

CSS + BOOTSTRAP +RESPONSIVE

Anatomi CSS

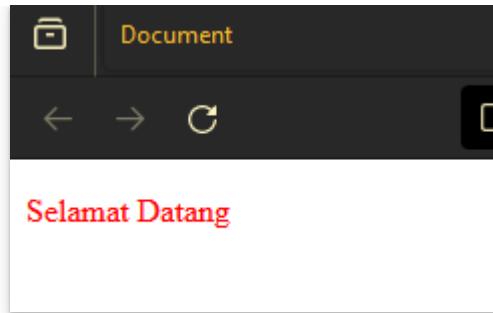
```
selektor{  
    /*Property dan Nilai */  
}  
  
/* Contoh */  
p{  
    color: white;  
}
```

- **p** Ini merupakan Selektor, selektor adalah bagian dari aturan CSS yang menentukan elemen HTML mana yang akan di-styling. Selektor bisa berupa nama elemen HTML, kelas, ID, atau kombinasi dari hal-hal tersebut
- **color** Merupakan property atribut sedangkan **white** adalah nilai dari atribut

Percobaan Pertama CSS

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Document</title>  
    <style>  
        p{  
            color: red;  
        }  
    </style>  
</head>  
<body>  
    <p>Selamat Datang</p>  
</body>  
</html>
```

- **<!DOCTYPE html>** Tag yang menentukan versi HTML yang digunakan adalah versi 5
- **<head>** Ini adalah bagian dari dokumen HTML yang berisi meta-informasi tentang dokumen, seperti karakter set yang digunakan, viewport, judul dokumen, dan referensi ke berkas CSS atau JavaScript.
- **<title>** Ini adalah judul dokumen yang akan ditampilkan di bilah judul atau tab peramban.
- **<style>** Tag ini digunakan untuk menyematkan gaya CSS ke dalam dokumen HTML.
- **<p>** Ini adalah selektor CSS yang memilih semua elemen **<p>** (paragraf) dalam dokumen HTML.
- **{ color: red; }** - Ini adalah blok deklarasi yang berisi satu properti CSS, yaitu **color**, yang ditetapkan ke nilai **red**. Artinya, semua teks dalam elemen **<p>** akan ditampilkan dengan warna merah.



Percobaan Kedua CSS

Kode

```
button {  
    width: 150px;  
    height: 150px;  
    background-color: crimson;  
    color: aqua;  
    font-size: 50pt;  
}
```

background-color

Before:



After:



Penjelasan:

- Ketika program dari tag button di beri atribut `background-color: crimson;` maka warna latar dari button akan berubah sesuai dengan yang di inginkan, terlihat perbedaannya sebelum di beri atribut `background-color: crimson;` warna dari latar button tetap berwarna putih alias default.

color:

Before:



After:



Penjelasan:

- Ketika program dari tag button di beri atribut `color: aqua;` maka warna font dari button tersebut berubah menjadi warna sesuai yang di inginkan terlihat perbedaan sebelum di beri atribut `color: aqua;` warma dari font button tersebut tetap berwana hitam alias default.

Font-size

Before:



After:



Penjelasan:

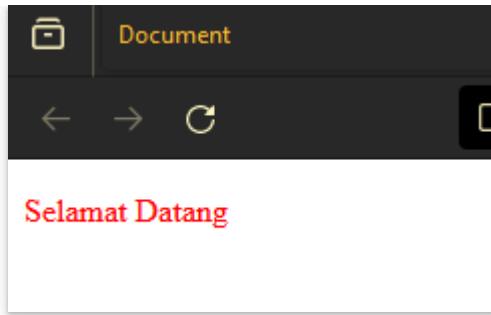
- Terlihat di atas perbedaannya ketika button di beri atribut `font-size: 50pt;` ukuran dari button tersebut langsung berubah menjadi besar karena atribut tersebut.

Cara Pemanggilan CSS

Inline

- **Inline CSS** Gaya CSS disematkan langsung ke dalam elemen HTML menggunakan atribut `style`. Contohnya:

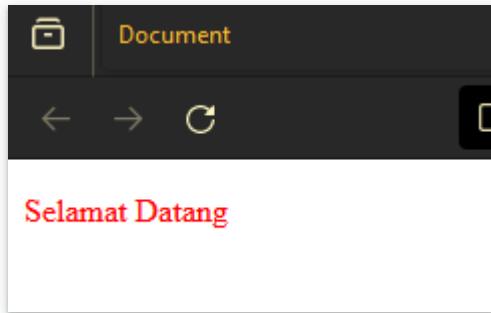
```
<p style="color: red;">Selamat Datang</p>
```



Internal

Internal CSS Aturan gaya CSS ditempatkan di dalam elemen `<style>` di bagian `<head>` dokumen HTML. Contohnya:

```
<style>
  p{
    color:red;
  }
</style>
```



External

External CSS Aturan gaya CSS ditempatkan di dalam berkas terpisah dengan ekstensi `.css` dan disematkan ke dalam dokumen HTML menggunakan tag `<link>`. Contoh jika aturan gaya disimpan dalam berkas bernama `style.css`.

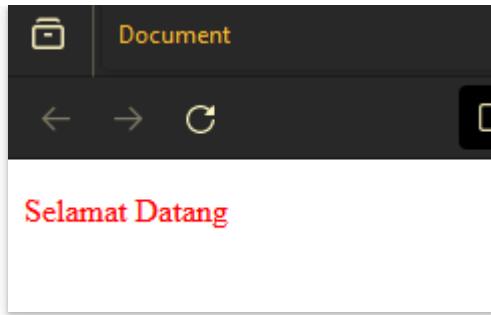
```
<link rel="stylesheet" type="text/css" href="style.css">
```

Dengan menggunakan metode eksternal CSS, Anda dapat memiliki satu berkas CSS yang diterapkan ke banyak halaman HTML, sehingga memungkinkan untuk mengelola gaya secara terpusat dan memisahkan struktur dari presentasi dalam halaman web Anda.

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

Dalam pemanggilan CSS, Anda dapat menetapkan gaya untuk berbagai elemen HTML menggunakan selektor CSS. Beberapa contoh selektor CSS yang umum meliputi:

- **Nama Elemen:** Misalnya, `p` untuk paragraf, `h1`, `h2`, dll., untuk judul.
- **Kelas:** Ditandai dengan titik `(.)` diikuti dengan nama kelas. Contoh: `.class-name`.
- **ID:** Ditandai dengan tanda pagar `(#)` diikuti dengan nama ID. Contoh: `#id-name`.



Selektor

Elemen

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

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

ID

```
#css {  
    color: red;  
}
```

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

Class

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

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

Text

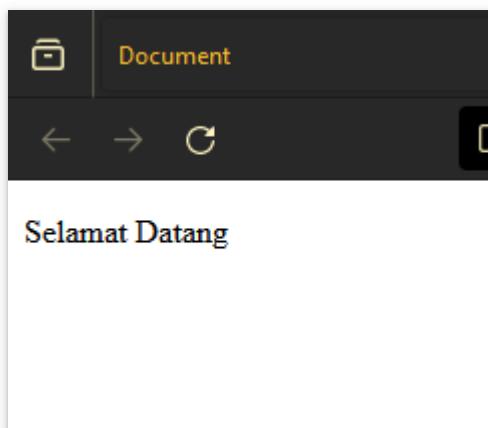
Text Align

- Digunakan untuk mengatur penataan horizontal teks di dalam sebuah elemen.
- Properti `text-align` dapat diatur ke nilai seperti `left`, `right`, `center`, atau `justify`.

```
.text{  
    text-align: /*Nilai*/;  
}
```

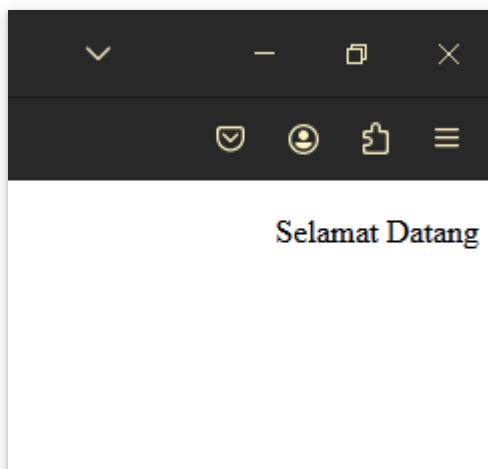
- **Left:** Mengatur teks agar berada di sebelah kiri elemen.

```
.text {  
    text-align: left;  
}
```



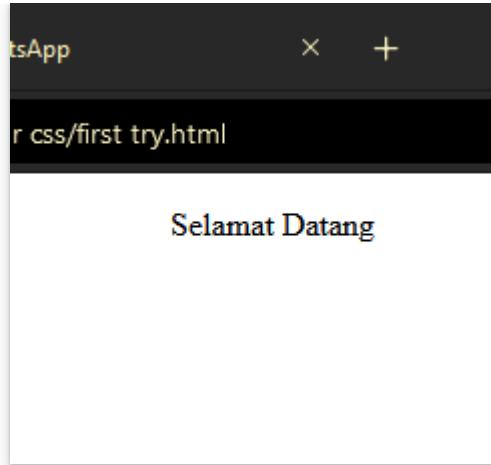
- **Right:** Mengatur teks agar berada di sebelah kanan elemen.

```
.text {  
    text-align: right;  
}
```



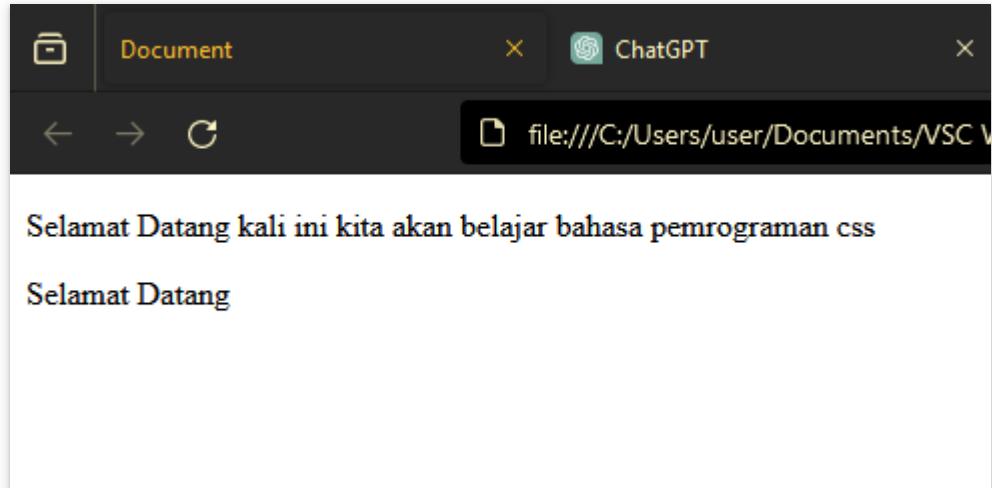
- **Center:** Mengatur teks agar berada di tengah elemen.

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



- **Justify:** Mengatur teks agar rata kiri dan kanan, kecuali pada baris terakhir yang bisa berada di sebelah kiri.

