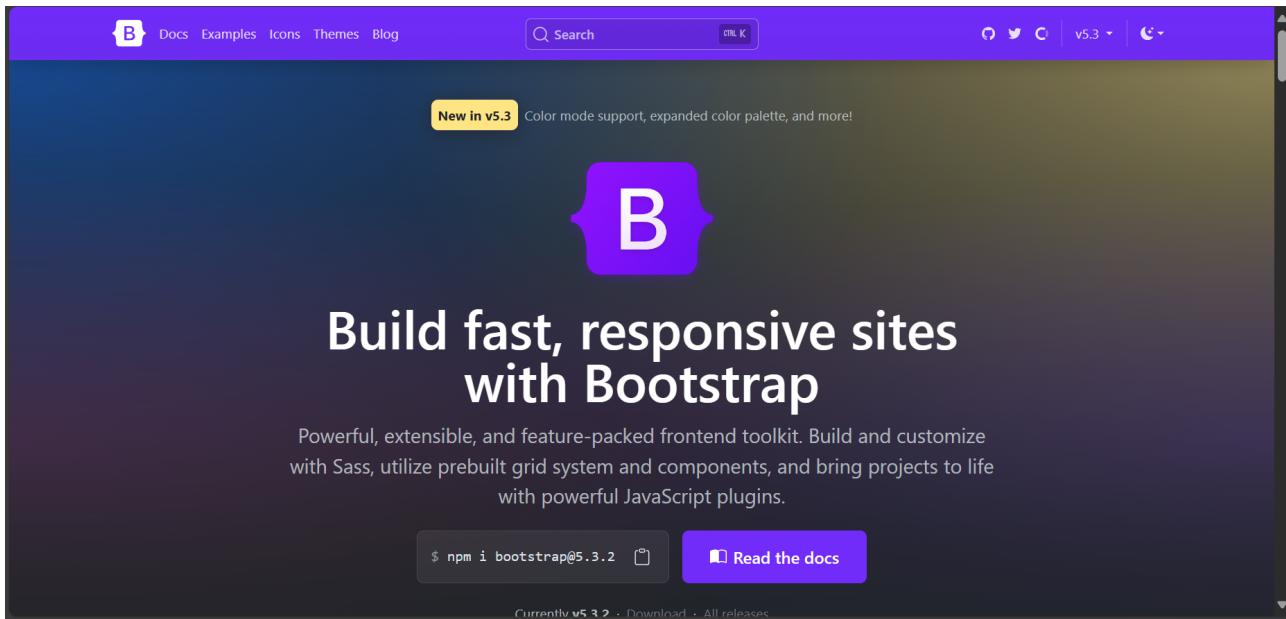
Jumbotron adalah komponen Bootstrap yang digunakan untuk menyoroti konten utama di halaman web. Biasanya ditempatkan di bagian atas halaman dengan judul besar dan Terima kasih atas klarifikasinya.Komponen carousel Bootstrap digunakan untuk membuat tampilan slide yang interaktif. kita dapat menambahkan gambar, teks, dan tombol navigasi ke dalam carousel. Carousel Bootstrap mendukung navigasi otomatis, kontrol manual, dan animasi transisi.

kita dapat menemukan daftar komponen lengkap dan dokumentasi resmi di situs web Bootstrap (<https://getbootstrap.com/docs/>).

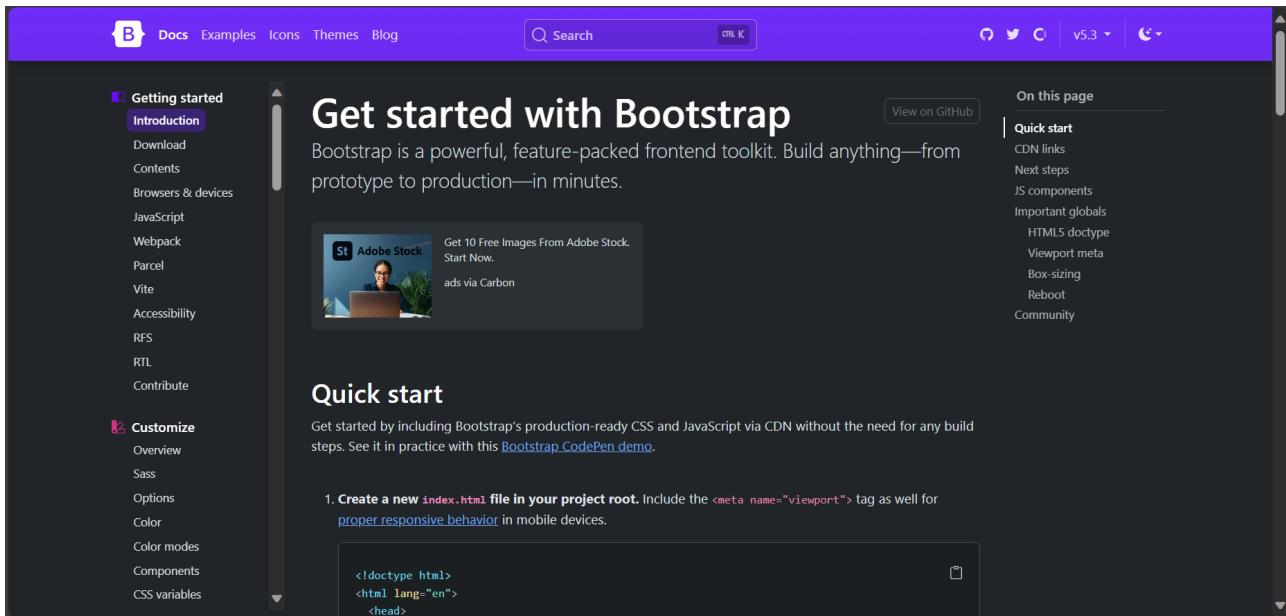
Contoh Penerapan Komponen Bootstrap

Typography

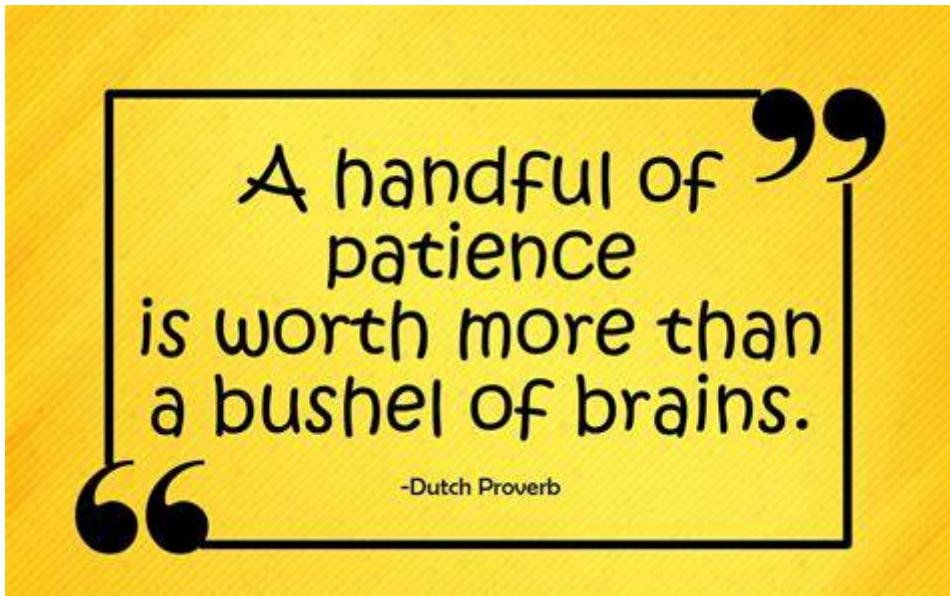
1. Buka situs resmi Bootstrap di getbootstrap.com



2. Klik "Read the docs". Maka kita akan diarahkan pada halaman web yang berisi berbagai macam Typografi (class Bootstrap).



3. Misalnya kita akan membuat sebuah teks quotes seperti di bawah ini:



4. Ketikan "Typography" di kolom search lalu enter. Maka kita akan masuk ke halaman berikut:

The screenshot shows the Bootstrap documentation website with a purple header. The left sidebar has a purple navigation bar with the following items: Breakpoints, Containers, Grid, Columns, Gutters, Utilities, Z-index, CSS Grid, Content (Reboot, Typography), Images, Tables, Figures, Forms (Overview, Form control, Select, Checks & radios, Range, Input group). The 'Typography' item is highlighted with a purple background. The main content area has a dark background. At the top, there's a search bar and a GitHub link. Below it, the title 'Typography' is displayed in a large, bold, white font. A sub-header says 'Documentation and examples for Bootstrap typography, including global settings, headings, body text, lists, and more.' To the right, there's a sidebar titled 'On this page' with a list of topics: Global settings, Headings, Customizing headings, Display headings, Lead, Inline text elements, Text utilities, Abbreviations, Blockquotes, Naming a source, Alignment, Lists, Unstyled, Inline, Description list alignment, Responsive font sizes, CSS, Sass variables, and Sass mixins. A small sidebar on the left shows icons for navigation and a message about design tips.

5. Carilah Subbab "Alignment":

Alignment

Use text utilities as needed to change the alignment of your blockquote.

A well-known quote, contained in ablockquote element.

— Someone famous in *Source Title*

HTML



```
<figure class="text-center">
  <blockquote class="blockquote">
    <p>A well-known quote, contained in ablockquote element.</p>
  </blockquote>
  <figcaption class="blockquote-footer">
    Someone famous in <cite title="Source Title">Source Title</cite>
  </figcaption>
</figure>
```

6. Salinlah kode yang tertera, lalu tempelkan di tag body pada halaman html yang telah kita hubungkan dengan Bootstrap kita

```
<figure class="text-center">
  <blockquote class="blockquote">
    <p>A well-known quote, contained in ablockquote element.</p>
  </blockquote>
  <figcaption class="blockquote-footer">
    Someone famous in
    <cite title="Source Title">Source Title </cite>
  </figcaption>
</figure>
```

```
< tes.html > < html > < body >
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4       <meta charset="UTF-8">
5       <meta name="viewport" content="width=device-width, initial-scale=1.0">
6       <title>Document</title>
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   </head>
9   <body>
10      <figure class="text-center">
11          <blockquote class="blockquote">
12              | <p>A well-known quote, contained in a blockquote element.</p>
13          </blockquote>
14          <figcaption class="blockquote-footer">
15              | Someone famous in <cite title="Source Title">Source Title</cite>
16          </figcaption>
17      </figure>
18  </html>
```

7. Hasilnya akan terlihat seperti berikut:

A well-known quote, contained in a blockquote element.
— Someone famous in *Source Title*

8. Kita bisa mengganti kata kata nya dengan cara mengganti teks yang ada di dalam tag p dan untuk teks kecil bagian bawah kita ganti pada bagian dalam tag figcaption

```
<figure class="text-center">
    <blockquote class="blockquote">
        <p>Sebenarnya otak kita sama aja, yang <br>
            membedakan hanya siapa yang <br>
            mulai belajar duluan dan siapa yang <br>
            belajar terus menerus </p>
    </blockquote>
```

```
<figcaption class="blockquote-footer">  
    Dea Afrizal Doroboka<cite title="Source Title">Doroboka</cite>  
</figcaption>  
</figure>
```

Hasilnya adalah sebagai berikut:

Sebenarnya otak kita sama aja, yang
membedakan hanya siapa yang
mulai belajar duluan dan siapa yang
belajar terus menerus
— Dea Afrizal Doroboka

Keterangan :

1. `[text-center]` pada `<figure>`: Kelas ini diterapkan pada elemen `<figure>`, yang mengakibatkan kontennya, termasuk elemen `<blockquote>` dan `<figcaption>`, akan diatur menjadi ketengah halaman maupun kontainer.
2. `blockquote` pada `<blockquote>` : Kelas ini memberikan gaya khusus pada elemen `<blockquote>`. Elemen ini digunakan untuk merinci sebuah kutipan atau teks yang dianggap signifikan. Penggunaan kelas ini dari Bootstrap mungkin memberikan tampilan tertentu, seperti memodifikasi gaya margin atau padding, untuk memberikan estetika yang lebih baik.
3. `blockquote-footer` pada `<figcaption>` : Kelas ini memberikan gaya khusus pada elemen `<figcaption>` yang berada di dalam `<figure>`. Elemen ini kemungkinan berisi informasi tambahan atau keterangan terkait elemen-elemen lain dalam `<figure>`.
 - `Dea Afrizal Doroboka` : Ini adalah teks yang berada di dalam elemen `<figcaption>`, memberikan informasi tambahan atau keterangan terkait dengan kutipan atau elemen-elemen lain dalam `<figure>`.
 - `<cite title="Source Title">Doroboka / cite>` : Ini adalah elemen `<cite>` yang memberikan judul sumber ("Source Title"). Penggunaan kelas `blockquote-footer` dari Bootstrap mungkin menyusun elemen ini dengan tata letak dan gaya tertentu, seperti menetapkannya sebagai teks kaki atau memberikan gaya yang konsisten dengan elemen lain dalam kelompok `<figure>`.

Keterangan:

Untuk bagian "Dea Afrizal" adalah nama seseorang yang mengatakan quotes tersebut dan

"Doroboka" adalah sumber dimana orang tersebut mengatakan kata-kata itu. Kata-kata yang dituliskan yaitu berada dalam tag `p`.

Navbar

1. Buka kembali getbootstrap.com lalu ketik "navbar" di kolom "search" dan klik enter

The screenshot shows the Bootstrap documentation website. The top navigation bar has links for Docs, Examples, Icons, Themes, and Blog. A search bar is present, along with social media icons and a version selector (v5.3). The left sidebar contains a navigation tree with categories like Breadcrumb, Buttons, Button group, Card, Carousel, Close button, Collapse, Dropdowns, List group, Modal, Nav (which is highlighted), Navs & tabs, Offcanvas, Pagination, Placeholders, Popovers, Progress, Scrollspy, Spinners, Toasts, and Tooltips. Below the sidebar, the main content area has a title 'Navbar' and a sub-section 'How it works'. It includes a list of bullet points about navbar behavior and styling, and a note about new features in v5.2.0. On the right side, there's a 'On this page' sidebar with links to various Bootstrap components and CSS sections.

2. Carilah jenis navbar yang diinginkan.

The screenshot shows the 'Image and text' section of the Bootstrap documentation. It displays an example of HTML code for a navbar containing an image and text. The code uses the `.d-inline-block` and `.align-text-top` utility classes. Below the code, there's a preview of how it looks. The right sidebar is identical to the one in the previous screenshot, showing the 'Image' link under the 'Supported content' section.

3. Jika misalnya kita telah menemukan jenis navbar yang kita inginkan, maka, salinlah kode program yang ada dibawah gambar contoh navbar tersebut

Navbar Home Features Pricing Dropdown link ▾

HTML



```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavDropdown">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Features</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Pricing</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" :>
            Dropdown link
          </a>
          <ul class="dropdown-menu">
            <li><a class="dropdown-item" href="#">Action</a></li>
```

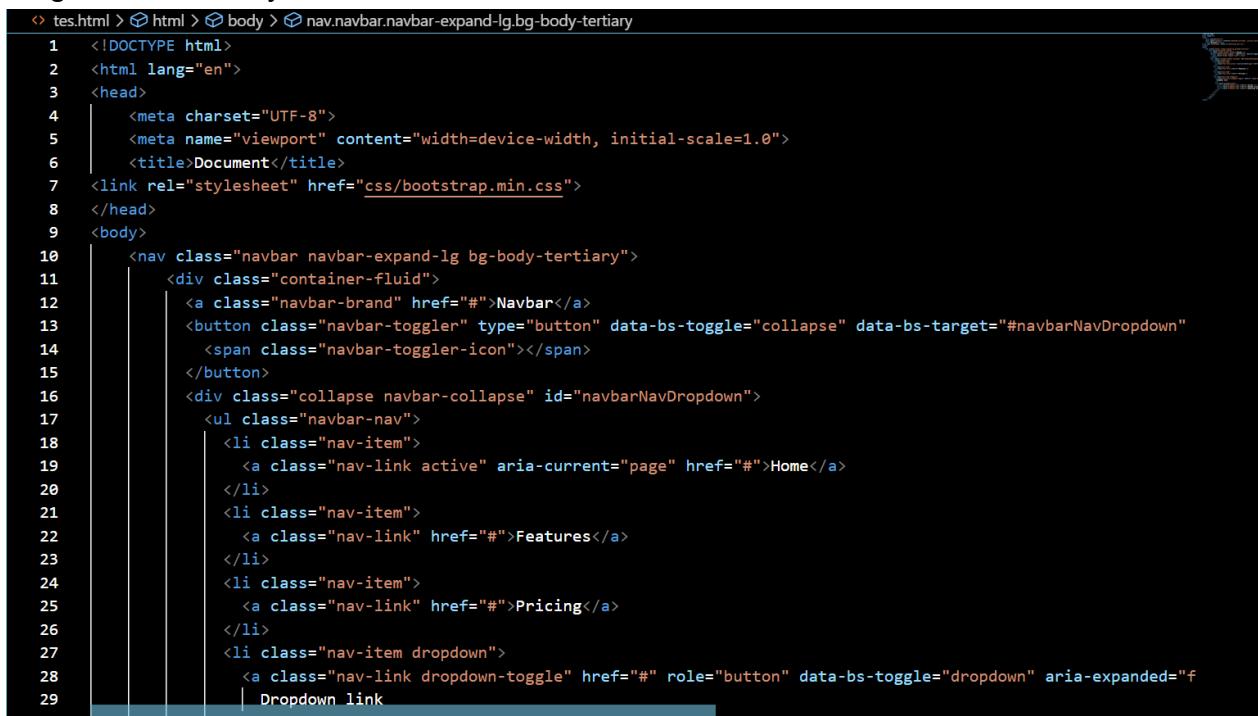
```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavDropdown">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Features</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Pricing</a>
        </li>
        <li class="nav-item dropdown">
```

```

        <a class="nav-link dropdown-toggle" href="#" role="button"
data-bs-toggle="dropdown" aria-expanded="false">
            Dropdown link
        </a>
        <ul class="dropdown-menu">
            <li><a class="dropdown-item" href="#">Action</a></li>
            <li><a class="dropdown-item" href="#">Another
action</a></li>
            <li><a class="dropdown-item" href="#">Something else
here</a></li>
        </ul>
    </li>
</ul>
</div>
</div>
</nav>

```

4. Tempelkanlah kode program yang terlah kita salin ke dalam tag body di file html kita. Jangan lupa untuk menautkan file html kita dengan bootstrap seperti pada langkah-langkah sebelumnya



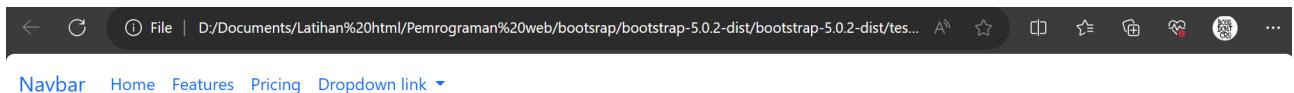
The screenshot shows a code editor with the following code:

```

1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="UTF-8">
5          <meta name="viewport" content="width=device-width, initial-scale=1.0">
6          <title>Document</title>
7          <link rel="stylesheet" href="css/bootstrap.min.css">
8      </head>
9      <body>
10         <nav class="navbar navbar-expand-lg bg-body-tertiary">
11             <div class="container-fluid">
12                 <a class="navbar-brand" href="#">Navbar</a>
13                 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown">
14                     <span class="navbar-toggler-icon"></span>
15                 </button>
16                 <div class="collapse navbar-collapse" id="navbarNavDropdown">
17                     <ul class="navbar-nav">
18                         <li class="nav-item">
19                             <a class="nav-link active" aria-current="page" href="#">Home</a>
20                         </li>
21                         <li class="nav-item">
22                             <a class="nav-link" href="#">Features</a>
23                         </li>
24                         <li class="nav-item">
25                             <a class="nav-link" href="#">Pricing</a>
26                         </li>
27                         <li class="nav-item dropdown">
28                             <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="f
| Dropdown link
29                         </li>

