

| Anatomi CSS

| kode program

```
P {  
  color: red;  
}
```

| penjelasan

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

| kesimpulan

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

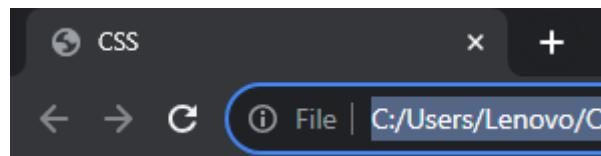
| Percobaan pertama CSS

| kode program

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

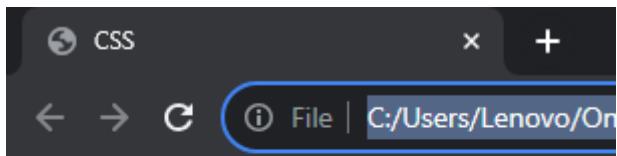
| hasil

| before



Welcome CSS

| after



Welcome CSS

| penjelasan

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

| kesimpulan

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

| Percobaan kedua CSS

| kode program

```
button{  
width: 150px;  
height: 50px;  
font-size: 20px;  
color: blue;  
background-color: #7949FF;  
}
```

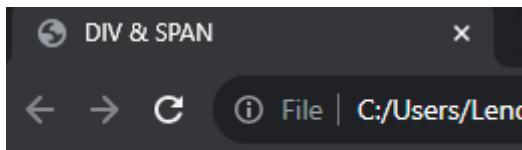
| Font-size

| PENJELASAN

Font size atau ukuran font dalam dunia pemformatan teks, merujuk pada ukuran fisik karakter huruf yang ditampilkan di layar.

| HASIL

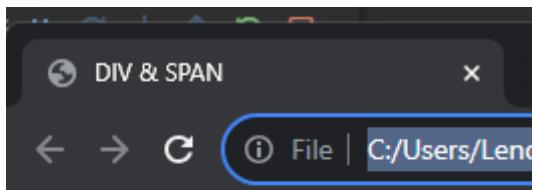
| BEFORE



Welcome CSS

klik aku!

AFTER



Welcome CSS

klik aku!

ANALISIS

- `Font-size` untuk memperbesar tulisan "KLIK AKU".

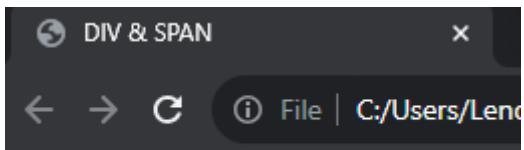
Color

PENJELASAN

Color adalah elemen fundamental dalam desain yang mempengaruhi persepsi, emosi, dan pemahaman audiens terhadap informasi yang disampaikan. Ini tidak hanya sekedar estetika, namun juga berperan penting dalam komunikasi visual.

HASIL

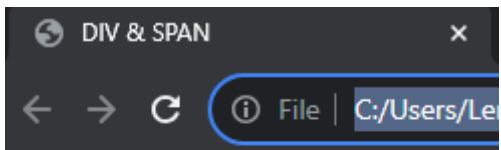
BEFORE



Welcome CSS

klik aku!

AFTER



Welcome CSS

klik aku!

ANALISIS

- `color` buat mewarnai tulisan

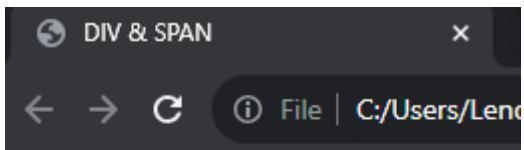
Background-color

PENJELASAN

Background color (warna latar belakang) merupakan aspek penting dalam desain visual. Ini mempengaruhi persepsi dan keterbacaan konten, serta membantu mengatur hierarki informasi.

HASIL

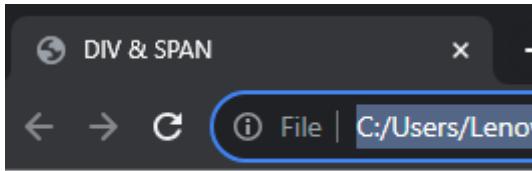
BEFORE



Welcome CSS

klik aku!

AFTER



Welcome CSS

klik akul!

ANALISIS

- `background-color`, menambahkan latar belakang pada kotak.

Deklarasi pemanggilan CSS

External

penjelasan

`<link rel="stylesheet" href="style1.css">` digunakan untuk menghubungkan file eksternal yang berisi kode Cascading Style Sheet (CSS) ke sebuah halaman web HTML. Mari kita lihat perannya:

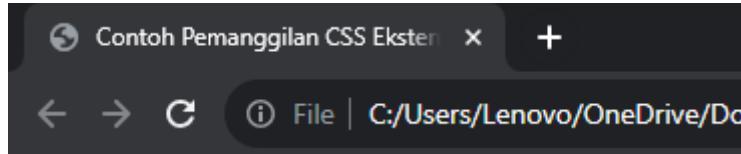
- `<link>`: Ini adalah tag HTML yang digunakan untuk menghubungkan resource eksternal ke halaman web.
- `rel="stylesheet"` : Atribut `rel` menentukan hubungan antara resource eksternal dengan dokumen HTML. Dalam hal ini, nilainya `stylesheet` menunjukkan bahwa resource yang dihubungkan adalah file stylesheet yang berisi aturan styling untuk halaman web.
- `href="style1.css"` : Atribut `href` menentukan lokasi resource eksternal tersebut. Dalam contoh ini, nilainya `"style1.css"` menunjukkan bahwa file stylesheet yang dihubungkan bernama `style1.css` dan kemungkinan berada di folder yang sama dengan file HTML atau di lokasi yang ditentukan pathnya.

kode program

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Contoh Pemanggilan CSS Eksternal</title>
    <link rel="stylesheet" type="text/css" href="styles.css">
</head>
```

```
<body>
    <p>Ini adalah contoh pemanggilan CSS eksternal.</p>
</body>
</html>
```

| hasil



Ini adalah contoh pemanggilan CSS eksternal.

| inline

| penjelasan

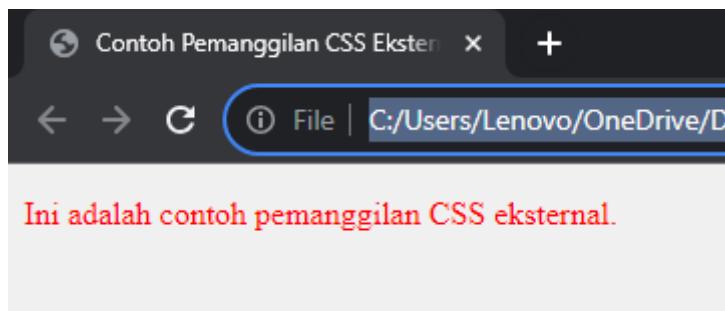
Inline merupakan sebuah konsep yang umum digunakan dalam pemrograman dan markup language, yang merujuk pada penyisipan atau penempatan elemen atau kode di dalam konteks tertentu.

`"inline"` sering digunakan untuk mengacu pada kode yang ditempatkan secara langsung di dalam baris kode lainnya, tanpa memerlukan pemanggilan fungsi atau blok pemisahan yang terpisah.

| kode program

```
<p style="color: red;">Ini adalah contoh pemanggilan CSS eksternal.</p>
```

| hasil



| Internal

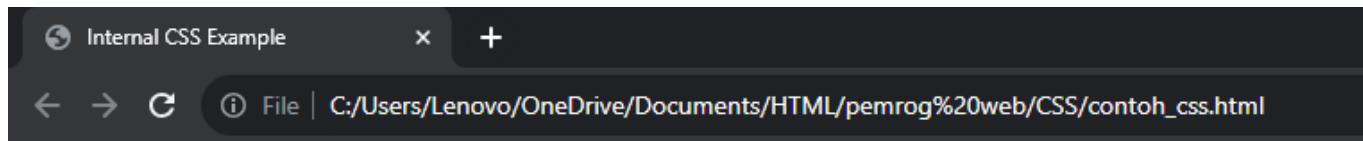
| penjelasan

Internal adalah istilah yang digunakan untuk mengacu pada sesuatu yang terjadi, terletak, atau berkaitan dengan suatu objek, sistem, atau organisasi secara dalam atau di dalamnya. Internal digunakan dalam berbagai konteks, tergantung pada domain atau bidang yang sedang dibahas.

| kode program

```
body {  
    font-family: Arial, sans-serif;  
    background-color:white;  
  
}  
h1 {  
    color:green;  
}  
  
p {  
    font-size: 16px;  
    line-height: 1.5;  
}
```

| hasil



ini adalah deklarasi pemanggilan css internal

| Selector

| Elemen selector

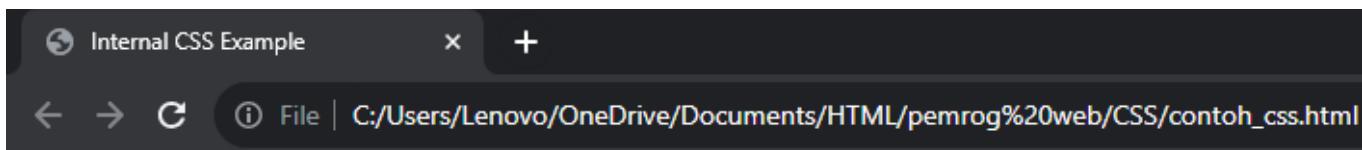
| penjelasan

selector adalah serangkaian aturan dari CSS yang memiliki fungsi yaitu memilih suatu elemen yang ingin kamu beri gaya.

| kode program

```
p{  
    text-decoration: overline;  
}
```

| hasil



ini adalah deklarasi pemanggilan css internal

| Text

| Text align

| PENJELASAN

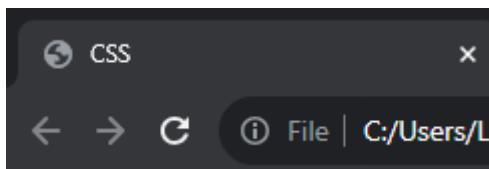
- `text-align:left` ; Text akan ditampilkan di sebelah kiri dokumen.
- `text-align:right` ; Text akan ditampilkan di sebelah kanan dokumen.
- `text-align:center` ; Text akan ditampilkan di tengah-tengah dokumen.
- `text-align:justify` ; Text akan ditampilkan dengan mengikuti spasi yang ada di antara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

| KODE PROGRAM

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

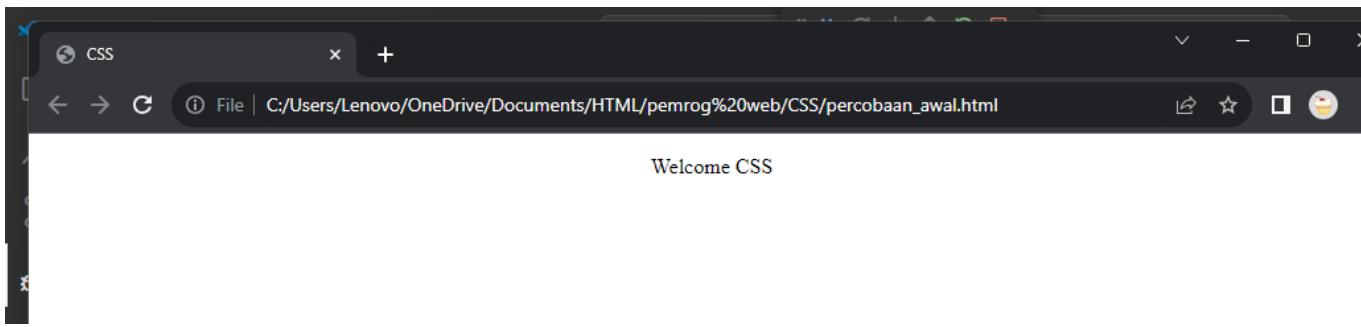
| HASIL

| BEFORE



Welcome CSS

| AFTER



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 decoration

PENJELASAN

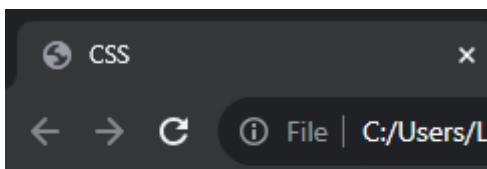
- `none` : menghilangkan dekorasi garis pada teks (pilihan default).
- `underline` : menambahkan garis bawah pada teks.
- `overline` : menambahkan garis atas pada teks.
- `line-through` : menambahkan garis horizontal yang menembus tengah teks, sehingga tampak tercoret.

KODE PROGRAM

```
p {  
  text-decoration: line-through;  
}
```

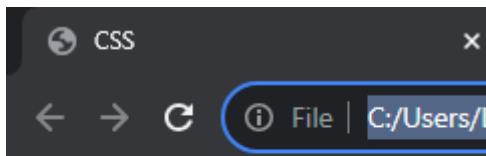
HASIL

BEFORE



Welcome CSS

AFTER



Welcome CSS

KESIMPULAN

Kesimpulan `text-decoration: line-through;` adalah bahwa teks yang diberi gaya dengan properti ini akan memiliki garis melintang di tengahnya, menunjukkan bahwa teks tersebut dianggap tidak relevan, usang, atau tidak berlaku lagi.

Text transform

PENJELASAN

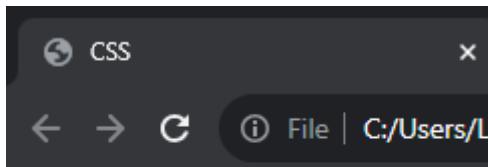
- `none`: (default): Menjaga kapitalisasi teks seperti pada penulisan aslinya.
- `uppercase`: Mengubah semua huruf menjadi huruf besar (uppercase).
- `lowercase`: Mengubah semua huruf menjadi huruf kecil (lowercase).
- `capitalize`: Mengubah huruf pertama setiap kata menjadi huruf besar (capitalize), mirip gaya penulisan judul.

KODE PROGRAM

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

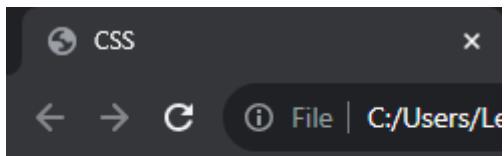
HASIL

BEFORE



Welcome CSS

AFTER



WELCOME CSS

KESIMPULAN

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

Text indent

PENJELASAN

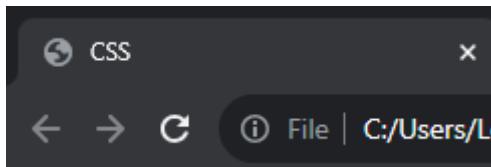
`Text indent` atau indentasi teks adalah proses membuat jarak antara teks dengan margin atas dari sebuah paragraf. Ini dapat membuat teks terlihat lebih rapi dan terstruktur. Teks dapat diatur dengan menggunakan tanda indentasi seperti tab atau spasi.

KODE PROGRAM

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

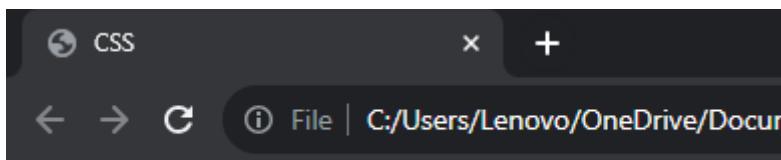
HASIL

BEFORE



Welcome CSS

AFTER



Welcome CSS

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.

Letter spacing

PENJELASAN

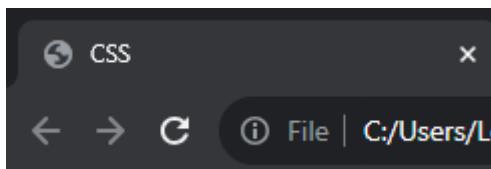
`letter-spacing` merujuk pada jarak horizontal antara huruf-huruf dalam sebuah teks. kata lain, letter spacing mengontrol seberapa dekat atau berjauhan huruf-huruf tersebut ditempatkan satu sama lain.

KODE PROGRAM

```
p {  
  letter-spacing: 50px;  
}
```

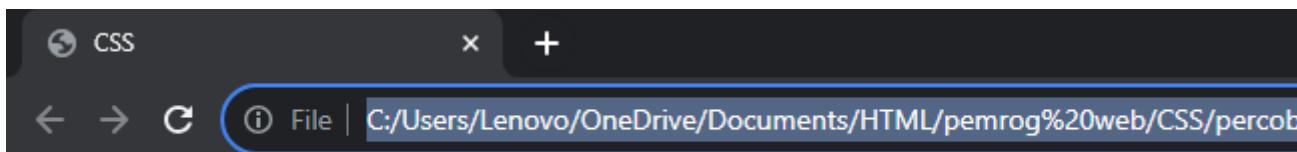
HASIL

BEFORE



Welcome CSS

AFTER



KESIMPULAN

Kesimpulan dari penggunaan properti CSS `letter-spacing: 50px;` adalah bahwa jarak antara setiap huruf dalam teks pada elemen `<p>` akan diperbesar sebesar 50 piksel. ini digunakan untuk mengatur spasi horizontal antara huruf-huruf dalam teks, menciptakan efek visual seperti peningkatan jarak antara huruf-huruf atau tampilan teks yang lebih terbaca dengan jelas.

Line height

PENJELASAN

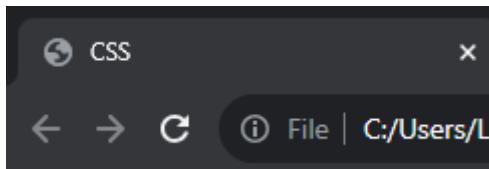
`Line-height` adalah properti dalam CSS yang menentukan **jarak vertikal** antara baris teks dalam sebuah elemen, biasanya elemen paragraf (`<p>`).

KODE PROGRAM

```
p {  
    line-height: 150px;  
}
```

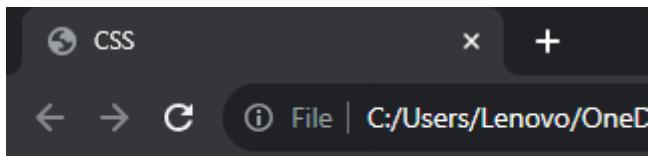
HASIL

BEFORE



Welcome CSS

AFTER



Welcome CSS

ANALISIS

Kesimpulan `"line-height"` diatur menjadi 150 piksel untuk elemen "p". `"line-height"` digunakan untuk mengatur jarak antara baris dalam elemen teks. nilai "150px" menunjukkan bahwa jarak antara setiap baris dalam elemen "p" akan menjadi 150 piksel.

Word spacing

PENJELASAN

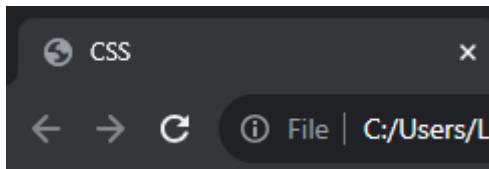
`Word spacing` merujuk pada pengaturan jarak antara kata dalam teks.

KODE PROGRAM

```
p {  
  word-spacing: 150px;  
}
```

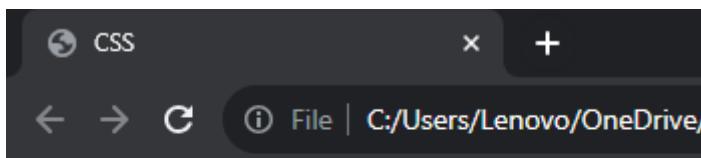
HASIL

BEFORE



Welcome CSS

AFTER



Welcome

CSS

ANALISIS

Kode `word-spacing: 150px;`, mengatur jarak antar kata dalam `<p>` elemen (paragraf) menjadi 150 piksel. Ini berarti akan ada jarak yang cukup besar antara setiap kata dalam paragraf, sehingga menciptakan tampilan visual yang berbeda. Properti ini dapat digunakan untuk berbagai tujuan desain, seperti membuat tata letak yang unik atau menekankan kata-kata individual.

Background

Background image

penjelasan

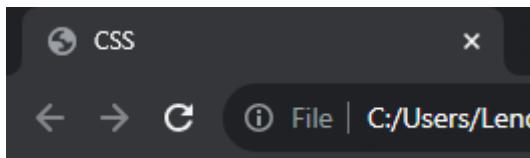
`Background image` adalah gambar atau visual yang diletakkan di belakang konten utama pada sebuah halaman web atau desain grafis.

KODE PROGRAM