```
.text {  
    text-align: justify;  
}
```



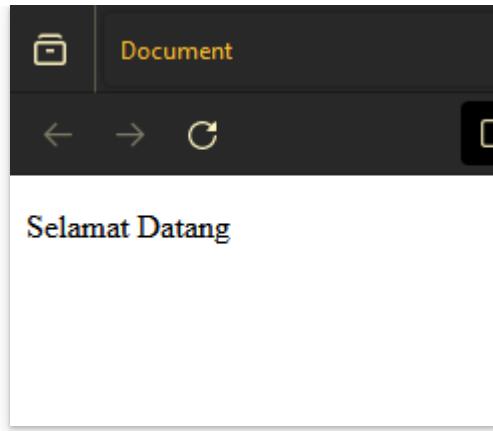
Text Decoration

- Mengatur dekorasi teks seperti garis bawah, garis atas, atau garis melintang.
- Properti `text-decoration` memiliki nilai seperti `none`, `underline`, `overline`, `line-through`, dan `inherit`.

```
.text{  
    text-decoration: /*Nilai*/;  
}
```

None: Tidak ada dekorasi pada teks.

```
.text {  
    text-decoration: none;  
}
```



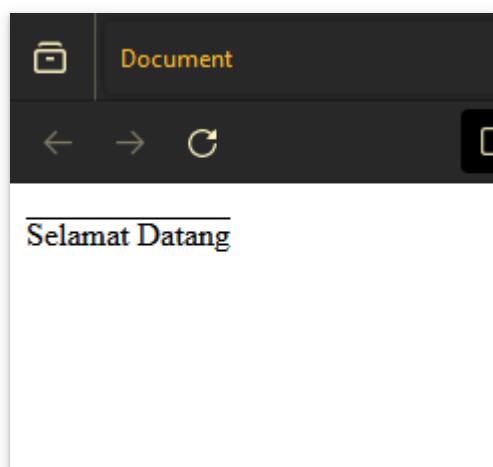
- **Underline:** Menambahkan garis bawah pada teks.

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



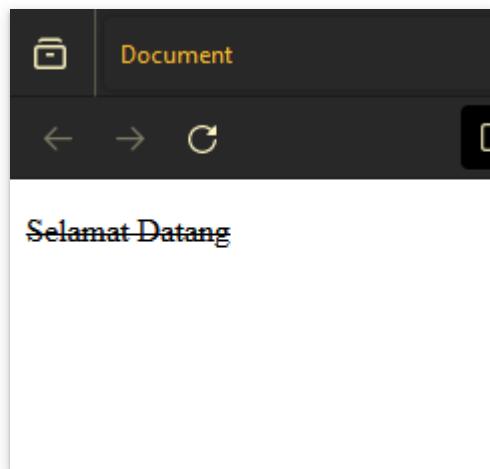
Overline: Menambahkan garis di atas teks.

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



Line-through: Menambahkan garis melintang pada teks.

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



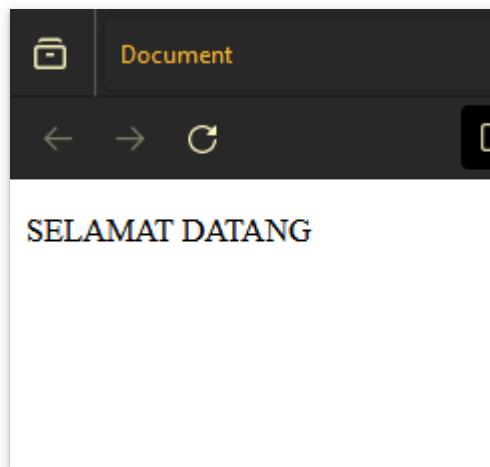
Text Transfrom

None: Tidak ada transformasi pada teks (kapitalisasi tidak diubah).

```
.text {  
    text-transform: none;  
}
```

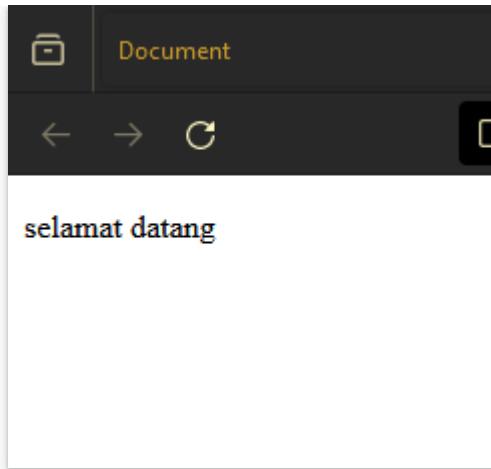
Uppercase: Mengubah semua huruf dalam teks menjadi huruf besar (uppercase).

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



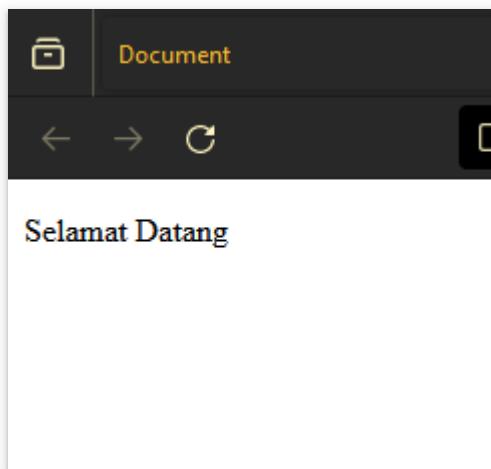
Lowercase: Mengubah semua huruf dalam teks menjadi huruf kecil (lowercase).

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



Capitalize: Mengubah setiap awal kata dalam teks menjadi huruf besar (uppercase).

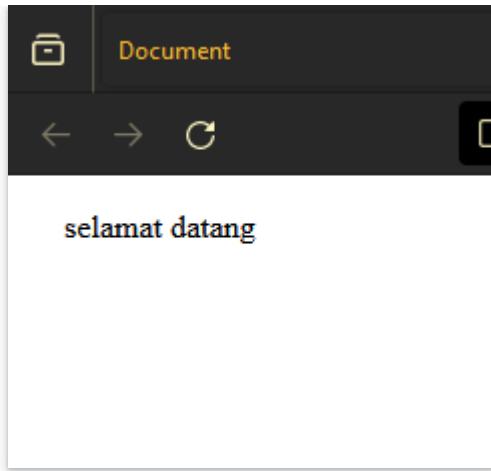
```
.text {  
    text-transform: capitalize;  
}
```



Text Indent

- Mengatur jarak indentasi dari awal baris pertama dalam sebuah paragraf atau elemen.
- Properti `text-indent` diukur dalam satuan seperti piksel atau persen.

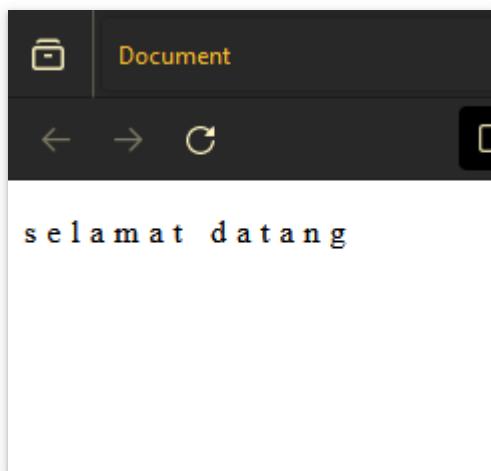
```
.text {  
    text-indent: 20px;  
}
```



Letter Spacing

- Mengatur jarak antara setiap huruf dalam teks.
- Properti `letter-spacing` dapat diatur dengan nilai positif (untuk meningkatkan jarak) atau negatif (untuk mengurangi jarak).

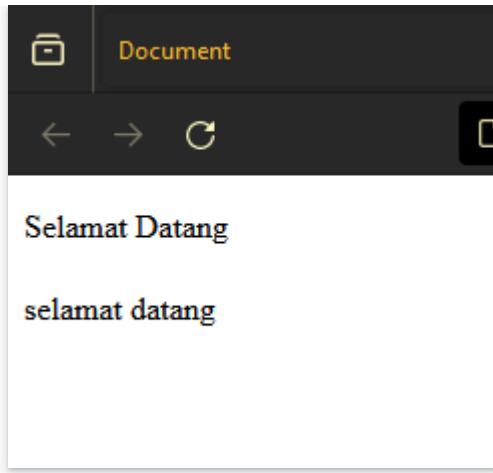
```
.text {  
  letter-spacing: 5px;  
}
```



Line Height

- Mengatur tinggi baris dalam teks.
- Properti `line-height` dapat diatur sebagai nilai numerik, persentase, atau menggunakan nilai `normal`.

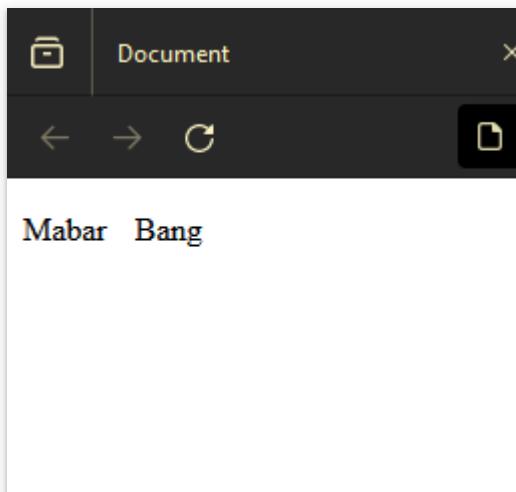
```
.text {  
  line-height: 1.5;  
}
```



Word Spacing

- Dan terakhir ialah word-spacing fungsinya untuk memberi jarak di antara setiap kata dan valuenya dapat di atur sesuai keinginan kita