```

5. Jalankanlah file html yang sudah ditempelki kode program bootstrap tadi melalui web browser. Maka lihatlah hasilnya

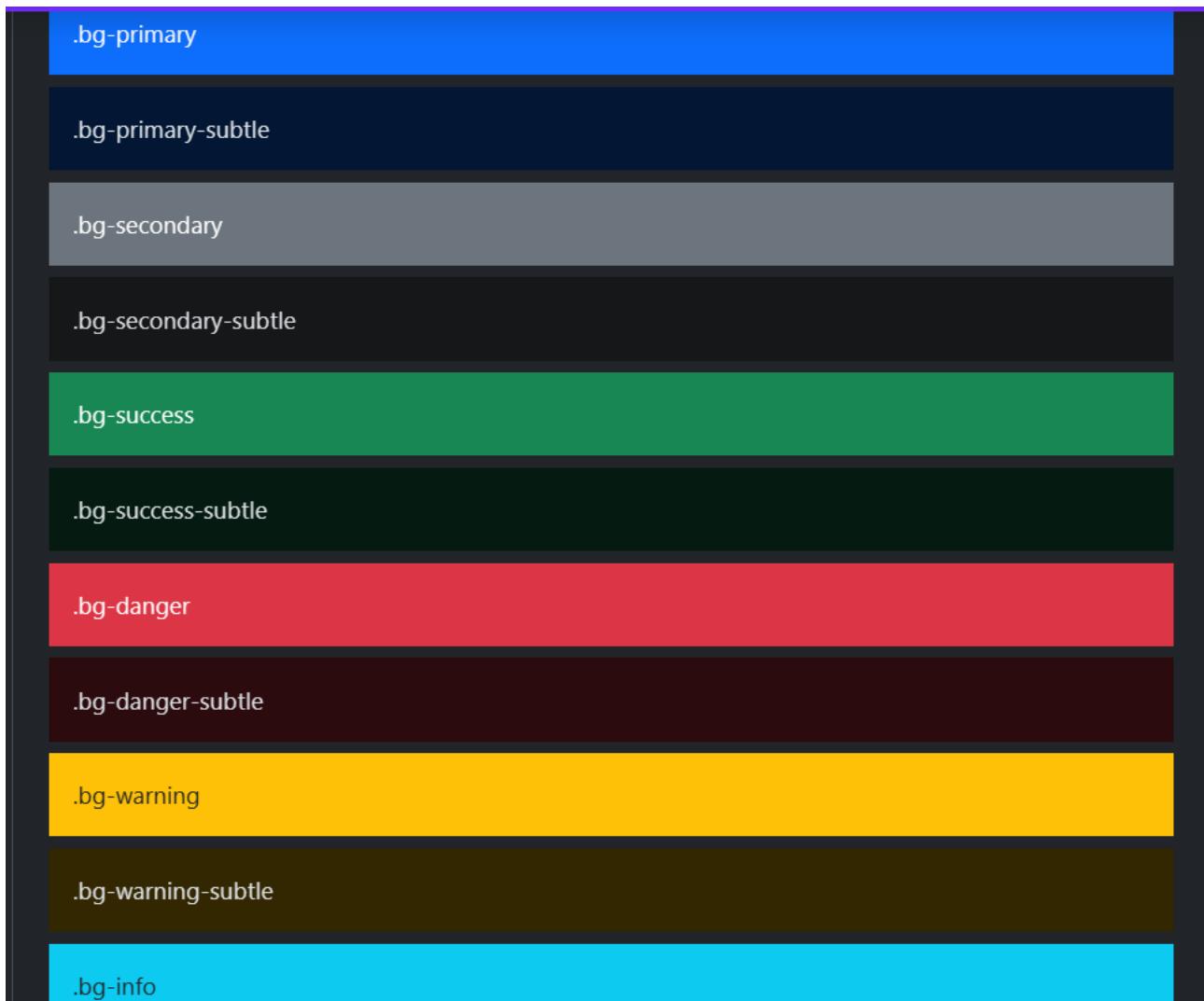


6. Misalnya kita ingin memberikan warna hijau pada background navbar kita. Pertama ketiklah "background" di kolom search

The screenshot shows the Bootstrap documentation site. The sidebar on the left has a tree view of utility classes, with 'Background' selected. The main content area has a dark background with white text. A callout box contains an accessibility tip: "Accessibility tip: Using color to add meaning only provides a visual indication, which will not be conveyed to users of assistive technologies like screen readers. Please ensure the meaning is obvious from the content itself (e.g., the visible text with a sufficient color contrast) or is included through alternative means, such as additional text hidden with the `.visually-hidden` class." At the bottom of the page, there is a blue bar containing the text ".bg-primary".

7. Carilah Warna yang kita inginkan. Misalnya dalam praktek ini adalah warna hijau





Terdapat banyak jenis warna dalam bootstrap.

8. Karena kita ingin mengkostumisasi warna background kita menjadi hijau maka kita akan memakai bg-success . ketikkan bg-success pada class tag pembuka seperti di bawah ini

```
① tes.html > ⌂ html > ⌂ body > ⌂ nav.navbar.navbar-expand-lg.bg-success
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Document</title>
7     <link rel="stylesheet" href="css/bootstrap.min.css">
8   </head>
9   <body>
10    <nav class="navbar navbar-expand-lg bg-success">
11      <div class="container-fluid">
```

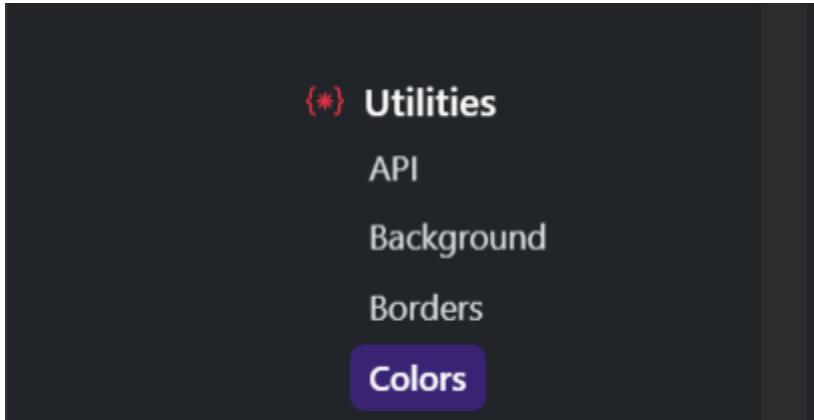
Keterangan:

Karena kita akan memberikan warna background pada navbar maka untuk menerapkannya, langsung terapkan class tag yang membungkus semua eleme-elemen yang ada pada tag tersebut.

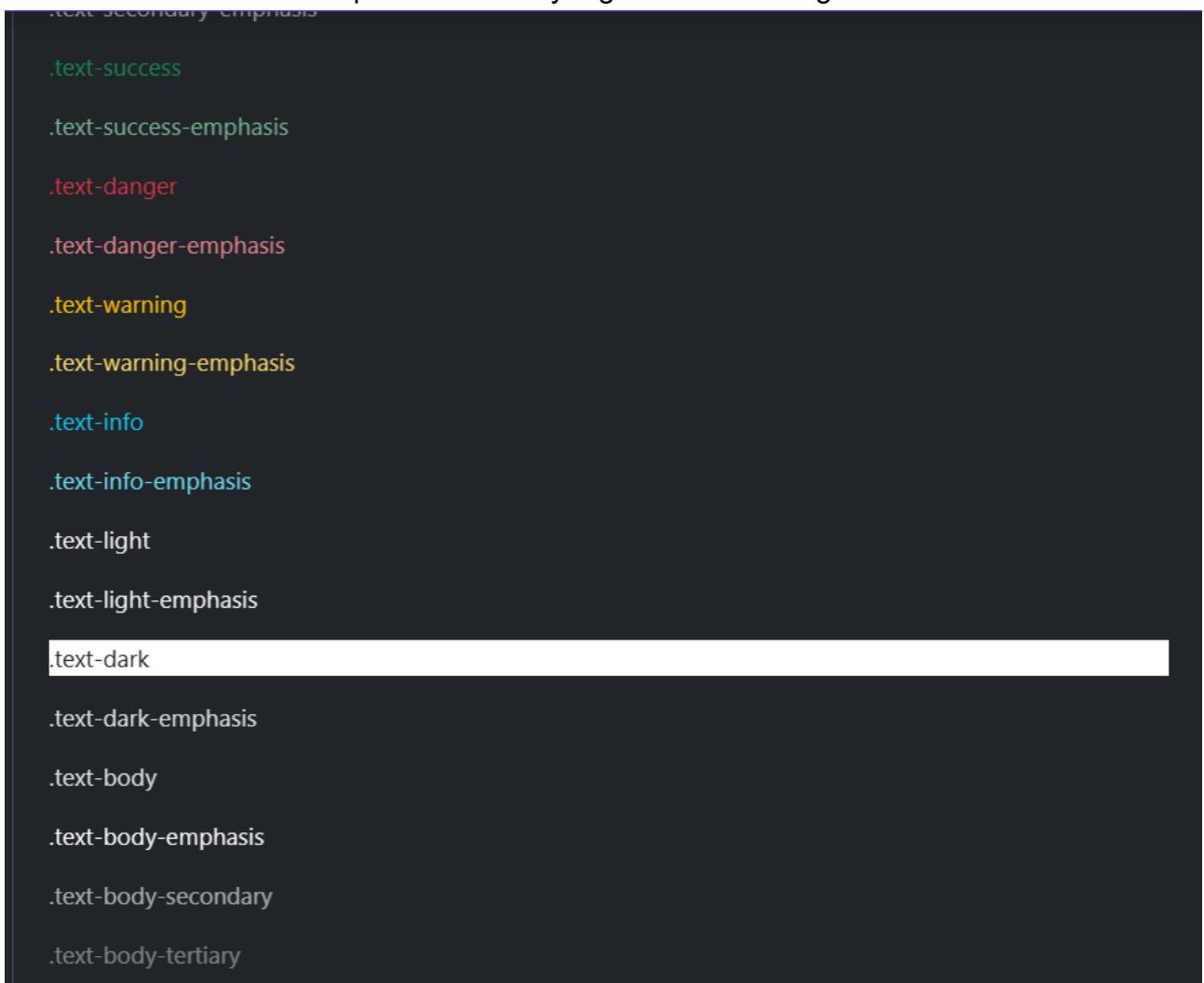
9. Jika sudah maka hasilnya akan seperti ini:



10. Sekarang kita akan mengubah warna font yang ada pada navbar. Langkah pertama klik "Colors" pada bagian Utilities



11. Maka kita akan diarahkan pada halaman yang memuat berbagai macam warna font



12. Carilah warna font yang kita inginkan. Misalnya pada kostumisasi navbar ini. Kita akan menggunakan warna font putih

```
.text-black
```

```
.text-white
```

```
.text-black-50
```

```
.text-white-50
```

13. Karena kita ingin menggunakan warna font putih maka perintah yang digunakan adalah "text-white"

14. Masuklah kembali ke dalam file html kita

15. pada bagian class sebuah tag yang membungkus text yang akan kita ganti warna, ketiklah text-white . Contohnya ada pada gambar berikut:

```
<nav class="navbar navbar-expand-lg bg-success">
|   <div class="container-fluid">
|   |   <a class="navbar-brand text-white" href="#">Navbar</a>
```

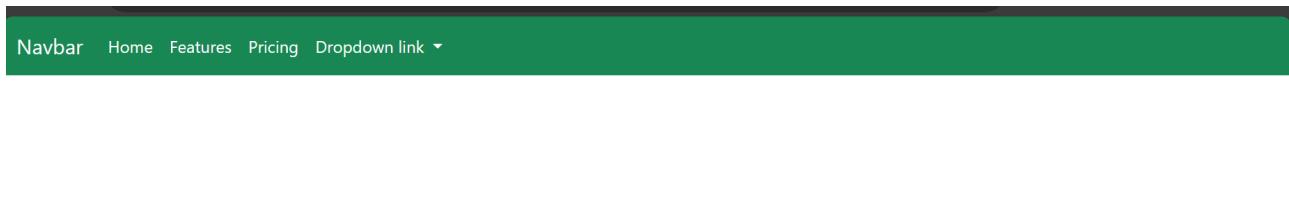
16. Hasilnya akan terlihat seperti di gambar ini:



17. Terapkanlah Text-White diseluruh class pada tag yang membungkus text-text itu seperti halnya pada text navbar tadi

"CSS/assets/btc-30.png" could not be found.

18. Jika file html dijalankan, hasilnya adalah sebagai berikut



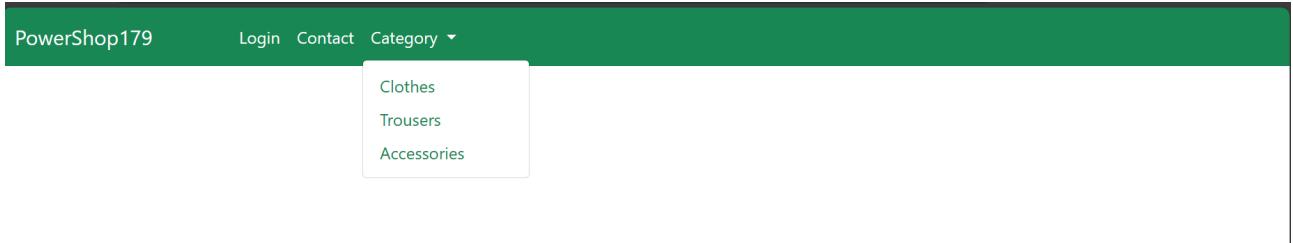
19. Sekarang kita akan mengganti text yang ada di navbar tersebut. Caranya sangat mudah, yaitu cukup kita ganti text-text tadi dengan kata kata yang kita inginkan. Misalnya adalah sebagai berikut:

```

10   <nav class="navbar navbar-expand-lg bg-success" >
11     <div class="container-fluid" >
12       <a class="navbar-brand text-white" href="#">PowerShop179</a>
13       <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" 
14         <span class="navbar-toggler-icon"></span>
15       </button>
16       <div class="collapse navbar-collapse" id="navbarNavDropdown">
17         <ul class="navbar-nav">
18           <li class="nav-item">
19             <a class="nav-link active text-success" aria-current="page" href="#">Home</a>
20           </li>
21           <li class="nav-item">
22             <a class="nav-link text-white" href="#">Login</a>
23           </li>
24           <li class="nav-item">
25             <a class="nav-link text-white" href="#">Contact</a>
26           </li>
27           <li class="nav-item dropdown">
28             <a class="nav-link dropdown-toggle text-white" href="#" role="button" data-bs-toggle="dropdown" aria-
29               | Category
30             </a>
31             <ul class="dropdown-menu">
32               <li><a class="dropdown-item text-success" href="#">Clothes</a></li>
33               <li><a class="dropdown-item text-success" href="#">Trousers</a></li>
34               <li><a class="dropdown-item text-success" href="#">Accessories</a></li>
35             </ul>
36           </li>
37         </ul>
38       </div>

```

20. Maka hasilnya adalah sebagai berikut:



21. Sekarang kita akan menambahkan kolom search pada navbar kita. pada bagian navbar di getbootstrap.com Scroll ke bawah hingga menemukan seperti pada gambar

```

<nav class="navbar bg-body-tertiary">
  <div class="container-fluid">
    <form class="d-flex" role="search">
      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>

```

22. Salin kodennya lalu tempelkan di bagian bawah kode navbar kita tadi. Jadi jangan disatukan ke dalam kode navbar tadi.

```

<nav class="navbar bg-body-tertiary">
    <div class="container-fluid">
        <form class="d-flex" role="search">
            <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
            <button class="btn btn-outline-success" type="submit">Search</button>
        </form>
    </div>
</nav>

```

```

<!-- tes.html -->
<html>
    <body>
        <nav class="navbar bg-body-tertiary">
            <div class="container-fluid">
                <ul class="nav-item dropdown">
                    <a class="nav-link text-white" href="#">Contact</a>
                </li>
                <li class="nav-item dropdown">
                    <a class="nav-link dropdown-toggle text-white" href="#" role="button" data-bs-toggle="dropdown" aria-pressed="false" data-bs-display="block">Category</a>
                    <ul class="dropdown-menu">
                        <li><a class="dropdown-item text-success" href="#">Clothes</a></li>
                        <li><a class="dropdown-item text-success" href="#">Trousers</a></li>
                        <li><a class="dropdown-item text-success" href="#">Accessories</a></li>
                    </ul>
                </li>
            </ul>
        </div>
    </nav>
    <nav class="navbar bg-body-tertiary">
        <div class="container-fluid">
            <form class="d-flex" role="search">
                <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
                <button class="btn btn-outline-success" type="submit">Search</button>
            </form>
        </div>
    </nav>
</html>

```

Keterangan:

- [**navbar**] : Kelas ini menunjukkan bahwa elemen [`<nav>`] adalah bagian dari komponen navbar. Navbar adalah bagian dari Bootstrap yang menyediakan navigasi di bagian atas halaman web. Dengan memberikan kelas navbar , kita memberitahu Bootstrap untuk menerapkan gaya dan tata letak khusus yang terkait dengan navigasi.
- [**bg-body-tertiary**] : Kelas ini memberikan warna latar belakang pada navbar. Dalam hal ini, warna latar belakangnya diberi warna sesuai dengan warna "hijau" Warna latar belakang ini dapat disesuaikan sesuai kebutuhan desain.
- [**container-fluid**] : Kelas ini diterapkan pada elemen [`<div>`] yang membungkus elemen-elemen dalam navbar. Kelas ini memberikan padding yang sesuai dan membuat elemen-elemen di dalamnya menjaga lebar penuh dari layar (menggunakan grid system Bootstrap). Dengan kata lain, kontennya akan merespons secara baik pada berbagai lebar perangkat.

4. `d-flex` : Kelas ini mengubah elemen `<form>` menjadi kontainer flex. Flexbox adalah teknik tata letak yang kuat di CSS yang memudahkan pengaturan dan penataan elemen dalam satu atau dua dimensi. Dengan memberikan kelas `d-flex` , elemen form dan anak-anaknya dapat disusun secara fleksibel.
5. `form-control` : Kelas ini diterapkan pada elemen `<input>` untuk memberikan gaya yang konsisten pada elemen formulir. Di sini, elemen input diberi gaya Bootstrap standar untuk mengubahnya menjadi kontrol formulir yang responsif dan berada dalam baris yang sesuai.
6. `me-2` : Ini adalah kelas Bootstrap yang memberikan margin kanan sebesar 2 pada elemen yang memilikinya. Dalam codingan ini, kelas ini diterapkan pada elemen input untuk memberikan jarak margin kanan.
7. `btn` dan `btn-outline-success` : Kelas ini memberikan gaya pada elemen `<button>` . Kelas `btn` mengindikasikan bahwa ini adalah elemen tombol Bootstrap, sedangkan `btn-outline-success` memberikan gaya tombol dengan warna tepi hijau ("success" dalam Bootstrap).
8. `type="submit"` pada tombol: Ini menentukan bahwa tombol tersebut bertindak sebagai tombol submit dalam formulir. Ketika ditekan, formulir akan dikirim.
9. Maka hasilnya adalah seperti berikut:



10. Untuk membuat agar navbar tersebut terlihat menyatu, berikan `background-color` yang sama pada navbar diatasnya, yaitu menggunakan kode `bg-success` . Letakkan di class tag yang membungkus seluruh elemen search tadi.

```
<nav class="navbar bg-success">
  <div class="container-fluid">
    <form class="d-flex" role="search">
      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>
```

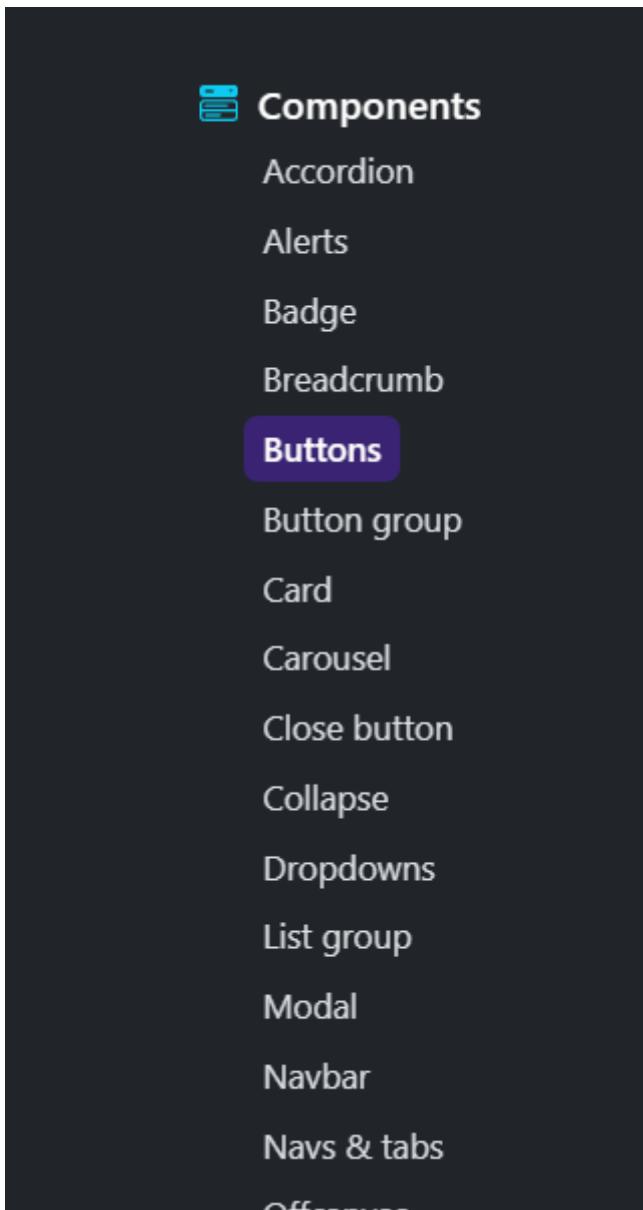
11. Maka hasil yang muncul akan seperti ini



Buttons

Pada praktik sebelumnya kita sudah mengatur navbar dan juga menambahkan kolom search. Namun ada masalah dengan tombolnya. Karena tombolnya memiliki warna yang sama dengan warna background yang digunakan yaitu hijau, Maka tombol tersebut tidak terlihat akibat warnanya menyatu. Pada bagian kali kita akan membahas cara mengatur sebuah tombol di bootstrap. Berikut caranya:

1. Permasalahan tadi ialah background color tombolnya yang menyatu dengan warna background navbarnya. Untuk mengatur warna tombol, pertama di situs getbootstrap.com, carilah bagian "buttons" di Components



2. Jika sudah menemukan klik bagian tersebut, maka kita akan diarahkan pada bagian buttons seperti pada gambar

Buttons

Use Bootstrap's custom button styles for actions in forms, dialogs, and more with support for multiple sizes, states, and more.