```
body {  
    background-image: url('5.jpg');  
}
```

HASIL

Before



Selamat Datang

After



| KESIMPULAN

- `body` mengacu pada area konten utama halaman web.
- `background-image` adalah properti di Cascading Style Sheets (CSS) yang mengontrol latar belakang suatu elemen.
- `url('5.jpg')` menentukan lokasi (URL) file gambar yang ingin Anda gunakan. Dalam hal ini, diasumsikan gambar berada dalam folder yang sama dengan file CSS, dan diberi nama "5.jpg".

| Background size

| PENJELASAN

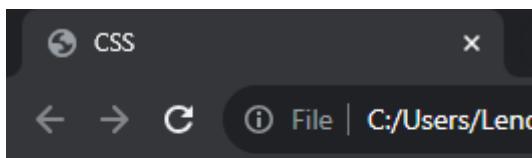
`Background size` kita mengatur size/ukuran dari background yang kita gunakan dan valuenya dapat kita atur sesuai dengan keinginan kita.

| KODE PROGRAM

```
body {  
    background-image: url('5.jpg');  
    width: 100%;  
    height: 100vh;  
  
    background-size: 50%;  
}
```

| HASIL

| Before



Selamat Datang

After



KESIMPULAN

- `body {`: Baris ini mendefinisikan gaya yang akan diterapkan pada `<body>` elemen dokumen HTML.
- `background-image: url('5.jpg');`: Ini menyetel gambar latar belakang elemen body ke file gambar bernama "5.jpg". Jalur gambar relatif terhadap lokasi file HTML.
- `width: 100%;`: Ini menyetel lebar elemen body menjadi 100% viewport (layar pengguna).
- `height: 100vh;`: Ini mengatur tinggi elemen body menjadi 100% dari tinggi viewport (VH adalah singkatan dari viewport height).
- `background-size: 50%;`: Ini mengatur ukuran gambar latar belakang menjadi 50% dari lebar dan tinggi elemen body.

Background Repeat

PENJELASAN

`Background-repeat` digunakan untuk mengatur bagaimana sebuah gambar background akan diulang atau tidak diulang.

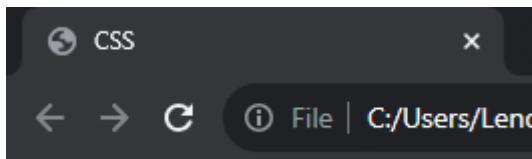
KODE PROGRAM

```
body {  
    background-image: url('5.jpg');
```

```
background-repeat: repeat;  
background-size: auto;  
}
```

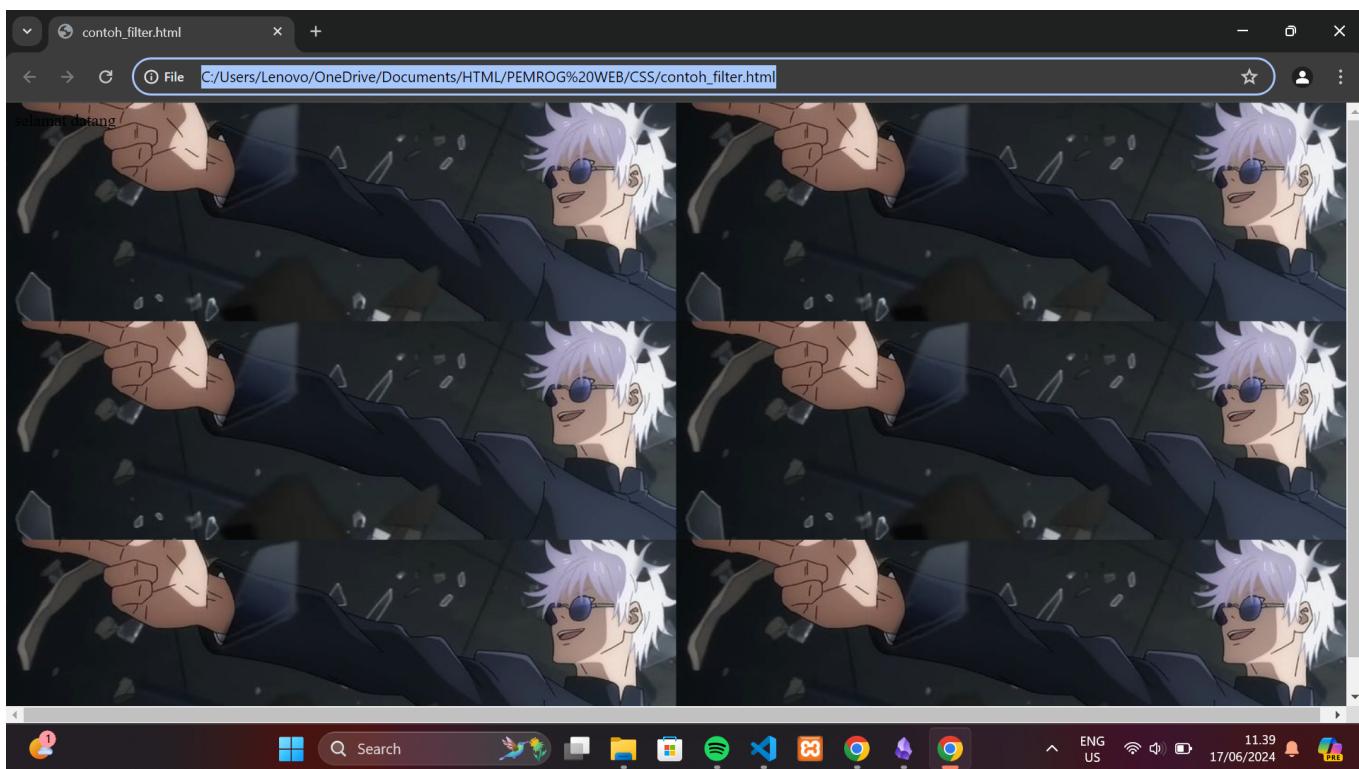
HASIL

Before



Selamat Datang

After



KESIMPULAN

- gambar latar belakang elemen `body` menggunakan gambar "5.jpg". Asumsi gambar berada di folder yang sama dengan file HTML (jika menggunakan path relatif).
- `background-repeat: repeat;` memerintahkan browser untuk mengulang gambar secara horizontal dan vertikal. Ini berarti gambar "5.jpg" akan memenuhi seluruh area latar belakang elemen `body`.

- `background-size: auto;` menentukan bahwa gambar akan mempertahankan proporsi aslinya saat menyesuaikan diri dengan elemen `body`. Jika gambar lebih besar dari elemen `body`, gambar akan diperkecil agar muat. Sebaliknya, jika gambar lebih kecil, gambar tidak akan diregang dan akan muncul di tengah.

| Background Attachment

| PENJELASAN

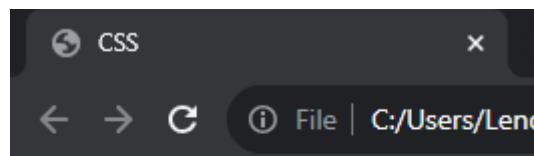
`Background-attachment` digunakan untuk mengatur bagaimana sebuah gambar background akan bergerak atau tidak bergerak dalam halaman.

| KODE PROGRAM

```
.background-fixed {  
    background-attachment: fixed;  
    background-image: url('5.jpg');
```

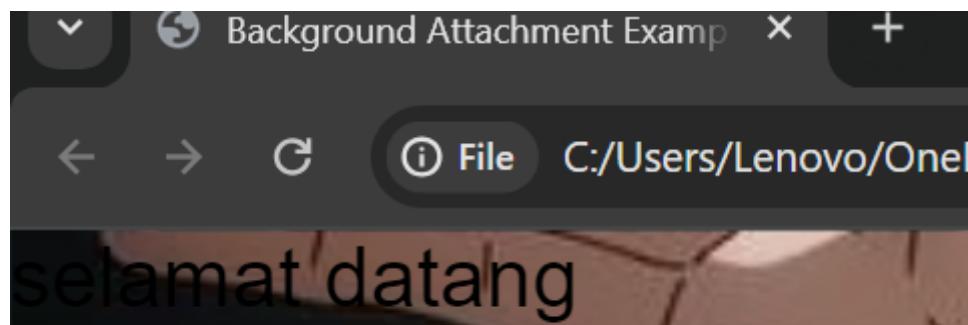
| HASIL

| Before



Selamat Datang

| After



| KESIMPULAN

- `background-attachment: fixed;` membuat gambar latar menempel pada posisi awal walaupun konten di dalam elemen tersebut digulir.
- `background-image:` Kode ini juga mengatur gambar latar belakang menjadi "5.jpg".

| Background Position

| PENJELASAN

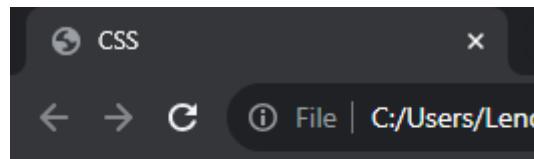
Background position adalah properti CSS yang digunakan untuk mengatur posisi gambar latar belakang pada suatu elemen.

| KODE PROGRAM

```
.center {  
    background-position: center center;  
}
```

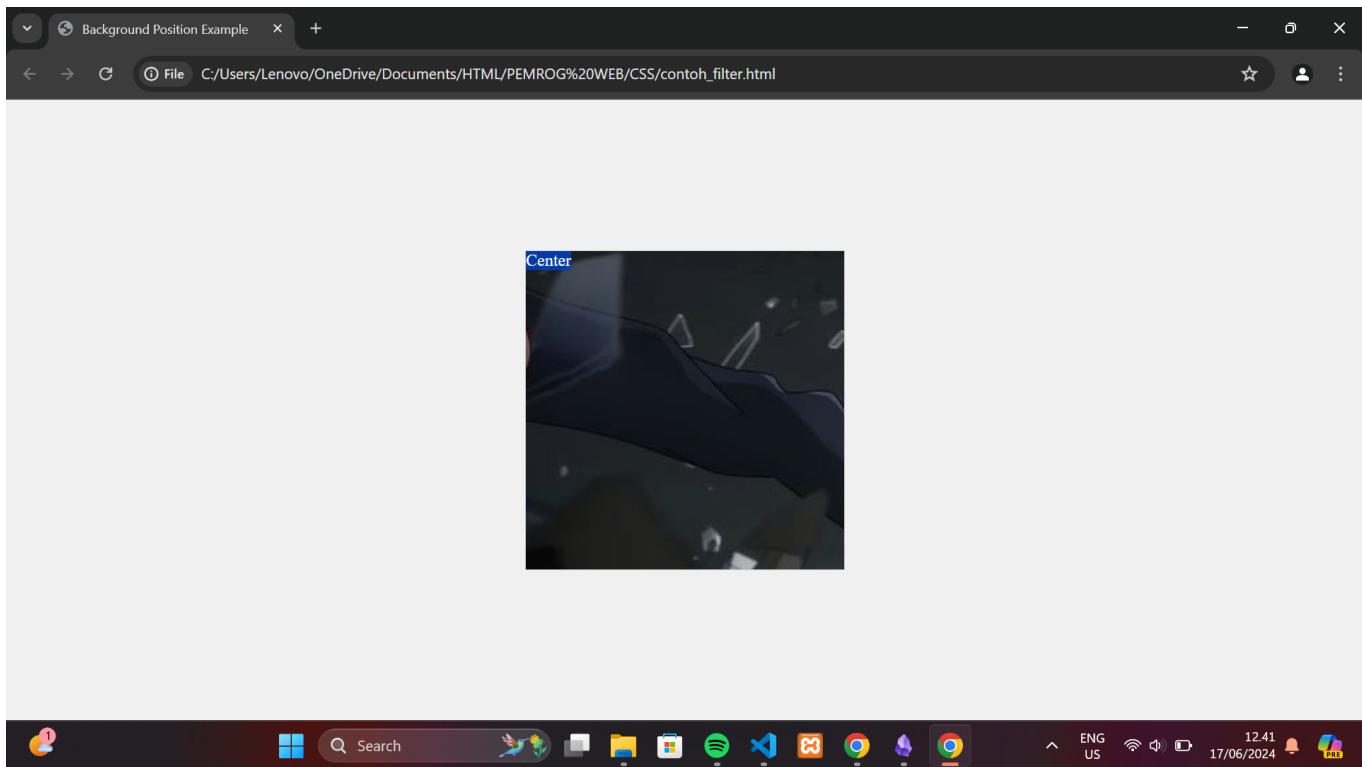
| HASIL

| Before



Selamat Datang

| After



| KESIMPULAN

`background-position: center center;` menginstruksikan browser untuk menempatkan gambar latar belakang di titik horizontal dan vertikal tengah elemen yang memiliki class `.center`.

| Font

| Font-size

| Penjelasan

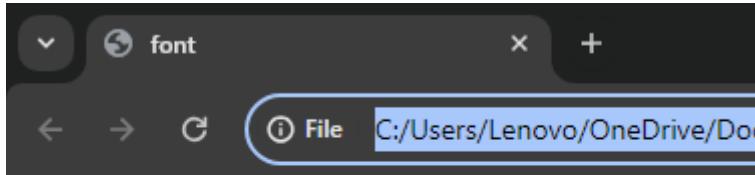
`font-size` digunakan untuk mengatur ukuran teks dalam elemen HTML. Properti ini dapat menerima berbagai jenis nilai untuk menentukan ukuran teks, termasuk panjang absolut, panjang relatif, kata kunci, dan unit persentase.

| Kode Program

```
<head>
    <title>font</title>
    <style>
        .size {
            font-size: 100px;
        }
    </style>
</head>
<body>
```

```
<p class="size">mada</p>
</body>
```

Hasil



mada

Kesimpulan

- Judul halaman:** Bagian `<title>font</title>` menetapkan judul halaman menjadi "font". Ini tidak berpengaruh terhadap tampilan teks "mada".
- Kelas CSS untuk ukuran font:** Bagian `<style>` mendefinisikan sebuah class CSS bernama `.size` yang mengatur `font-size` (ukuran font) menjadi 100px.
- Menerapkan class CSS:** Paragraf `<p>` memiliki class "size". Ini berarti paragraf tersebut akan mengikuti ketentuan yang diatur di class `.size`, yaitu ukuran font menjadi 100px.
- Teks "mada":** Karena paragraf menggunakan class ".size", maka teks "mada" akan ditampilkan dengan ukuran font 100px, membuatnya terlihat sangat besar.

Font-weight

Penjelasan

`font-weight` digunakan untuk mengatur ketebalan font. Properti ini memungkinkan Anda untuk menyesuaikan berat atau ketebalan teks dalam elemen HTML.

Kode Program

```
<head>
  <title>css</title>
```

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

| Hasil

mada

| Kesimpulan

- Dalam halaman ini, satu-satunya elemen yang diberi gaya adalah elemen `<p>` (paragraf).
- Semua paragraf di halaman ini akan memiliki teks yang ditampilkan dalam tebal (bold) karena aturan gaya CSS yang ditentukan.
- Penggunaan CSS di dalam tag `<style>` memungkinkan penentuan gaya yang konsisten dan terpusat untuk elemen-elemen tertentu di seluruh halaman web.

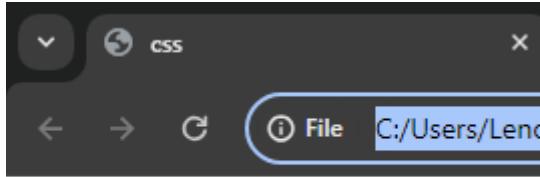
| Font-style

| Penjelasan

| Kode Program

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

| Hasil



mada

| Kesimpulan

- Dalam halaman ini, semua elemen `<p>` (paragraf) akan memiliki teks yang ditampilkan dalam gaya miring (italic) karena aturan gaya CSS yang ditentukan.
- Penggunaan CSS di dalam tag `<style>` memungkinkan penentuan gaya yang konsisten untuk elemen-elemen tertentu di seluruh halaman web.

| Font-family

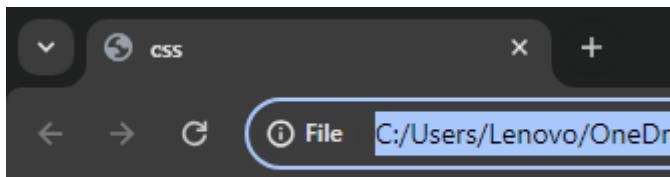
| Penjelasan

`font-family` digunakan untuk menentukan jenis font yang akan digunakan untuk teks di dalam elemen HTML.

| Kode Program

```
p {  
    font-family:'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif  
}
```

| Hasil



mada

| Kesimpulan

- Aturan gaya di atas mengatur jenis font untuk semua elemen paragraf di halaman web.
- Dengan urutan ini, desainer mencoba untuk memastikan bahwa teks di halaman akan menggunakan jenis font yang memiliki gaya yang diinginkan, mulai dari preferensi tertentu hingga jenis font generik sebagai cadangan.

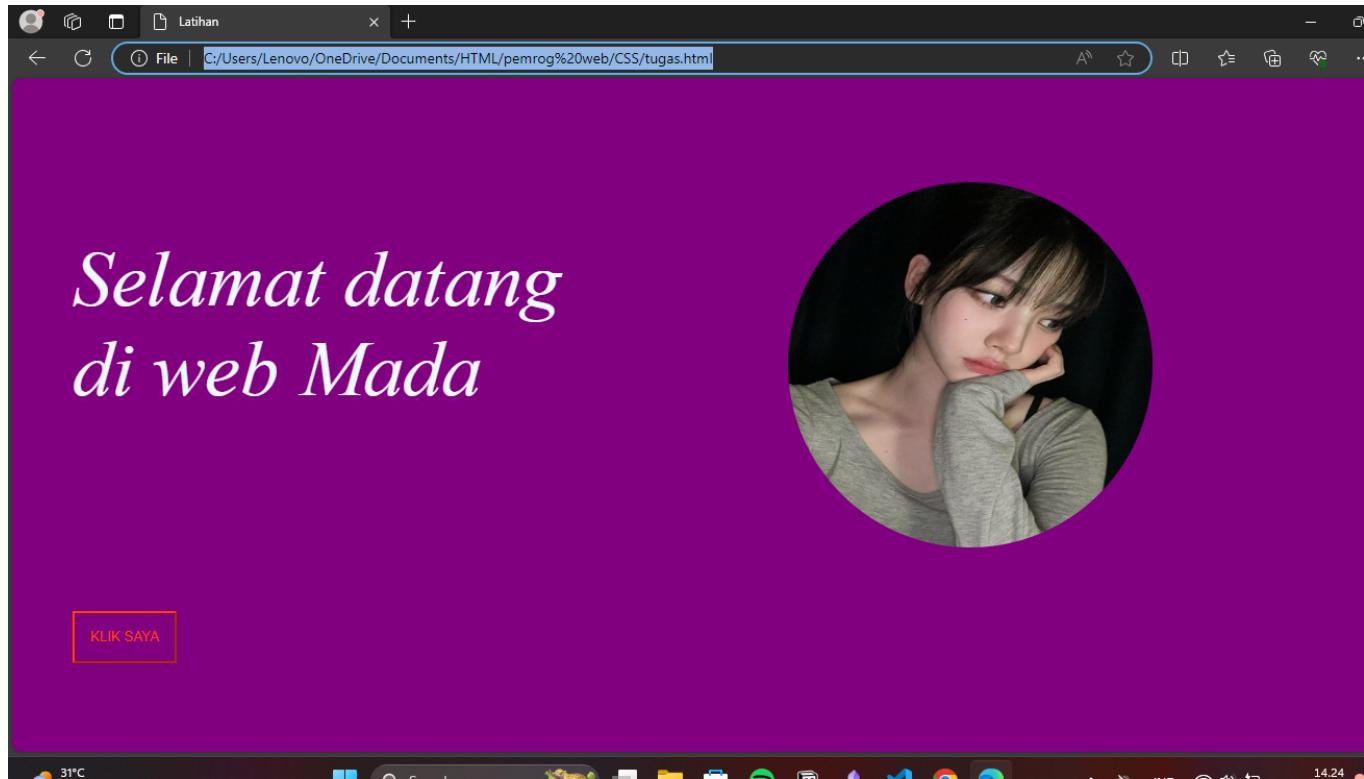
| Latihan box model

| Kode program