```
.test{  
    word-spacing: 10px;  
}
```



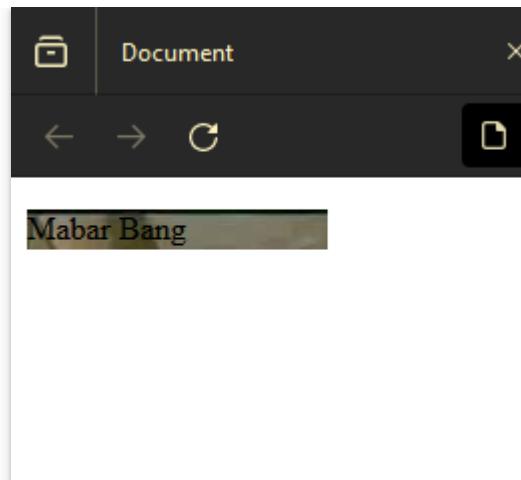
Background

- Mempelajari value dan atribut dari materi background

Background image

- Atribut ini digunakan untuk memberi gambar di tampilan web dan bisa di sempurnakan bentuknya menggunakan atribut lain, hasil di bawah ini berbeda karena hanya menggunakan satu atribut

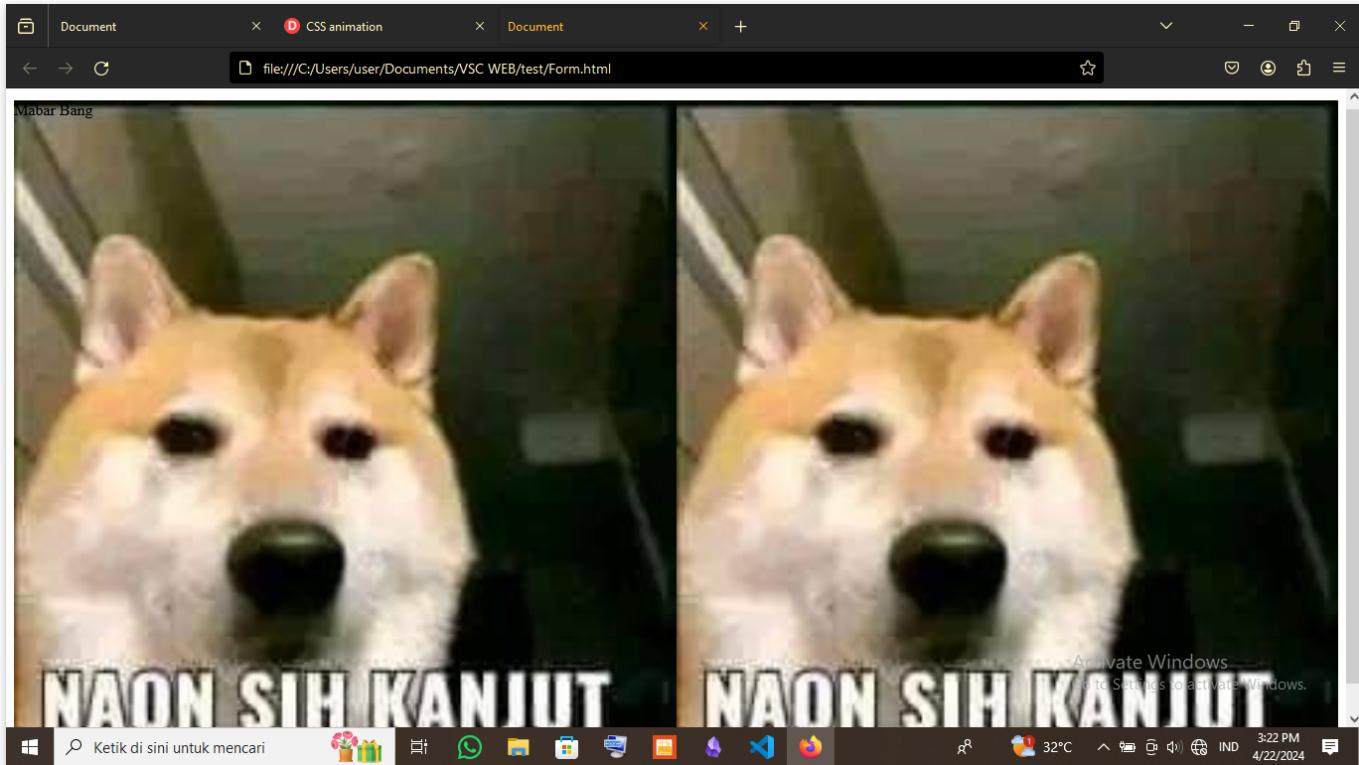
```
.test{  
    background-image: url(naon.jpg);  
}
```



Background Size

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

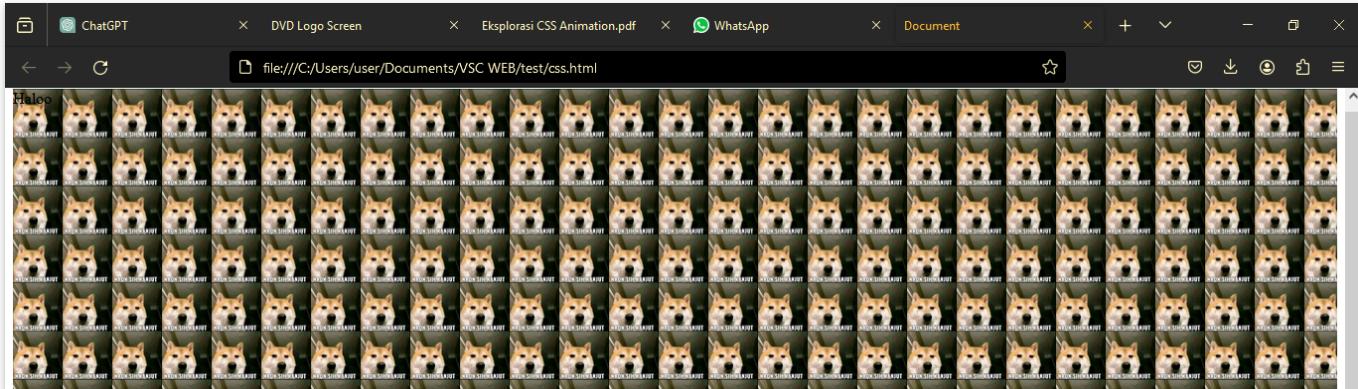
```
.test{  
background-image: url(naon.jpg);  
width: 100%;  
height: 100VH;  
  
background-size: 50%;
```



Background Repeat

- Background-repeat ini mengulang gambar dalam ukurannya dan untuk penggunaannya

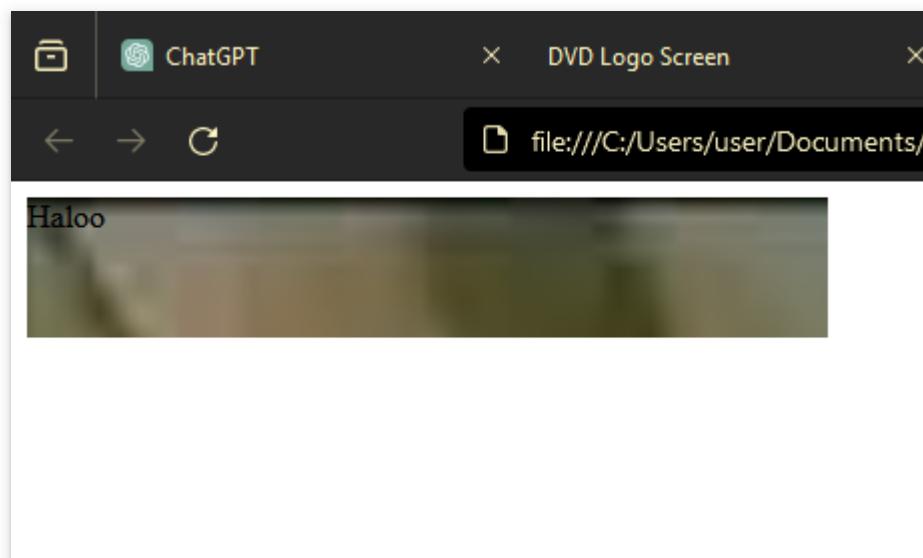
```
.container{  
background-image: url("./naon.jpg");  
width: 100%;  
height: 100vh;  
background-size: 50px;  
background-repeat: repeat;  
}
```



Background Attachment

- background-attachment ini mengatur ketetapan image yang di gunakan contohnya ialah seperti penggunaan value "fixed" yang membuat gambar tetap berada di posisinya walau di scroll

```
.container{  
background-image: url(naon.jpg);  
width: 400px;  
height: 70px;  
background-size: 90%;  
background-attachment: fixed;  
}
```



Background Position

- dan yang terakhir ialah background-position dan fungsinya ialah untuk mengatur posisi dari background yang digunakan, untuk contoh penggunaanya aka seperti dibawah ini :

```
.container {  
    background-position: 10% , 50%;  
}
```



Font

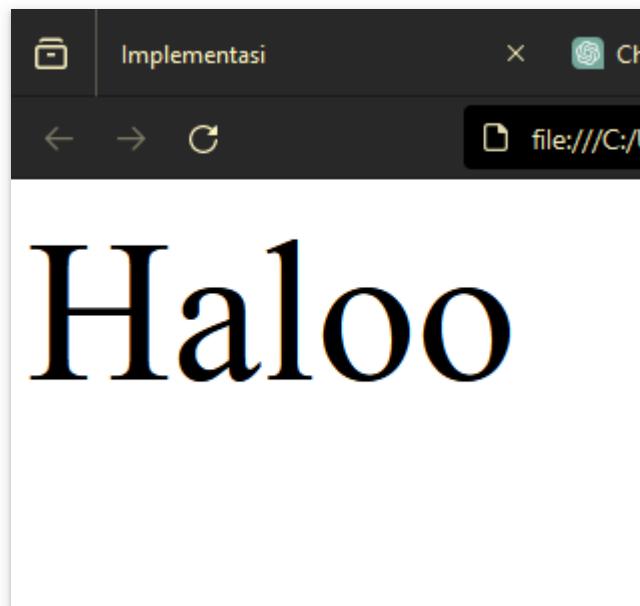
Font Size

Font-size adalah properti CSS yang digunakan untuk mengatur ukuran font teks dalam sebuah elemen HTML.

Kode Program

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

Hasil



Kesimpulan

properti font-size: 100px;, ukuran font dalam elemen tersebut akan diperbesar menjadi 100 piksel.

Font-style italic

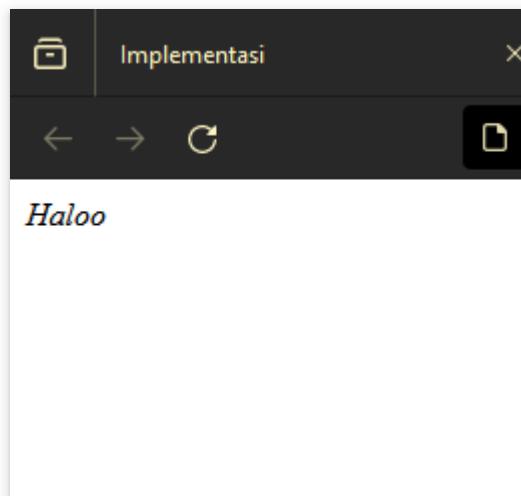
Font-style adalah properti CSS yang digunakan untuk menentukan gaya huruf (font style) dari sebuah teks.

italic: Menampilkan teks dalam gaya miring (italic), jika font tersebut mendukung.

Kode program