I've spent 2 years learning DNS while building NSLookup.io. Now, I'm teaching everything I know.
ads via Carbon

[View on GitHub](#)

On this page

Base class

Variants
Disable text wrapping

Button tags

Outline buttons

Sizes

Disabled state

Link functionality caveat

Block buttons

Button plugin

Toggle states

Methods

CSS

Variables

Sass variables

Sass mixins

Sass loops

Base class

Bootstrap has a base `.btn` class that sets up basic styles such as padding and content alignment. By default, `.btn` controls have a transparent border and background color, and lack any explicit focus and hover styles.

Base class

HTML



```
<button type="button" class="btn">Base class</button>
```

3. Scroll lah ke bawah hingga menemukan bagian "Outlines Buttons"

Outline buttons

In need of a button, but not the hefty background colors they bring? Replace the default modifier classes with the `.btn-outline-*` ones to remove all background images and colors on any button.

Primary

Secondary

Success

Danger

Warning

Info

Light

HTML



```
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-secondary">Secondary</button>
<button type="button" class="btn btn-outline-success">Success</button>
<button type="button" class="btn btn-outline-danger">Danger</button>
<button type="button" class="btn btn-outline-warning">Warning</button>
<button type="button" class="btn btn-outline-info">Info</button>
<button type="button" class="btn btn-outline-light">Light</button>
<button type="button" class="btn btn-outline-dark">Dark</button>
```

4. Saat ini kita akan menggunakan jenis warna putih agar tidak lagi menyatu dengan warna background navabrnya. Pada outline buttons yang berwarna putih, tertulis "Light". Oleh karena itu salinlah kode program yang ada tulisan "Light".

```
<button type="button" class="btn btn-outline-light">Light</button>
```

5. Blok lah terlebih dahulu kode program yang mengatur tombol pada bagian search tadi

```
<nav class="navbar bg-success">
  <div class="container-fluid">
    <form class="d-flex" role="search">
      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>
```

6. Tempelkanlah kode program outline light, untuk menggantikan kode program button yang lama

```
<nav class="navbar bg-success">
  <div class="container-fluid">
    <form class="d-flex" role="search">
      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
      <button type="button" class="btn btn-outline-light">Light</button>
    </form>
  </div>
</nav>
```

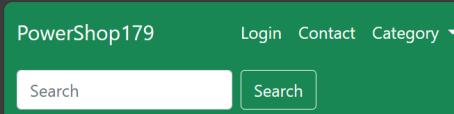
7. Maka hasilnya akan bagus seperti pada gambar dibawah ini:



8. Ubahlah Text "Light" menjadi "Search"

```
<button type="button" class="btn btn-outline-light">Search</button>
```

9. Maka hasilnya akan seperti ini:



Cards dan Form

Sekarang kita akan membuat sebuah form, dimana form tersebut kita asumsikan sebagai login page dari web kita tadi. Dalam pembuatan form login ini, kita akan menggunakan cards sebagai layout/tata letak yang akan mengatur form login tadi. Berikut langkah-langkahnya:

1. Yang pertama kita tentunya harus menentukan layout cards nya terlebih dahulu. Hal ini bertujuan agar jika tata letak card sudah ada, maka kita tinggal mengatur form nya di dalam card tadi. Masuk ke situs bootstrap tadi lalu pilih cards

Cards

[View on GitHub](#)

Bootstrap's cards provide a flexible and extensible content container with multiple variants and options.



Get 10 free Adobe Stock photos. Start
downloading amazing royalty-free
stock photos today.

ads via Carbon

About

A **card** is a flexible and extensible content container. It includes options for headers and footers, a wide variety of content, contextual background colors, and powerful display options. If you're familiar with Bootstrap 3, cards replace our old panels, wells, and thumbnails. Similar functionality to those components is available as modifier classes for cards.

Example

Cards are built with as little markup and styles as possible, but still manage to deliver a ton of control and customization. Built with flexbox, they offer easy alignment and mix well with other Bootstrap components. They

On this page

About

Example

Content types

Body

Titles, text, and links

Images

List groups

Kitchen sink

Header and footer

Sizing

Using grid markup

Using utilities

Using custom CSS

Text alignment

Navigation

Images

Image caps

Image overlays

Horizontal

Card styles

Background and color

Border

Mixins utilities

- Scrol ke bawah hingga menemukan layout yang kita inginkan, misalnya seperti pada gambar:

Featured

Special title treatment

With supporting text below as a natural lead-in to additional content.

[Go somewhere](#)

HTML

```
<div class="card">
  <div class="card-header">
    Featured
  </div>
  <div class="card-body">
    <h5 class="card-title">Special title treatment</h5>
    <p class="card-text">With supporting text below as a natural lead-in to additional content...
      <a href="#" class="btn btn-primary">Go somewhere</a>
    </p>
  </div>
</div>
```

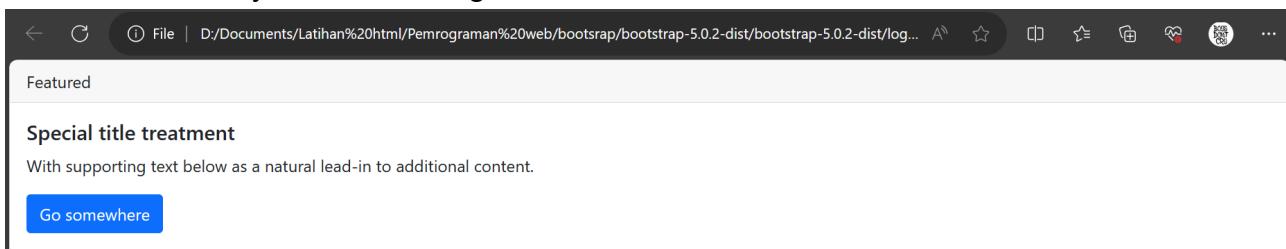
- Pada bagian bawah contoh, terdapat kode program. Salinlah kode program tersebut

```

<div class="card">
    <div class="card-header">
        Featured
    </div>
    <div class="card-body">
        <h5 class="card-title">Special title treatment</h5>
        <p class="card-text">With supporting text below as a natural lead-in to additional content.</p>
        <a href="#" class="btn btn-primary">Go somewhere</a>
    </div>
</div>

```

4. Tempelkanlah kode program itu pada halaman html terkait. Tentu saja dengan bootstrap yang sudah ditautkan, baik offline, maupun online.
5. Maka hasil awalnya adalah sebagai berikut:



6. karena jarak atas antara card dengan batas website terlalu dekat maka pada class card tambahkan mt-2 .

```

<div class="card mt-2">
    <div class="card-header">
        Featured
    </div>
    <div class="card-body">
        <h5 class="card-title">Special title treatment</h5>
        <p class="card-text">With supporting text below as a natural lead-in to additional content.</p>
        <a href="#" class="btn btn-primary">Go somewhere</a>
    </div>
</div>

```

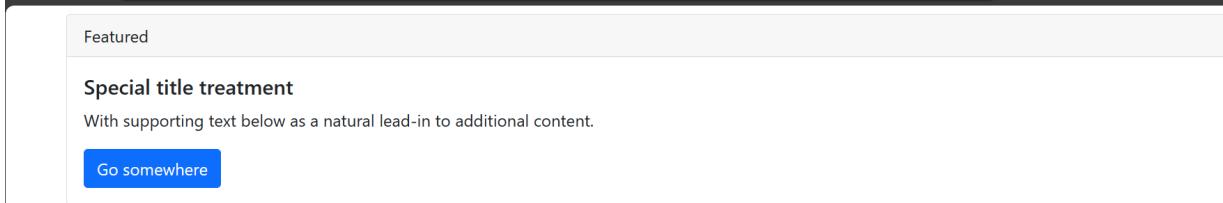
7. Maka hasilnya adalah seperti ini:



8. Sekarang kita ingin membuatnya terlihat ke tengah. Maka buatlah sebuah div dengan class yaitu "container", lalu salin masuklah kode program card tadi. Dengan kata lain, bungkuslah kode program card dengan sebuah div dengan class="container".

```
<div class="container">
  <div class="card mt-2">
    <div class="card-header">
      Featured
    </div>
    <div class="card-body">
      <h5 class="card-title">Special title treatment</h5>
      <p class="card-text">With supporting text below as a natural lead-in to additional content.</p>
      <a href="#" class="btn btn-primary">Go somewhere</a>
    </div>
  </div>
</div>
```

9. Berikut hasil dari di bungkusnya cards tadi.



10. Sekarang kita akan membuat form nya. Pertama-tama, kita perlu membuat judul form. Caranya adalah ganti tulisan "Featured" di html kita dengan kata yang kita inginkan. misalnya pada program ini kita akan mengganti tulisan menjadi "Form Login".

```
<div class="container">
  <div class="card mt-2">
    <div class="card-header">
      | <h5>Form Login</h5>
    </div>
    <div class="card-body">
      | <h5 class="card-title">Special title treatment</h5>
      | <p class="card-text">With supporting text below as a natural lead-in to additional content.</p>
      | <a href="#" class="btn btn-primary">Go somewhere</a>
    </div>
  </div>
</div>
```

```
<div class="container">
  <div class="card mt-2">
    <div class="card-header">
      <h5>Form Login</h5>
    </div>
    <div class="card-body">
      <h5 class="card-title">Special title treatment</h5>
```

```

<p class="card-text">With supporting text below as a natural lead-in to additional content.</p>
    <a href="#" class="btn btn-primary">Go somewhere</a>
</div>
</div>
</div>

```

Keterangan:

- [container] pada [div class="container"] : Kelas ini memberikan tata letak yang terkandung pada lebar tertentu (dalam pixel) dan ditengahkan di tengah halaman atau elemen yang memuatnya. Ini membantu dalam mengatur konten agar sesuai dengan standar desain Bootstrap.
- [card] pada [div class="card mt-2"] : Kelas ini memberikan gaya dan tata letak khusus untuk elemen [div] yang merupakan kartu (card). Kartu adalah elemen Bootstrap yang sering digunakan untuk menampilkan konten atau informasi dalam satu unit terpisah dengan gaya yang konsisten.
 - mt-2 : Kelas ini memberikan margin atas (margin-top) sebesar 2 unit. Ini membantu memberikan ruang di bagian atas kartu dan memisahkannya dari elemen-elemen sekitarnya.
- [card-header] pada [div class="card-header"] : Kelas ini memberikan gaya khusus untuk elemen [div] yang berfungsi sebagai header kartu. Header kartu biasanya berisi judul atau informasi lain yang menandai atau menjelaskan konten kartu.
 - <h5>Form Login / h5> : Ini adalah elemen judul level 5 (h5) di dalam header kartu yang memberikan judul "Form Login" pada kartu.
- [card-body] pada [div class="card-body"] : Kelas ini memberikan gaya khusus untuk elemen [div] yang berisi tubuh atau konten utama kartu. Ini membantu memisahkan dan memvisualisasikan konten utama kartu.
 - <h5 class="card-title">Special title treatment / h5> : Ini adalah elemen judul level 5 (h5) di dalam tubuh kartu yang memberikan judul khusus.
 - <p class="card-text">With supporting text below as a natural lead-in to additional content. / p> : Ini adalah elemen paragraf di dalam tubuh kartu yang memberikan teks pendukung atau keterangan untuk konten utama kartu.
 - Go somewhere / a> : Ini adalah elemen anchor (tautan) yang merupakan tombol dengan kelas Bootstrap "btn" dan "btn-primary", memberikan tampilan dan warna tertentu sesuai dengan desain Bootstrap.

Perbedaan Container dengan Container-fluid:

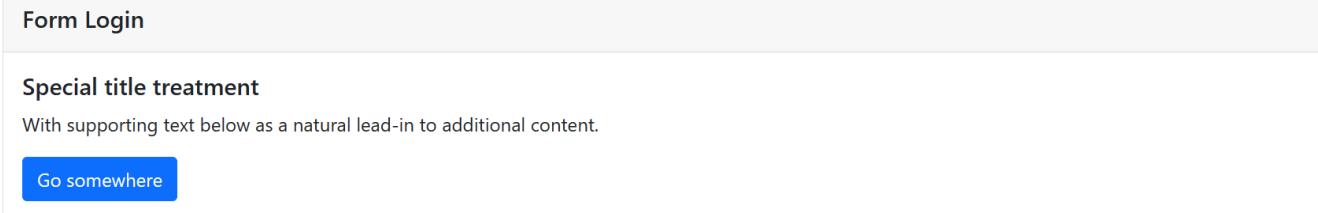
container:

- [container] memberikan tata letak yang terpusat dan memiliki lebar yang tetap (fixed-width).
- Lebar [container] diatur dalam satuan piksel dan tetap konstan terlepas dari ukuran layar pengguna.
- Digunakan untuk membuat tata letak yang terkonsentrasi di tengah halaman dan memberikan batasan pada lebar kontennya.

container-fluid:

- `container-fluid` memberikan tata letak yang penuh lebar, mengisi seluruh lebar layar.
- Lebar `container-fluid` disesuaikan dengan lebar layar pengguna, sehingga konten dapat memanfaatkan seluruh lebar tampilan tanpa batasan piksel tertentu.
- Digunakan ketika Anda ingin membuat tata letak yang menyesuaikan diri dengan lebar layar dan memberikan tampilan responsif.

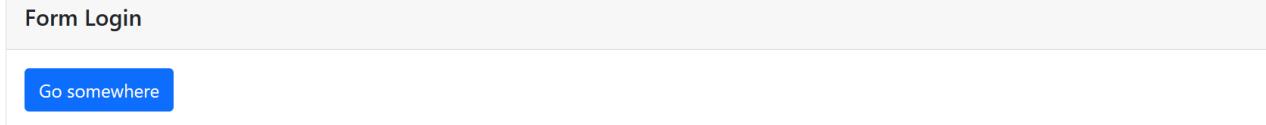
11. Hasilnya adalah sebagai berikut



12. Sekarang pada bagian dalam div card-body hapus semua elemen yang ada didalamnya kecuali button:

```
<div class="container">
  <div class="card mt-2">
    <div class="card-header">
      <h5>Form Login</h5>
    </div>
    <div class="card-body">
      <a href="#" class="btn btn-primary">Go somewhere</a>
    </div>
  </div>
</div>
```

13. Hasilnya akan seperti ini



14. Langkah selanjutnya adalah menambahkan elemen-elemen form ke dalam card-body untuk membuat formulir login. Anda dapat menggunakan elemen-elemen HTML seperti form , input , dan button untuk ini. bukalah kembali getbootstrap.com dan pilihlah form yang kita inginkan:

Email address

We'll never share your email with anyone else.

Password

Check me out

Submit

HTML

```
<form>
  <div class="mb-3">
    <label for="exampleInputEmail1" class="form-label">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">
    <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>
  </div>
```

Copy to clipboard

15. Salinlah kode program dibawah contoh forms tadi:

```
<form>
  <div class="mb-3">
    <label for="exampleInputEmail1" class="form-label">Email
address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" aria-
describedby="emailHelp">
    <div id="emailHelp" class="form-text">We'll never share your email
with anyone else.</div>
  </div>
  <div class="mb-3">
    <label for="exampleInputPassword1" class="form-label">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1">
  </div>
  <div class="mb-3 form-check">
    <input type="checkbox" class="form-check-input" id="exampleCheck1">
    <label class="form-check-label" for="exampleCheck1">Check me
out</label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

16. Tempelkanlah didalam cards-body tadi