```
p {  
    font-size: 75px;  
    font-family: 'times new roman';  
    margin-top: 150px;  
    margin-bottom: 100px;  
    margin-left: 50px;  
    margin-right: 100px;  
    font-style: italic;  
    color: aliceblue;  
}  
  
img {  
    margin-right: 200px;  
    margin-top: -50px;  
    border: 10px;  
    border-radius: 1500px 1500px;  
}  
  
button {  
    background-color: purple;  
    width: 100px;  
    height: 50px;  
    border-width: 2px;  
    color: orangered;  
}
```

```
border-color: orangered;  
}
```

| Hasil



| Kesimpulan

Paragraf :

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

image :

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

button :

- **Warna ungu**: Background color `purple` memberikan kesan profesional dan modern pada tombol.
- **Ukuran sedang**: Lebar 100px dan tinggi 50px membuat tombol berukuran sedang dan mudah diklik.
- **Garis tepi oranye**: Border width 2px dan color `orangered` membuat garis tepi tombol berwarna oranye dan terlihat jelas.

| Box-model

| Border

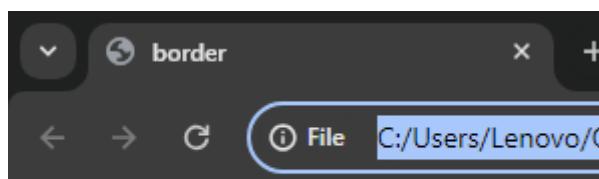
| Penjelasan

`border` digunakan untuk menambahkan garis tepi (border) di sekitar elemen HTML. Properti ini memungkinkan Anda untuk mengontrol gaya (jenis, warna, lebar) dan posisi dari garis tepi tersebut.

| Kode program

```
Button {  
    border-color: plum;  
    border-width: 10px;  
    border-style: solid;  
}
```

| Hasil



| Kesimpulan

- Aturan CSS di atas mengatur elemen `<button>` di halaman web untuk memiliki border dengan warna `plum`, lebar 10 piksel, dan gaya solid.

- Pengaturan ini akan memberikan tampilan yang jelas dan terdefinisi pada tombol di halaman web Anda, membedakannya dari elemen lain dan menyesuaikan dengan desain visual yang diinginkan.

Margin

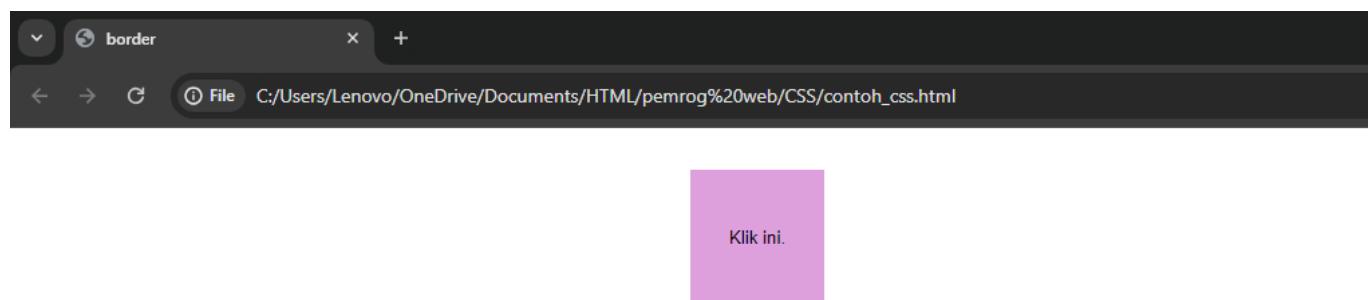
Penjelasan

`margin` digunakan untuk mengatur ruang atau jarak di sekeliling elemen HTML. Margin adalah area kosong antara batas luar elemen dan elemen lain di sekitarnya. Properti `margin` memungkinkan Anda untuk mengatur ruang atas, kanan, bawah, dan kiri dari suatu elemen.

Kode program

```
Button {  
    background-color: plum;  
    width: 100px;  
    height: 100px;  
    border: 0px;  
    margin-top: 600px;  
    margin-bottom: 600px;  
    margin-left: 500px;  
    margin-right: 100px;  
}
```

Hasil



Kesimpulan

- CSS di atas mengatur tampilan tombol (`<button>`) di halaman web dengan latar belakang ungu muda (`plum`), ukuran 100x100 piksel, tanpa border, dan dengan margin yang cukup besar di sekitarnya.
- Margin yang besar seperti yang didefinisikan (`600px` untuk atas dan bawah, `500px` untuk kiri, dan `100px` untuk kanan) akan menyebabkan tombol terletak jauh dari elemen lain di halaman web.

| Padding

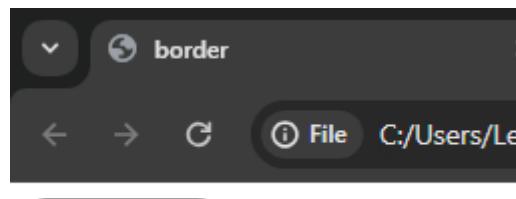
| Penjelasan

`padding` digunakan untuk menentukan ruang atau jarak di sekitar konten dalam suatu elemen HTML. Padding adalah area kosong di antara batas dalam elemen dan konten di dalamnya. Properti `padding` memungkinkan Anda untuk mengatur ruang atas, kanan, bawah, dan kiri dari suatu elemen.

| Kode program

```
Button {  
    background-color: plum;  
    width: 100px;  
    height: 50px;  
    border: 5px solid black;  
    border-radius: 10px 15px;  
    padding-left: 0px;  
    padding-bottom: 50px;  
    padding-right: 50px;  
    padding-top: 50px;  
}
```

| Hasil



| Kesimpulan

- CSS di atas mengatur tampilan tombol (`<button>`) di halaman web dengan latar belakang ungu muda (`plum`), ukuran 100x50 piksel, border hitam dengan sudut bulat, dan padding yang disesuaikan untuk menciptakan tampilan dan tata letak yang diinginkan.
- Penggunaan `border-radius` untuk sudut bulat dan `padding` yang berbeda-beda memberikan detail visual yang menarik pada tombol.

| Pseudo class

| Haver

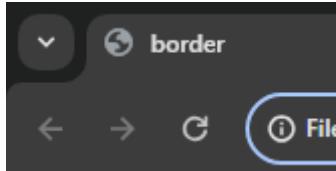
- `button:hover`: adalah pseudo-class selector yang menargetkan tombol (button) ketika pengguna mengarahkan kursor ke atasnya (hover) atau diklik.
- `background-color: red;` : Properti ini menetapkan warna latar belakang tombol menjadi merah ketika tombol sedang di-hover.
- `color: white;` : Properti 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.

| Kode Program

```
button:hover{  
    background-color:red;  
    color:white;  
    height:100px;  
    width:100px;  
}
```

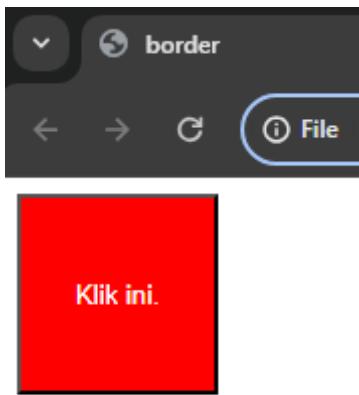
| Hasil

| Before



Klik ini.

| After



| Kesimpulan

| Active

`button:active` : adalah pseudo-class selector yang menargetkan tombol (`button`) ketika tombol tersebut sedang dalam keadaan "active", atau tombol sedang ditekan.

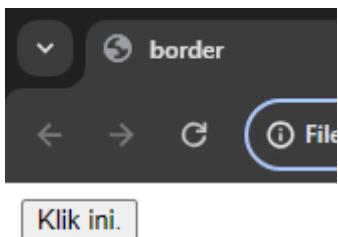
`color: yellow;` : Properti memberikan warna kuning pada teks ketika tombol sedang active atau tombol sedang ditekan.

| Kode Program

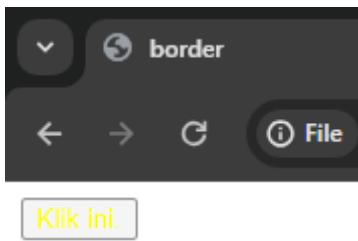
```
button:active{  
    color: yellow ;  
}
```

| Hasil

| Before



| After



| Kesimpulan

Teks klik akan berganti warna menjadi kuning jika ditekan dan berputar.

| Transition

| Penjelasan

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

Transition-property menentukan property apa yang akan diberi efek Transition, seperti width,color, atau all(semua property).

transition-duration menentukan waktu berapa lama efek Transitionnya di ukur dalam detik seperti 0,5s.

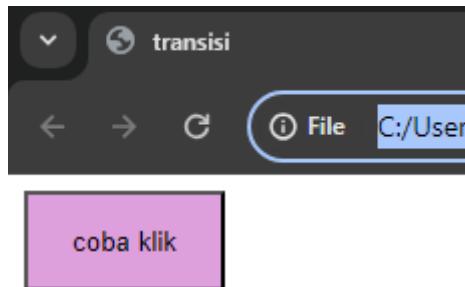
transition-timing-function menentukan bagaimana kecepatan perubahan nilai properti selama transisi. Seperti **ease** : lambat diawal, cepat di tengah, dan lambat di akhir **ease-in** : dimulai dengan lambat dan semakin cepat seiring waktu **ease-out** : cepat diawal dan lambat diakhir **ease-in-out** : Kombinasi dari ease-in dan ease-out, menciptakan efek transisi mulai lambat, cepat di tengah, dan melambat kembali **linear** : memberikan perubahan langsung secara konstan

| Kode Program

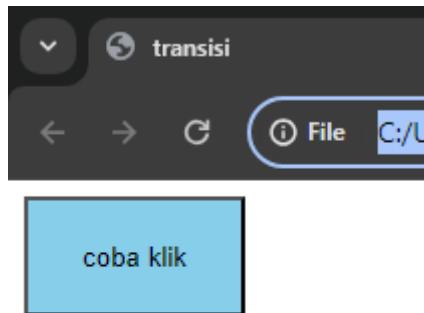
```
button {  
    height: 50px;  
    width: 100px;  
    background-color: plum;  
}  
  
button:hover {  
    height: 60px;  
    width: 110px;  
    background-color: skyblue;  
    transition: all 0.5s ease-in-out;  
}
```

| Hasil

| before



| after



| Tantangan Transition

| Kode Program

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

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

```
.ody {
background-color: rgb(45, 178, 255);
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 pl;
border-radius: 1500px 1500px;
}

button {
background-color: lightblue;
width: 150px;
height: 50px;
border-width: 2px;
color: rgb(138, 138, 229);
border-color: rgba(243, 107, 182, 0.685);
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 Mada !

klik saya



| Kesimpulan

1. `body`:

- Warna latar belakang diatur menjadi ungu (`background-color: purple`).

2. `span`:

- Teks dalam elemen span berwarna putih (`color: white`).
- Ukuran font diatur menjadi 30 piksel (`font-size: 30px`).

3. `button`:

- Teks pada tombol berwarna oranye-merah (`color: orangered`).
- Latar belakang tombol juga ungu (`background-color: purple`).
- Batas tombol berwarna oranye-merah (`border-color: orangered`).
- Ukuran tombol adalah 100 piksel untuk lebar dan 50 piksel untuk tinggi.
- Ada margin kiri 100 piksel dan margin bawah 1 piksel.
- Pada saat tombol di-hover (diseleksi dengan mouse), tinggi bertambah menjadi 60 piksel, lebar menjadi 110 piksel, latar belakang berubah menjadi biru langit (`background-color: skyblue`), dan transisi terjadi dalam 0.5 detik.

1. `img`:

- Lebar dan tinggi gambar diatur menjadi 130 piksel.
- Gambar memiliki bentuk bulat dengan radius 100 piksel (`border-radius: 100px`).
- Warna batas gambar hitam dengan gaya solid.
- Margin kiri 250 piksel dan margin atas negatif 150 piksel.

| Transform

| Scale

| Penjelasan

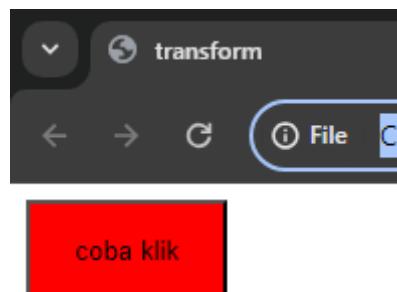
digunakan untuk mengubah tampilan suatu elemen, seperti memindahkannya, memutarinya, atau mengubah ukurannya. nilai Scale didefinisikan dengan menyebut faktor perubahannya seperti Scale(2) akan memperbesar elemen menjadi 2 kali lipat.

| Kode Program

```
button {
    height: 50px;
    width: 100px;
    background-color: red;
}

button:hover {
    height: 60px;
    width: 110px;
    background-color: plum;
    transform:scale(0.5);
}
```

| Hasil



| Kesimpulan

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

| ScaleX

| Penjelasan

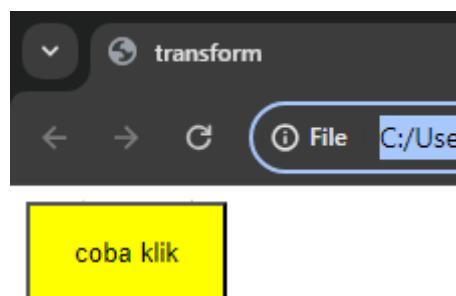
ScaleX Property CSS yang digunakan untuk mengubah skala elemen hanya pada sumbu horizontal saja. Misalnya, transform: scaleX(2) akan membuat elemen menjadi dua kali lebih lebar dari ukurannya.

| Kode Program

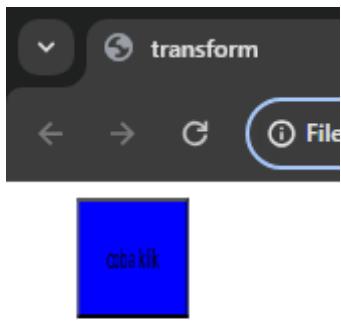
```
button {  
    height: 50px;  
    width: 100px;  
    background-color: yellow;  
}  
  
button:hover {  
    height: 60px;  
    width: 110px;  
    background-color: blue;  
    transform:scaleX(0.5);  
}
```

| hasil

| Before



| after



| Kesimpulan

Efek hover ini mengubah ukuran tombol dan warna saat kursor mouse ditempatkan di atasnya, memberikan interaksi visual yang menarik bagi pengguna.

| Posision

| Position relativ

| Penjelasan

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

| Kode program

```
.container {  
    position: relative;  
    width: 400px;  
    height: 200px;  
    border: 2px solid #333;  
}  
  
.box {  
    position: relative;  
    width: 100px;  
    height: 100px;  
    background-color: #333;  
    color: #fff;  
    text-align: center;  
    line-height: 100px;  
}  
...  
### Hasil  
![] (aset/potret42.png)
```

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

## Position absolute

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

### Kode program
```css
.container {
 position: relative;
 width: 400px;
 height: 300px;
 border: 1px solid black;
}

.box {
 position: absolute;
 top: 50px;
 left: 50px;
 width: 200px;
 height: 100px;
 background-color: white;
 border: 1px solid red;
}
}
```

## | Hasil

“css53.png” could not be found.

## | Kesimpulan

Dengan demikian, aturan gaya di atas akan membuat elemen dengan kelas ".box" ditempatkan secara absolut di dalam elemen dengan kelas ".container". Elemen ".container" memiliki lebar 400 piksel dan tinggi 300 piksel dengan border hitam 1 piksel. Elemen ".box" memiliki lebar 200 piksel, tinggi 100 piksel, latar belakang putih, dan border merah 1 piksel. Elemen ".box" akan ditempatkan 50 piksel dari atas dan 50 piksel dari kiri elemen ".container".

## | Position fixed

## | Penjelasan

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

## | Kode program

```
.fixed-element {
 position: fixed;
 top: 20px;
 left: 20px;
 width: 200px;
 height: 100px;
 background-color: #f0f0f0;
 border: 1px solid #ccc;
 padding: 10px;
}
```

## | Hasil

"css54.png" could not be found.

## | Kesimpulan

Jadi, aturan gaya di atas akan membuat elemen dengan kelas ".fixed-element" ditempatkan secara tetap di jendela penampilan dengan jarak 20 piksel dari atas dan 20 piksel dari kiri jendela. Elemen tersebut akan memiliki lebar 200 piksel, tinggi 100 piksel, latar belakang abu-abu muda, border dengan garis abu-abu muda, dan jarak isi sebesar 10 piksel dari tepi border.

## | Position sticky

### | Penjelasan

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

## | Kode program

```
.header {
 background-color: #333;
 color: #fff;
 padding: 10px 0;
 text-align: center;
 position: sticky;
 top: 0;
 z-index: 1000;
}
```

```
.content {
 padding: 20px;
 height: 1000px;
}
```

## | Hasil

### | Before

"css55.png" could not be found.

### | After

"css56.png" could not be found.

## | Kesimpulan

Jadi, aturan gaya di atas akan memberikan elemen dengan kelas ".header" latar belakang gelap, teks putih, jarak isi, dan teks yang rata tengah. Elemen tersebut juga akan melekat di bagian atas jendela penampilan, memiliki tingkatan tumpukan 1000, dan tidak akan terpengaruh oleh pengguliran halaman. Elemen dengan kelas ".content" akan memiliki jarak isi dan tinggi tertentu sesuai dengan properti yang ditentukan.

## | Tantangan Posision

### | 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
Lorem Ipsum</h1>
 <p class="item box-4">Sed posuere consectetur est at lobortis.