```
.element{  
    font-style: italic ;  
}
```

Hasil



Kesimpulan

properti `font-style: italic;`, teks dalam elemen tersebut akan ditampilkan dengan gaya huruf miring.

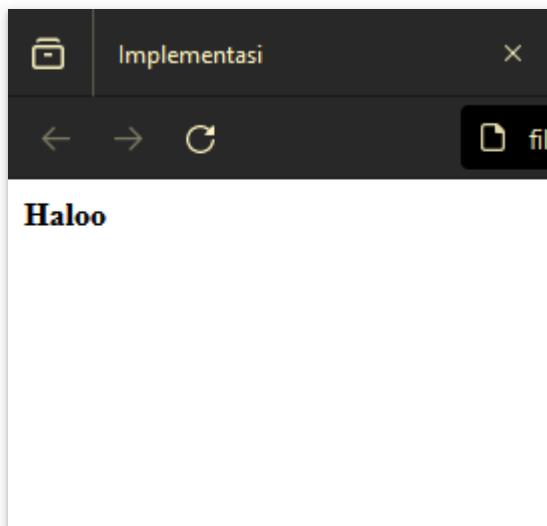
Font-weight Bold

`font-weight: bold;` dalam CSS, teks dalam elemen yang diberi aturan tersebut akan ditampilkan dengan ketebalan huruf yang lebih tebal dari biasanya.

Kode program

```
.element{  
    font-weight: bold ;  
}
```

Hasil



Kesimpulan

`font-weight: bold;`, teks dalam elemen tersebut akan memiliki gaya huruf tebal.

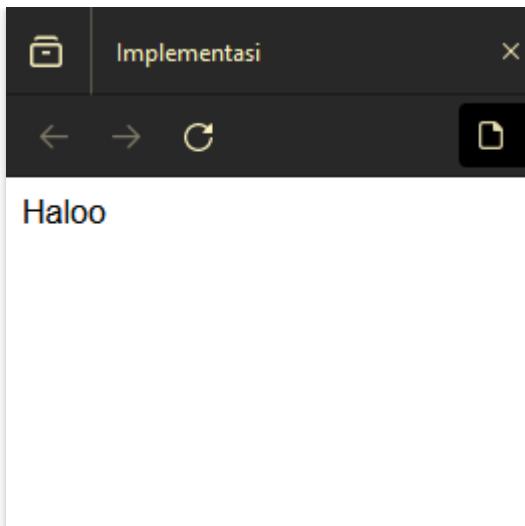
Font-family Arial

`Font-family justify` properti ini mengatur teks agar rata di kedua sisi, sehingga ruang antara kata-kata disesuaikan untuk mengisi seluruh lebar elemen dengan cara yang seragam.

Kode Program

```
.element{  
    font-family: arial;  
}
```

Hasil



Kesimpulan

atribut `font-family: arial;` akan mengubah gaya text dari default ke gaya arial

Box Model

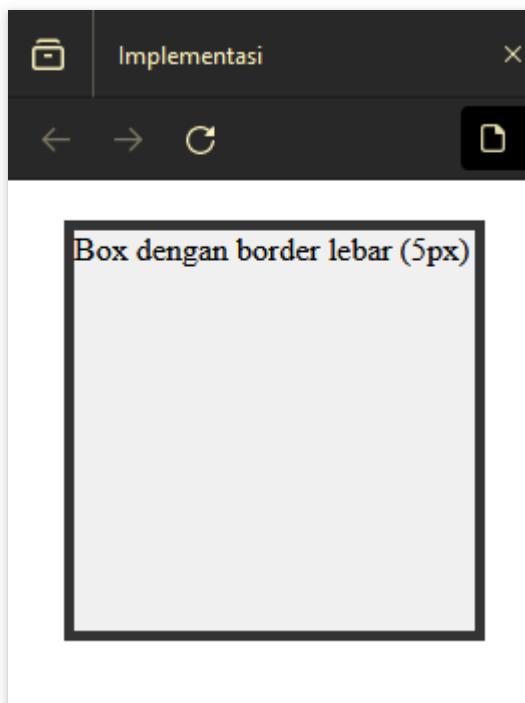
Border-Width

`border-width` digunakan untuk menentukan lebar border dari sebuah elemen.

Kode Program

```
.element{  
    border-width: 5px;  
}
```

Hasil



Kesimpulan

Dalam CSS, properti `border-width` digunakan untuk mengatur lebar dari border sebuah elemen

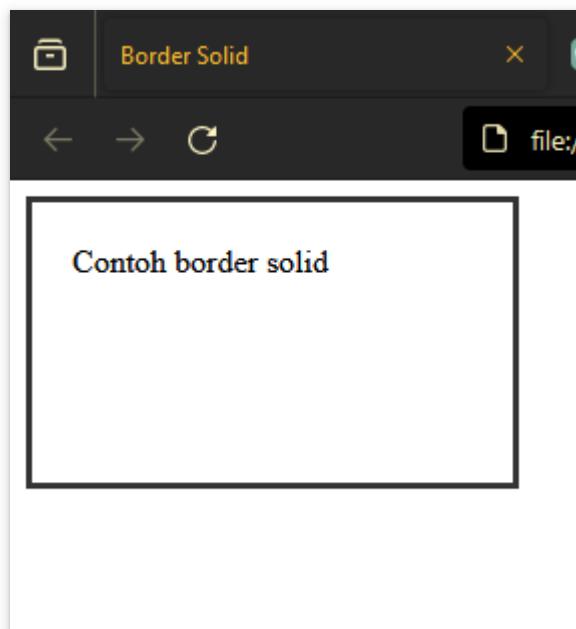
Border-style solid

`border-style: solid;` adalah properti CSS yang mengatur gaya border menjadi garis lurus dan solid.

Kode program

```
.element{  
    border-style: solid;  
}
```

Hasil



Kesimpulan

`border-style: solid;` adalah properti CSS yang mengatur gaya border menjadi garis lurus dan solid.

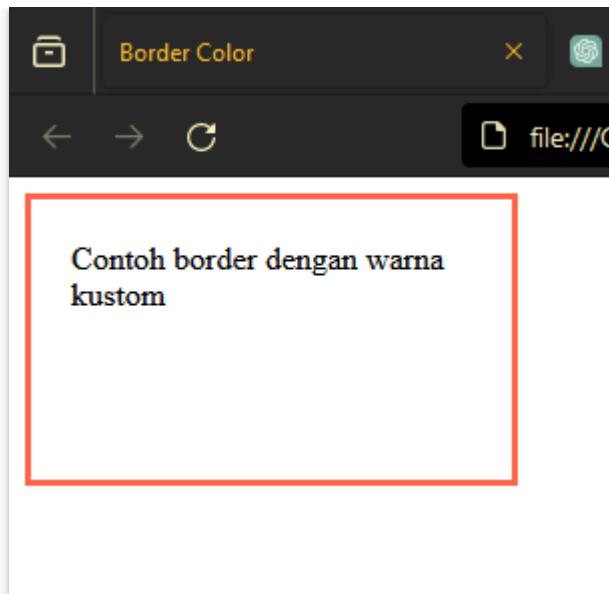
Border-color orangered

`border-color: orangered;` adalah properti CSS yang mengatur warna border dari sebuah elemen menjadi oranye merah (orangered).

Kode program

```
.element{  
    border-color: orangered;  
}
```

Hasil



Kesimpulan

Mengatur warna atau memberikan warna orangered pada pinggiran border

Box Model (Tantangan)

Penjelasan HTML

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

Penjelasan CSS

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

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

Kode Program

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

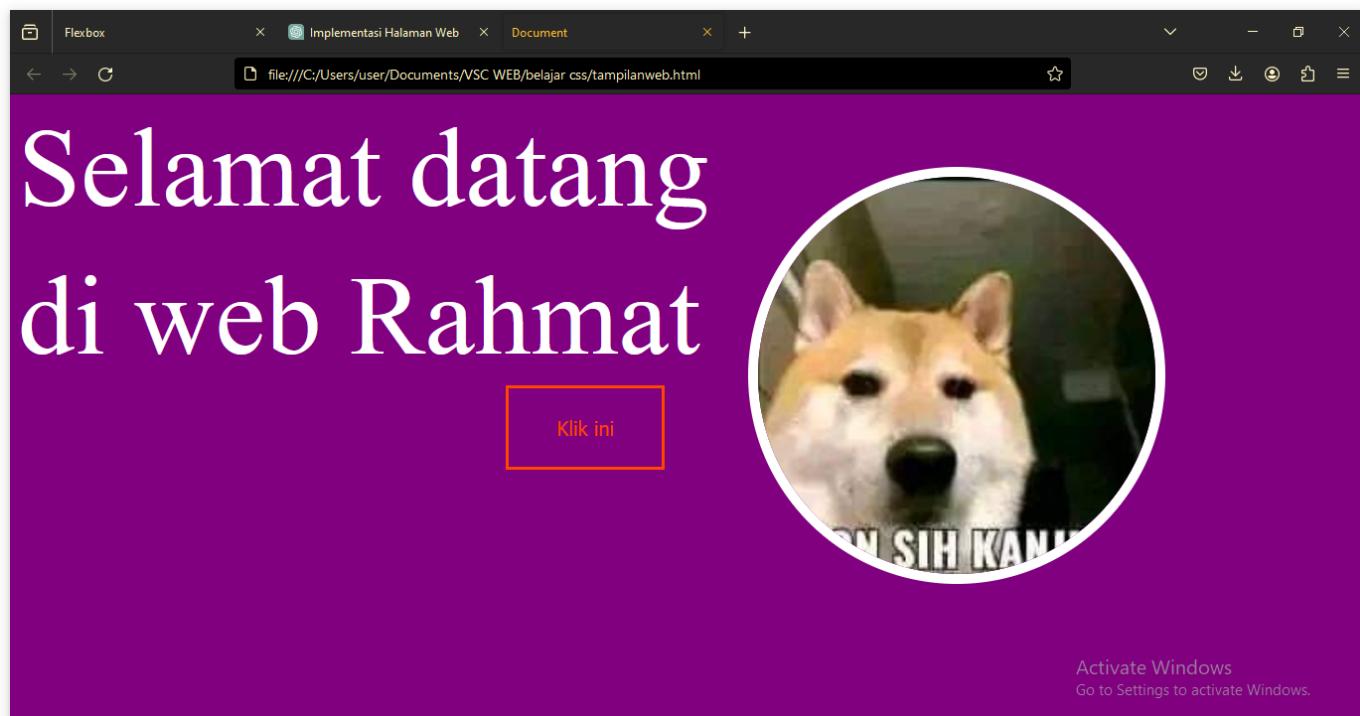
```
.text{
    font-family: 'Times New Roman';
    font-size: 30mm;
    color: white;
}

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

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

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

Hasil



Padding

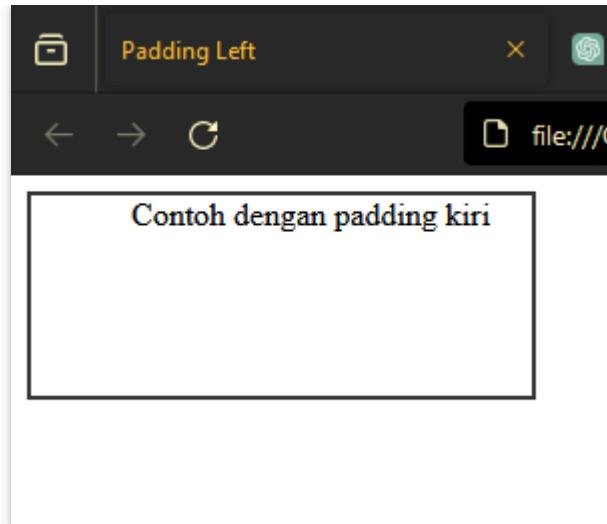
Padding-left

padding-left adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di sebelah kiri konten dalam sebuah elemen.