```

<body>
  <div class="container">
    <div class="card mt-2">
      <div class="card-header">
        <h5>Form Login</h5>
      </div>
      <div class="card-body">
        <form>
          <div class="mb-3">
            <label for="exampleInputEmail1" class="form-label">Email address</label>
            <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">
            <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>
          </div>
          <div class="mb-3">
            <label for="exampleInputPassword1" class="form-label">Password</label>
            <input type="password" class="form-control" id="exampleInputPassword1">
          </div>
          <div class="mb-3 form-check">
            <input type="checkbox" class="form-check-input" id="exampleCheck1">
            <label class="form-check-label" for="exampleCheck1">Check me out</label>
          </div>
          <button type="submit" class="btn btn-primary">Submit</button>
        </form>
      </div>
    </div>
  </div>

```

17. Maka hasilnya adalah seperti ini:

The screenshot shows a simple login form titled "Form Login". It includes fields for "Email address" and "Password", a checkbox labeled "Check me out", and a blue "Submit" button.

Dengan demikian kita sudah berhasil membuat layoutcards dimana form login sebagai kontennya

Grid For Responsive Web/Layouting

Bootstrap memiliki sistem tata letak yang kuat yang bernama grid. Grid berguna untuk membuat tata letak yang responsif di halaman web kita. Sistem grid Bootstrap berdasarkan konsep kolom yang dapat diatur dalam baris. Berikut adalah beberapa hal penting yang perlu kita ketahui tentang grid dan layouting di Bootstrap:

1. KONTAINER (Container)

Grid Bootstrap harus ditempatkan dalam elemen kontainer. Terdapat dua jenis kontainer yang tersedia, yaitu .container dan .container-fluid. .container memiliki lebar terbatas dan

akan disesuaikan dengan lebar layar. .container ini digunakan ketika kita ingin membuat tata letak yang terpusat dan terbatas pada lebar tertentu. .container-fluid memiliki lebar penuh dan akan mengisi seluruh lebar layar. Kontainer ini digunakan ketika kita ingin membuat tata letak yang menyesuaikan dengan lebar layar penuh.

2. BARIS (Row)

Baris digunakan untuk mengelompokkan kolom-kolom dalam tata letak. kita dapat menambahkan class .row pada elemen yang berfungsi sebagai wadah kolom. Baris ini akan memastikan bahwa kolom-kolom di dalamnya akan diatur secara horizontal.

3. KOLOM (Column):

Kolom adalah bagian dasar dari sistem grid Bootstrap. Kolom- kolom ditempatkan di dalam baris dan digunakan untuk membagi horizontal ruang dalam grid. Setiap baris dipecah menjadi 12 kolom, yang dapat kita bagi sesuai kebutuhan.

Catatan: kita dapat menggunakan kelas seperti .col- , .col-sm- , .col-md- , .col-lg- , atau .col-xl- untuk menentukan bagaimana kolom akan berperilaku di berbagai ukuran perangkat. Misalnya, .col-sm-6 akan membuat kolom tersebut memiliki lebar setengah dari baris pada ukuran layar kecil. kita dapat menggabungkan kelas kolom untuk ukuran layar yang berbeda untuk menciptakan tata letak yang responsif. Misalnya, .col-md-6 .col-lg-4 akan membuat kolom tersebut memiliki lebar setengah dari baris pada ukuran layar medium, dan memiliki lebar sepertiga dari baris pada ukuran layar besar.

4. OFFSET

Kita dapat menggunakan kelas offset untuk memindahkan kolom ke samping. Misalnya, .offset-md-2 akan memindahkan kolom 2 satuan ke kanan pada layar dengan ukuran medium. Offset digunakan ketika Kita ingin membuat ruang kosong di antara kolom-kolom.

5. Perilaku pada Ukuran Layar yang Berbeda

Kita dapat menggunakan kelas-kelas grid Bootstrap yang berbeda untuk mengontrol tampilan kolom pada ukuran layar yang berbeda. Misalnya, Kita dapat menggunakan .colsm- untuk ukuran layar kecil, .col-md- untuk ukuran layar medium, dan sebagainya. Dengan cara ini, Kita dapat membuat tata letak yang responsif untuk berbagai perangkat.

6. NESTING

Nesting memungkinkan Kita menempatkan baris dan kolom di dalam kolom lainnya. Dengan cara ini, Kita dapat membuat tata letak yang lebih kompleks dengan komponen-komponen yang terkait. Misalnya, Kita dapat menempatkan sebuah baris di dalam kolom yang ada di dalam baris lainnya

7. ORDER

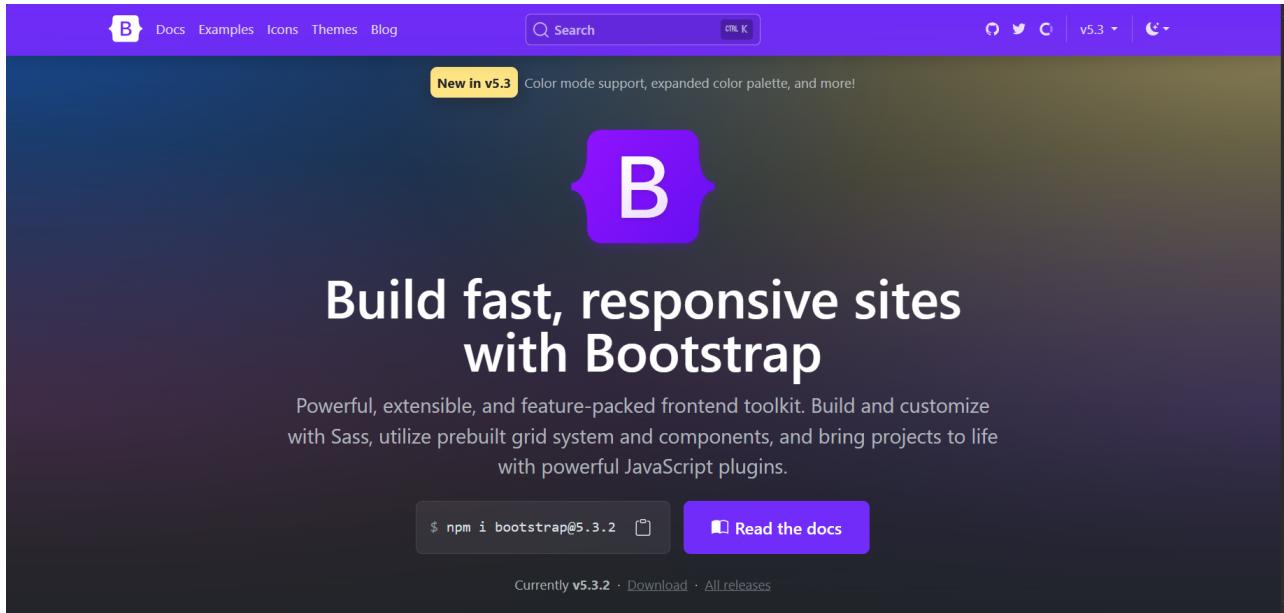
Kita dapat menggunakan kelas .order- untuk mengubah urutan kolom pada ukuran layar tertentu. Misalnya, .order-first akan memindahkan kolom ke posisi pertama, dan .orderlast akan memindahkan kolom ke posisi terakhir. Dengan menggunakan kelas ini, Kita dapat mengatur ulang urutan kolom untuk mencapai tata letak yang diinginkan pada berbagai ukuran layar.

Itu adalah beberapa konsep penting dalam sistem grid Bootstrap. Dengan menggunakan sistem grid ini, kita dapat membuat tata letak yang responsif dan menyesuaikan tampilan elemen-elemen pada berbagai ukuran layar dengan mudah.

Contoh Penggunaan dan Penjelasannya

Sekarang kita akan mencoba penggunaan grid yang dengan bantuan Bootstrap. Berikut langkah-langkahnya:

1. Masuklah kembali ke web getbootstrap.com.



2. carilah "Grid" di Kolom search lalu klik enter. Maka kita akan diarahkan pada halaman web yang berisi tentang grid.

A screenshot of the Bootstrap Grid system documentation page. The left sidebar has a dark theme with categories like Sass, Options, Color, etc., and a "Layout" section with "Grid" selected. The main content area has a title "Grid system" with a "View on GitHub" button. It includes a sub-section "Example #". A sidebar on the right lists various topics under "On this page", such as Example, How it works, Grid options, Auto-layout columns, Equal-width, Setting one column width, Variable width content, Responsive classes, All breakpoints, Stacked to horizontal, Mix and match, Row columns, Nesting, CSS, Sass variables, Sass mixins, Example usage, Customizing the grid, Columns and gutters, and Grid tiers.

3. Carilah layout atau tata letak grid yang kita inginkan. Misalnya pada praktek kali ini, kita akan menggunakan layout "Row columns".

Row columns

Use the responsive `.row-cols-*` classes to quickly set the number of columns that best render your content and layout. Whereas normal `.col-*` classes apply to the individual columns (e.g., `.col-md-4`), the row columns classes are set on the parent `.row` as a shortcut. With `.row-cols-auto` you can give the columns their natural width.

Use these row columns classes to quickly create basic grid layouts or to control your card layouts.

Column	Column
Column	Column

HTML



```
<div class="container text-center">
  <div class="row row-cols-2">
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
  </div>
</div>
```

4. Salinlah kode program yang terletak di kode bawah contohnya.

```
<div class="container text-center">
  <div class="row row-cols-2">
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
  </div>
</div>
```

5. Tempelah pada file html yang tentunya telah dihubungkan dengan tautan bootstrap, baik secara offline maupun online.

```

41 </nav>
42
43 <nav class="navbar bg-success">
44   <div class="container-fluid">
45     <form class="d-flex" role="search">
46       <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
47       <button type="button" class="btn btn-outline-light">Search</button>
48     </form>
49   </div>
50 </nav>
51
52 <div class="container text-center">
53   <div class="row row-cols-2">
54     <div class="col">Column</div>
55     <div class="col">Column</div>
56     <div class="col">Column</div>
57     <div class="col">Column</div>
58   </div>
59 </div>
60
61 </body>
62 </html>

```

6. Maka hasilnya akan seperti berikut:



Keterangan:

Jadi tulisan "Column" adalah isi konten yang sudah teratur dalam grid ini. kita bisa mengganti "konten" ini dengan konten yang kita inginkan

7. Untuk mengganti kontennya hapuslah tulisan "Column" lalu gantilah dengan konten yang kita inginkan.

```

<div class="container text-center">
  <div class="row row-cols-2">
    <div class="col">
      <div>
        
        <h3>HTML</h3>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
        <br>Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam
        impedit laboriosam earum, <br> possimus consequatur quisquam dolore ex ipsa
        eaque cupiditate neque.</p>
      </div>
    </div>
    <div class="col">
      <div>
        
      </div>
    </div>
  </div>
</div>

```

```

<h3>CSS</h3>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br>Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam
impedit laboriosam earum, <br> possimus consequatur quisquam dolore ex ipsa
eaque cupiditate neque.</p>
    </div>
</div>
<div class="col">
    <div>
        
            <h3>JavaScrip</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br>Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam
impedit laboriosam earum, <br> possimus consequatur quisquam dolore ex ipsa
eaque cupiditate neque.</p>
        </div>
    </div>
    <div class="col">
        <div>
            
                <h3>JQuery</h3>
                <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br> Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam
impedit laboriosam earum, <br> possimus consequatur quisquam dolore ex ipsa
eaque cupiditate neque.</p>
            </div>
        </div>
    </div>
</div>

```

8. Hasil nya akan menjadi seperti berikut:



Lorem ipsum dolor sit amet consectetur adipisicing elit.
 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,
 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.



Lorem ipsum dolor sit amet consectetur adipisicing elit.
 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,
 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.



Lorem ipsum dolor sit amet consectetur adipisicing elit.
 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,
 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.

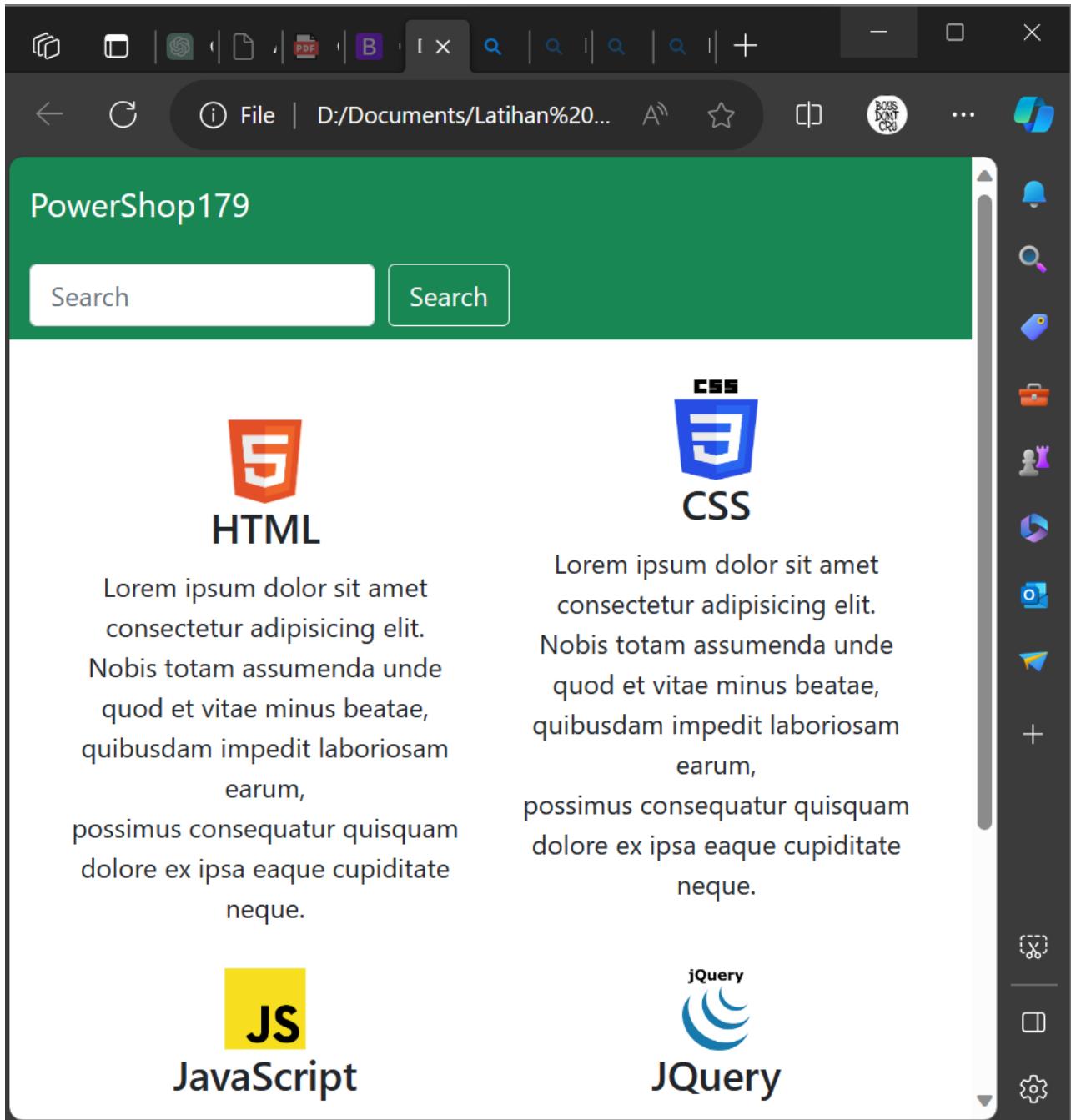


Lorem ipsum dolor sit amet consectetur adipisicing elit.
 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,
 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.

Keterangan:

Jadi Grid tadi mengatur Row dan Column dimana terdapat 2 baris berisi 4 kolom sebagai kontennya. Kontennya tadi kita ganti dengan konten yang kita inginkan

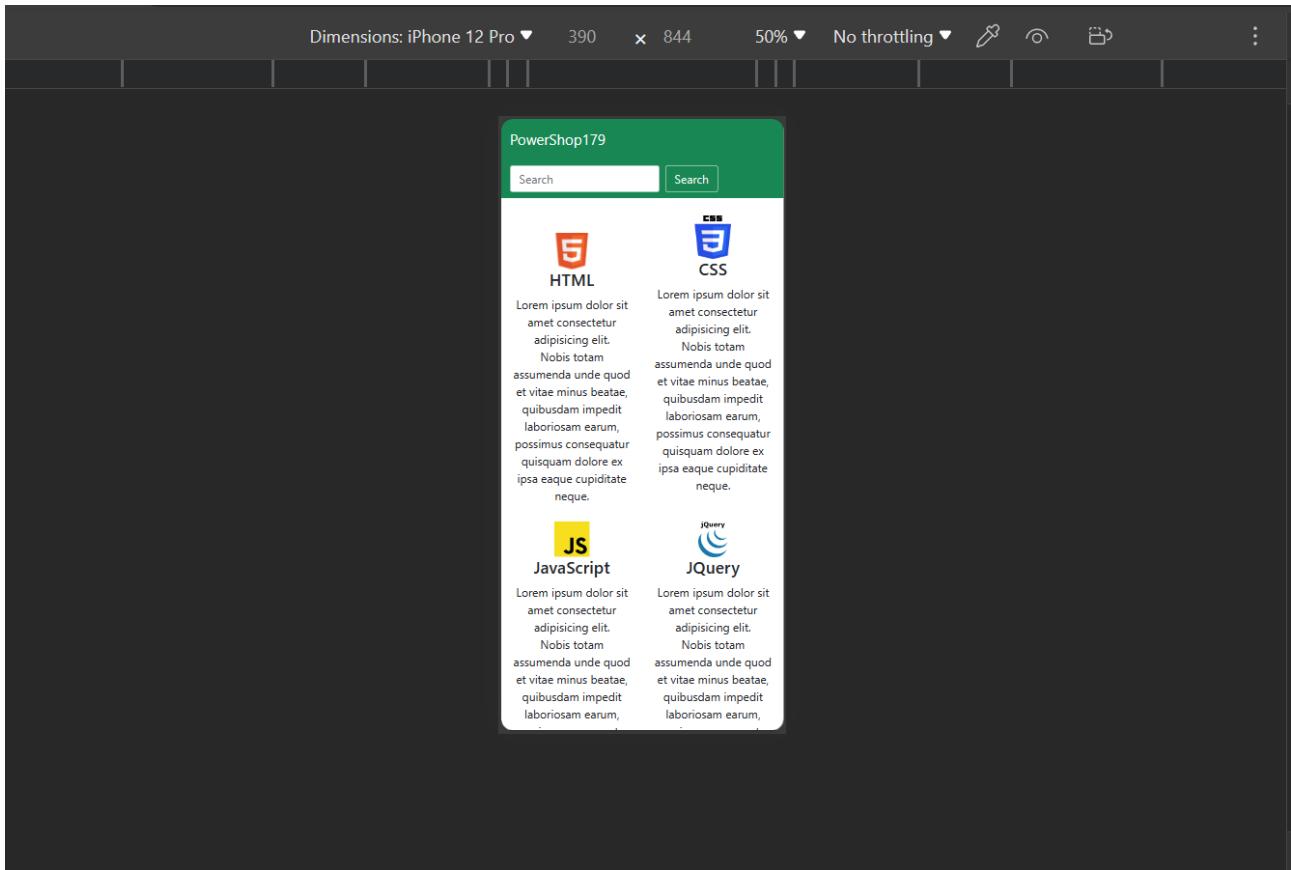
- Apabila layar yang kita kecilkan, maka grid akan menyesuaikan ukurannya seperti pada gambar ini



Keterangan:

Jadi grid ini sudah mengatur mengenai penyesuaian ukuran layar kita, sehingga, dapat menyesuaikan di segala ukuran monitor.

- Apabila kita melihatnya di perangkat, maka akan terlihat seperti ini:



Keterangan:

Jadi karena grid yang disediakan oleh bootstrap telah mendukung fitur responsive, maka ketika user melihat tampilan web kita di ponsel, maka tata letak akan mengikutinya sesuai dengan ukuran ponsel, seperti yang terlihat pada gambar di atas

Contoh Penggunaan `col-md` , `col-lg` dan `col-sm`

Kelas `col-md` , `col-lg` , dan `col-sm` adalah kelas-kelas kolom dalam Bootstrap yang digunakan untuk meresponsifkan tata letak halaman web sesuai dengan ukuran layar perangkat. Bootstrap menggunakan sistem grid yang terdiri dari 12 kolom, dan kelas-kelas ini memungkinkan Anda mengontrol sejauh mana elemen-elemen tersebut harus meluas pada layar yang berbeda.

- `col-sm` : Digunakan untuk layar kecil (small), seperti pada perangkat mobile atau tablet dalam orientasi potret. Kolom ini akan berlaku untuk layar dengan lebar 576px atau lebih.
- `col-md` : Digunakan untuk layar sedang (medium), yang mencakup perangkat seperti tablet dalam orientasi landscape. Kolom ini akan berlaku untuk layar dengan lebar 768px atau lebih.
- `col-lg` : Digunakan untuk layar besar (large), yang mencakup desktop. Kolom ini akan berlaku untuk layar dengan lebar 992px atau lebih.

Contoh:

Sekarang kita akan coba untuk membuat sebuah tampilan dengan menggunakan class diatas tadi. Berikut contohnya:

1. Bukalah kembali web getbootstrap lalu di kolom pencarian carilah "grid".

The screenshot shows the Bootstrap documentation homepage. A search bar at the top contains the word 'grid'. Below it, a sidebar on the left lists various documentation sections like Webpack, Parcel, Vite, Accessibility, RFS, RTL, Contribute, Customize, Overview, Sass, Options, Color, Color modes, Components, CSS variables, Optimize, Layout, Breakpoints, Containers, Grid, Columns, and Grids. The main content area features a large heading 'Get started with Bootstrap' and several sections: 'Grid system' (with 'Example' selected), 'How it works', 'Grid options', 'Auto-layout columns', and 'Responsive classes'; and 'CSS Grid' (with 'How it works' selected). On the right side, there's a sidebar titled 'On this page' with links to 'Quick start', 'CDN links', 'Next steps', 'JS components', 'Important globals', 'HTML5 doctype', 'Viewport meta', 'Box-sizing', 'Reboot', and 'Community'. The footer includes a copyright notice and a link to 'View on GitHub'.

2. Cari dan copylah kembali codingan grid yang kita gunakan pada praktik sebelumnya.

The screenshot shows the 'Row columns' section of the Bootstrap documentation. It features a title 'Row columns' and a paragraph explaining how to use responsive .row-cols-* classes to set the number of columns. It contrasts these with individual .col-* classes. Below this, a heading says 'Use these row columns classes to quickly create basic grid layouts or to control your card layouts.' A visual representation shows a 2x2 grid of four purple boxes labeled 'Column' in white. Below this, an 'HTML' code block displays the following code:

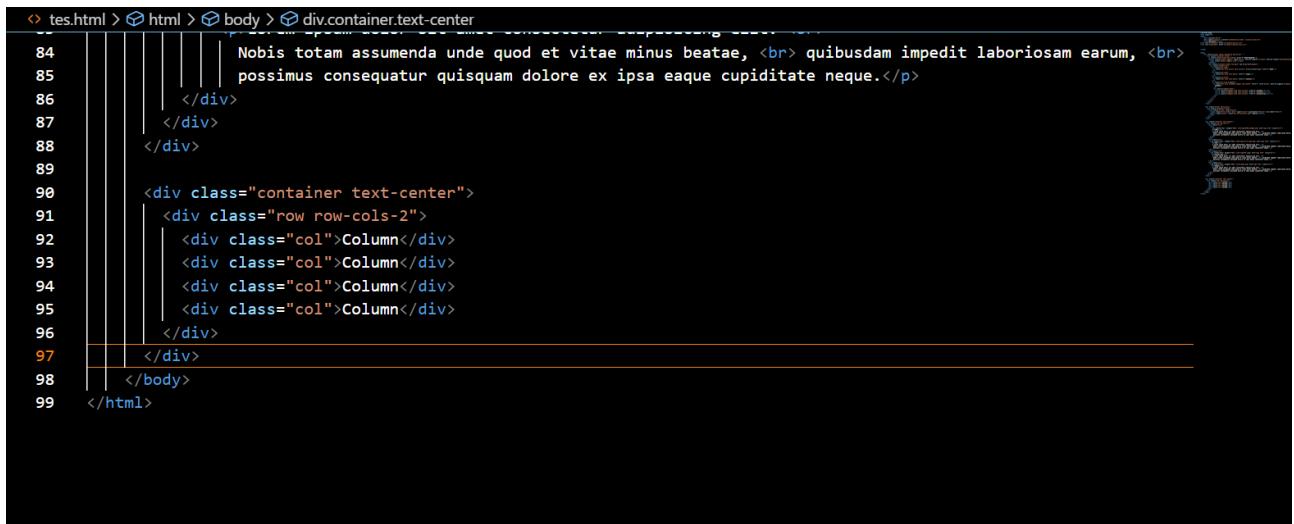
```
<div class="container text-center">
  <div class="row row-cols-2">
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="col">Column</div>
  </div>
</div>
```