Aeneen eu leo quam</p>
 <p class="box-5">Read more </p>
 </div>
</body>
```

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

```
.container {
 position: static;
 display: flex;
 flex-direction: column;
 width: 100%;
 height: 580px;
 background-color: rgba(122, 122, 247, 0.628);
}

.box {
 width: 250px;
 height: 350px;
 background-color: white;
 align-items: center;
 align-content: center;
 border-radius: 10px;
 align-self: center;
 margin-top: 150px;
 margin-bottom: 200px;
}

.item {
 width: 100%;
 color: black;
}

.box-1 {
 height: 225px;
 width: 100%;
 border-radius: 10px 10px 0px 0px;
}

.box-2 {
 font-size: 10px;
 margin-left: 20px;
 margin-top: 20px;
 padding-top: 10px;
 font-family: Arial, Helvetica, sans-serif;
}

.box-3 {
 font-size: 18px;
 margin-left: 20px;
 margin-top: 10px;
```

```

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

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

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

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

.icon {
 background-color: skyblue;
 position: relative;
 left: 200px;
 bottom: 35px;
 width: 30px;
 height: 30px;
 border-radius: 1500px;
}

```

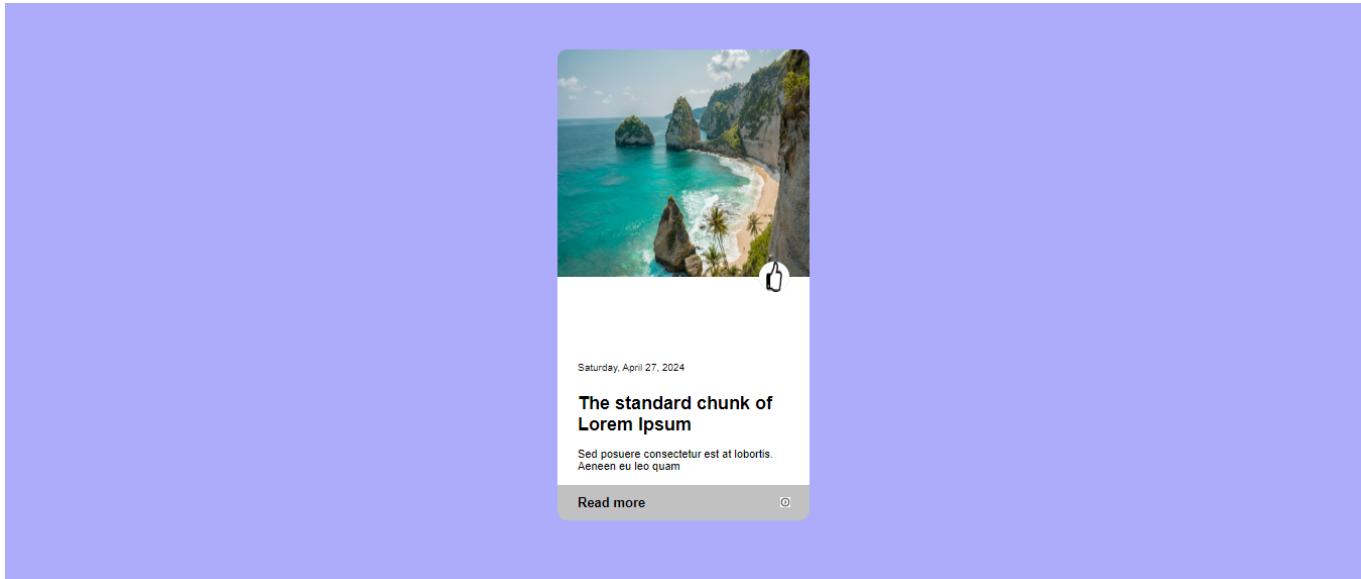
## | Penjelasan

1. **.container**: Ini adalah kelas CSS yang diterapkan pada elemen kontainer. Properti yang diatur di sini adalah:
  - **position: static;** : Menetapkan posisi elemen kontainer sebagai statis, yang berarti elemen tersebut akan mengikuti aliran dokumen normal.
  - **display: flex;** : Mengatur elemen kontainer sebagai kontainer fleksibel yang menggunakan model tata letak flexbox.
  - **flex-direction: column;** : Menentukan arah tata letak fleksibel menjadi kolom, sehingga elemen-elemen dalam kontainer akan ditampilkan dalam kolom vertikal.
  - **width: 100%;** : Mengatur lebar kontainer menjadi 100% dari lebar elemen induknya.

- `height: 580px;` : Mengatur tinggi kontainer menjadi 580 piksel.
  - `background-color: rgba(122, 122, 247, 0.628);` : Mengatur warna latar belakang kontainer menggunakan kode warna RGBA.
2. `.box`: Ini adalah kelas CSS yang diterapkan pada elemen "box". Properti yang diatur di sini adalah:
- `width: 250px;` : Mengatur lebar elemen "box" menjadi 250 piksel.
  - `height: 350px;` : Mengatur tinggi elemen "box" menjadi 350 piksel.
  - `background-color: white;` : Mengatur warna latar belakang elemen "box" menjadi putih.
  - `align-items: center;` : Mengatur penempatan horisontal item-item dalam elemen "box" menjadi di tengah.
  - `align-content: center;` : Mengatur penempatan vertikal item-item dalam elemen "box" menjadi di tengah.
  - `border-radius: 10px;` : Mengatur radius lengkung sudut elemen "box" menjadi 10 piksel.
  - `align-self: center;` : Mengatur penempatan elemen "box" secara horizontal menjadi di tengah relatif terhadap kontainer.
  - `margin-top: 150px;` : Mengatur margin atas elemen "box" sebesar 150 piksel.
  - `margin-bottom: 200px;` : Mengatur margin bawah elemen "box" sebesar 200 piksel.
3. `.item`: Ini adalah kelas CSS yang diterapkan pada elemen "item". Properti yang diatur di sini adalah:
- `width: 100%;` : Mengatur lebar elemen "item" menjadi 100% dari lebar elemen "box".
  - `color: black;` : Mengatur warna teks elemen "item" menjadi hitam.
4. `.box-1, .box-2, .box-3, .box-4, .box-5`: Ini adalah kelas-kelas CSS yang diterapkan pada elemen-elemen di dalam elemen "box". Properti yang diatur di sini bervariasi tergantung pada elemen yang bersangkutan, seperti tinggi, lebar, ukuran font, margin, dan warna teks.
5. `.item-1`: Ini adalah kelas CSS yang diterapkan pada elemen "item-1". Properti yang diatur di sini adalah:
- `width: 10px;` : Mengatur lebar elemen "item-1" menjadi 10 piksel.
  - `padding-left: 130px;` : Mengatur padding kiri elemen "item-1" sebesar 130 piksel.
6. `.icon`: Ini adalah kelas CSS yang diterapkan pada elemen "icon". Properti yang diatur di sini adalah:
- `background-color: skyblue;` : Mengatur warna latar belakang elemen "icon" menjadi biru langit.
  - `position: relative;` : Menetapkan posisi elemen "icon" sebagai relatif terhadap posisi normalnya.
  - `left: 200px;` : Menggeser elemen "icon" sebesar 200 piksel ke kanan.
  - `bottom: 35px;` : Menggeser elemen "icon" sebesar 35 piksel ke bawah.
  - `width: 30px;` : Mengatur lebar elemen "icon" menjadi 30 piksel.
  - `height: 30px;` : Mengatur tinggi elemen "icon" menjadi 30 piksel.
- elemen "icon" dalam gaya CSS yang Anda berikan adalah elemen dengan latar belakang biru langit (`background-color: skyblue;`). Elemen ini memiliki posisi relatif (`position: relative;`) dan diposisikan 200

piksel ke kanan (`left: 200px;`) dan 35 piksel ke bawah (`bottom: 35px;`) dari posisi normalnya. Elemen "icon" memiliki lebar 30 piksel (`width: 30px;`) dan tinggi 30 piksel (`height: 30px;`). Elemen ini juga memiliki bentuk bulat dengan radius border 1500 piksel (`border-radius: 1500px;`), sehingga menghasilkan tampilan lingkaran.

## | Hasil



## | Kesimpulan

1. Elemen dengan kelas `.container` adalah kontainer utama. Kontainer ini memiliki tinggi 580 piksel, latar belakang berwarna biru muda dengan transparansi, dan menggunakan model tata letak flexbox dengan arah kolom.
2. Elemen dengan kelas `.box` adalah kotak di dalam kontainer. Kotak ini memiliki lebar 250 piksel, tinggi 350 piksel, latar belakang putih, dan ditempatkan di tengah horizontal dan vertikal. Kotak juga memiliki radius sudut 10 piksel, dan dimargin-top sebesar 150 piksel dan margin-bottom sebesar 200 piksel relatif terhadap kontainer.
3. Elemen dengan kelas `.item` adalah item di dalam kotak. Item ini memiliki lebar 100% dari kotak dan teks berwarna hitam.
4. Elemen dengan kelas `.box-1` adalah bagian pertama dari kotak. Bagian ini memiliki tinggi 225 piksel, lebar 100%, dan radius sudut 10 piksel di sudut atas.
5. Elemen dengan kelas `.box-2`, `.box-3`, dan `.box-4` adalah bagian teks di dalam kotak. Masing-masing memiliki ukuran font, margin, dan font-family yang berbeda.
6. Elemen dengan kelas `.box-5` adalah bagian bawah kotak. Bagian ini memiliki latar belakang abu-abu, padding di sisi kiri, atas, dan bawah, dan margin bawah 90 piksel. Bagian ini juga memiliki radius sudut 10 piksel di sudut bawah.
7. Elemen dengan kelas `.item-1` adalah item khusus di dalam kotak. Item ini memiliki lebar 10 piksel dan padding kiri sebesar 130 piksel.
8. Elemen dengan kelas `.icon` adalah ikon di dalam kotak. Ikon ini memiliki latar belakang biru langit, posisi relatif, dan diposisikan 200 piksel ke kanan dan 35 piksel ke bawah dari posisi

normalnya. Ikon memiliki lebar 30 piksel, tinggi 30 piksel, dan bentuk bulat dengan radius border 1500 piksel.

## | FLEX-BOX

## | FLEX-CONTAINER

### | Penjelasan

Flex container adalah elemen induk yang mengatur tata letak flex item-nya. Untuk membuat elemen menjadi flex container, kita perlu mengatur properti display menjadi flex atau inline-flex. Untuk mengatur tata letak flex item, kita dapat menggunakan properti flex: Properti shorthand untuk flex-grow, flex-shrink, dan flex-basis. Urutan nilai yang diberikan adalah flex-grow, flex-shrink, dan flex-basis.

### | Kode Program

```
.container {
 display: flex;
 flex-direction: column;
 border: 2px solid black;
 height: 200px;
 width: 400px;
}
.item {
 background-color: green;
 border: 2px solid black;
 height: 50px;
 width: 150px;
}
```

### | Hasil

## | 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/>.
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

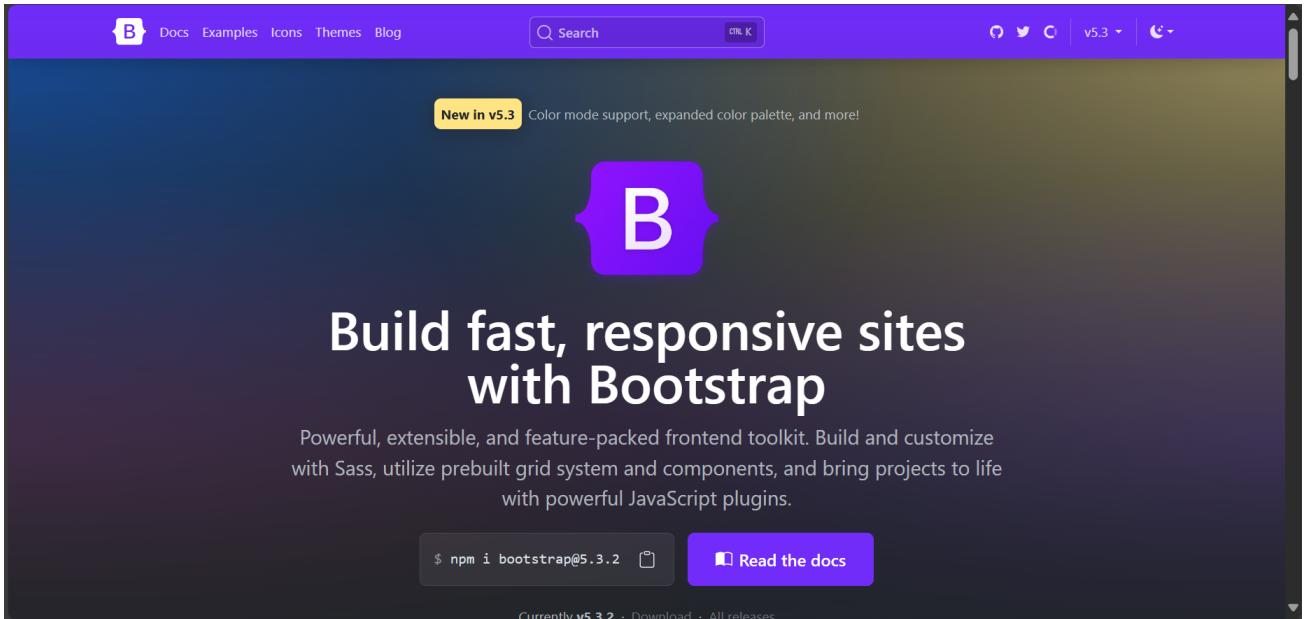
Jumbotron adalah komponen Bootstrap yang digunakan untuk menyoroti konten utama di halaman web. Biasanya ditempatkan di bagian atas halaman dengan judul besar dan terima kasih atas klarifikasiannya. Komponen carousel Bootstrap digunakan untuk membuat tampilan slide yang interaktif. kita dapat menambahkan 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/>).

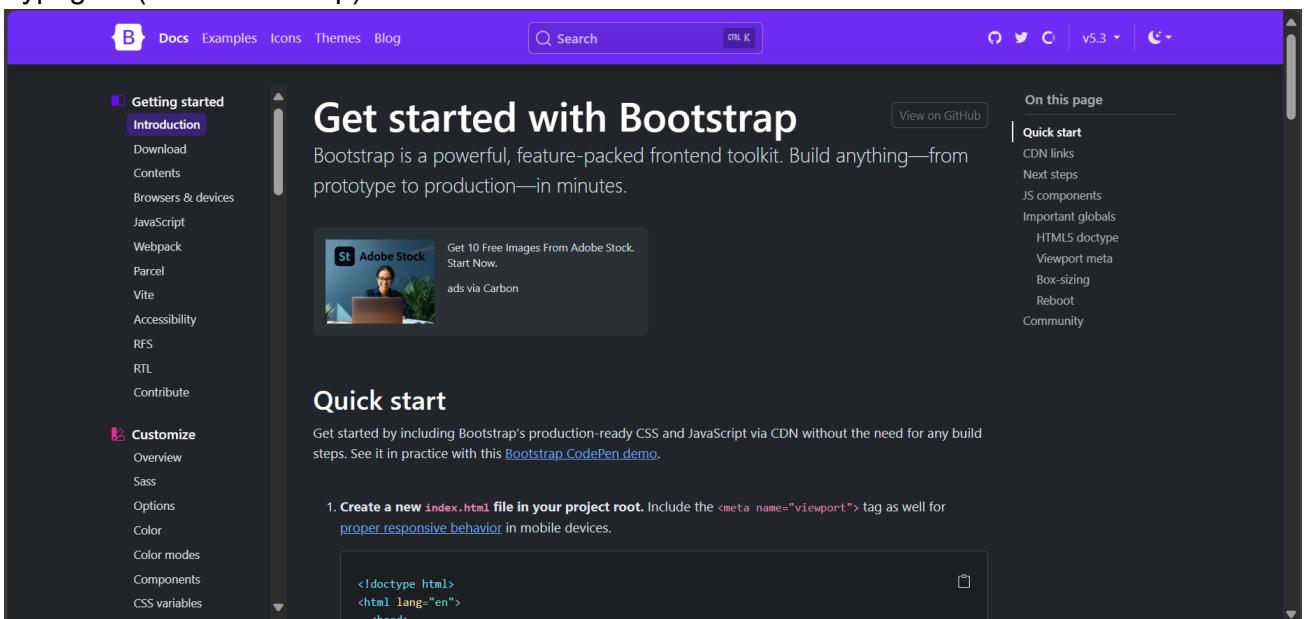
## | Contoh Penerapan Komponen Bootstrap

### | Typography

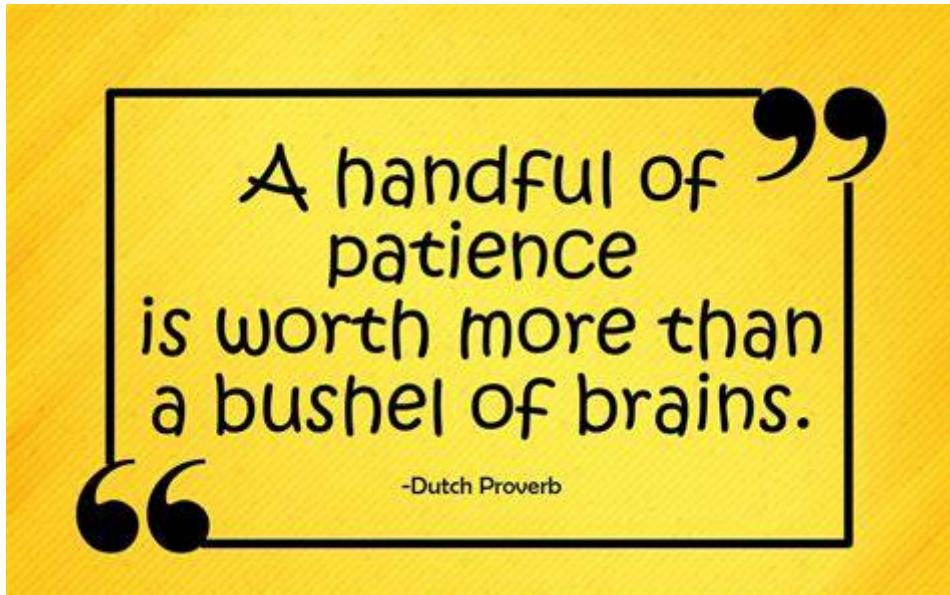
1. Buka situs resmi Bootstrap di [getbootstrap.com](https://getbootstrap.com)



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's 'Typography' page. The left sidebar has 'Typography' selected under 'Content'. The main content area shows the 'Typography' heading and its documentation. A sidebar on the right lists 'On this page' topics like Global settings, Headings, and Lists.

5. Carilah Subbab "Alignment":

# Alignment

Use text utilities as needed to change the alignment of yourblockquote.

A well-known quote, contained in a `blockquote` element.

— Someone famous in *Source Title*

HTML



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

```
<!-- tes.html -->
<!DOCTYPE html>
<html lang="en">
 <head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Document</title>
 <link rel="stylesheet" href="css/bootstrap.min.css">
 </head>
 <body>
 <figure class="text-center">
 <blockquote class="blockquote">
 | <p>A well-known quote, contained in a blockquote element.</p>
 </blockquote>
 <figcaption class="blockquote-footer">
 | Someone famous in <cite title="Source Title">Source Title</cite>
 </figcaption>
 </figure>
 </body>
</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

 membedakan hanya siapa yang

 mulai belajar duluan dan siapa yang

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

```
</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 / <code>`: Ini adalah elemen `<code>` 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 mengatakn kata-kata itu. Kata- kata yang dituliskan yaitu berada dalam tag `p`.

## Navbar

1. Buka <https://getbootstrap.com> lalu ketik "navbar" di kolom "search" dan klik enter

The screenshot shows the Bootstrap documentation for the 'Navbar' component. The left sidebar has 'Navbar' selected. The main content area has a large title 'Navbar' and a sub-section 'How it works' with a bulleted list of instructions. A note at the bottom says 'New in v5.2.0: Navbars can be themed with CSS variables that are scoped to the .navbar base class. .navbar-'.

2. Carilah jenis navbar yang diinginkan.

The screenshot shows the Bootstrap documentation for 'Image and text'. It includes an 'HTML' code editor with the following code:

```
<nav class="navbar bg-body-tertiary">
 <div class="container">

 </div>
</nav>
```

Below the code, there's a section titled 'Image and text' with a note about adding `.d-inline-block` and `.align-text-top` to the `<img>` tag. There are two examples: one with the logo centered and another where the logo is aligned to the top of the text.

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">
 Navbar
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown">

 </button>
 <div class="collapse navbar-collapse" id="navbarNavDropdown">
 <ul class="navbar-nav">
 <li class="nav-item">
 Home

 <li class="nav-item">
 Features

 <li class="nav-item">
 Pricing

 <li class="nav-item dropdown">

 Dropdown link

 <ul class="dropdown-menu" id="navbarDropdownMenuLink">
 Action
```

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
 <div class="container-fluid">
 Navbar
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">

 </button>
 <div class="collapse navbar-collapse" id="navbarNavDropdown">
 <ul class="navbar-nav">
 <li class="nav-item">
 Home