Kode Program

```
.element{  
    padding-left: 29px;  
}
```

Hasil



Kesimpulan

elemen HTML yang memiliki kelas "element" akan memiliki ruang tambahan di bagian kiri sebesar 29 piksel. Dan membuat tulisan klik disini tidak terlalu ke kiri.

Padding-bottom

padding-bottom adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di bagian bawah konten dalam sebuah elemen.

Kode Program

```
.element{  
    padding-bottom: 25px;  
}
```

Hasil



Kesimpulan

elemen HTML yang memiliki kelas "element" akan memiliki ruang tambahan di bagian bawah sebesar 25 piksel. Dan membuat tulisan klik disini tidak terlalu ke bawah.

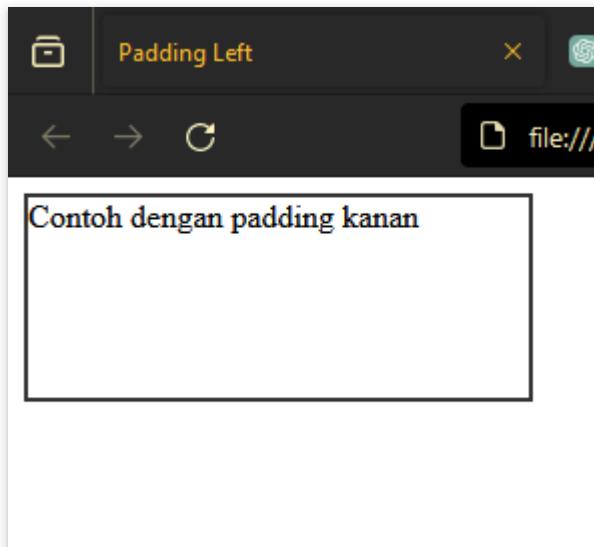
Padding-Right

padding-right adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di sebelah kanan konten dalam sebuah elemen.

Kode Program

```
.element{  
    padding-right: 20px;  
}
```

Hasil



Kesimpulan

elemen HTML yang memiliki kelas "element" akan memiliki ruang tambahan di bagian kanan sebesar 20 piksel. Dan membuat tulisan klik disini tidak terlalu ke kanan.

Padding-top

padding-top adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di bagian atas konten dalam sebuah elemen.

Kode program

```
.element{  
    padding-top: 20px;  
}
```

Hasil



Kesimpulan

elemen HTML yang memiliki kelas "element" akan memiliki ruang tambahan di bagian atasnya sebesar 20 piksel. Dan membuat tulisan klik disini tidak terlalu ke atas.

Margin

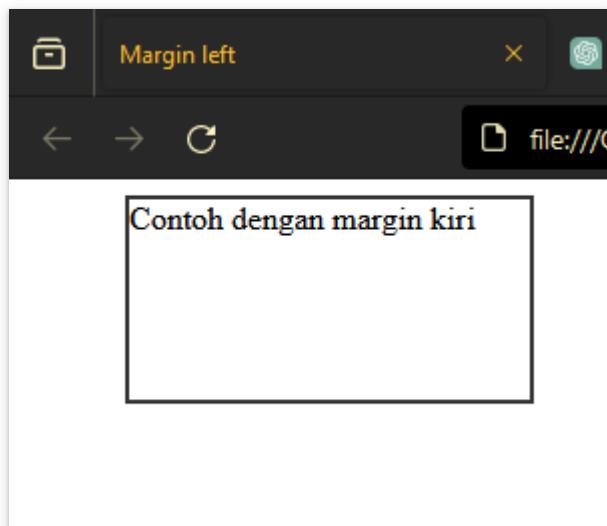
Margin-Left

margin-left adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di sebelah kiri dari sebuah elemen terhadap elemen lain di sekitarnya.

Kode Program

```
.element{  
    margin-left: 600px;  
}
```

Hasil



Margin-Top

margin-top adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di bagian atas dari sebuah elemen terhadap elemen lain di atasnya.

Kode Program

```
.element{  
    margin-top: 50px;  
}
```

Hasil



Kesimpulan

aturan gaya ini memengaruhi tata letak (layout) elemen-elemen HTML dengan selector class "element" dengan menambahkan margin di bagian atas sebesar 60 piksel.

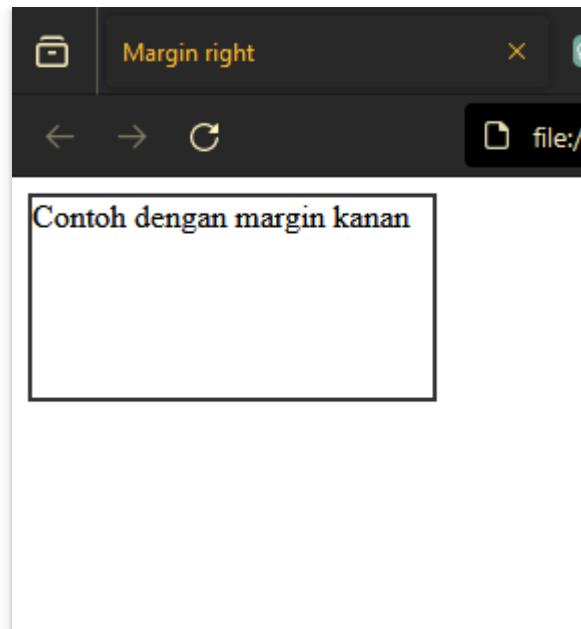
Margin-Right

margin-right adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di sebelah kanan dari sebuah elemen terhadap elemen lain di sekitarnya.

Kode program

```
.element{  
    margin-right: 60px;  
}
```

Hasil



Kesimpulan

elemen-elemen HTML dengan class "element" akan memiliki jarak atau spasi tambahan sebesar 60 piksel di sebelah kanan.

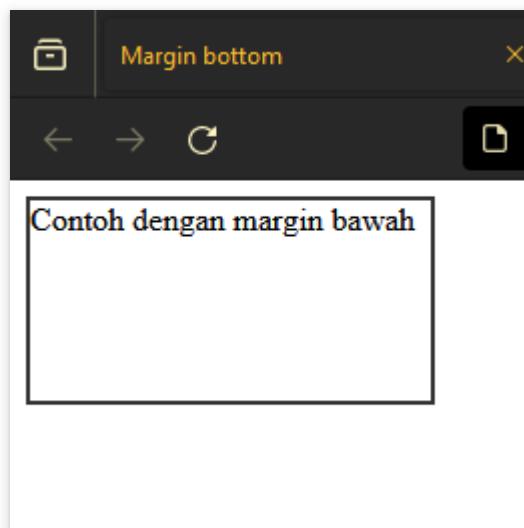
Margin-Bottom

margin-bottom adalah properti CSS yang digunakan untuk menentukan jarak atau ruang tambahan di bagian bawah dari sebuah elemen terhadap elemen lain di bawahnya.

Kode program

```
.element{  
    Margin-bottom: 90px  
}
```

Hasil



Kesimpulan

elemen-elemen yang memiliki kelas "element" akan memiliki jarak atau spasi sebesar 90 piksel di bagian bawahnya.

Pseudo-classes

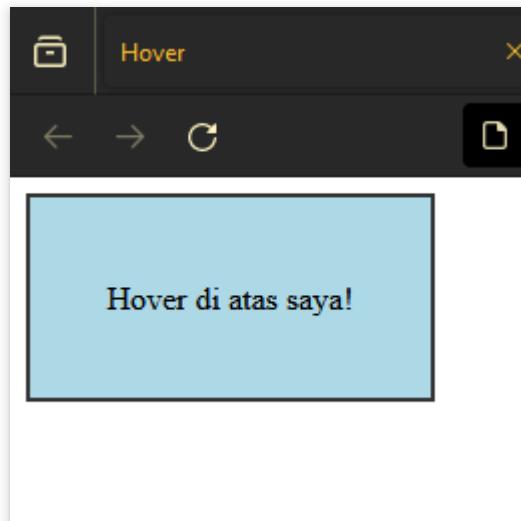
Hover

- Dalam HTML, kita memiliki sebuah div dengan kelas `.box`.
- Dalam CSS, kita memberikan gaya kepada `.box`, termasuk ukuran, latar belakang, border, dan lainnya.
- Dengan menggunakan selector `:hover`, kita menentukan perubahan gaya saat pengguna mengarahkan kursor ke atas elemen dengan kelas `.box`.
- Pada contoh ini, ketika pengguna melakukan hover, latar belakang elemen `.box` berubah menjadi lightblue dengan transisi yang halus.

Kode program

```
.box {  
    width: 200px;  
    height: 100px;  
    background-color: #f0f0f0;  
    border: 2px solid #333;  
    text-align: center;  
    line-height: 100px;  
    transition: background-color 0.3s ease;  
}  
  
.box:hover {  
    background-color: lightblue;  
}
```

Hasil



Kesimpulan

Sebuah kotak dengan latar belakang abu-abu, tetapi saat kursor mengarah ke atasnya, latar belakangnya berubah menjadi lightblue.

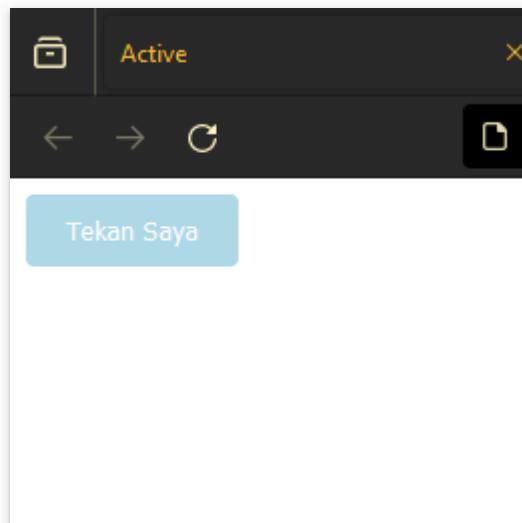
Active

- Dalam HTML, kita memiliki sebuah tombol (`<button>`) dengan kelas `.btn`.
- Dalam CSS, kita memberikan gaya kepada `.btn`, termasuk ukuran, warna, dan gaya border.
- Dengan menggunakan selector `:active`, kita menentukan perubahan gaya saat pengguna menekan tombol.
- Pada contoh ini, ketika tombol ditekan, latar belakangnya berubah menjadi abu-abu gelap (#666) dengan transisi yang halus.