```

<div class="container text-center">
    <div class="row row-cols-2">
        <div class="col">Column</div>
        <div class="col">Column</div>
        <div class="col">Column</div>
        <div class="col">Column</div>
    </div>
</div>

```

3. Tempelkanlah codingan tadi di visual studio kode pada file tempat kita mempraktekkan grid pada praktikum sebelumnya, agar kita bisa menggunakan satu file saja untuk praktek.



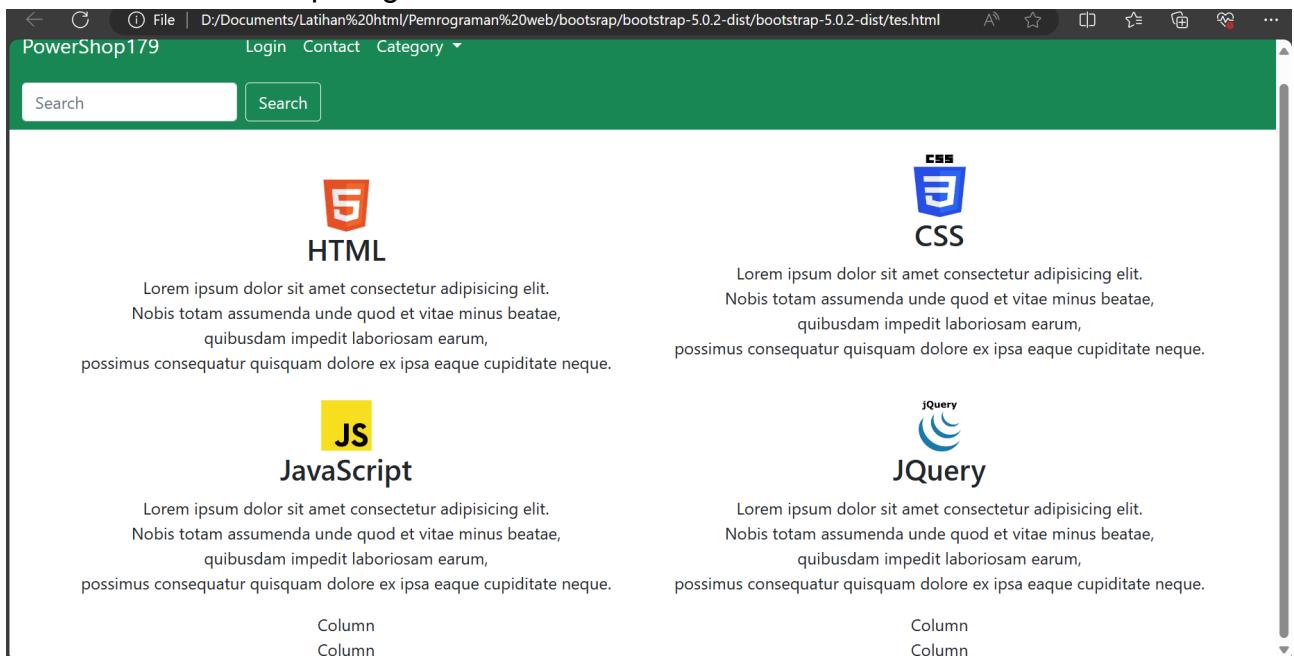
The screenshot shows a code editor window with the file 'tes.html' open. The code is a simple HTML structure with a container and a row of four columns. The code is as follows:

```

<div class="container text-center">
    <div class="row row-cols-2">
        <div class="col">Column</div>
        <div class="col">Column</div>
        <div class="col">Column</div>
        <div class="col">Column</div>
    </div>
</div>

```

4. Maka akan terlihat seperti gambar di bawah ini:



Keterangan:

Perhatikanlah di bawah kolom JavaScript dan JQuery, terdapat tulisan column sebanyak 4

dan bersusun hal ini dikarenakan pada kodingan tadi, teks yang dipakai sebagai template di bootstrap adalah "Column", nanti pada tahap selanjutnya kita akan mengganti "Column" itu dengan text atau item yang kita inginkan.

5. Sekarang, pada codingan yang kita ambil dari web bootstrap tadi pada bagian `<div class="col">` yang pertama, tambahkanlah -md sehingga menjadi `<div class="col-md">`. Kemudian pada `<div class="col">` yang kedua, tambahkanlah -lg sehingga menjadi `<div class="col-lg">`, dan terakhir pada `<div class="col">` yang ketiga, tambahkanlah -sm sehingga menjadi `<div class="col-sm">`, sementara pada `<div class="col">` yang terakhir kita biarkan saja seperti bawaan.

```
<div class="container text-center">
  <div class="row row-cols-2">
    <div class="col-md">Column</div>
    <div class="col-lg">Column</div>
    <div class="col-sm">Column</div>
    <div class="col">Column</div>
  </div>
</div>
```

The screenshot shows a web page with a green header containing a search bar and navigation links for PowerShop179, Login, Contact, and Category. The main content area has a container with a row of four columns. The first column (col-md) contains an HTML icon and text. The second column (col-lg) contains a CSS icon and text. The third column (col-sm) contains a JavaScript icon and text. The fourth column (col) contains a jQuery icon and text. Each column section includes placeholder text from the Bootstrap documentation.

Column	Column	Column	Column
HTML Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.	CSS Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.	JavaScript Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.	JQuery Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.

6. Maka hasilnya akan menjadi seperti ini:

Sebelum:

Column
Column

Column
Column

Sesudah:

Keterangan:

- `col-md` : Dengan menambahkan -md pada codingan tadi akan membuat kolom tersebut mengambil sebagian besar lebar tersedia (12 kolom) pada layar dengan lebar medium (768px atau lebih).
- `col-lg` : Sama seperti col-md , tetapi ditujukan untuk layar lebar (large, 992px atau lebih).
`col-sm` : Sama seperti col-md , tetapi ditujukan untuk layar kecil (small, 576px atau lebih).

7. Tambahkanlah teks ataupun item dari yang kita inginkan dengan cara hapuslah teks "Column" tadi, dan ditempat teks "Column" yang sudah dihapus, tempelkan lah item atau teks yang kita inginkan

```
<div class="container text-center">
  <div class="row row-cols-2">
    <div class="col-md">
      <div>
        
        <h3>HTML</h3>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br>
          Nobis totam assumenda unde quod et vitae minus beatae,
<br> quibusdam impedit laboriosam earum, <br>
          possimus consequatur quisquam dolore ex ipsa eaque
cupiditate neque.</p>
      </div>
    </div>
    <div class="col-lg">
      <div>
        
        <h3>CSS</h3>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br>
          Nobis totam assumenda unde quod et vitae minus beatae,
<br> quibusdam impedit laboriosam earum, <br>
          possimus consequatur quisquam dolore ex ipsa eaque
cupiditate neque.</p>
      </div>
    </div>
    <div class="col-sm">
      <div>
```

```

        
        <h3>JavaScript</h3>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br>
        Nobis totam assumenda unde quod et vitae minus beatae,
<br> quibusdam impedit laboriosam earum, <br>
        possimus consequatur quisquam dolore ex ipsa eaque
cupiditate neque.</p>
        </div>
    </div>
    <div class="col">
        <div>
            
            <h3>JQuery</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
<br>
        Nobis totam assumenda unde quod et vitae minus beatae,
<br> quibusdam impedit laboriosam earum, <br>
        possimus consequatur quisquam dolore ex ipsa eaque
cupiditate neque.</p>
            </div>
        </div>
    </div>
</div>

```

The screenshot shows the browser's developer tools with the 'Element' tab selected. The address bar at the top lists several files: 'Welcome', 'home.html', 'tes.html' (which is the active tab), 'bootstrap.min.css 4', and 'login.html'. The main area displays the DOM structure of the 'tes.html' page. The code is color-coded for syntax highlighting. The DOM structure includes a container with two columns ('col-md') and two rows ('row-cols-2'). Each card in the grid has a logo image and a title ('HTML', 'CSS', 'JavaScript') followed by a paragraph of placeholder text ('Lorem ipsum...'). The right side of the developer tools interface shows a detailed view of the current element being inspected.

```

<div class="container text-center">
    <div class="row row-cols-2">
        <div class="col-md">
            <div>
                
                <h3>HTML</h3>
                <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. <br>
                    Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam impedit laboriosam earum, <br>
                    possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
            </div>
        </div>
        <div class="col-lg">
            
            <h3>CSS</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. <br>
                Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam impedit laboriosam earum, <br>
                possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
        </div>
        <div class="col-sm">
            
            <h3>JavaScript</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. <br>
                Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam impedit laboriosam earum, <br>
                possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
        </div>
        <div class="col">
            
            <h3>JQuery</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. <br>

```

8. Maka hasilnya akan seperti ini:

 HTML	 CSS	 JavaScript	 jQuery
<p> Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>	<p> Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>	<p> Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>	<p> Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>

Keterangan:

Jadi pada praktek sebelumnya tanpa adanya tambahan pada class bootstrap tadi maka elemen akan menyesuaikan lebar pada pembungkus menjadi full, sedangkan untuk praktek yang saat ini kita tampilkan, item dan teks nya menyesuaikan ukurannya sesuai dengan tambahannya. Untuk penjelasan ukurannya **Penjelasannya telah kami sediakan di langkah ke-6.**

Penggunaan Tema

Dalam Bootstrap, tema dasarnya terdiri dari sejumlah variabel CSS yang mengontrol aspek-aspek seperti warna, ukuran teks, jarak, dan banyak lagi. Untuk melakukan kostumisasi tema, kita dapat mengganti nilai-nilai variabel ini sesuai dengan preferensi kita.

Berikut adalah langkah-langkah umum untuk melakukan kostumisasi tema Bootstrap:

- Menentukan gaya desain:** Pertama, tentukan gaya desain yang ingin kita terapkan pada situs web kita. Apakah kita ingin tampilan yang lebih minimalis, warna yang cerah, atau mungkin tampilan yang lebih kustom dengan elemen desain yang unik? Menentukan gaya desain akan membantu kita memutuskan bagaimana mengubah tema Bootstrap.
- Menggunakan Sass atau CSS:** Bootstrap menyediakan versi Sass (Syntactically Awesome Style Sheets) dari file sumbernya. Sass adalah bahasa pemrograman yang memungkinkan kita menulis CSS dengan sintaks yang lebih kuat dan fleksibel. Jika kita memiliki pengetahuan tentang Sass, kita dapat mengunduh versi Sass Bootstrap dan mengedit variabel-variabelnya. Jika kita tidak familiar dengan Sass, kita masih dapat mengedit file CSS Bootstrap langsung.
- Membuat file kustom:** Buat file kustom yang akan berisi kostumisasi tema kita. Dalam file ini, kita dapat menetapkan nilai-nilai variabel yang ingin kita ubah. Misalnya, kita dapat mengubah warna primer, warna latar belakang, ukuran huruf, dan lain sebagainya.

4. **Mengganti variabel:** Temukan variabel yang relevan dalam file sumber Bootstrap dan ubah nilainya sesuai dengan preferensi kita. Misalnya, jika kita ingin mengubah warna primer, cari variabel `$primary-color` atau sejenisnya dalam file sumber dan ubah nilainya menjadi warna yang diinginkan.
5. **Mengompilasi tema kustom:** Setelah kita selesai mengedit file kustom, kita perlu mengompilasinya ke dalam file CSS yang dapat digunakan pada situs web kita. Jika kita menggunakan Sass, kita perlu mengompilasi file Sass menjadi CSS . Jika kita mengedit file CSS langsung, kita dapat menggunakan file tersebut langsung.
6. **Menerapkan tema kustom:** Setelah kita memiliki file CSS tema kustom, kita perlu menggantikan file Bootstrap default dengan file tema kustom kita pada situs web kita. Pastikan untuk menghubungkan file CSS tema kustom kita setelah file Bootstrap default sehingga kostumisasi kita akan ditimpa pada tema default. Kita dapat membuat tema kostuminasi Bootstrap yang sesuai dengan kebutuhan dan gaya desain kita. Penting untuk diingat bahwa jika kita menggunakan versi Bootstrap yang lebih baru, beberapa variabel atau struktur file mungkin telah berubah.

Kita dapat membuat tema kostuminasi Bootstrap yang sesuai dengan kebutuhan dan gaya desain kita. Penting untuk diingat bahwa jika kita menggunakan versi Bootstrap yang lebih baru, beberapa variabel atau struktur file mungkin telah berubah.

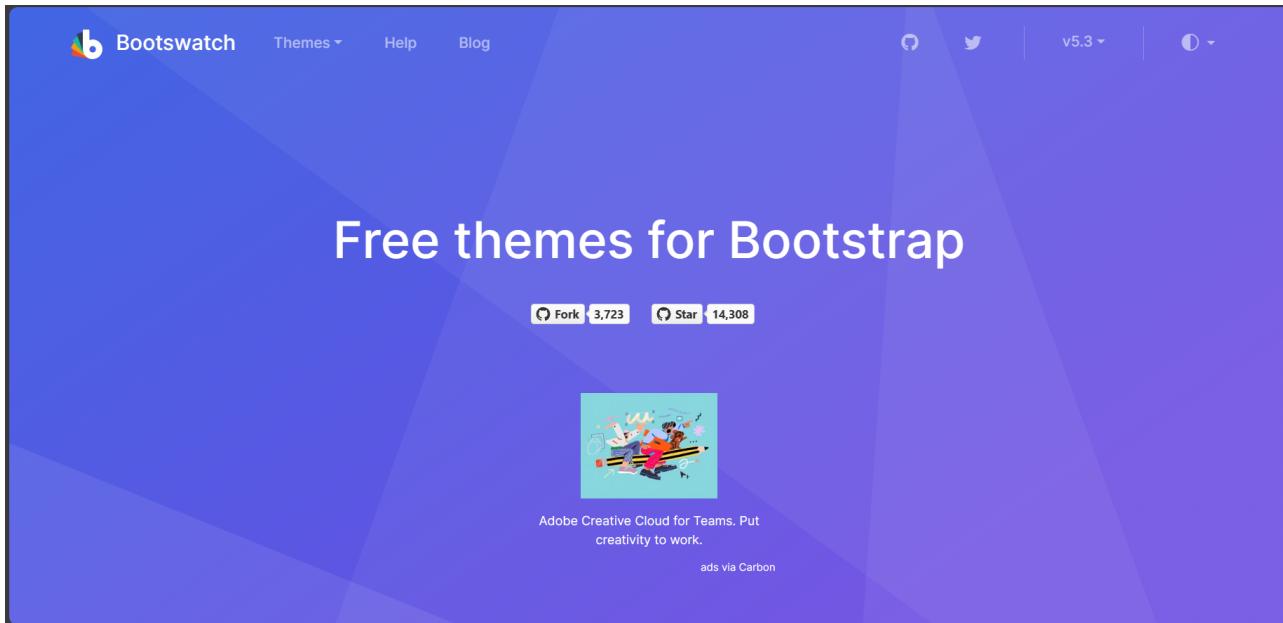
Contoh Penggunaan Tema dalam Bootstrap

1. Jadi misalnya kita ingin memberikan tema pada website yang telah kita buat sebelumnya

The screenshot shows a website with a green header bar. The header contains the text "PowerShop179" and navigation links for "Login", "Contact", and "Category". Below the header is a search bar with a "Search" button. The main content area features four cards arranged in a grid. Each card has a title icon and some placeholder text.

- HTML Card:** Features the HTML logo (a red square with white text) and the word "HTML". Below it is placeholder text: "Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque."
- CSS Card:** Features the CSS logo (a blue square with white text) and the word "CSS". Below it is placeholder text: "Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque."
- JS Card:** Features the JS logo (a yellow square with white text) and the word "JavaScript". Below it is placeholder text: "Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque."
- JQuery Card:** Features the JQuery logo (a blue circular icon with white text) and the word "JQuery". Below it is placeholder text: "Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque."

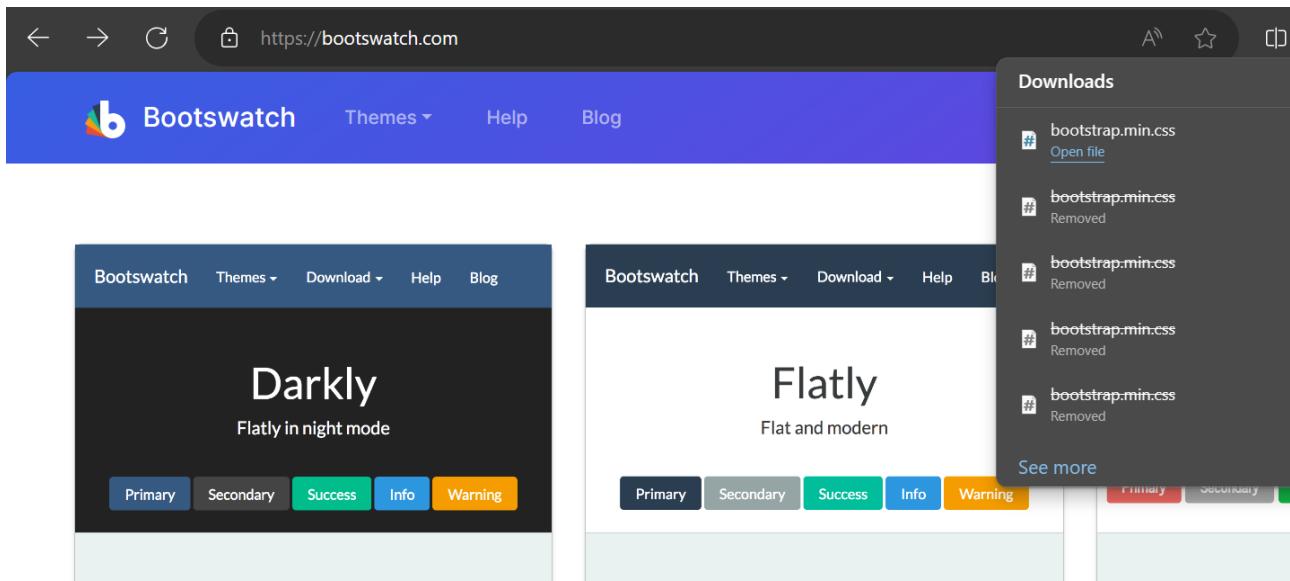
2. Bukalah website bootswatch.com dimana website ini menyediakan tema tema yang dapat digunakan untuk web bootstrap kia.



3. Scroll lah ke bawah hingga kita menemukan kumpulan tema tema yang telah disediakan oleh website ini

This screenshot shows a scroll view of the Bootswatch theme library. It displays three cards for different themes: 'Darkly', 'Flatly', and 'Journal'. Each card includes a preview image, a title, a subtitle, a color palette, and download buttons for 'Preview' and 'Download'. The 'Darkly' card is currently selected, showing its dark mode preview. The URL 'https://bootswatch.com/flatly/' is visible at the bottom left of the scroll area.

4. Misalnya kita akan memilih tema "Darkly", oleh karena itu, klik download pada bagian bawah contoh tema tersebut.



5. Jika sudah mendownloadnya maka kita telah mendapatkan file css untuk tema tersebut

Name	Date modified	Type	Size
css	10/22/2023 5:04 PM	File folder	
img	11/15/2023 4:38 PM	File folder	
js	10/22/2023 5:04 PM	File folder	
home	11/15/2023 1:01 AM	Microsoft Edge HTM...	12 KB
login	11/15/2023 4:04 PM	Microsoft Edge HTM...	1 KB
tes	11/16/2023 4:13 AM	Microsoft Edge HTM...	6 KB
# bootstrap.min	11/16/2023 4:16 AM	CSS Source File	229 KB

6. Pindahkanlah file css tersebut ke dalam folder yang berisi file html yang akan kita berikan tema

Name	Date modified	Type	Size
css	10/22/2023 5:04 PM	File folder	
img	11/15/2023 4:38 PM	File folder	
js	10/22/2023 5:04 PM	File folder	
home	11/15/2023 1:01 AM	Microsoft Edge HTM...	12 KB
login	11/15/2023 4:04 PM	Microsoft Edge HTM...	1 KB
tes	11/16/2023 4:13 AM	Microsoft Edge HTM...	6 KB
# bootstrap.min	11/16/2023 4:16 AM	CSS Source File	229 KB

7. Sekarang, masuklah ke dalam file html yang akan kita berikan tema bootstrap, lalu panggilah secara eksternal file css tema tadi ke dalam file html tersebut