 <li class="nav-item">
 Features

 <li class="nav-item">
 Pricing

 <li class="nav-item dropdown">

```

```

data-bs-toggle="dropdown" aria-expanded="false">
 Dropdown link

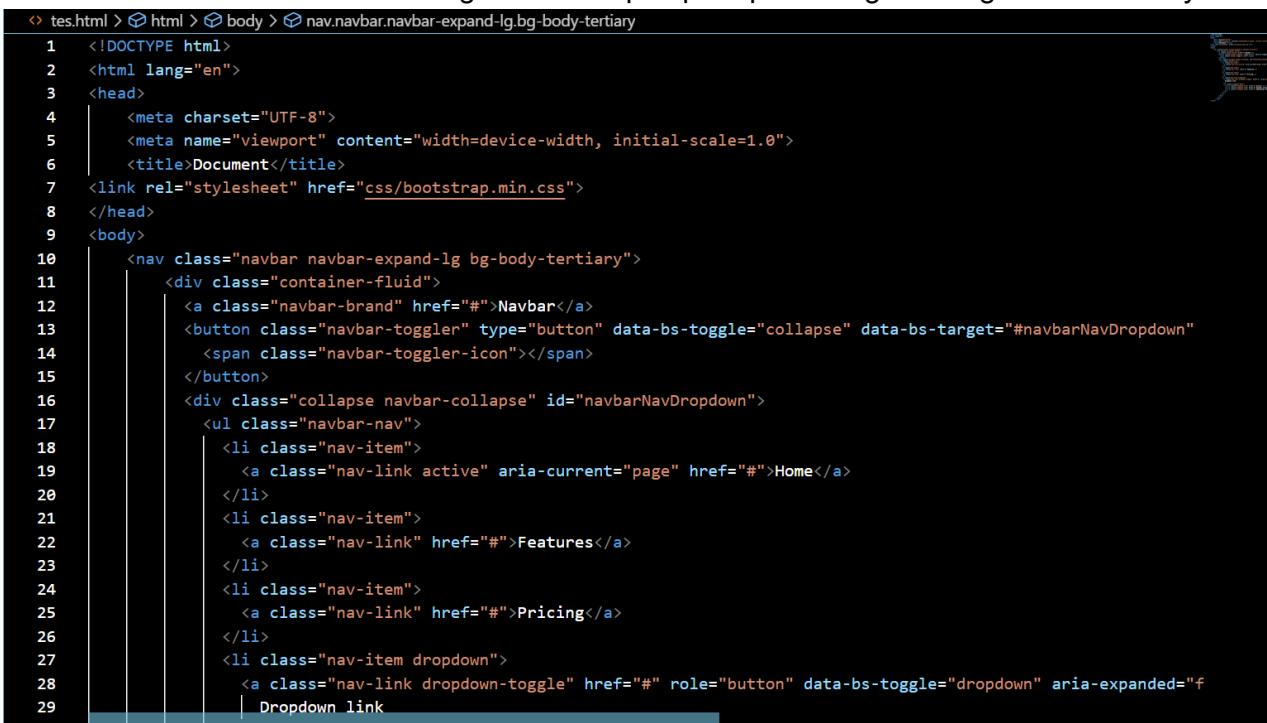
<ul class="dropdown-menu">
 Action
 Another action

 Something else here

</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



```

<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Document</title>
 <link rel="stylesheet" href="css/bootstrap.min.css">
</head>
<body>
 <nav class="navbar navbar-expand-lg bg-body-tertiary">
 <div class="container-fluid">
 Navbar
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown">

 </button>
 <div class="collapse navbar-collapse" id="navbarNavDropdown">
 <ul class="navbar-nav">
 <li class="nav-item">
 Home

 <li class="nav-item">
 Features

 <li class="nav-item">
 Pricing

 <li class="nav-item dropdown">
 <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="f
 | Dropdown link

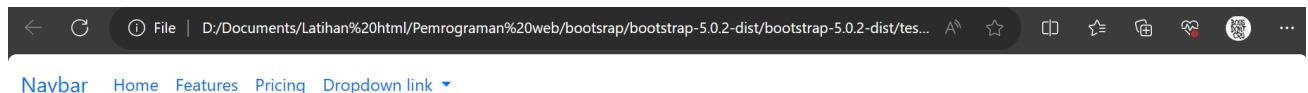
 <ul class="dropdown-menu">
 Action
 Another action

 Something else here

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

```

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

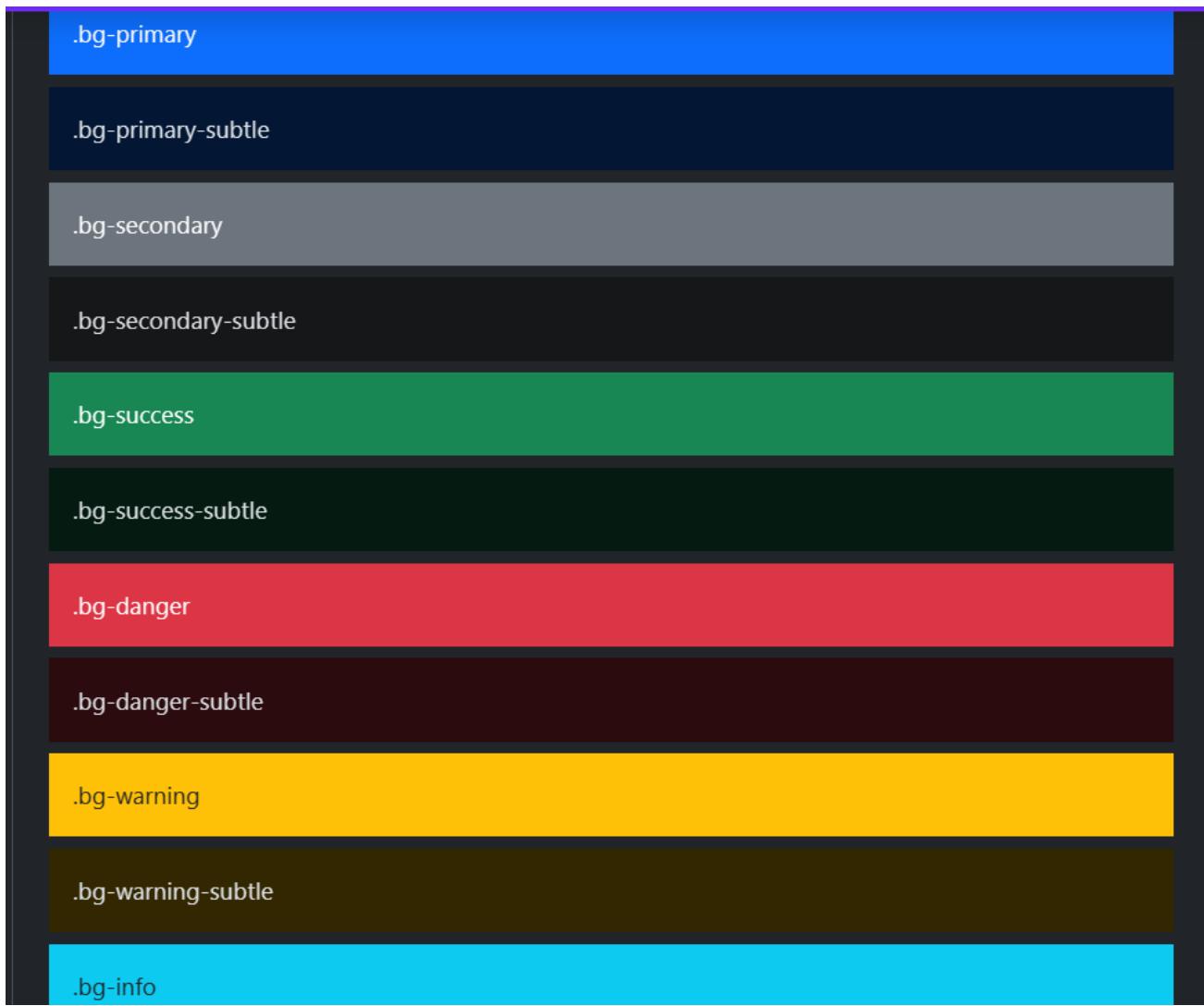


6. Misalnya kita ingin memberikan warna hijau pada background navbar kita. Pertama ketiklah "background" di kolom search

The screenshot shows the Bootstrap documentation for the 'Background' section. The sidebar on the left has 'Background' selected under the 'Utilities' category. The main content area features a large heading 'Background' and a sub-section titled 'Background color'. A tip box states: '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.' On the right side, there's a sidebar titled 'On this page' with links to 'Background color', 'Background gradient', 'Opacity', 'How it works', 'Example', and 'CSS' sections.

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





Terdapat banyak jenis warna dalam bootstrap.

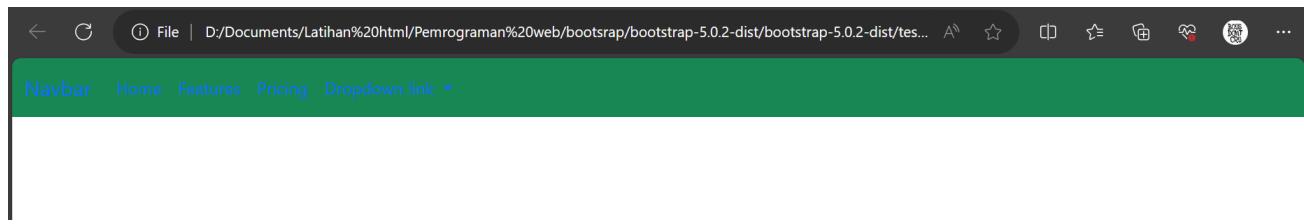
8. Karena kita ingin mengkostumisasi warna background kita menjadi hijau maka kita akan memakai bg-success . ketikkan bg-success pada class tag pembuka seperti di bawah ini

```
① <!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">
```

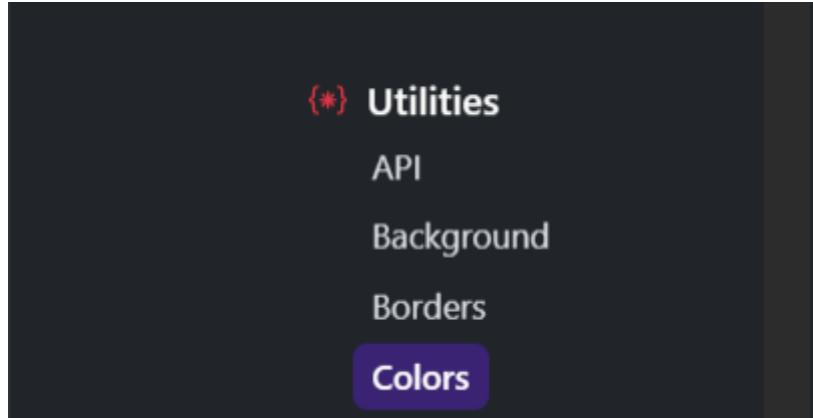
#### Keterangan:

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

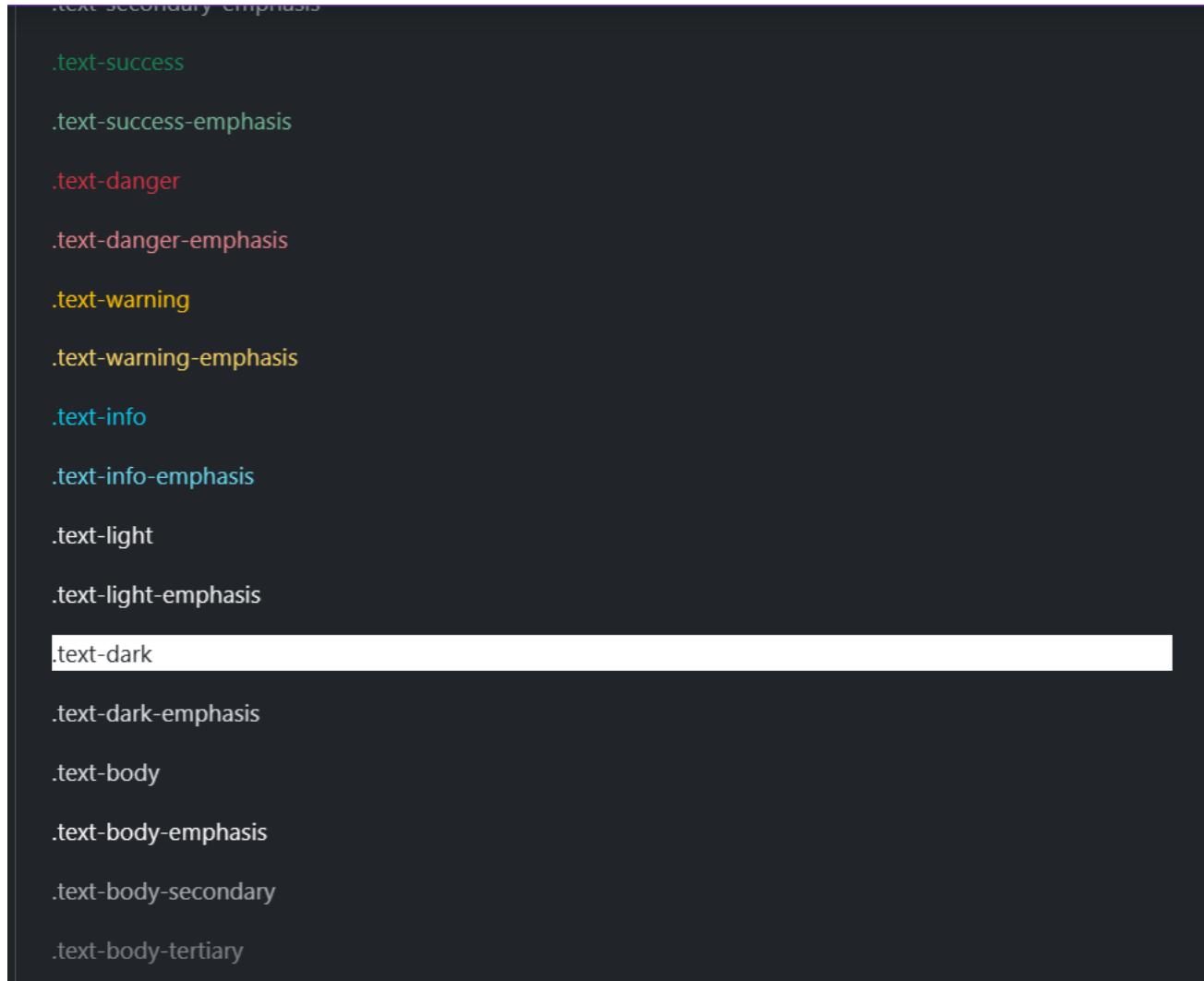
9. Jika sudah maka hasilnya akan seperti ini:



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
```

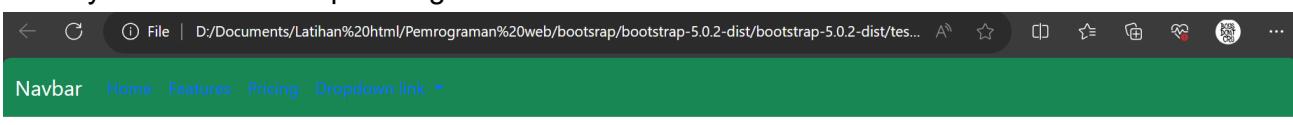
13. Karena kita ingin menggunakan warna font putih maka perintah yang digunakan adalah "text-white"

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">
 Navbar
```

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

```
19 Home
20
21 <li class="nav-item">
22 Features
23
24 <li class="nav-item">
25 Pricing
26
27 <li class="nav-item dropdown">
28 <a class="nav-link dropdown-toggle text-white" href="#" role="button" data-bs-toggle="dropdown" aria-
29 | Dropdown link
30
```

18. Jika file html dijalankan, hasilnya adalah sebagai berikut



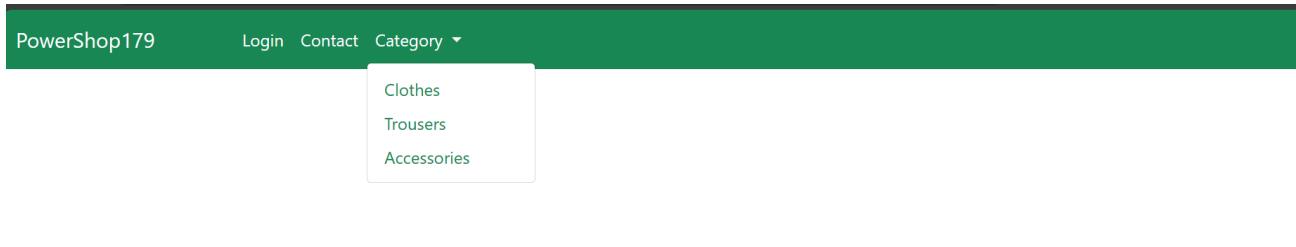
19. Sekarang kita akan mengganti text yang ada di navbar tersebut. Caranya sangat mudah, yaitu cukup kita ganti text-text tadi dengan kata kata yang kita inginkan. Misalnya adalah sebagai berikut:

```

10 <nav class="navbar navbar-expand-lg bg-success" >
11 <div class="container-fluid" >
12 PowerShop179
13 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" >
14
15 </button>
16 <div class="collapse navbar-collapse" id="navbarNavDropdown" >
17 <ul class="navbar-nav" >
18 <li class="nav-item" >
19 Home
20
21 <li class="nav-item" >
22 Login
23
24 <li class="nav-item" >
25 Contact
26
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
31 <ul class="dropdown-menu" >
32 Clothes
33 Trousers
34 Accessories
35
36
37
38 </div>

```

20. Maka hasilnya adalah sebagai berikut:



21. Sekarang kita akan menambahkan kolom search pada navbar kita. pada bagian navbar di [getbootstrap.com](https://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.navbar.bg-body-tertiary
25 | Contact
26 |
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 | Clothes
33 | Trousers
34 | Accessories
35 |
36 |
37 |
38 | </div>
39 | </div>
40 </nav>
41
42 <nav class="navbar bg-body-tertiary">
43 <div class="container-fluid">
44 <form class="d-flex" role="search">
45 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
46 <button class="btn btn-outline-success" type="submit">Search</button>
47 </form>
48 </div>
49 </nav>
50 </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.
- 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.
- 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

## Components

Accordion

Alerts

Badge

Breadcrumb

### **Buttons**

Button group

Card

Carousel

Close button

Collapse

Dropdowns

List group

Modal

Navbar

Navs & tabs

Offcanvas

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.

The DNS course  
for developers



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>
```

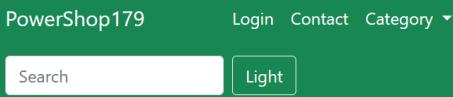
5. Blok lah terlebih dahulu kode program yang mengatur tombol pada bagian search tadi

```
<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

## 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

## 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

2. 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.</p>
 Go somewhere
 </div>
</div>
```

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

```
<div class="card">
 <div class="card-header">
```

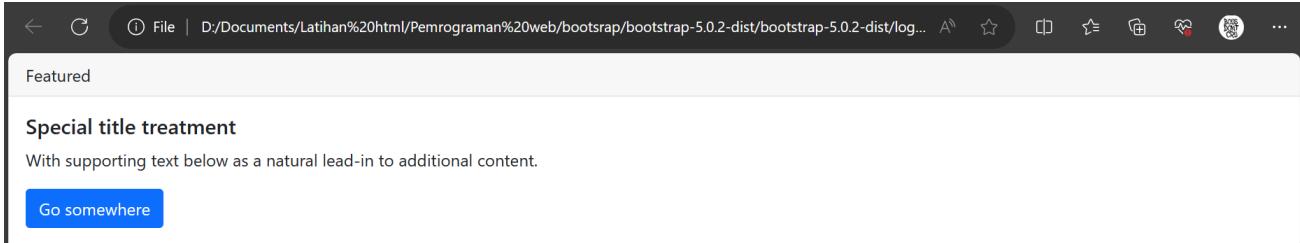
```

 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>
 Go somewhere
 </div>
</div>

```

4. Tempelkanlah kode program itu pada halaman html terkait. Tentu saja dengan bootstrap yang sudah ditutup, baik offline, maupun online.

5. Maka hasil awalnya adalah sebagai berikut:



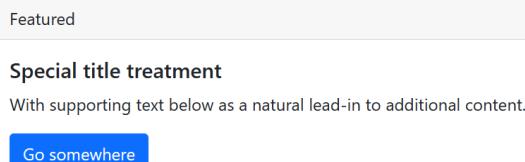
6. karena jarak atas antara card dengan batas website terlalu dekat maka pada class card tambahkan mt-2 .