Kode program

```
.btn {  
    padding: 10px 20px;  
    background-color: #333;  
    color: #fff;  
    border: none;  
    border-radius: 4px;  
    cursor: pointer;  
    transition: background-color 0.3s ease; /* transisi perubahan warna latar belakang */  
}  
  
.btn:hover {  
    background-color: lightblue; /* saat hover, latar belakang berubah menjadi lightblue */  
}  
  
.btn:active {  
    background-color: #666; /* saat ditekan, latar belakang berubah menjadi abu-abu gelap */  
}
```

Hasil



Kesimpulan

Hasilnya adalah sebuah tombol dengan latar belakang abu-abu saat ditekan.

Transition

Penjelasan

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

Kode program

```
.btn {  
  padding: 10px 20px;  
  background-color: #333;  
  color: #fff;  
  border: none;  
  border-radius: 4px;  
  cursor: pointer;  
  transition: background-color 0.3s ease;  
}  
  
.btn:hover {  
  background-color: lightblue;  
}  
}
```

Hasil



Transition(Tantangan)

Kode Program

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Document</title>
    <link rel="stylesheet" href="tugasflex.css">
</head>
<body bgcolor="purple">
    <div class="main-container">
        <div class="hero-container">
            <div class="item p">
                <p> Selamat Datang <br>
                    di <b>Web Rahmat</b> </P>
                    <button> Klik saya </button>
            </div>
            <div>
                <span class="img">
                    
                </span>
            </div>
        </div>
    </body>
</html>
```

Penjelasan HTML

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

```

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

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

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

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

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

```

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

button:hover {
    background-color: orange;
    color: white;
    width: 150px;
    transition: all 0.3s ease-in;
    cursor: pointer;
}

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

Penjelasan CSS

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

Hasil



Transform

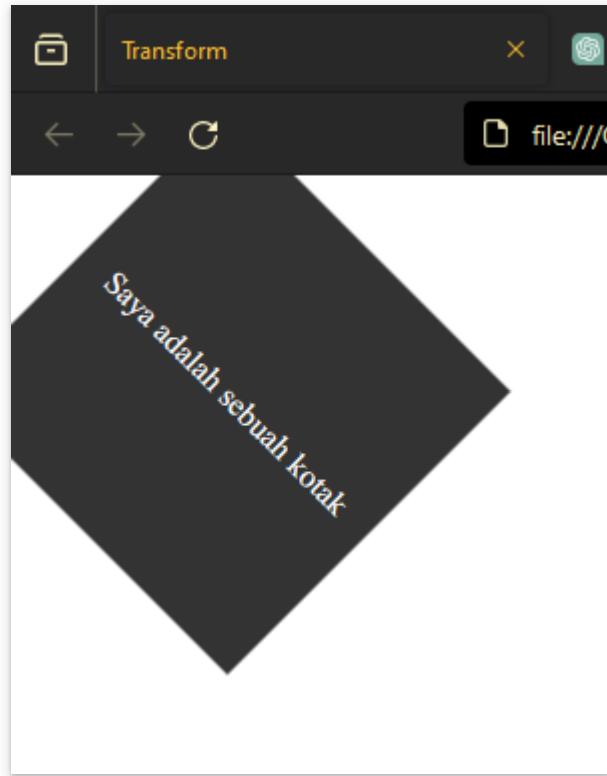
Penjelasan

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

Kode Program

```
.box {  
    width: 200px;  
    height: 200px;  
    background-color: #333;  
    color: #fff;  
    text-align: center;  
    line-height: 200px;  
    transform: rotate(45deg);  
}
```

Hasil



Kesimpulan

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

Flexbox

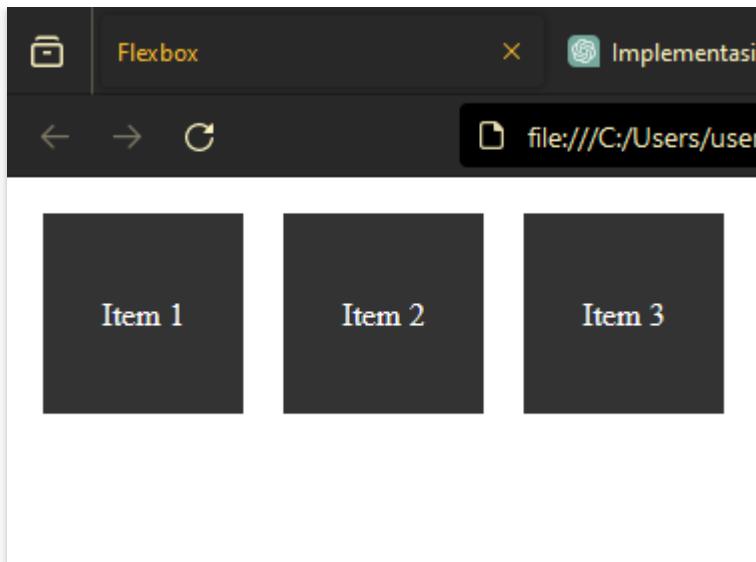
Display Flex

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

Kode Program

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

Hasil



Kesimpulan

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

Flex-Direction

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

Kode Program

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

Hasil



Kesimpulan

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

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

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

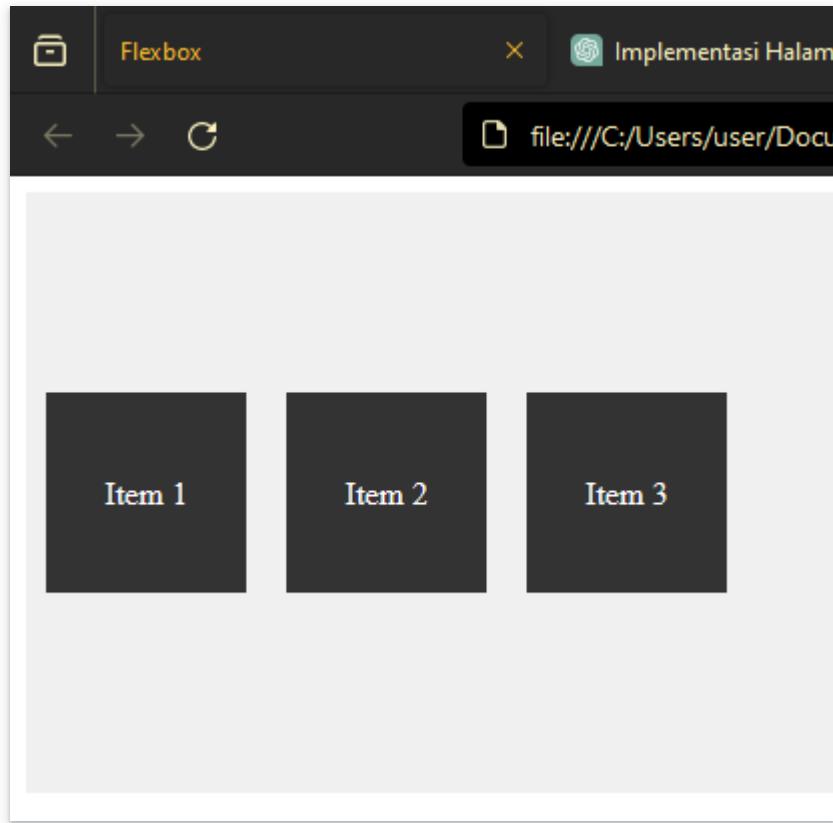
Align-Items

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

Kode Program

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

Hasil



Kesimpulan

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

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

Justify-Content

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

terakhir, tetapi tidak ada ruang di antara fleks item yang berdekatan.

Kode Program

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

Hasil



Kesimpulan

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

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

Flex Box(Tantangan)

Kode Program

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <title>Document</title>
```

```

<link rel="stylesheet" href="styles.css">
</head>
<body bgcolor="purple">
    <div class="main-container">
        <div class="hero-container">
            <div class="item p">
                <p> Selamat Datang <br>
                    di <b>Web Rahmat</b> </P>
                <button> Klik saya </button>
            </div>
            <div>
                <span class="img">
                    
                </span>
            </div>
        </div>
    </body>
</html>

```

Penjelasan HTML

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

```

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

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

```

```

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

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

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

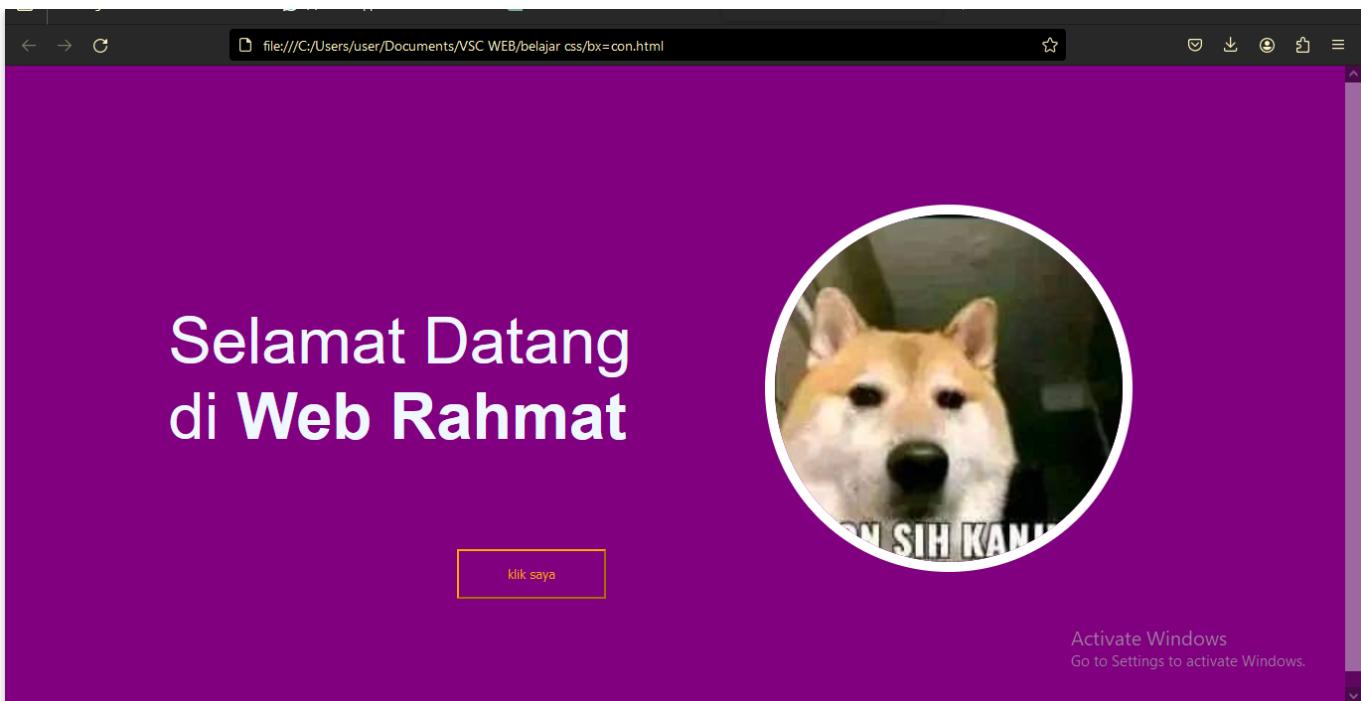
button {
    background-color: purple;
    width: 150px;
    height: 50px;
    border-width: 2px;
    color: orange;
    border-color: orange;
    margin-bottom: 20px;
    margin-left: 290px;
    box-shadow: 20px;
}

```

Penjelasan CSS

- **.main-container, .hero-container**: Membuat tata letak menggunakan flexbox dengan menentukan dimensi, warna latar belakang, dan penempatan.
- **.item**: Mengatur dimensi dan warna latar belakang untuk item.
- **.p**: Mengatur gaya teks untuk paragraf.
- **img**: Mengatur margin dan gaya border untuk gambar.
- **button**: Mengatur gaya untuk tombol, termasuk dimensi, warna latar belakang, warna teks, dan margin.