```

<tes.html> </html> </head> </link>
1   <!DOCTYPE html>
2   <html lang="en">
3     <head>
4       <meta charset="UTF-8">
5       <meta name="viewport" content="width=device-width, initial-scale=1.0">
6       <title>Document</title>
7       <link rel="stylesheet" href="css/bootstrap.min.css">
8       <link rel="stylesheet" href="css/bootstrap-grid.min.css">
9       <link rel="stylesheet" href="b">
10      # bootstrap.min.css
11    </head>

```

8. Maka hasilnya adalah sebagai berikut dengan demikian kita sudah berhasil memberikan tema gelap pada tampilan html kita menggunakan tema dark pada bootstrap

The screenshot shows a dark-themed web page with a green header. The header includes a navigation bar with 'PowerShop179' and links for 'Login', 'Contact', and 'Category'. Below the header, there are four cards arranged in a grid. Each card has a dark background with white text. The first card is for 'HTML' with a red '5' icon, the second for 'CSS' with a blue '3' icon, the third for 'JS' with a yellow 'S' icon, and the fourth for 'JQuery' with a blue circular icon. Each card contains a brief description of its purpose.

Card	Icon	Title	Description
HTML	Red '5'	HTML	Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.
CSS	Blue '3'	CSS	Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.
JS	Yellow 'S'	JavaScript	Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum,
JQuery	Blue circular icon	JQuery	Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum,

dengan demikian kita sudah berhasil memberikan tema gelap pada tampilan html kita menggunakan tema dark pada bootstrap

Modifikasi Template Bootstrap

Kostumasisasi dalam bootstrap adalah menggunakan template bootstrap kemudian kita akan memodifikasi template tersebut sesuai dengan keinginan kita.

Template dalam Bootstrap adalah struktur dasar atau kerangka kerja yang telah dirancang sebelumnya dan dapat digunakan sebagai dasar untuk membangun halaman web. Bootstrap adalah sebuah framework front-end yang populer digunakan untuk pengembangan web responsif. Dalam konteks Bootstrap, template menyediakan struktur HTML dan gaya CSS dasar yang dapat digunakan sebagai dasar untuk membangun halaman web yang responsif dan mudah diatur.

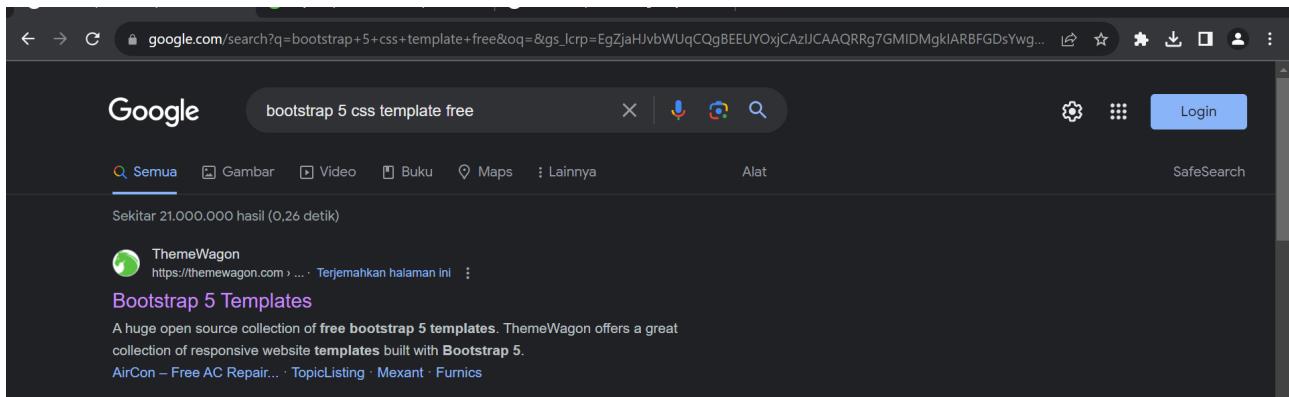
Dengan menggunakan template Bootstrap, kita dapat menghemat waktu dan usaha dalam merancang tata letak dan gaya dasar dari suatu proyek web. Template ini biasanya sudah mencakup elemen-elemen umum seperti navigasi, grid system, tombol, formulir, dan banyak

lagi. Dengan cara ini, kita dapat fokus pada konten dan fitur unik dari situs web kita tanpa harus memulai dari awal.

Template dalam Bootstrap juga mendukung konsep responsivitas, yang berarti halaman web atau aplikasi yang dibangun dengan menggunakan template ini dapat menyesuaikan diri dengan berbagai ukuran layar, mulai dari perangkat seluler hingga desktop, tanpa perlu penyesuaian tambahan. Ini membuat pengembangan web lebih efisien dan memastikan pengalaman pengguna yang konsisten di berbagai perangkat.

Berikut cara mengkostumisasi template bootstrap:

1. Pertama adalah kita akan membuka sebuah website yang menyediakan template bootstrap. Kali ini web akan kita gunakan adalah themewagon.com



2. Masuklah ke dalam website tersebut

The screenshot shows the themewagon.com website's main theme catalog. At the top, there's a navigation bar with links for Admin & Dashboard, Bootstrap 5, eCommerce, Tailwind CSS, Landing Pages, Business & Corporate, Portfolio, Educational, and a prominent 'Bundle — Save 80%' button. Below the navigation is a search bar with the placeholder 'shop'. The main content area is titled 'Themes' and features a grid of template previews. Each preview includes a thumbnail image, the template name, a download count (e.g., 4931 Downloads for MultiShop), and a five-star rating. The templates shown are MultiShop, CAKEZONE, E Shopper, Zay, and Aroma.

3. Carilah template yang ingin kita sesuaikan dengan cara mensearch ataupun menscroll hingga menemukan tema yang sesuai

themeWAGON Shop

Browse Themes Premium Freebies [Sign in](#) [Sign up](#) [Hire us](#)

Admin & Dashboard Bootstrap 5 eCommerce Tailwind CSS Landing Pages Business & Corporate Portfolio Educational [Bundle — Save 80%](#)

Browse Themes Clear filter Free (20) Premium (0) Filter (3) Relevance

4. Jika sudah menemukan template yang kita inginkan, masuk ke template tersebut lalu klik download pada bagian kanan sebelah, contoh gambar dibawah ini.

themeWAGON Search themes...

Browse Themes Premium Freebies [Sign in](#) [Sign up](#) [Hire us](#)

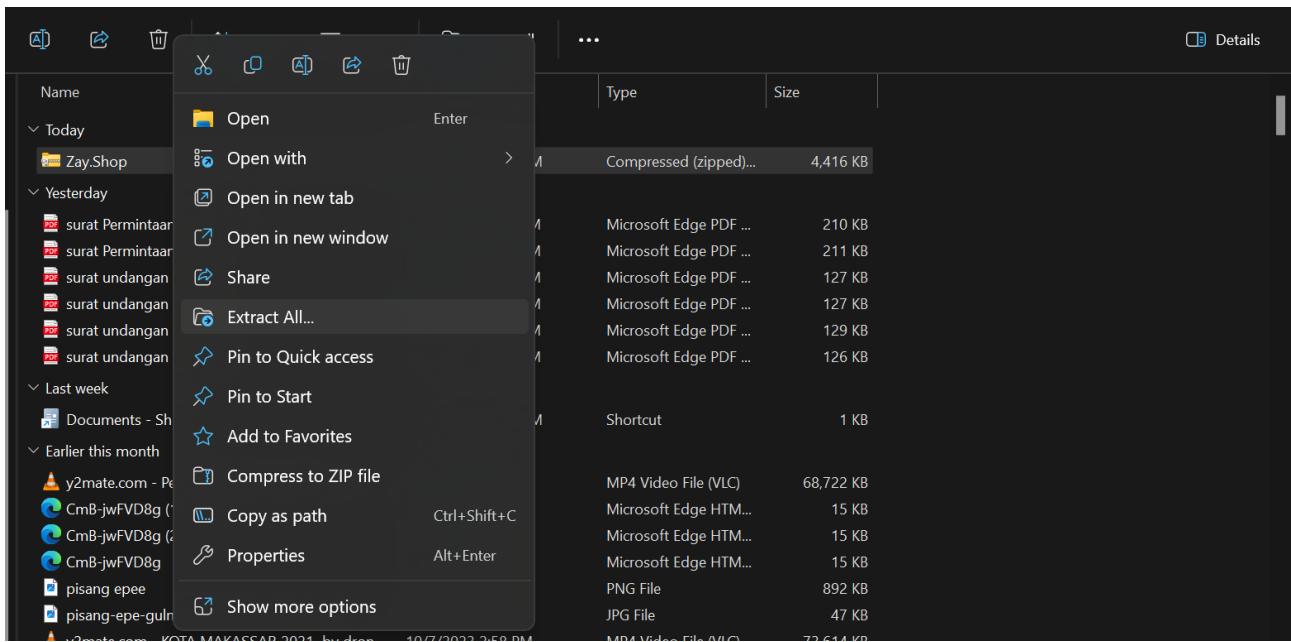
Admin & Dashboard Bootstrap 5 eCommerce Tailwind CSS Landing Pages Business & Corporate Portfolio Educational [Bundle — Save 80%](#)

Zay Shop – Free Bootstrap 5 HTML5 Responsive Ecommerce Website Template

★★★★★ 0 customers reviews

5. Jika sudah mendownloadnya maka kita telah mendapatkan file template bootstrap kita dalam bentuk zip. File ini berisi asset gambar, style css, index tml, maupun javascript yang menyusun komponen template ini

6. Ekstraklah file tersebut hingga berbentuk seperti file biasa dengan cara klik kanan pada file zip tadi lalu pilih "Extract All".



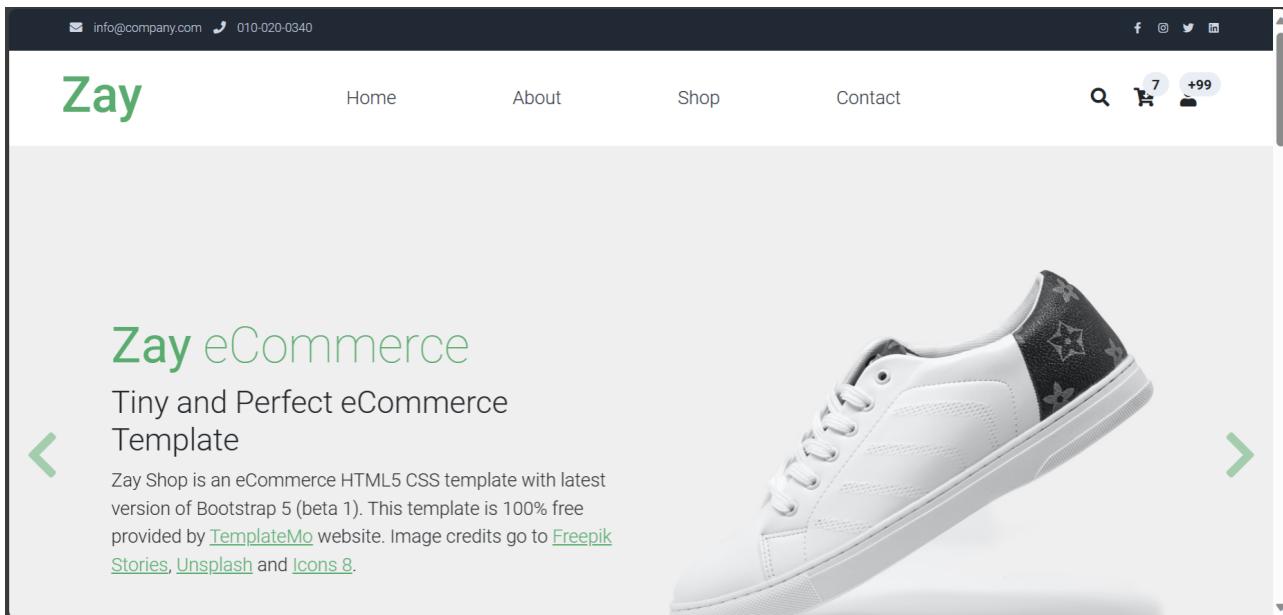
7. Jika proses ekstrak sudah selesai maka kita telah mendapatkan folder template tersebut dalam bentuk folder biasa



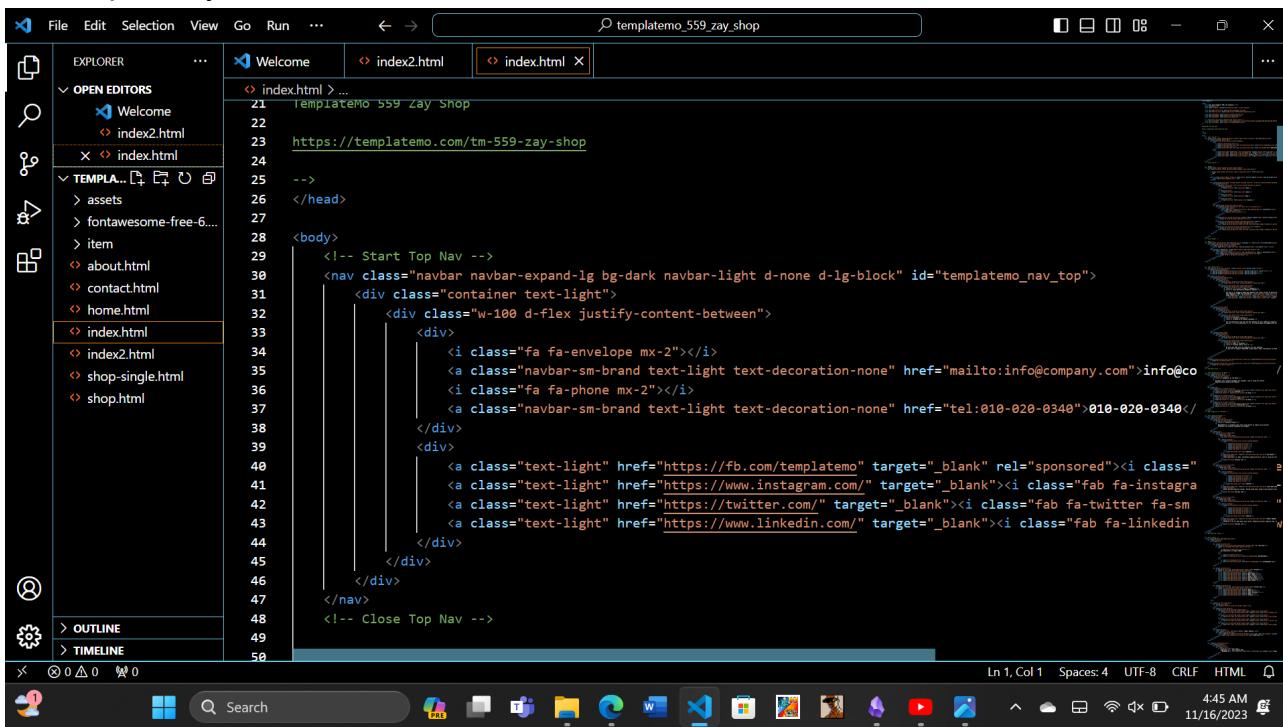
8. Masuklah ke dalam folder template yang telah diekstrak tadi

Name	Date modified	Type	Size
assets	10/18/2023 4:42 AM	File folder	
fontawesome-free-6.4.2-web	10/18/2023 9:42 AM	File folder	
item	10/18/2023 8:50 AM	File folder	
about	10/18/2023 4:42 AM	Microsoft Edge HTM...	21 KB
contact	10/18/2023 4:42 AM	Microsoft Edge HTM...	16 KB
home	10/23/2023 4:20 AM	Microsoft Edge HTM...	8 KB
index	10/18/2023 4:42 AM	Microsoft Edge HTM...	24 KB
index2	10/18/2023 8:58 AM	Microsoft Edge HTM...	1 KB
shop	10/18/2023 4:42 AM	Microsoft Edge HTM...	49 KB
shop-single	10/18/2023 4:42 AM	Microsoft Edge HTM...	58 KB

9. Masuklah ke dalam folder index.html. didalam sana terdapat tampilan template kita



10. Sekarang kita akan mengkostumasisasi tampilan web template ini. Caranya adalah buka file template nya di visual studio code



11. Mulailah mengganti konten-konten seperti teks, gambar dan lainnya seperti yang kita inginkan. Misalnya saya akan mengganti teks yang ada di gambar ini



12. Buka kembali visual studio dan masuk ke file index.html tersebut lalu carilah penyusun konten itu

```
:class="fa fa-envelope mx-2">></i>
:class="navbar-sm-brand text-light text-decoration-none" href="mailto:info@company.com">info@company.com</a>
:class="fa fa-phone mx-2">></i>
:class="navbar-sm-brand text-light text-decoration-none" href="tel:010-020-0340">010-020-0340</a>
```

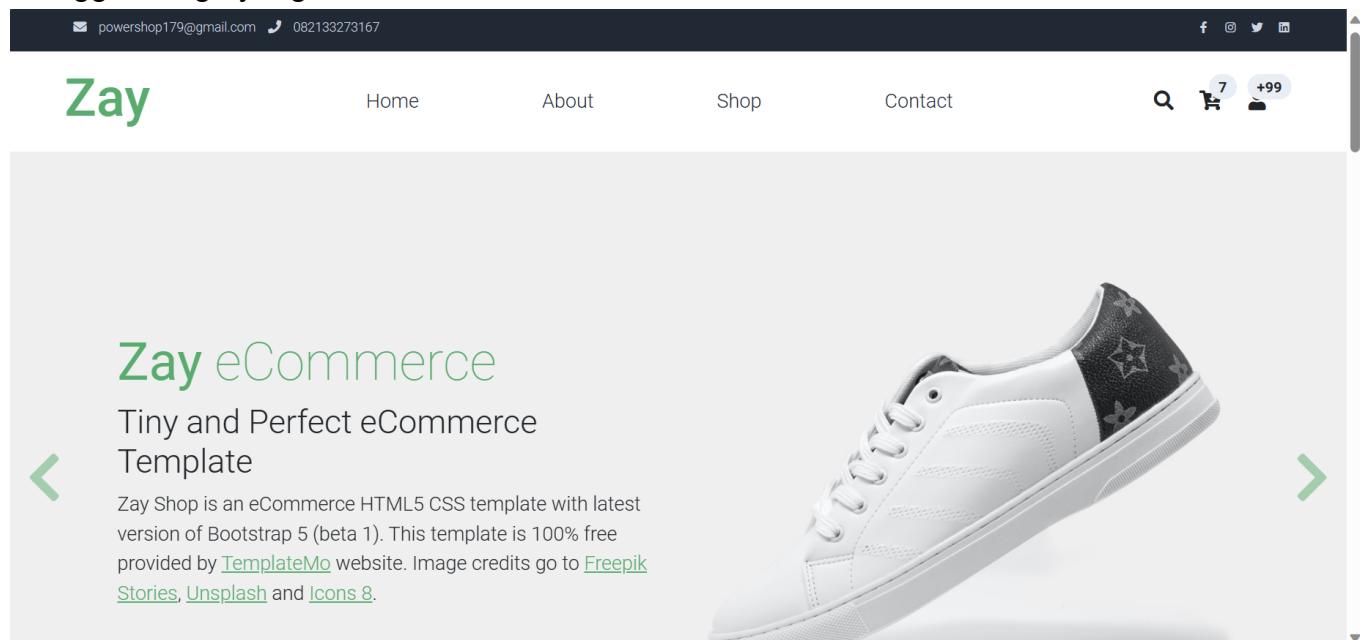
13. Gantilah kalimat dan tujuan email dan nomor telp itu dengan yang kita inginkan

```
<a class="navbar-sm-brand text-light text-decoration-none"  
href="mailto:powershop179@gmail.com">powershop179@gmail.com</a>  
<i class="fa fa-phone mx-2"></i>  
<a class="navbar-sm-brand text-light text-decoration-none"  
href="tel:082133273167">082133273167</a>
```

Maka hasilnya akan terlihat seperti dibawah ini:



14. Mulailah mengganti elemen-elemen lainnya sesuai keinginan kita. Misalnya kita ingin juga mengganti logo yang ada di website ini



```
<a class="navbar-brand text-success logo h1 align-self-center"  
href="index.html">  
Powershop179  
</a>
```

15. Hasil setelah diganti adalah seperti ini

Zay eCommerce

Tiny and Perfect eCommerce Template

Zay Shop is an eCommerce HTML5 CSS template with latest version of Bootstrap 5 (beta 1). This template is 100% free provided by [TemplateMo](#) website. Image credits go to [Freepik Stories](#), [Unsplash](#) and [Icons 8](#).



16. Sekarang kita akan mengganti Nama web kita yang ada di slide carousel

Zay eCommerce

Tiny and Perfect eCommerce Template

Zay Shop is an eCommerce HTML5 CSS template with latest version of Bootstrap 5 (beta 1). This template is 100% free provided by [TemplateMo](#) website. Image credits go to [Freepik Stories](#), [Unsplash](#) and [Icons 8](#).