```

<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>
 Go somewhere
 </div>
</div>

```

7. Maka hasilnya adalah seperti ini:



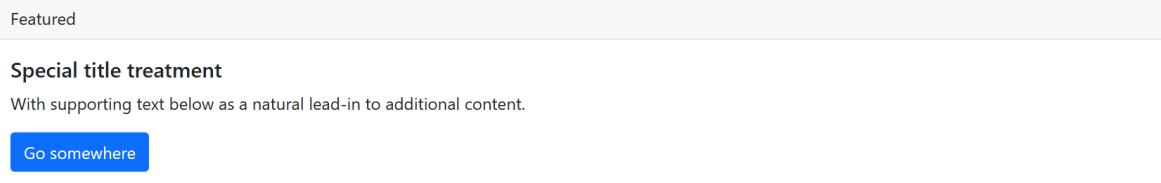
8. Sekarang kita ingin membuatnya terlihat ke tengah. Maka buatlah sebuah div dengan class yaitu "container", lalu salin masuklah kode program card tadi. Dengan kata lain, bungkuslah kode program card dengan sebuah div dengan class="container".

```

<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>
 Go somewhere
 </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>
 Go somewhere
 </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>
 Go somewhere
 </div>
 </div>
</div>

```

### Keterangan:

1. `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.
2. `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.
3. `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.
4. `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.
  - `<a href="#" class="btn btn-primary">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

## Form Login

### Special title treatment

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

Go somewhere

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">
 Go somewhere
 </div>
 </div>
</div>
```

13. Hasilnya akan seperti ini

Form Login

Go somewhere

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**

**Copy to clipboard**

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>
```

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">
 Form Login
 </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:

Dengan demikian kita sudah berhasil membuat layout cards 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 .col-sm- 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**

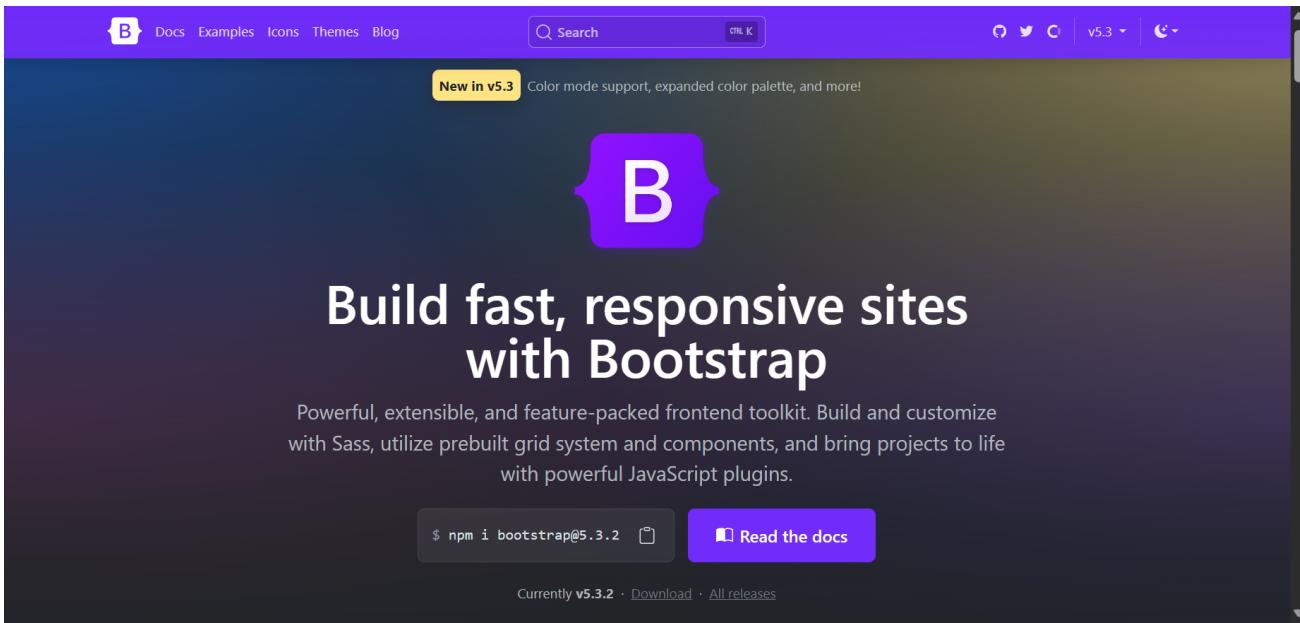
Kita dapat menggunakan kelas .order- untuk mengubah urutan kolom pada ukuran layar tertentu. Misalnya, .order-first akan memindahkan kolom ke posisi pertama, dan .order-last akan memindahkan kolom ke posisi terakhir. Dengan menggunakan kelas ini, kita dapat mengatur ulang urutan kolom untuk mencapai 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](https://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. On the left, there's a sidebar with a navigation menu: 'Sass', 'Options', 'Color', 'Color modes', 'Components', 'CSS variables', 'Optimize', 'Layout' (which is expanded), 'Breakpoints', 'Containers', 'Grid' (which is selected and highlighted in purple), 'Columns', 'Gutters', 'Utilities', 'Z-index', 'CSS Grid', 'Content' (which is expanded), 'Reboot', 'Typography', 'Images', and 'Tables'. The main content area has a heading 'Grid system' with a 'View on GitHub' button. It includes a sub-headline 'Example #', a paragraph explaining the grid system's purpose, and a note about flexbox. To the right, there's a sidebar titled 'On this page' with a list of topics: '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 praktik 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.

File Explorer:

- Welcome
- home.html
- tes.html**
- login.html

Code View:

```

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.

Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit
 laboriosam earum,
 possimus consequatur quisquam dolore ex ipsa eaque
 cupiditate neque.</p>
 </div>
 </div>
 <div class="col">
 <div>

 <h3>CSS</h3>
 <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>
 </div>
</div>
<div class="col">
 <div>

 <h3>JavaScript</h3>
 <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>
 </div>
 </div>
<div class="col">
 <div>

 <h3>JQuery</h3>
 <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>
 </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.

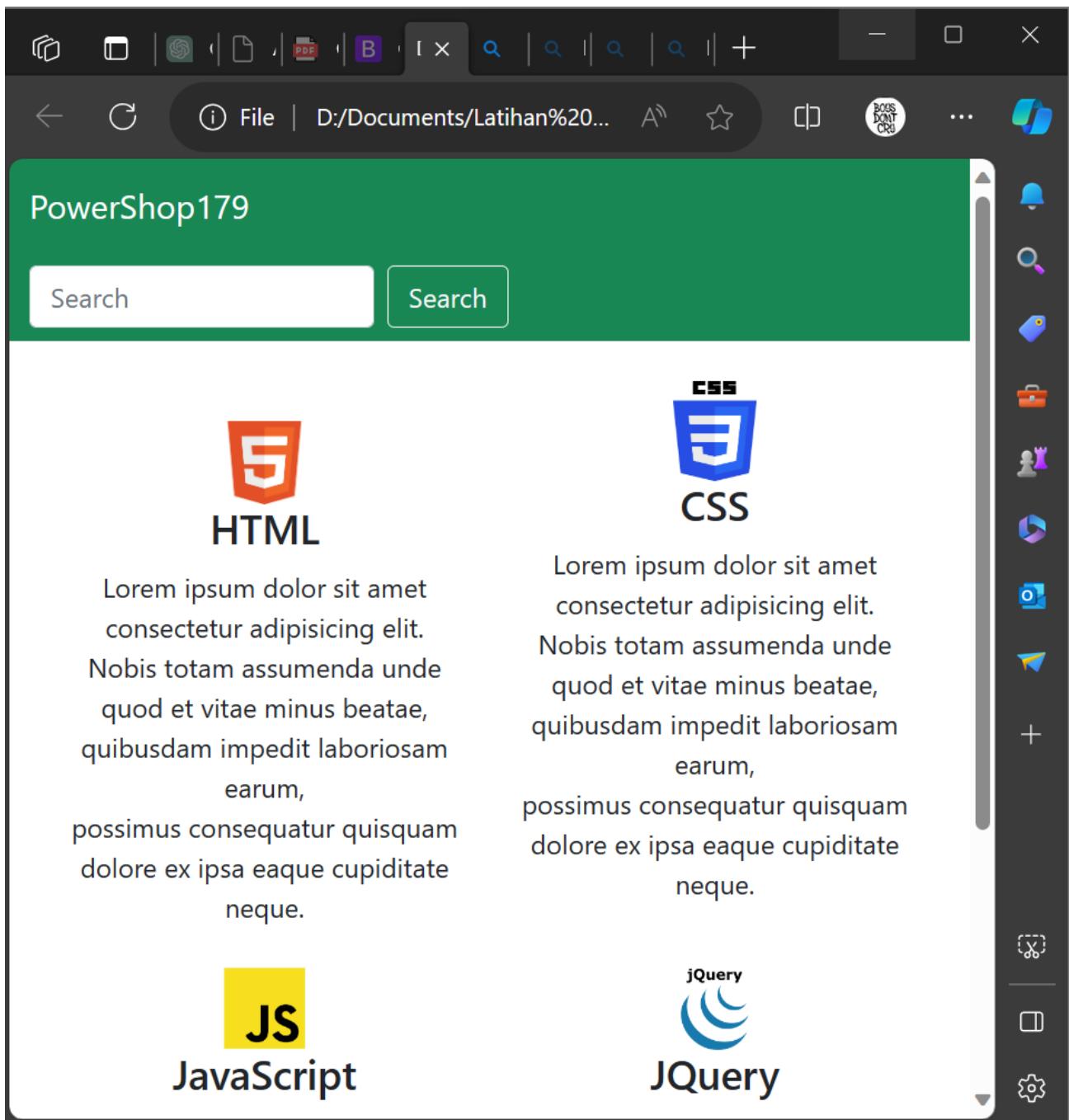


Lorem ipsum dolor sit amet consectetur adipisicing elit.  
 Nobis totam assumenda unde quod et vitae minus beatae,  
     quibusdam impedit laboriosam earum,  
     possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.

### 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

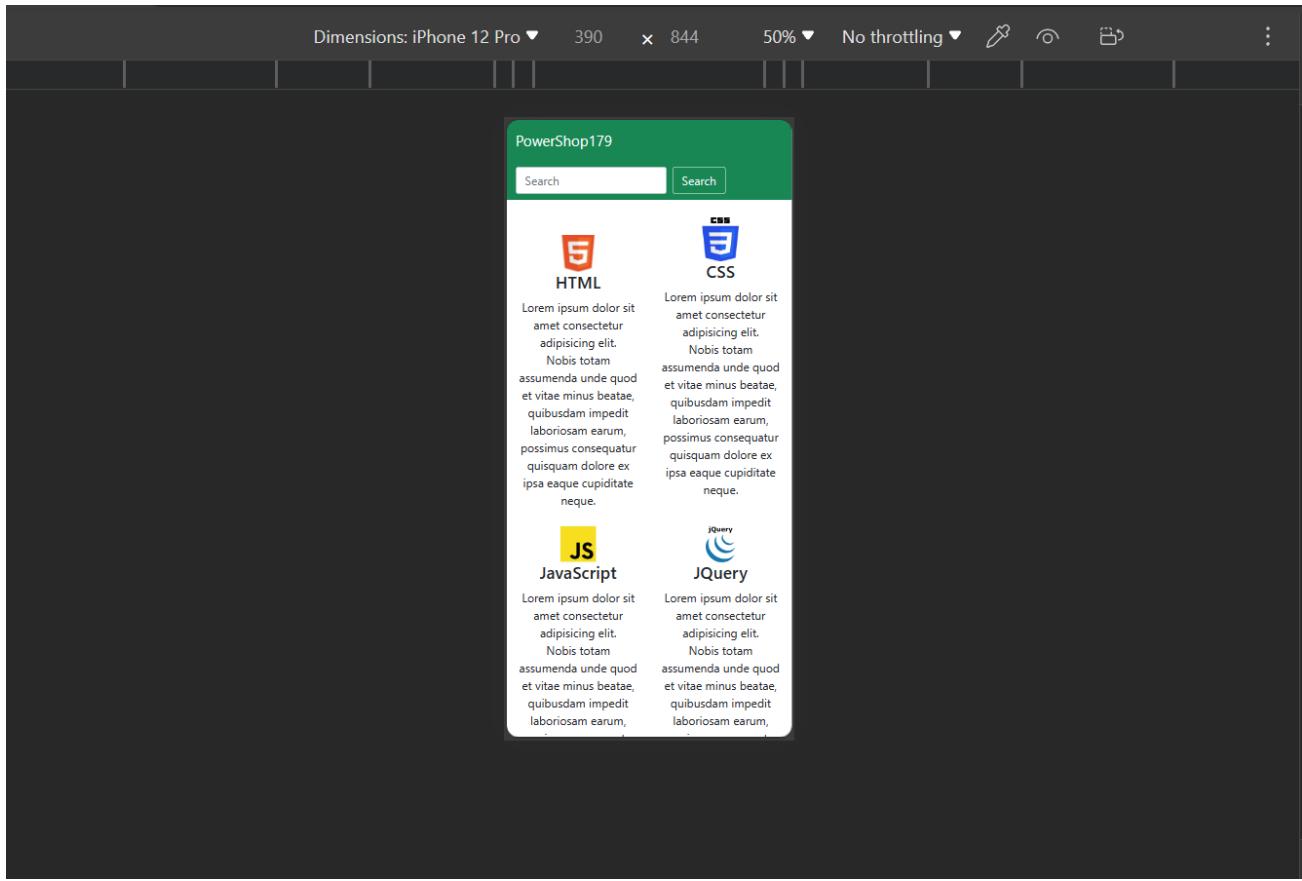
9. Apabila layar yang kita kecilkan, maka grid akan menyesuaikan ukurannya seperti pada gambar ini



**Keterangan:**

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

10. 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 5.3 documentation homepage. A search bar at the top contains the query 'grid'. Below it, a sidebar on the left lists various documentation sections like 'Grid system', 'CSS Grid', 'Layout', 'Customize', and 'Components'. The main content area features a large search result card for 'grid' with a purple header labeled 'Example'. This card contains links to 'How it works', 'Grid options', 'Auto-layout columns', and 'Responsive classes'. Below this is another card for 'CSS Grid' with links to 'How it works' and 'Key differences'. At the bottom of the search results, there's a note about using the keyboard to navigate and a search bar by Algolia.

- Cari dan copylah kembali codingan grid yang kita gunakan pada praktek sebelumnya.

## 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.

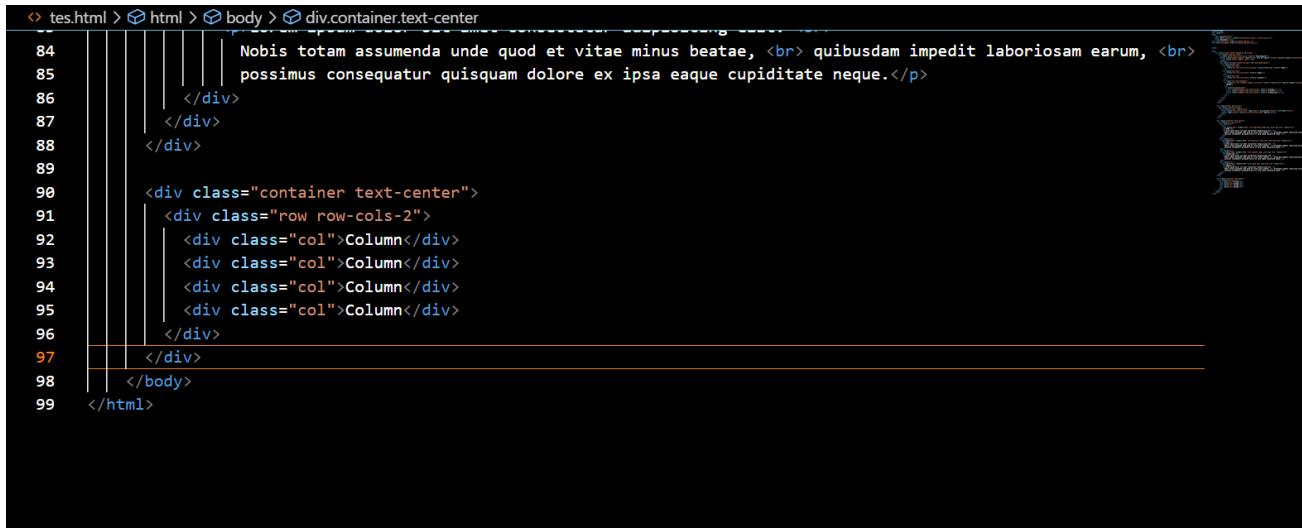
This screenshot shows the 'Row columns' section of the Bootstrap documentation. It includes a visual representation of a 2x2 grid with four columns labeled 'Column' and a snippet of HTML code below it.

```
<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>
```

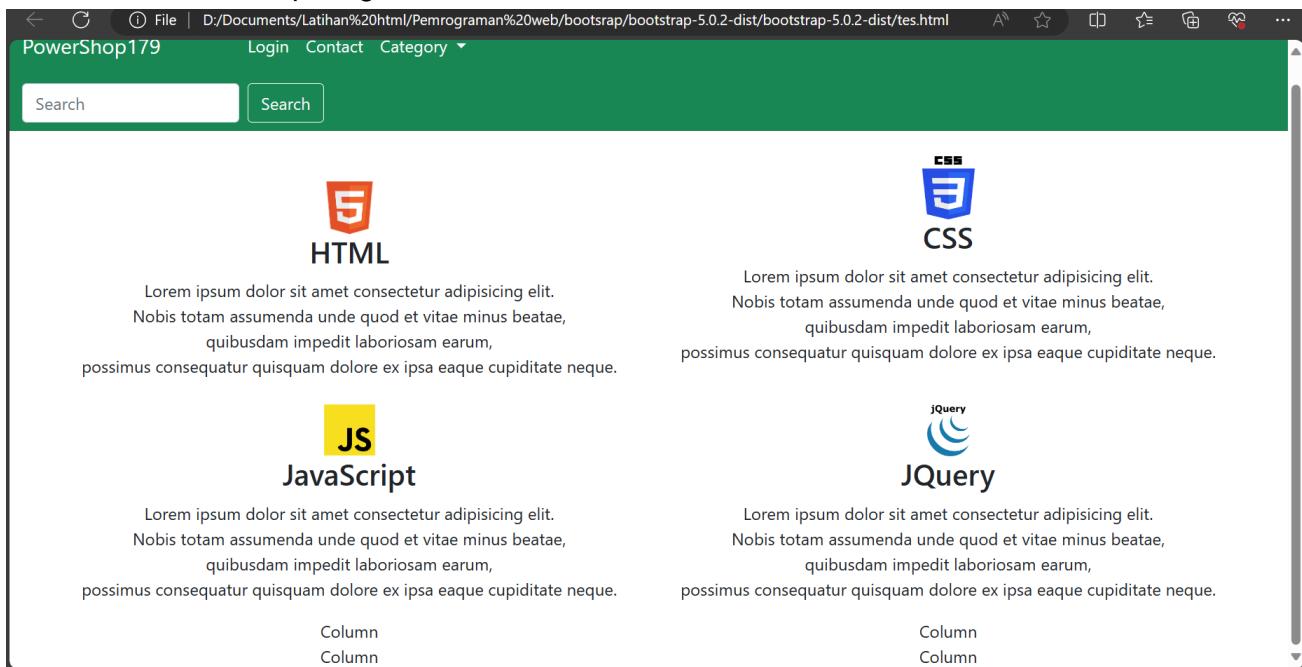
3. Tempelkanlah codingan tadi di visual studio kode pada file tempat kita mempraktekkan grid pada praktikum sebelumnya, agar kita bisa menggunakan satu file saja untuk praktek.



```
tes.html > html > body > div.container.text-center
84 | | | Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,

85 | | | possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
86 | | |</div>
87 | | |</div>
88 | |</div>
89 |
90 | <div class="container text-center">
91 | <div class="row row-cols-2">
92 | <div class="col">Column</div>
93 | <div class="col">Column</div>
94 | <div class="col">Column</div>
95 | <div class="col">Column</div>
96 | </div>
97 | </div>
98 | </body>
99 </html>
```

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. Below the header, there are four columns. The first column contains an HTML logo and placeholder text. The second column contains a CSS logo and placeholder text. The third column contains a JS logo and placeholder text. The fourth column contains a JQuery logo and placeholder text. Each column has a 'Column' label below it.