Hasil



Kesimpulan

Program ini menggabungkan HTML untuk struktur konten dan CSS untuk merancang tampilan. Elemen-elemen seperti teks, gambar, dan tombol diberikan gaya agar terlihat menarik dan responsif. Penggunaan flexbox dalam CSS memungkinkan penataan elemen-elemen secara fleksibel dalam layout halaman. Dengan demikian, program ini menciptakan halaman web sederhana dengan desain yang menarik dan responsif.

Position

Position Relative

Penjelasan

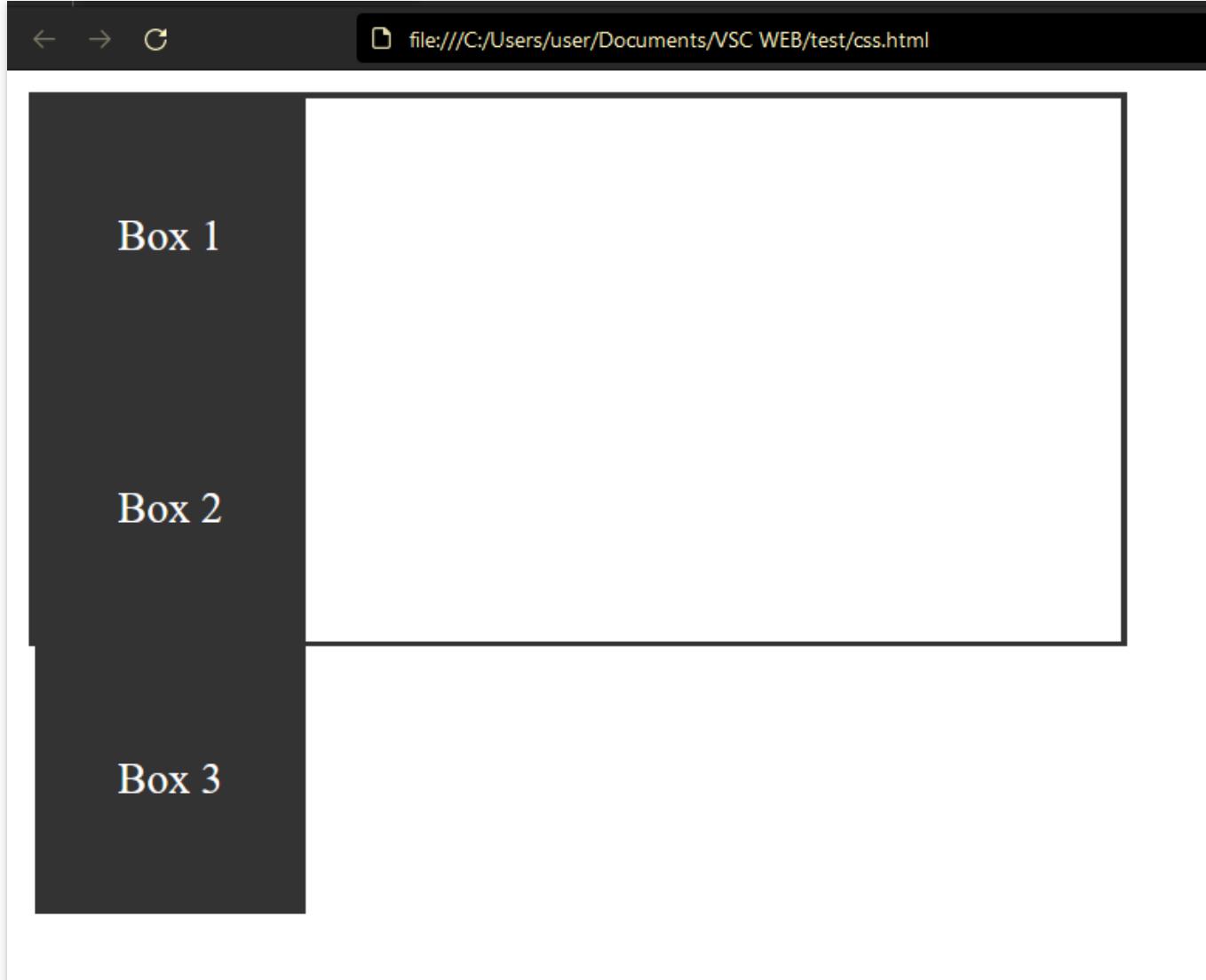
- Dalam HTML, kita memiliki sebuah div dengan kelas `.container` yang berisi tiga div dengan kelas `.box`.
- Dalam CSS, kita menerapkan `position: relative` pada `.container` untuk membuatnya menjadi referensi bagi posisi kotak-kotak anaknya yang memiliki `position: absolute`.
- Setiap kotak diberi properti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

Kode Program

```
.container {  
  position: relative;  
  width: 400px;  
  height: 200px;  
  border: 2px solid #333;  
}
```

```
.box {  
  position: relative;  
  width: 100px;  
  height: 100px;  
  background-color: #333;  
  color: #fff;  
  text-align: center;  
  line-height: 100px;  
}
```

Hasil



Kesimpulan

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

Position Absolute

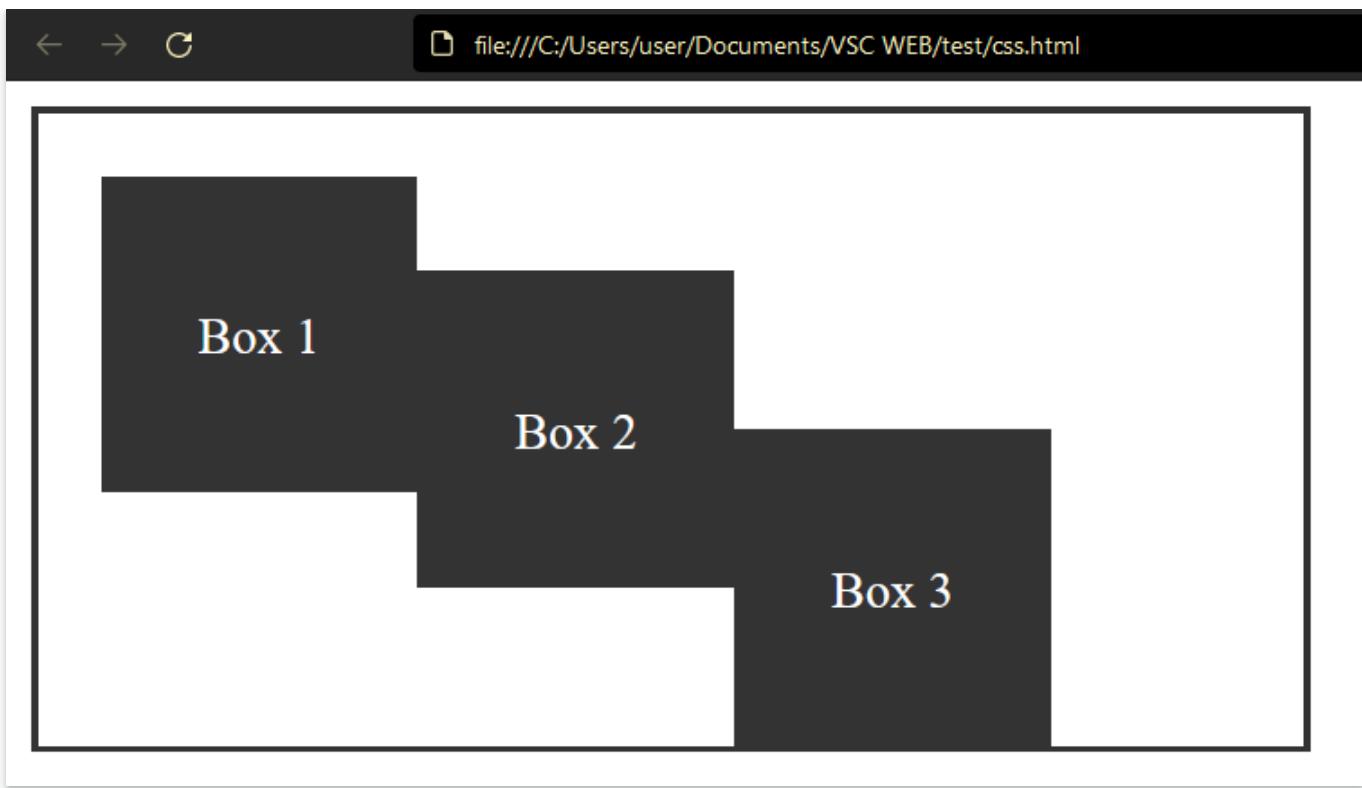
Penjelasan

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

Kode Program

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

Hasil



Kesimpulan

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

Position Fixed

Kode Program

```
.fixed-box {  
    position: fixed; /* Menggunakan posisi fixed */  
    top: 80px;  
    left: 50px;  
    width: 200px;  
    height: 100px;  
    background-color: #333;  
    color: #fff;  
    padding: 10px;  
}
```

Hasil



Kesimpulan

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

Position Sticky

Penjelasan

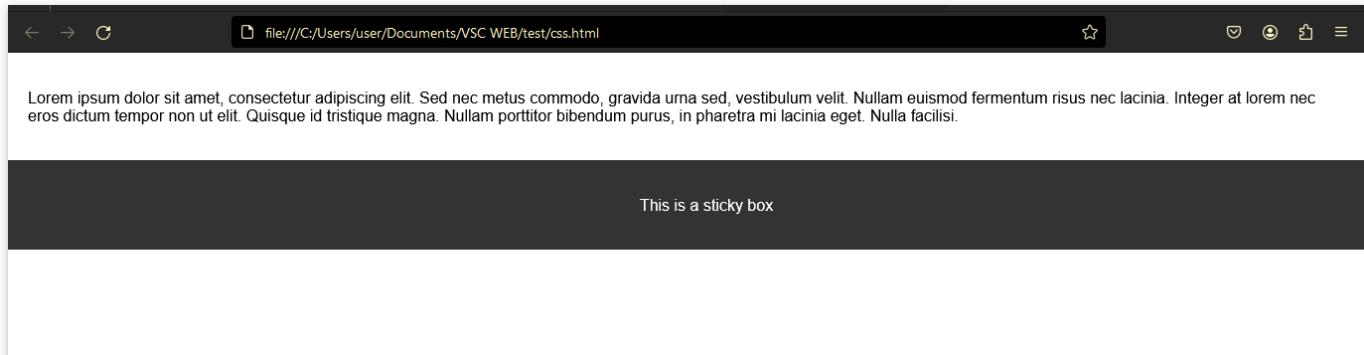
- Kami memiliki sebuah div untuk konten (`content`) di HTML.
- Di CSS, kami memberikan padding pada `.content` untuk memberikan ruang di sekitar konten.
- Kami menambahkan sebuah div dengan kelas `.sticky-box`, yang akan dijadikan elemen sticky. Properti `position: sticky` diterapkan untuk membuatnya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