```
<div class="col-lg-6 mb-0 d-flex align-items-center">
  <div class="text-align-left align-self-center">
    <h1 class="h1 text-success"><b>PowerShop179</b> eCommerce</h1>
    <h3 class="h2">Tiny and Perfect eCommerce Template</h3>
    <p>
      PowerShop179 is an eCommerce HTML5 CSS template with the latest
      version of Bootstrap 5 (beta 1).
      This template is 100% free provided by <a rel="sponsored"
      class="text-success" href="https://templatemo.com"
      target="_blank">TemplateMo</a> website. Image credits go to
      <a rel="sponsored" class="text-success"
      href="https://stories.freepik.com/" target="_blank">Freepik Stories</a>,
      <a rel="sponsored" class="text-success"
      href="https://unsplash.com/" target="_blank">Unsplash</a> and
      <a rel="sponsored" class="text-success" href="https://icons8.com/"
      target="_blank">Icons 8</a>.
    </p>
  </div>
```

```
</div>
```

Hasilnya akan menjadi seperti ini:

The screenshot shows a clean, modern eCommerce website layout. At the top, there's a navigation bar with links for Home, About, Shop, and Contact, along with a search icon, a shopping cart icon showing 7 items, and a user icon with +99 notifications. The main header features the brand name "Powershop179" in a large, bold, green font. Below the header, there's a large central image of a white sneaker with black star patterns on the side. To the left of the shoe, there's a green left arrow icon and the text "Tiny and Perfect eCommerce Template". To the right, there's a green right arrow icon. The background of the main content area is light gray.

17. bagaimana jika kita ingin mengganti gambar? Misalnya kita akan mengganti gambar yang ada di card ini

The screenshot displays three product cards in a grid format. Each card includes an image, a title, a short description, a rating, a price, and a review count. The products are:

- Gym Weight**: An image of a silver cylindrical weight. Rating: 4 stars. Price: \$240.00. Reviews: (24).
- Cloud Nike Shoes**: An image of a black Apple Watch with a red woven strap. Rating: 4 stars. Price: \$480.00. Reviews: (48).
- Summer Addides Shoes**: An image of a Polaroid OneStep+ camera. Rating: 5 stars. Price: \$360.00. Reviews: (74).

18. Masuklah ke visual studio code dan ke index.html tadi lalu carilah kode penyusun untuk gambar ini, kemudian gantilah di bagian src gambar yang sudah ada dengan gambar yang kita inginkan

```
<a href="shop-single.html">  

```


Maka hasilnya akan menjadi seperti ini:



★★★☆☆

\$240.00

Gym Weight

Lore ipsum dolor sit amet, consectetur adipisicing elit. Sunt in culpa qui officia deserunt.

Reviews (24)



★★★☆☆

\$480.00

Cloud Nike Shoes

Aenean gravida dignissim finibus. Nullam ipsum diam, posuere vitae pharetra sed, commodo ullamcorper.

Reviews (48)



★★★★★

\$360.00

Summer Addides Shoes

Curabitur ac mi sit amet diam luctus porta. Phasellus pulvinar sagittis diam, et scelerisque ipsum lobortis nec.

Reviews (74)

Demikianlah kostumisasi template bootstrap ini. Kesimpulanya kita dapat mengganti text, gambar, link tujuan, atau menambahkan elemen tertentu sesuai dengan kebutuhan kita. Ini tentunya sangat menghemat waktu dan tenaga, dimana kita hanya fokus pada konten yang kita sajikan dibanding harus menyusun grid, responsif, dan lainnya dari awal

Studi Kasus Pengaplikasian Bootstrap dalam Desain Web

Contoh Desain Web Yang Akan di Implementasikan:

Jobless

Home Discover Post a Job Contact Us Sign Up

Get Connected to the Best Remote Jobs in Your Field

Discover a wide range of remote opportunities on our platform and take control of your career

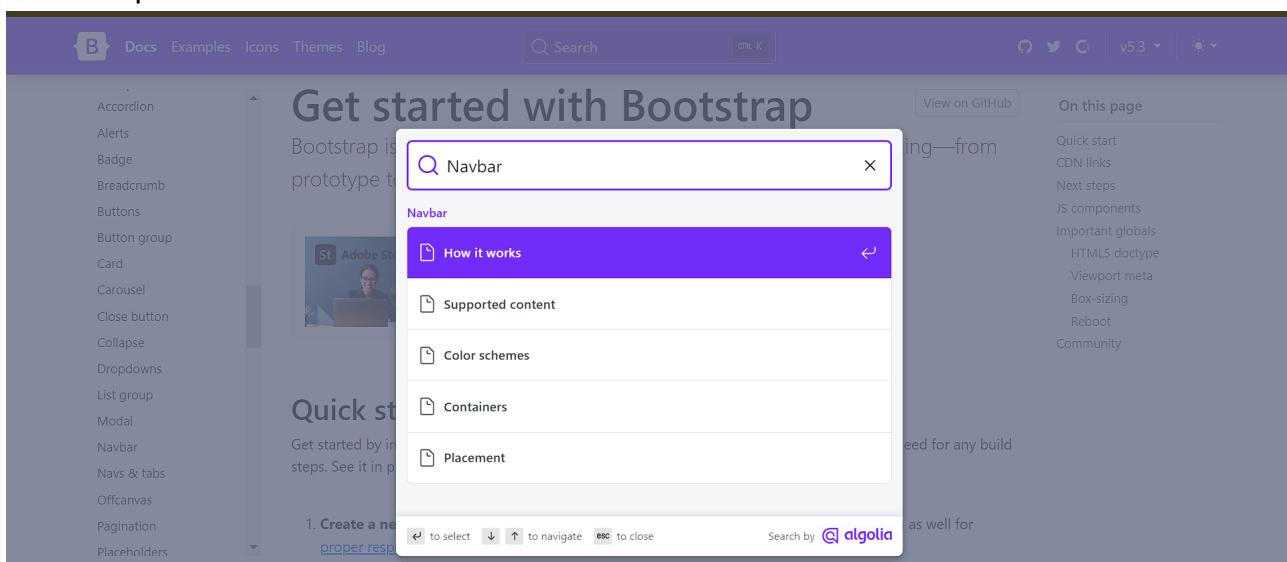
Explore Remote Jobs How it works?

1. dalam desain web tersebut terdapat beberapa komponen, yaitu
 - Navbar, Hero Section, Button dll
2. Pertama-tama kita perlu Inisialisasikan class `container` agar dapat memberikan tata letak konten yang sesuai dengan desainnya.

```
<div class="container">  
</div>
```

Keterangan:

- pada `<div class="container">`: Kelas ini memberikan tata letak yang terkandung pada lebar tertentu (dalam pixel) dan ditengahkan di tengah halaman atau elemen yang memuatnya. Ini membantu dalam mengatur konten agar sesuai dengan standar desain Bootstrap.
3. Selanjutnya kita akan implementasikan navbarnya, bukalah kembali web getbootstrap lalu di kolom pencarian carilah "navbar".



4. Cari dan copylah codingan komponen navbar yang mungkin sesuai dengan desainnya.

The screenshot shows the Bootstrap documentation website. The sidebar on the left is titled "Components" and lists various UI components: Buttons, Button group, Card, Carousel, Close button, Collapse, Dropdowns, List group, Modal, Navbar (which is highlighted), Navs & tabs, Offcanvas, Pagination, Placeholders, Popovers, Progress, Scrollspy, Spinners, Toasts, and Tooltips. The main content area displays the HTML code for a navbar. The code uses Bootstrap's utility classes like `bg-body-tertiary` and `container-fluid` along with standard HTML tags like `nav`, `a`, `button`, and `ul`. The code also includes Bootstrap's JavaScript behaviors such as `data-bs-toggle="collapse"`, `data-bs-target="#navbarSupportedContent"`, and `data-bs-toggle="dropdown"`. A search bar at the top right is visible.

```

<nav class="navbar navbar-expand-lg bg-body-tertiary"> <div class="container-fluid"> <a class="navbar-brand" href="#">Navbar</a> <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"> <span class="navbar-toggler-icon"></span> </button> <div class="collapse navbar-collapse" id="navbarSupportedContent"> <ul class="navbar-nav me-auto mb-2 mb-lg-0"> <li class="nav-item"> <a class="nav-link active" aria-current="page" href="#">Home</a> </li> <li class="nav-item"> <a class="nav-link" href="#">Link</a> </li> <li class="nav-item dropdown"> <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"> Dropdown </a> <ul class="dropdown-menu"> <li><a class="dropdown-item" href="#">Action</a></li> <li><a class="dropdown-item" href="#">Another action</a></li> <li> <hr class="dropdown-divider"> </li> <li> <a class="dropdown-item" href="#">Something else here</a></li> </ul> </li> <li class="nav-item"> <a class="nav-link disabled" aria-disabled="true">Disabled</a> </li> </ul> <form class="d-flex" role="search"> <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search"> <button class="btn btn-outline-success" type="submit">Search</button> </form> </div> </div> </nav>

```

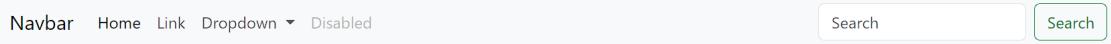
Keterangan:

1. `[navbar]` : Kelas ini menunjukkan bahwa elemen `<nav>` adalah bagian dari komponen navbar. Navbar adalah bagian dari Bootstrap yang menyediakan navigasi di bagian atas halaman web. Dengan memberikan kelas navbar , kita memberitahu Bootstrap untuk menerapkan gaya dan tata letak khusus yang terkait dengan navigasi.
2. `[bg-body-tertiary]` : Kelas ini memberikan warna latar belakang pada navbar. Dalam hal ini, warna latar belakangnya diberi warna sesuai dengan warna "hijau" Warna latar belakang ini dapat disesuaikan sesuai kebutuhan desain.
3. `[container-fluid]` : Kelas ini diterapkan pada elemen `<div>` yang membungkus elemen-

elemen dalam navbar. Kelas ini memberikan padding yang sesuai dan membuat elemen-elemen di dalamnya menjaga lebar penuh dari layar (menggunakan grid system Bootstrap). Dengan kata lain, kontennya akan merespons secara baik pada berbagai lebar perangkat.

4. `[d-flex]` : Kelas ini mengubah elemen `<form>` menjadi kontainer flex. Flexbox adalah teknik tata letak yang kuat di CSS yang memudahkan pengaturan dan penataan elemen dalam satu atau dua dimensi. Dengan memberikan kelas `d-flex` , elemen form dan anak-anaknya dapat disusun secara fleksibel.
5. `[me-2]` : Ini adalah kelas Bootstrap yang memberikan margin kanan sebesar 2 pada elemen yang memilikinya. Dalam codingan ini, kelas ini diterapkan pada elemen input untuk memberikan jarak margin kanan.

5. Tempelkanlah kode program itu didalam class `[container]` sebelumnya pada halaman html terkait. Tentu saja dengan bootstrap yang sudah ditautkan, baik offline, maupun online.
6. Maka hasilnya akan seperti ini:



7. Sekarang pada bagian dalam div `navbar-collapse` hapus elemen `form`:

```
<form class="d-flex" role="search"> <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search" /> <button class="btn btn-outline-success" type="submit">Search</button> </form>
```

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid"> <a class="navbar-brand" href="#">Navbar</a> <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"> <span class="navbar-toggler-icon"></span> </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav me-auto mb-2 mb-lg-0">
      <li class="nav-item"> <a class="nav-link active" aria-current="page" href="#">Home</a> </li>
      <li class="nav-item"> <a class="nav-link" href="#">Link</a> </li>
      <li class="nav-item dropdown"> <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"> Dropdown </a>
        <ul class="dropdown-menu">
          <li> <a class="dropdown-item" href="#">Action</a></li>
          <li> <a class="dropdown-item" href="#">Another action</a></li>
          <li> <hr class="dropdown-divider">
          </li>
          <li> <a class="dropdown-item" href="#">Something else here</a></li>
        </ul>
      </li>
      <li class="nav-item"> <a class="nav-link disabled" aria-disabled="true">Disabled</a> </li>
    </ul>
  </div>
</div>
</nav>
```

8. Hasilnya akan seperti ini:

9. Langkah Selanjutnya kita perlu mengganti list list serta nama logo pada navbarnya

```
<ul class="navbar-nav me-auto mb-2 mb-lg-0"> <li class="nav-item"> <a class="nav-link active fw-bold" aria-current="page" href="#">Home</a> </li>
<li class="nav-item"> <a class="nav-link text-muted" href="#">Discover</a>
</li> <li class="nav-item"> <a class="nav-link text-muted" href="#">Post a Job</a> </li> <li class="nav-item"> <a class="nav-link text-muted" href="#">Contact Us</a> </li> </ul>
```

Keterangan:

1. Disini kita mengganti text list item, dari sebelumnya default dari bootstrap ke desain yang akan diimplementasikan seperti: home,discover,post a job dan contact us
2. `fw-bold` dalam Bootstrap digunakan untuk memberikan tebal pada teks. Kelas ini merupakan bagian dari utilitas `font-weight` yang disediakan oleh Bootstrap, yang memungkinkan Anda untuk dengan cepat mengubah berat font teks dengan kelas `.fw-*`. Kelas `fw-bold` setara dengan berat font 700
3. `text-muted` digunakan untuk memberi gaya pada teks dengan warna yang redup. Kelas ini merupakan bagian dari utilitas `text-color` yang disediakan oleh Bootstrap, yang memungkinkan Anda untuk dengan mudah mengubah warna teks dengan kelas `.text-*`. Kelas `text-muted` setara dengan warna `$gray-600`, yang merupakan warna abu-abu yang redup.

10. Kemudian tambah tombol `sign-up` di sebelah kanan, sama seperti form namun kita berikan tag anchor

```
<a href="/sign-up" class="btn btn-outline-dark rounded-0 fw-bold"> Sign Up
</a>
```

Keterangan:

1. class `btn` digunakan untuk mengatur tampilan tombol. Class ini menambahkan beberapa stil ke tampilan tombol, seperti padding, margin, dan warna latar belakang.
2. `btn-outline-dark` digunakan untuk mengatur tampilan tombol dengan latar belakang transparan dan warna teks yang hitam.

3. `rounded-0` digunakan untuk mengatur bentuk tombol dengan sudut yang tidak ada (0).
4. `fw-bold` digunakan untuk mengatur font teks dengan berat font yang tebal.
5. `text-muted` digunakan untuk mengatur warna teks dengan warna yang redup.

Jobless Home Discover Post a Job Contact Us

Sign Up

11. Selanjutnya ganti background navbarnya dari `bg-body-tertiary` menjadi transparan dengan menggunakan class `bg-transparent`

```
<nav class="navbar navbar-expand-lg bg-transparent">
  <div class="container-fluid"> <a class="navbar-brand" href="#">Navbar</a> <button class="navbar-toggler"
    type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"
    aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"> <span
      class="navbar-toggler-icon"></span> </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav me-auto mb-2 mb-lg-0">
      <li class="nav-item"> <a class="nav-link active fw-bold" aria-current="page" href="#">Home</a>
      </li>
      <li class="nav-item"> <a class="nav-link text-muted" href="#">Discover</a> </li>
      <li class="nav-item"> <a class="nav-link text-muted" href="#">Post a Job</a> </li>
      <li class="nav-item"> <a class="nav-link text-muted" href="#">Contact Us</a> </li>
    </ul>

    <a href="/sign-up" class="btn btn-outline-dark rounded-0 fw-bold"> Sign Up </a>
  </div>
</div>
</nav>
```

12. Sekarang kita akan menengahkan list navbarnya dengan menggunakan class `me-auto`

Jobless Home Discover Post a Job Contact Us

Sign Up

sebelum

```
<div class="container">

    <nav class="navbar navbar-expand-lg bg-transparent">
        <div class="container-fluid">
            <a class="navbar-brand" href="#">Jobless</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
                data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
                aria-expanded="false" aria-label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarSupportedContent">
                <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                    <li class="nav-item"> <a class="nav-link active" aria-current="page" href="#">Home</a> </li>
                    <li class="nav-item"> <a class="nav-link" href="#">Discover</a> </li>
                    <li class="nav-item"> <a class="nav-link" href="#">Post a Job</a> </li>
                    <li class="nav-item"> <a class="nav-link" href="#">Contact Us</a> </li>
                </ul>
                <a href="/sign-up" class="btn btn-outline-dark rounded-0">
                    Sign Up
                </a>
            </div>
        </div>
    </nav>

</div>
```

sesudah

```
<div class="container">

    <nav class="navbar navbar-expand-lg bg-transparent">
        <div class="container-fluid">
            <a class="navbar-brand" href="#">Jobless</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
                data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
                aria-expanded="false" aria-label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarSupportedContent">
                <ul class="navbar-nav m-auto mb-2 mb-lg-0">
                    <li class="nav-item"> <a class="nav-link active" aria-current="page" href="#">Home</a> </li>
                    <li class="nav-item"> <a class="nav-link" href="#">Discover</a> </li>
                    <li class="nav-item"> <a class="nav-link" href="#">Post a Job</a> </li>
                    <li class="nav-item"> <a class="nav-link" href="#">Contact Us</a> </li>
                </ul>
                <a href="/sign-up" class="btn btn-outline-dark rounded-0">
                    Sign Up
                </a>
            </div>
        </div>
    </nav>

</div>
```

13. Hasilnya akan seperti ini:

14. Sekarang kita akan implementasikan komponen hero section, tuliskan kode berikut.

```
<section id="hero"></section>
```

15. dapat kita lihat pada komponen hero terdapat text dan tombol posisinya ditengah, nah disini kita bisa menggunakan grid system bootstrap agar dapat mengatur posisinya.

```
<div class="row h-100 justify-content-center align-items-center text-center">
<div class="col-12"> <h1 class="fw-bold">Get Connected to the Best Remote <br
/>Jobs in Your Field </h1> <p class="text-muted">Discover a wide range of
remote opportunities on our platform and<br />take control of your career</p>
</div> </div>
```

Keterangan:

1. `row` digunakan untuk mengatur tata letak konten dalam satu baris. Class ini menambahkan beberapa stil ke tata letak konten, seperti mengatur margin dan padding untuk konten yang berada di sisi.
2. `h-100` adalah class Bootstrap yang digunakan untuk mengatur tinggi konten dalam sebuah halaman web. Class ini menambahkan stil yang memastikan konten berada pada tinggi penuh dari bagian atas ke bawah.
3. `justify-content-center` adalah class Bootstrap yang digunakan untuk mengatur posisinya konten di tengah bagian horizontal.
4. class `align-items-center` adalah class Bootstrap yang digunakan untuk mengatur posisinya konten di tengah bagian vertikal.
5. class `text-center` adalah class Bootstrap yang digunakan untuk mengatur posisi teks di tengah bagian konten.
6. class `col-12` adalah class Bootstrap yang digunakan untuk mengatur lebar konten. Class ini

menambahkan stil yang memastikan konten berada pada lebar penuh dari bagian atas ke bawah.

16. Hasilnya akan seperti ini:



17. Disini text nya hanya ke tengah secara horizontal, agar dapat ke tengah secara vertikal perlu kita atur tinggi pada hero sectionnya dengan external style

```
<style> #hero { height: 85vh; } </style>
```

```
<head>
  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-s<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" integrity="sha384-QWTKZyjpPEjISv5WaRU90FeRpok6YctnYmDr5pI" />
  <title>Bootstrap Implementasi Web Desain</title>

  <style>
    #hero {
      height: 85vh;
    }
  </style>
</head>
```

Keterangan:

ini mengatur tinggi dari elemen yang memiliki ID `hero` menjadi 85% dari tinggi viewport (viewport height).