6. Maka hasilnya akan menjadi seperti ini:

**Sebelum:**

Column  
Column

Column  
Column

**Sesudah:**

Column      Column      Column      Column

**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.

 Nobis totam assumenda unde quod et vitae minus beatae,

 quibusdam impedit laboriosam earum,

 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.

 Nobis totam assumenda unde quod et vitae minus beatae,

 quibusdam impedit laboriosam earum,

 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.

 Nobis totam assumenda unde quod et vitae minus beatae,

 quibusdam impedit laboriosam earum,

 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.

```

```

Nobis totam assumenda unde quod et vitae minus beatae,

quibusdam impedit laboriosam earum,

possimus consequatur quisquam dolore ex ipsa eaque cupiditate
neque.</p>
</div>
</div>
</div>
</div>

```

File navigation bar: Welcome, home.html, tes.html (highlighted), bootstrap.min.css 4, login.html, ...

File structure: tes.html > html > body > div.container.text-center > div.row.row-cols-2 > div.col-md > div

```

90 <div class="container text-center">
91 <div class="row row-cols-2">
92 <div class="col-md">
93 <div>
94
95 <h3>HTML</h3>
96 <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.

97 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,

98 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
99 </div>
100 </div>
101 <div class="col-lg">
102
103 <h3>CSS</h3>
104 <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.

105 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,

106 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
107 </div>
108 <div class="col-sm">
109
110 <h3>JavaScript</h3>
111 <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.

112 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,

113 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.</p>
114 </div>
115 <div class="col">
116
117 <h3>JQuery</h3>
118 <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.


```

8. Maka hasilnya akan seperti ini:



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.

### 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:

1. **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.
2. **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.
3. **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

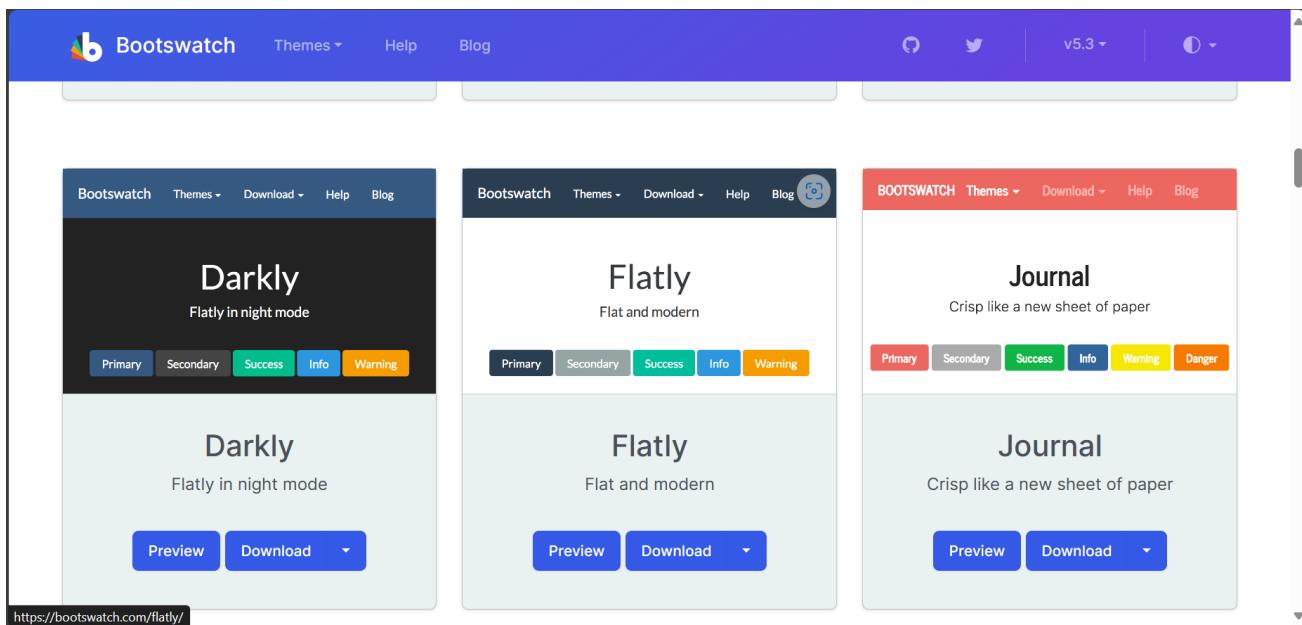
The screenshot shows a website with a green header containing the text "PowerShop179", "Login", "Contact", "Category", a search input field, and a search button. Below the header are four cards arranged in a 2x2 grid:

- HTML**: Features the HTML5 logo and a brief placeholder text.
- CSS**: Features the CSS logo and a brief placeholder text.
- JS**: Features the JavaScript logo and a brief placeholder text.
- jQuery**: Features the jQuery logo and a brief placeholder text.

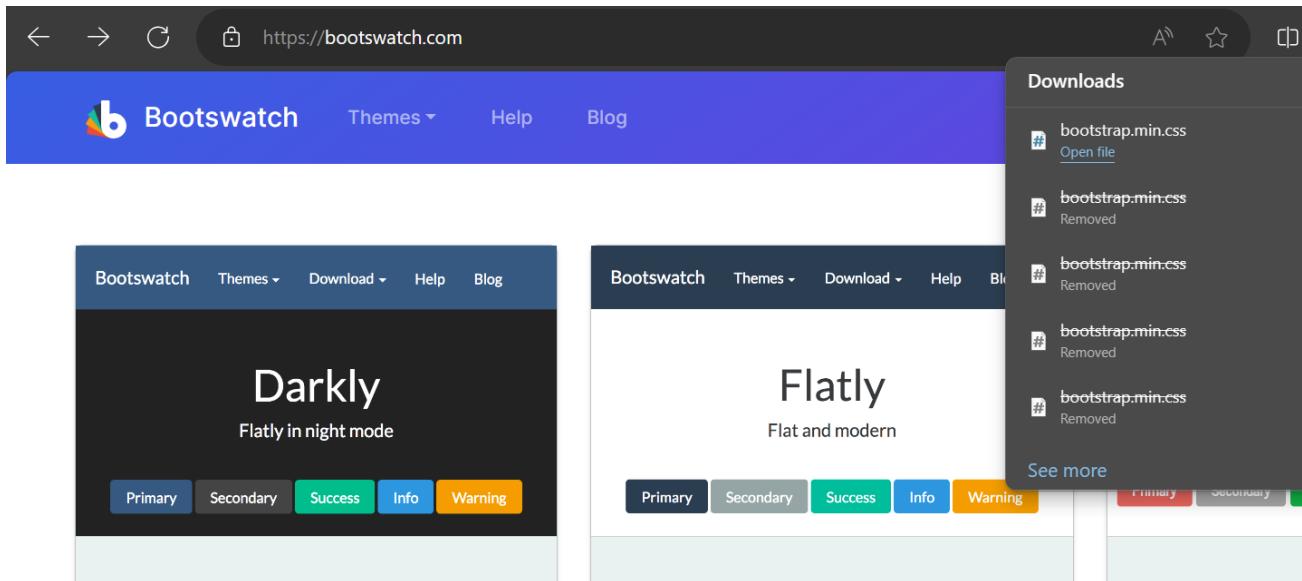
2. Bukalah website [bootswatch.com](https://bootswatch.com) dimana website ini menyediakan tema tema yang dapat digunakan untuk web bootstrap kia.

The screenshot shows the homepage of bootswatch.com. The header includes the "Bootswatch" logo, navigation links for "Themes", "Help", and "Blog", and social media icons for GitHub, Twitter, and a dropdown menu. The main content area has a purple background and features the text "Free themes for Bootstrap" in large white letters. Below this, there are two small stats: "Fork 3,723" and "Star 14,308". A small image of a cartoon character is displayed, followed by the text "Adobe Creative Cloud for Teams. Put creativity to work." and a note "ads via Carbon".

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



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

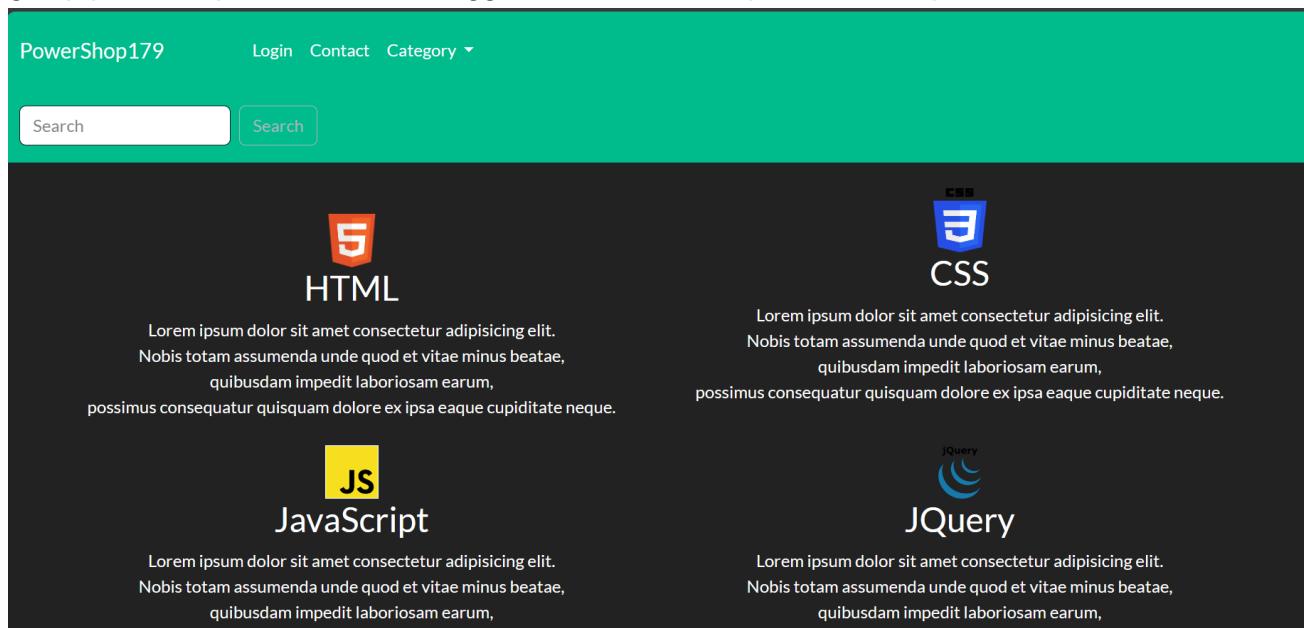
7. Sekarang, masuklah ke dalam file html yang akan kita berikan tema bootstrap, lalu panggilah secara eksternal file css tema tadi ke dalam file html tersebut

```

 <!DOCTYPE html>
 <html lang="en">
 <head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Document</title>
 <link rel="stylesheet" href="css/bootstrap.min.css">
 <link rel="stylesheet" href="css/bootstrap-grid.min.css">
 <link rel="stylesheet" href="b">
 # bootstrap.min.css
 </head>

```

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



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.

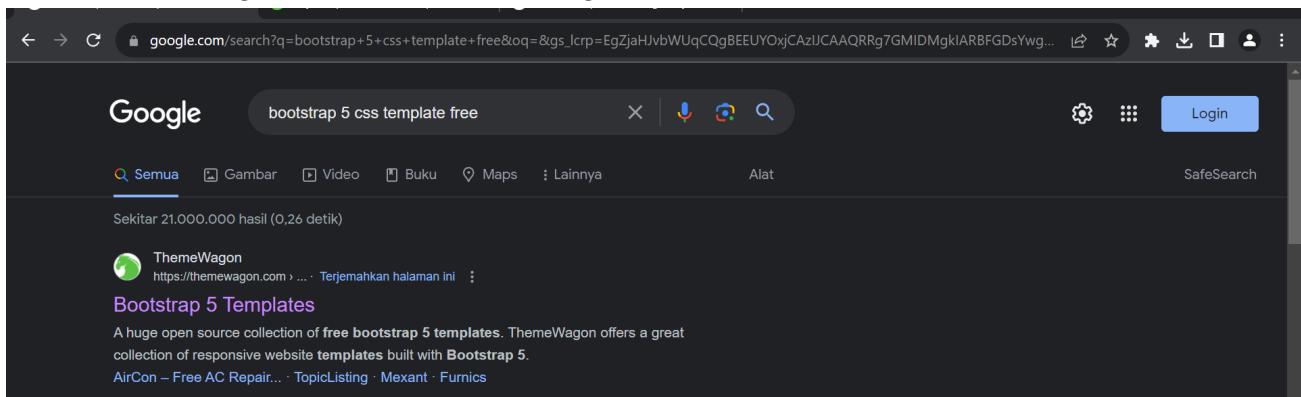
Template dalam Bootstrap adalah struktur dasar atau kerangka kerja yang telah dirancang sebelumnya dan dapat digunakan sebagai dasar untuk membangun halaman web. Bootstrap adalah sebuah framework front-end yang populer digunakan untuk pengembangan web responsif. Dalam konteks Bootstrap, template menyediakan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

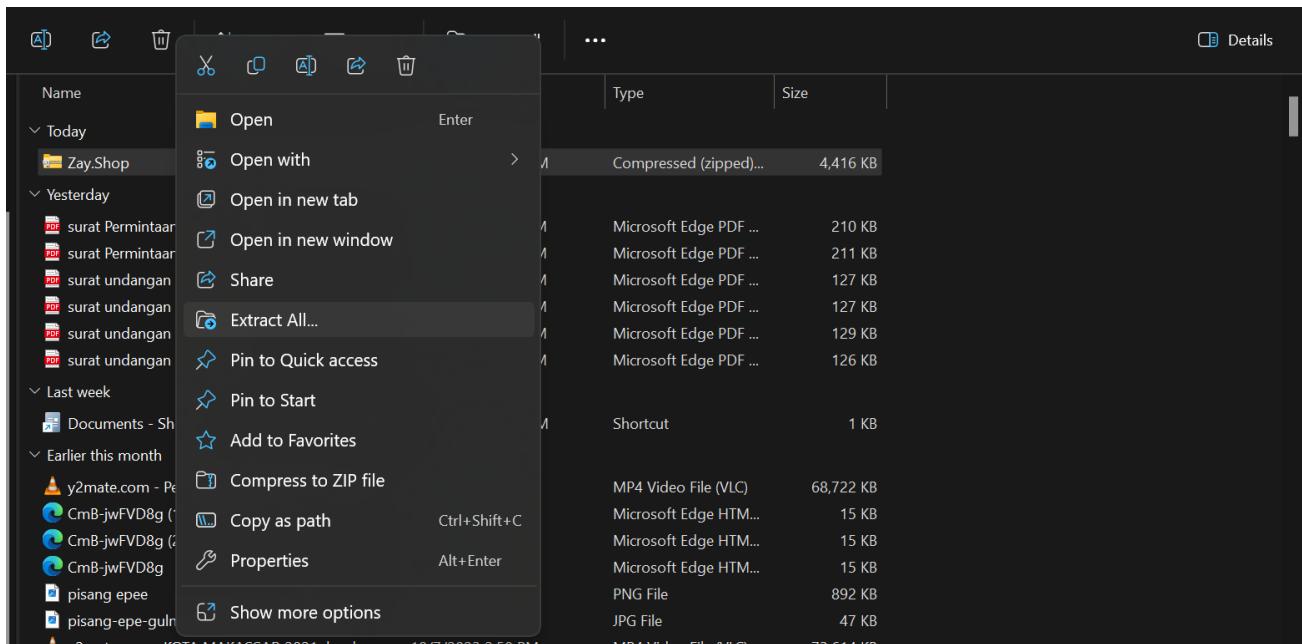
A screenshot of the themewagon.com website. The header includes the theme logo, a search bar, and navigation links for "Browse Themes", "Premium", "Freebies", "Sign in", "Sign up", and "Hire us". Below the header is a navigation bar with links for "Admin &amp; Dashboard", "Bootstrap 5", "eCommerce", "Tailwind CSS", "Landing Pages", "Business &amp; Corporate", "Portfolio", "Educational", and a "Bundle — Save 80%" button. The main content area displays a grid of free Bootstrap themes. Each theme card includes a preview image, the theme name, a "Free" badge, download statistics (e.g., "4931 Downloads"), and a star rating. The themes shown are "MultiShop – Free Bootstrap 4 E...", "CakeZone – Free Bootstrap 5 HT...", "Eshopper – Free Bootstrap 4 E...", and "Zay".

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

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

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".



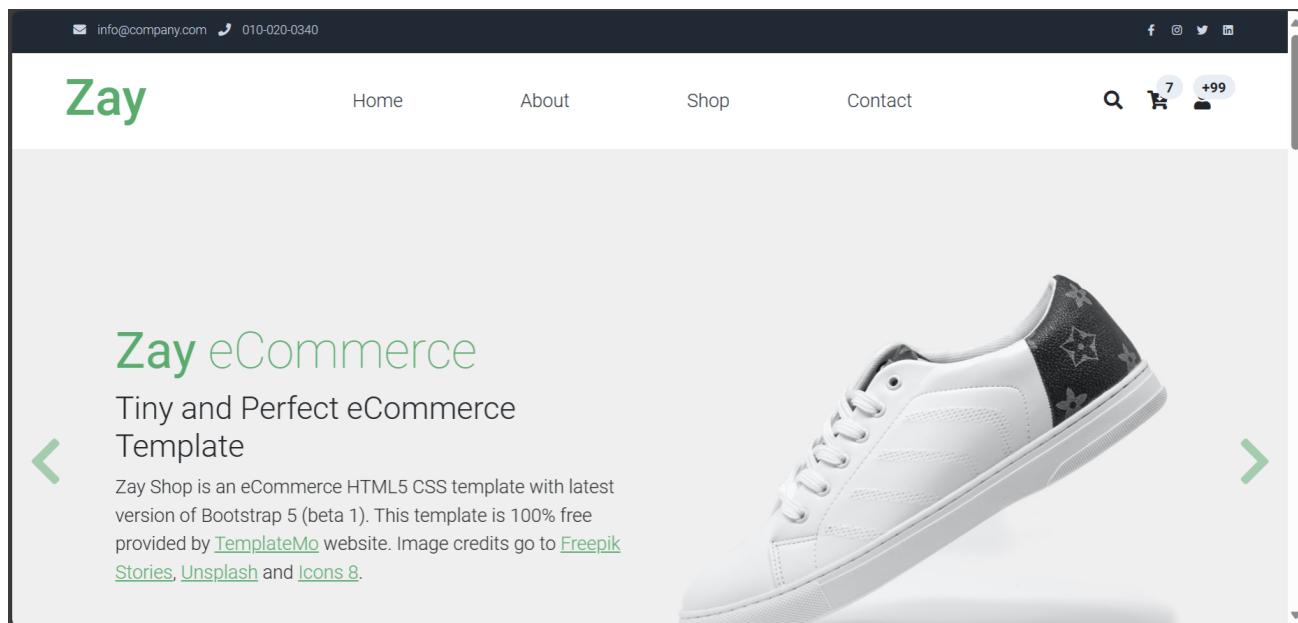
7. Jika proses ekstrak sudah selesai maka kita telah mendapatkan folder template tersebut dalam bentuk folder biasa



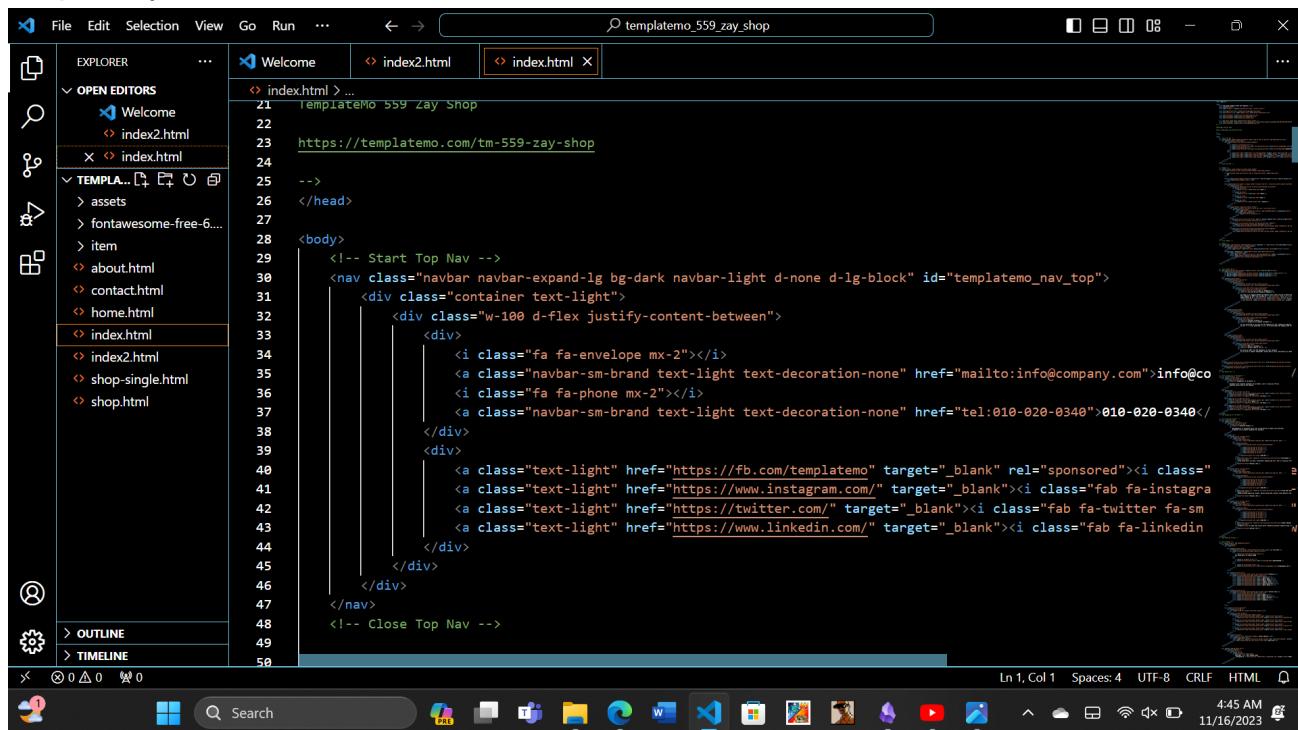
8. Masuklah ke dalam folder template yang telah diekstrak tadi

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
:class="fa fa-phone mx-2">></i>
:class="navbar-sm-brand text-light text-decoration-none" href="tel:010-020-0340">010-020-0340
```