18. Hasilnya akan seperti ini:

Get Connected to the Best Remote Jobs in Your Field

Discover a wide range of remote opportunities on our platform and take control of your career

19. Sekarang kita akan implementasikan komponen button nya, dapat kita lihat pada desainnya terdapat dua button sejajar secara horizontal, maka kita perlu memakai flexbox yang disediakan oleh bootstrap.

```
<div class="d-flex gap-4 justify-content-center align-items-center"> <a href="#" class="">Explore Remote Jobs</a> <a href="#" class="">How it works? </a> </div>
```

Keterangan:

1. `d-flex` digunakan untuk mengatur tata letak konten dalam satu baris. Class ini menambahkan stil yang memastikan konten berada pada satu baris dan tidak berpindah ke baris baru.
2. `gap-4` digunakan untuk mengatur jarak antara konten dalam satu baris. Class ini menambahkan stil yang membuat jarak antara konten berada 4 unit.
3. `justify-content-center` digunakan untuk mengatur posisinya konten di tengah bagian horizontal.
4. `align-items-center` digunakan untuk mengatur posisinya konten di tengah bagian vertikal.

Get Connected to the Best Remote Jobs in Your Field

Discover a wide range of remote opportunities on our platform and take control of your career

[Explore Remote Jobs](#) [How it works?](#)

20. Selanjutnya kita perlu beri space dari setiap buttonnya, dan beri warna pada button sebelah kiri

```
<div class="d-flex gap-4 justify-content-center align-items-center"> <a href="#" class="p-2 px-3 text-decoration-none bg-primary rounded-5 text-white shadow-sm ">Explore Remote Jobs</a> <a href="#" class="p-2 px-3 text-primary rounded-5 fw-semibold">How it works?</a> </div>
```

Keterangan:

1. `p-2` digunakan untuk mengatur padding pada tautan. Class ini menambahkan stil yang membuat padding pada tautan berada 0.5 unit pada sisi atas dan bawah, serta 0.75 unit pada sisi kiri dan kanan.
2. `px-3` digunakan untuk mengatur padding pada tautan. Class ini menambahkan stil yang membuat padding pada tautan berada 0.75 unit pada sisi kiri dan kanan.
3. `text-decoration-none` digunakan untuk mengatur dekorasi teks pada tautan. Class ini menghilangkan dekorasi garis bawah yang biasanya muncul pada tautan.
4. `bg-primary` digunakan untuk mengatur warna latar belakang pada tautan. Class ini menambahkan stil yang membuat latar belakang tautan berwarna hitam.
5. `rounded-5` digunakan untuk mengatur bentuk tautan. Class ini menambahkan stil yang membuat bentuk tautan berbentuk lingkaran dengan jari-jari 5 unit.
6. `text-white` digunakan untuk mengatur warna teks pada tautan. Class ini menambahkan stil yang membuat warna teks tautan berwarna putih.
7. `shadow-sm` digunakan untuk mengatur gaya bagian atas tautan. Class ini menambahkan stil yang membuat bagian atas tautan berwarna abu-abu dengan efek shadow.
8. `fw-semibold` digunakan untuk mengatur font teks dengan berat font yang semibold.
9. `text-primary` digunakan untuk mengatur warna teks pada tautan. Class ini menambahkan stil yang membuat warna teks tautan berwarna hitam.

Get Connected to the Best Remote Jobs in Your Field

Discover a wide range of remote opportunities on our platform and take control of your career

[Explore Remote Jobs](#)[How it works?](#)

21. Sekarang kita akan mengganti background body nya

```
<style> body { background-color: #F6FAFF; background-image: linear-gradient(90deg, #0500FF 0%, #F8FB76 35%, #4EFF75 64%, #00FFD1 100%); background-blend-mode: overlay; } </style>
```

```
<style>
  body {
    background-color: #f6fafaf;
    background-image: linear-gradient(120deg, #0500FF 0%, #F8FB76 35%, #4EFF75 64%, #00FFD1 100%);
    background-blend-mode: overlay;
  }

  #hero {
    height: 85vh;
  }
</style>
</head>
```

22. Hasilnya akan seperti ini:

Get Connected to the Best Remote Jobs in Your Field

Discover a wide range of remote opportunities on our platform and take control of your career

[Explore Remote Jobs](#)

[How it works?](#)

Best Practices dan Tips

Gunakan Grid Sistem:

Manfaatkan grid sistem Bootstrap untuk mengatur tata letak (layout) situs Anda dengan baik. Gunakan kelas-kelas seperti `container`, `row`, dan `col` untuk membagi konten Anda menjadi bagian-bagian yang sesuai.

Gunakan Komponen Bootstrap:

Manfaatkan komponen-komponen yang disediakan oleh Bootstrap seperti navbar, card, form, dll. Komponen-komponen ini sudah dirancang dan diuji untuk memastikan konsistensi dan fungsionalitas yang baik.

Uji pada Berbagai Browser:

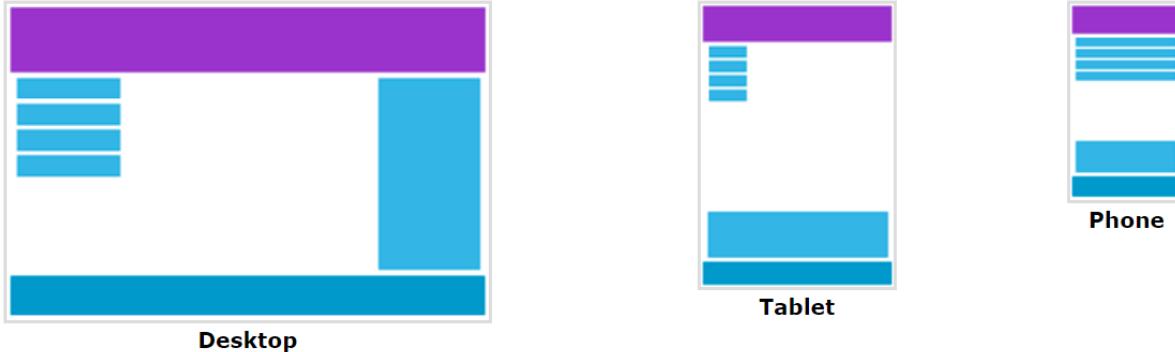
Pastikan situs Anda berfungsi dengan baik di berbagai browser, termasuk versi lama dan baru, dengan melakukan pengujian silang browser.

Pengenalan Responsive Web Design

Responsive web design atau desain web responsif adalah sebuah teknik atau metode bagi web designer untuk membuat suatu layout website yang dapat menyesuaikan diri sesuai dengan ukuran layar pengguna. Baik dari ukuran huruf, user interface, gambar dan tata letak akan

menyesuaikan dengan lebar layar dan resolusi device yang digunakan.

Selain dapat memengaruhi apa yang ditampilkan pada perangkat tertentu, ini juga untuk meningkatkan kenyamanan pengguna dalam mengunjungi suatu website.



Mengapa perlu CSS Responsive?

Pernahkah Anda mengunjungi situs web di smartphone Anda dan halaman webnya berantakan atau teksnya terlalu kecil untuk dibaca? Ini artinya situs web tersebut belum responsif. Dengan memakai css responsif maka web kita memungkinkan tampil optimal di berbagai perangkat, mulai dari desktop, laptop, tablet, hingga smartphone. Berikut alasan beberapa alasan mengapa css responsif penting:

- **Tampilan yang Bagus di Semua Perangkat**

CSS responsif memungkinkan tampilan situs web beradaptasi dengan baik terhadap berbagai perangkat dan ukuran layar, sehingga pengguna akan memiliki pengalaman yang konsisten dan optimal, baik mereka mengakses situs melalui desktop, tablet, atau ponsel.

- **Mudah Diakses**

Situs yang responsif bisa diakses dengan mudah oleh lebih banyak orang, termasuk yang menggunakan ponsel untuk browsing.

- **Interaksi yang Lebih Baik**

Situs yang responsif cenderung membuat orang merasa lebih nyaman dan lebih mudah berinteraksi. Ini bisa meningkatkan kemungkinan orang melakukan hal-hal seperti membeli produk atau menghubungi Anda.

Teknik Dasar CSS Responsif

Konsep Dasar

- **Grid System:** Menggunakan grid system dalam CSS untuk mempermudah penataan elemen pada halaman web.

- **Flexbox:** Memahami penggunaan flexbox untuk menata elemen secara responsif.

Media Query

Media query adalah fitur CSS yang memungkinkan Anda menargetkan perangkat dan ukuran layar yang berbeda dengan aturan CSS yang spesifik. Hal ini sangat penting untuk membuat website yang responsif, yaitu website yang dapat menyesuaikan tata letak dan desainnya secara otomatis agar terlihat optimal di semua perangkat, mulai dari desktop, laptop, tablet, hingga smartphone.

Cara Kerja Media Query

Media query menggunakan aturan `@media` untuk menargetkan perangkat dan ukuran layar yang berbeda. Aturan ini terdiri dari dua bagian:

- **Media type:** Menentukan jenis perangkat yang ingin Anda targetkan, seperti `screen` (untuk desktop, laptop, tablet, dan smartphone) atau `print` (untuk printer).
- **Media feature:** Menentukan karakteristik perangkat yang ingin Anda targetkan, seperti `max-width` (lebar layar maksimum), `min-width` (lebar layar minimum), `device-width` (lebar perangkat), dan `orientation` (orientasi perangkat).

Breakpoints

Breakpoints adalah titik-titik di mana tata letak halaman web akan berubah. Breakpoints biasanya ditentukan berdasarkan lebar layar perangkat.

- **320px:** Lebar minimum untuk smartphone.
- **768px:** Lebar minimum untuk tablet.
- **1024px:** Lebar minimum untuk laptop.
- **1200px:** Lebar minimum untuk desktop.

```
@media screen and (max-width: 768px) {  
    /* Aturan CSS untuk perangkat mobile */  
    body {  
        font-size: 16px;  
    }  
    .container {  
        width: 100%;  
    }  
}
```

Unit Relatif

Unit relatif adalah unit yang digunakan dalam CSS untuk menentukan ukuran elemen secara proporsional. Berbeda dengan unit absolut seperti piksel dan sentimeter, unit relatif tidak memiliki nilai tetap dan dapat berubah tergantung pada konteksnya.

Keuntungan

- **Fleksibilitas:** Unit relatif memungkinkan Anda membuat website yang responsif dan dapat menyesuaikan dengan berbagai ukuran layar dan perangkat.
- **Kemudahan Penggunaan:** Unit relatif lebih mudah digunakan untuk menentukan ukuran elemen yang proporsional dan konsisten di seluruh website.
- **Pemeliharaan:** Website yang menggunakan unit relatif lebih mudah dipelihara dan diubah karena ukuran elemen akan menyesuaikan secara otomatis.

Jenis-jenis Unit Relatif:

- **em:** Ukuran font elemen induk
- **rem:** Ukuran font elemen root (biasanya elemen `<html>`)
- **vw:** Lebar viewport (layar perangkat pengguna)
- **vh:** Tinggi viewport (layar perangkat pengguna)
- **vmin:** Nilai terkecil antara vw dan vh
- **vmax:** Nilai terbesar antara vw dan vh

Viewport Meta Tag

Viewport meta tag adalah sebuah elemen HTML yang memberitahu browser web bagaimana cara menampilkan website Anda pada perangkat pengguna. Tag ini sangat penting untuk website responsif, yaitu website yang dapat menyesuaikan tata letak dan desainnya secara otomatis agar terlihat optimal di semua perangkat, mulai dari desktop, laptop, tablet, hingga smartphone.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

- **width=device-width:** Menentukan lebar awal website agar sama dengan lebar perangkat pengguna.
- **initial-scale=1.0:** Mengatur skala awal website menjadi 100%.

Best Practices

Gunakan Mobile-First

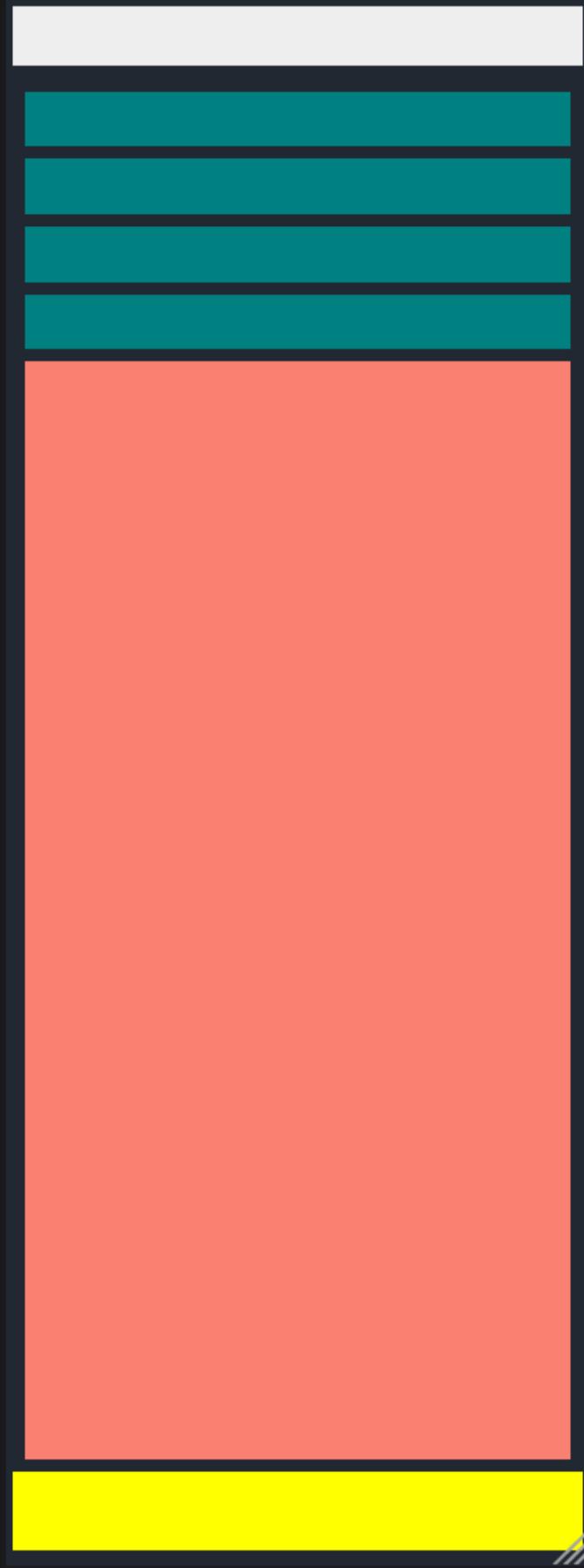
Saat mendesain website, mulailah dengan mempertimbangkan tampilan website pada perangkat mobile. Hal ini akan membantu Anda membuat website yang lebih ringkas dan mudah digunakan di semua perangkat.

Implementasi CSS Responsif

Sebelum Responsif



Sesudah Responsif



Kode

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="style.css" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>CSS Responsif</title>
  </head>
  <body>
    <header></header>
    <section class="container">
      <div class="list">
        <div class="item"></div>
        <div class="item"></div>
        <div class="item"></div>
        <div class="item"></div>
      </div>
      <aside></aside>
    </section>
    <footer></footer>
  </body>
</html>
```

CSS

```
body {
  background-color: #222831;
  color: white;
}

header {
  width: 100%;
  height: 77px;
  background-color: #EEEEEE;
}

.container {
  display: flex;
  justify-content: space-between;
}

.item {
```

```

width: 250px;
height: 70px;
background-color: teal;
margin: 1em 0;
}

aside {
  width: 300px;
  height: 70vh;
  background-color: salmon;
}

footer {
  width: 100%;
  height: 100px;
  background-color: yellow;
}

@media screen and (max-width: 768px) {
  .container {
    padding: 1rem;
    flex-direction: column;
  }
  .item {
    width: 100%;
  }
  aside {
    width: 100%;
  }
}

```

Penjelasan Singkat

```

body {
  background-color: #222831;
  color: white;
}

```

- body : Ini adalah elemen HTML yang menandakan seluruh konten halaman web.
- background-color: #222831 : Ini mengatur warna latar belakang dari elemen "body" menjadi warna gelap dengan kode warna #222831 (hampir hitam).
- color: white : Ini mengatur warna teks di dalam elemen "body" menjadi putih.

```
header {  
    width: 100%;  
    height: 77px;  
    background-color: #EEEEEE;  
}
```

- `header` : elemen HTML yang menandakan bagian atas halaman web.
- `width: 100%` : membuat lebar elemen "header" sejajar dengan lebar area yang mengandungnya, yang mungkin adalah lebar jendela browser atau lebar elemen induk lainnya. Dengan memberikan nilai 100%, elemen "header" akan merentang ke seluruh lebar area tersebut.
- `height: 77px` : menetapkan tinggi elemen "header" menjadi 77 piksel. Jadi, meskipun lebarnya mengikuti lebar area induknya (100%), tingginya tetap diberikan secara spesifik dengan 77 piksel. Ini bertujuan untuk memastikan bahwa tinggi header tetap konsisten meskipun konten di dalamnya berubah.
- `background-color: #EEEEEE` : Ini mengatur warna latar belakang dari elemen "header" menjadi abu-abu muda dengan kode warna # EEEE.

```
.container {  
    display: flex;  
    justify-content: space-between;  
}
```

- `.container` : Ini adalah kelas CSS yang digunakan untuk menargetkan elemen HTML dengan kelas "container".
- `display: flex` : membuat elemen dengan kelas "container" menggunakan model tata letak flexbox.
- `justify-content: space-between` : Ini mengatur jarak antara elemen-elemen di dalam flexbox container agar rata di sepanjang sumbu utama (biasanya sumbu horizontal), dengan menjaga jarak yang sama antara elemen-elemen dan meletakkan mereka pada posisi ujung container.

```
.item {  
    width: 250px;  
    height: 70px;  
    background-color: teal;  
    margin: 1em 0;  
}
```

- `.item` : Ini adalah sebuah selektor CSS yang memilih elemen dengan kelas "item".

- `width: 250px` : Ini mengatur lebar elemen menjadi 250 piksel.
- `height: 70px` : Ini mengatur tinggi elemen menjadi 70 piksel.
- `background-color: teal` : Ini mengatur warna latar belakang elemen menjadi warna teal.
- `margin: 1em 0` : Ini mengatur jarak antara elemen dengan elemen lainnya, dengan 1 em (ukuran relatif terhadap font-size) pada bagian atas dan bawah, dan 0 pada bagian kanan dan kiri.

```
aside {
  width: 300px;
  height: 70vh;
  background-color: salmon;
}
```

- `aside` : Ini adalah sebuah elemen HTML yang biasanya digunakan untuk menunjukkan konten tambahan di samping konten utama.
- `width: 300px` : Ini mengatur lebar elemen "aside" menjadi 300 piksel.
- `height: 70vh` : Ini mengatur tinggi elemen "aside" menjadi 70% dari tinggi viewport.
- `background-color: salmon` : Ini mengatur warna latar belakang elemen "aside" menjadi warna salmon.

```
footer {
  width: 100%;
  height: 100px;
  background-color: yellow;
}
```

- `footer` : Ini adalah sebuah elemen HTML yang biasanya digunakan untuk menampilkan informasi tambahan di bagian bawah halaman web.
- `width: 100%` : Ini mengatur lebar elemen "footer" agar mengisi seluruh lebar dari parent element (biasanya body atau container lainnya).
- `height: 100px` : Ini mengatur tinggi elemen "footer" menjadi 100 piksel.
- `background-color: yellow` : Ini mengatur warna latar belakang elemen "footer" menjadi warna kuning.

```
@media screen and (max-width: 768px) {
  .container {
    padding: 1rem;
    flex-direction: column;
  }
}
```

```
.item {  
    width: 100%;  
}  
aside {  
    width: 100%;  
}  
}
```

- `@media screen and (max-width: 768px)` : Ini adalah aturan media query yang diterapkan ketika lebar layar kurang dari atau sama dengan 768 piksel.
- `.container` : Mengatur tampilan untuk elemen dengan kelas "container" ketika media query aktif.
 - `padding: 1rem;` : Memberikan padding sebesar 1 rem di sekitar elemen "container".
 - `flex-direction: column;` : Mengubah arah penataan elemen menjadi vertikal (kolom), sehingga elemen akan ditata dari atas ke bawah.
- `.item` : Mengatur tampilan untuk elemen dengan kelas "item" ketika media query aktif.
 - `width: 100%;` : Memberikan lebar sebesar 100% dari parent element, sehingga elemen akan mengisi lebar penuh.
- `aside` : Mengatur tampilan untuk elemen "aside" ketika media query aktif.
 - `width: 100%;` : Memberikan lebar sebesar 100% dari parent element, sehingga elemen "aside" akan mengisi lebar penuh.