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

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

The screenshot shows the Zay eCommerce website. The logo 'Zay' has been replaced with 'Powershop179'. The hero section features a white and black sneaker with a star pattern. The text in the hero section is as follows:

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](#).

```

 Powershop179

```

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">PowerShop179 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
 website. Image credits go to
 <a rel="sponsored" class="text-success"
 href="https://stories.freepik.com/" target="_blank">Freepik Stories,
 <a rel="sponsored" class="text-success" href="https://unsplash.com/"
 target="_blank">Unsplash and
 <a rel="sponsored" class="text-success" href="https://icons8.com/"
 target="_blank">Icons 8.
 </p>
 </div>
```

```
</div>
```

Hasilnya akan menjadi seperti ini:

The screenshot shows the homepage of the Powershop179 eCommerce template. At the top, there is 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 notification bubble with +99. The main content area features a large image of a white and black star-patterned sneaker on the right. To the left of the shoe, the text "Powershow179" and "eCommerce" is displayed, followed by "Tiny and Perfect eCommerce Template". Below this text is a brief description of the template: "PowerShop179 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](#)". There are also left and right navigation arrows.

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

The screenshot shows three product cards from the Powershop179 template. Each card has a large image at the top, followed by a rating, price, title, a short description, and a reviews link. The first card is for a "Gym Weight" priced at \$240.00. The second card is for "Cloud Nike Shoes" priced at \$480.00. The third card is for "Summer Addides Shoes" (a placeholder name) priced at \$360.00. The products shown in the cards are a grey cylindrical device, a black smartwatch with a red woven strap, and a white Polaroid camera.

Product	Rating	Price	Description	Reviews
Gym Weight	★★★★★	\$240.00	Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sunt in culpa qui officia deserunt.	Reviews (24)
Cloud Nike Shoes	★★★★★	\$480.00	Aenean gravida dignissim finibus. Nullam ipsum diam, posuere vitae pharetra sed, commodo ullamcorper.	Reviews (48)
Summer Addides Shoes	★★★★★	\$360.00	Curabitur ac mi sit amet diam luctus porta. Phasellus pulvinar sagittis diam, et scelerisque ipsum lobortis nec.	Reviews (74)

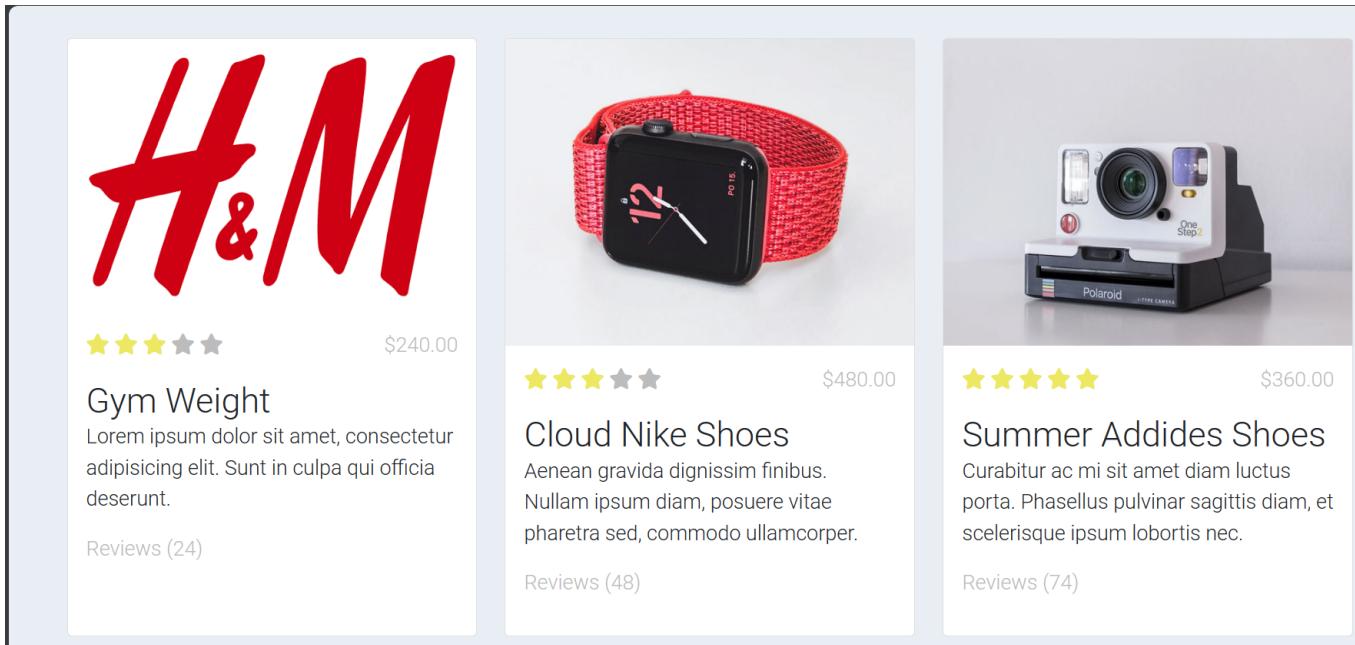
18. Masuklah ke visual studio code dan ke index.html tadi lalu carilah kode penyusun untuk gambar ini, kemudian gantilah di bagian src gambar yang sudah ada dengan gambar yang kita inginkan

```



```

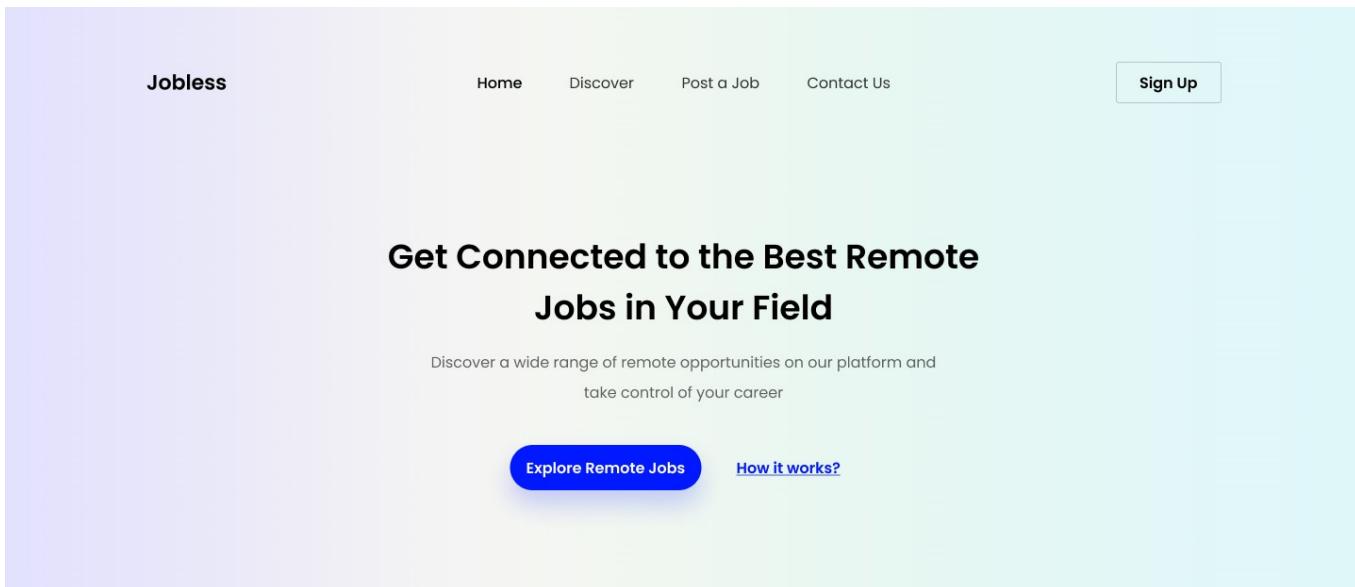
Maka hasilnya akan menjadi seperti ini:



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:**



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.

- Selanjutnya kita akan implementasikan navbarnya, bukalah kembali web getbootstrap lalu kolom pencarian carilah "navbar".

The screenshot shows the Bootstrap documentation homepage. A search bar at the top contains the query "Navbar". Below the search bar, a sidebar on the left lists various Bootstrap components like Accordion, Alerts, Badge, etc. The main content area features a large heading "Get started with Bootstrap" and a search result card for "Navbar". The card has a purple header with the title "How it works" and a list of sections: "Supported content", "Color schemes", "Containers", and "Placement". To the right of the card, there's a sidebar titled "On this page" with links to "Quick start", "CDN links", "Next steps", etc. At the bottom of the card, there are navigation instructions and a search bar labeled "Search by algolia".

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

This screenshot shows the Bootstrap documentation page for the "Navbar" component. The sidebar on the left has "Navbar" selected. The main content area displays the "HTML" section of the Navbar documentation, which includes a screenshot of a browser showing a navigation bar with a brand logo, a toggle button, and a dropdown menu. Below the screenshot is the corresponding HTML code:

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
 <div class="container-fluid">
 Navbar
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent">

 </button>
 <div class="collapse navbar-collapse" id="navbarSupportedContent">
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">
 <li class="nav-item">
 Home

 <li class="nav-item">
 Link

 <li class="nav-item dropdown">
 Dropdown
 <ul class="dropdown-menu">
 Action

 </div>
 </div>
</nav>
```

```
<nav class="navbar navbar-expand-lg bg-body-tertiary"> <div class="container-
fluid"> Navbar <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"> </button> <div class="collapse navbar-collapse" id="navbarSupportedContent">
<ul class="navbar-nav me-auto mb-2 mb-lg-0">
 <li class="nav-item">
 Home

 <li class="nav-item">
 Link

 <li class="nav-item dropdown">
 <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-
 expanded="false" aria-label="Toggle navigation"> Dropdown
 <ul class="dropdown-menu">
 Action

</div>
</div>
</nav>
```

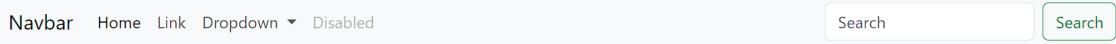
```

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

```

### 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"> Navbar <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"> </button>
 <div class="collapse navbar-collapse" id="navbarSupportedContent">
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">
 <li class="nav-item"> Home
 <li class="nav-item"> Link
 <li class="nav-item dropdown"> Dropdown
 <ul class="dropdown-menu">
 Action
 Another action

 <hr class="dropdown-divider">

 Something else here

 <li class="nav-item"> Disabled

 <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>
```

## 8. Hasilnya akan seperti ini:

Navbar Home Link Dropdown ▾ Disabled

## 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"> Home <li class="nav-item"> Discover <li class="nav-item"> Post a Job <li class="nav-item"> Contact Us

```

### 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.

Jobless Home Discover Post a Job Contact Us

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

```
 Sign Up
```

#### 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"> Navbar <button class="navbar-toggler"
 type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"
 aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"> </button>
 <div class="collapse navbar-collapse" id="navbarSupportedContent">
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">
 <li class="nav-item"> Home

 <li class="nav-item"> Discover
 <li class="nav-item"> Post a Job
 <li class="nav-item"> Contact Us

 Sign Up
 </div>
</div>
</nav>
```

12. Sekarang kita akan menengahkan list navbarnya dengan menggunakan class `mx-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">
 Jobless
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
 data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
 aria-expanded="false" aria-label="Toggle navigation">

 </button>
 <div class="collapse navbar-collapse" id="navbarSupportedContent">
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">
 <li class="nav-item"> Home
 <li class="nav-item"> Discover
 <li class="nav-item"> Post a Job
 <li class="nav-item"> Contact Us

 Sign Up

 </div>
 </div>
 </nav>

</div>
```

sesudah

```
<div class="container">

 <nav class="navbar navbar-expand-lg bg-transparent">
 <div class="container-fluid">
 Jobless
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
 data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
 aria-expanded="false" aria-label="Toggle navigation">

 </button>
 <div class="collapse navbar-collapse" id="navbarSupportedContent">
 <ul class="navbar-nav m-auto mb-2 mb-lg-0">
 <li class="nav-item"> Home
 <li class="nav-item"> Discover
 <li class="nav-item"> Post a Job
 <li class="nav-item"> Contact Us

 Sign Up

 </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
Jobs in Your Field </h1> <p class="text-muted">Discover a wide range of remote
opportunities on our platform and
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:

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

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</pre>
```

```
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" integrity="sha384-QWTKZyjpPEjISv5WaRU90FeRpok6YctnYmDr5pI" rel="stylesheet">
```

```
 <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"> Explore Remote Jobs How it works? </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"> Explore Remote Jobs How it works? </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: #f6faf9;
 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

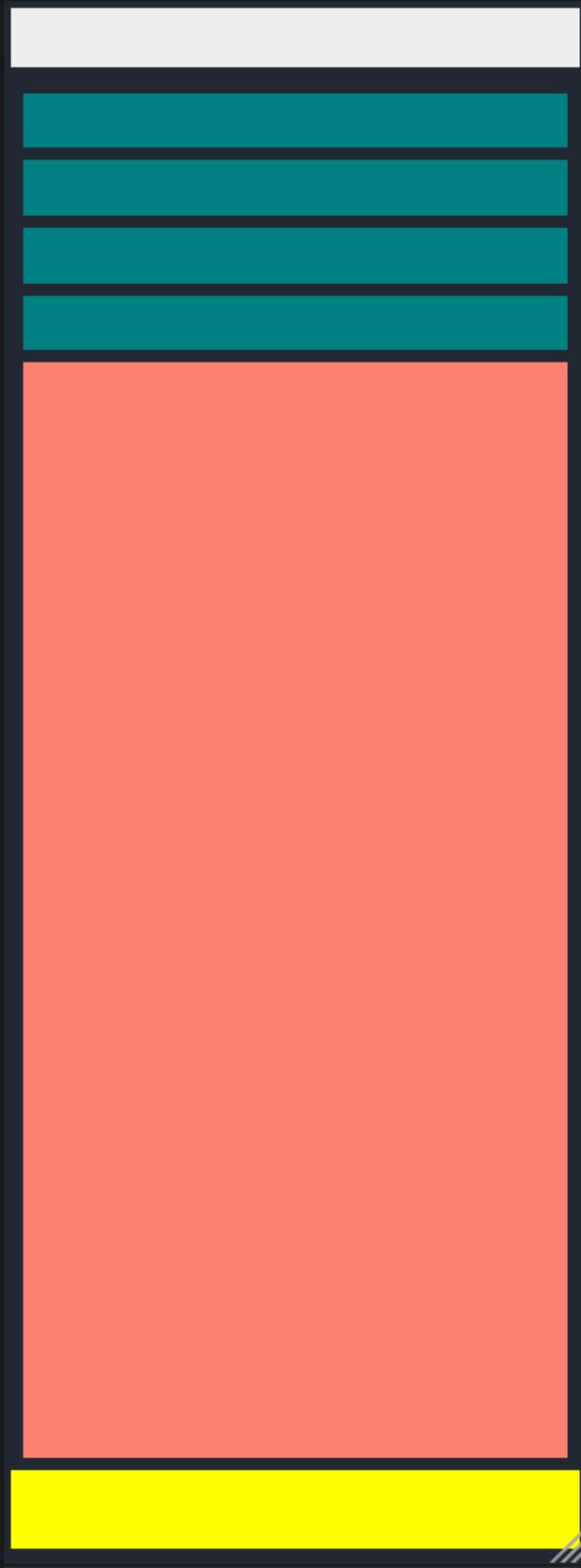
Saat mendesain website, mulailah dengan mempertimbangkan tampilan website pada perangkat mobile. Hal ini akan membantu Anda membuat website yang lebih ringkas dan mudah digunakan di semua perangkat.

### | 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.