Kode Program

```
body {  
    margin: 0;  
    font-family: Arial, sans-serif;  
}  
  
.content {  
    padding: 20px;  
}  
  
.sticky-box {  
    background-color: #333;  
    color: #fff;  
    text-align: center;  
    position: sticky;  
    top: 0;  
}
```

```
    width: 100%;  
    padding: 20px;  
}
```

Hasil



Kesimpulan

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

Positon(Tantangan)

Kode Program

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

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

Penjelasan HTML

- `<!DOCTYPE html>`: Mendefinisikan tipe dokumen HTML.
- `<html lang="en">`: Menandakan awal dari dokumen HTML dengan bahasa Inggris.
- `<head>`: Bagian kepala dokumen HTML yang berisi metadata dan referensi ke stylesheet.
- `<meta charset="UTF-8">`: Menentukan pengkodean karakter dokumen sebagai UTF-8.
- `<meta http-equiv="X-UA-Compatible" content="IE=edge">`: Menentukan versi IE yang kompatibel.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`: Menyesuaikan tampilan halaman dengan lebar perangkat.
- `<title>Tantangan Position</title>`: Memberikan judul halaman.
- `<style>`: Mulai dari bagian CSS yang didefinisikan dalam dokumen HTML.
- `<body>`: Bagian utama dari dokumen HTML yang berisi konten yang akan ditampilkan di browser.
- `<div class="container">`: Sebuah div yang bertindak sebagai wadah untuk semua konten. Ini memiliki beberapa div anak yang akan di-styling.
- `<div class="item1">`, `<div class="item2">`, `<div class="item3">`: Div-div yang bertindak sebagai item atau bagian-bagian dari konten.
- ``: Elemen gambar untuk menampilkan gambar.
- `<p>`, `<h4>`: Elemen paragraf dan heading untuk menampilkan teks.
- `<button>`: Tombol untuk berbagai fungsi, seperti menyukai konten.
- `<div class="like">`: Sebuah div yang berisi tombol "suka" atau "like".

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

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

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

img {
```

```
width:100%,
height:100%,
border-radius: 5px 5px 0px 0px;
}

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

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

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

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

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

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

button {
width: 62px;
height: 55px;
background-color: transparent;
border: none;
border-radius: 100px 100px 100px 100px;
```

```
    position: fixed;
    background-repeat: no-repeat;
    top: 350px;
    right: 520px;
}

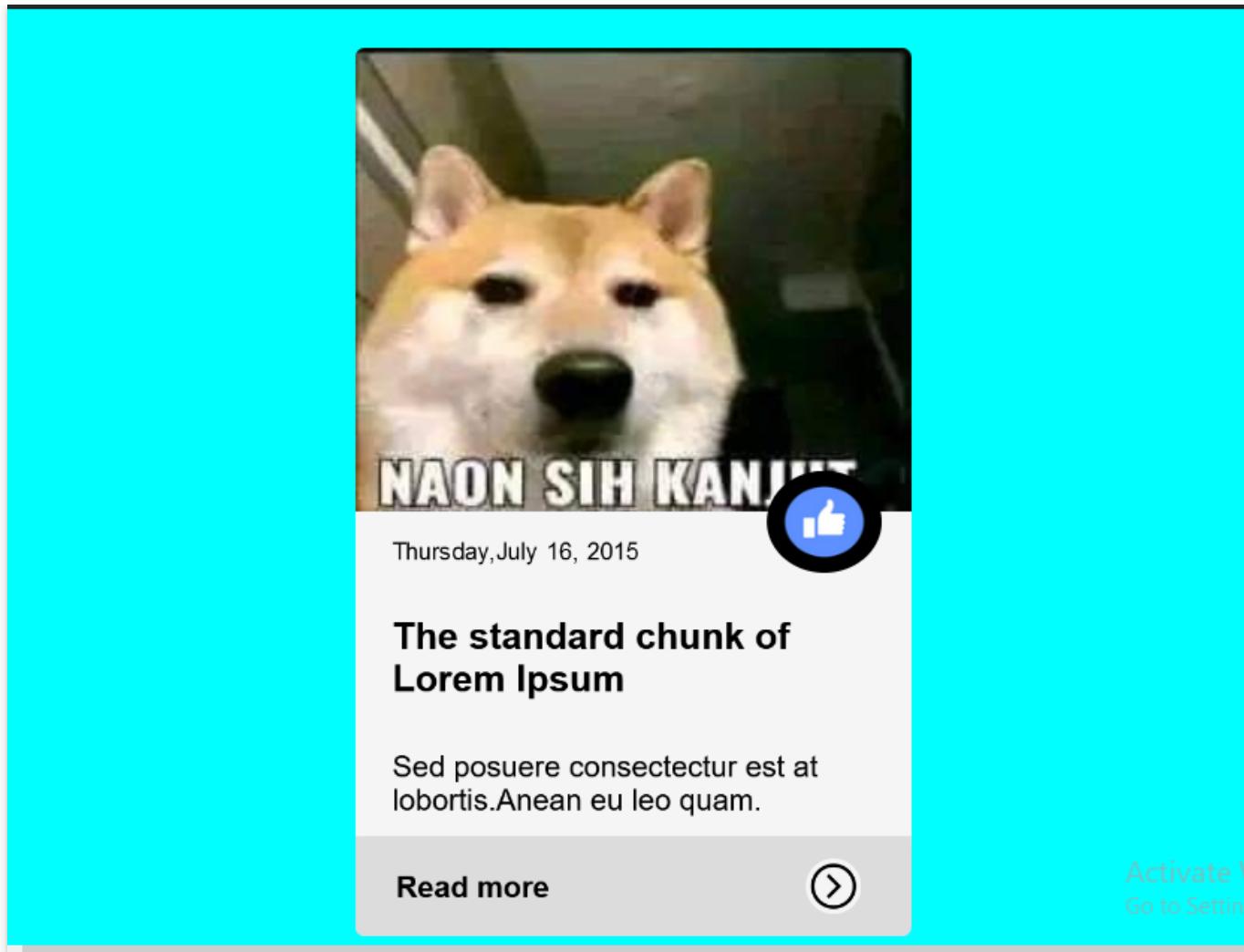
.like img {
    border-radius: 100% ;
    margin-right: 110px;
    margin-left: 135px;
    position: absolute;
}

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

Penjelasan CSS

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

Hasil



Kesimpulan

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

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 klarifikasi. Komponen carousel Bootstrap digunakan untuk membuat tampilan slide yang interaktif. kita dapat menambahkan gambar, teks, dan tombol navigasi ke dalam carousel. Carousel Bootstrap mendukung navigasi otomatis, kontrol manual, dan animasi transisi.

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

Contoh Penerapan Komponen Bootstrap

Typography

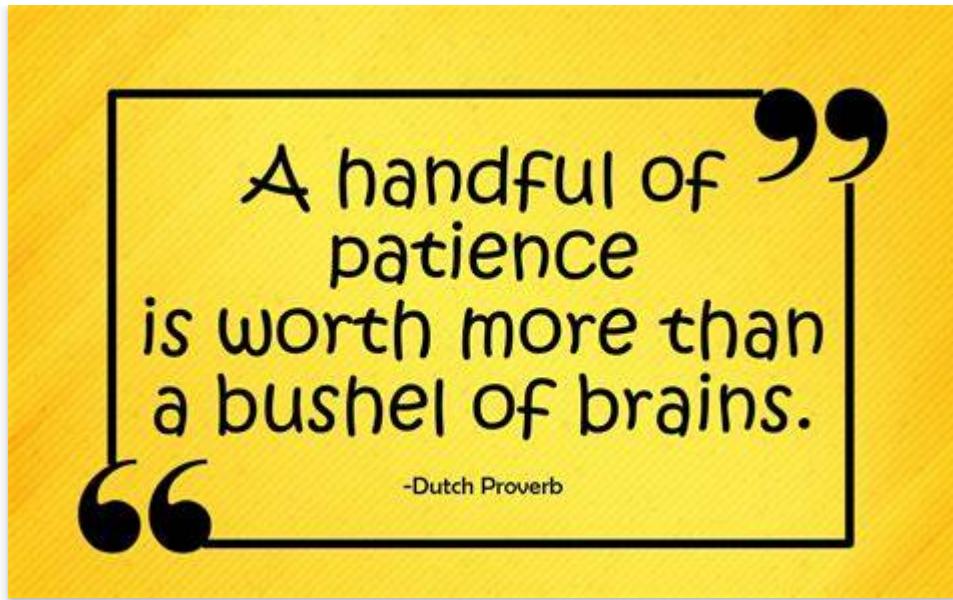
1. Buka situs resmi Bootstrap di getbootstrap.com



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

This screenshot shows the 'Get started with Bootstrap' page. On the left, there's a sidebar with a 'Getting started' section containing 'Introduction' (which is highlighted in purple) and other items like 'Download', 'Contents', 'Browsers & devices', 'JavaScript', 'Webpack', 'Parcel', 'Vite', 'Accessibility', 'RFCS', 'RTL', and 'Contribute'. Below that is a 'Customize' section with 'Overview', 'Sass', 'Options', 'Color', 'Color modes', 'Components', and 'CSS variables'. The main content area has a large heading 'Get started with Bootstrap'. It includes a sub-section titled 'Quick start' with instructions and sample code. To the right, there's a sidebar titled 'On this page' with links to 'Quick start', 'CDN links', 'Next steps', 'JS components', 'Important globals', 'HTML5 doctype', 'Viewport meta', 'Box-sizing', 'Reboot', and 'Community'. An advertisement for Adobe Stock is also visible on the page.

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



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

The screenshot shows the Bootstrap documentation website with a dark theme. The top navigation bar includes links for Docs, Examples, Icons, Themes, and Blog, along with a search bar and version information (v5.3). The left sidebar has a navigation menu with sections like Breakpoints, Containers, Grid, Columns, Gutters, Utilities, Z-index, CSS Grid, Content (with sub-links for Reboot, Typography, Images, Tables, Figures), and Forms (with sub-links for Overview, Form control, Select, Checks & radios, Range, Input group). The main content area features a large heading 'Typography' with a sub-subtitle 'Documentation and examples for Bootstrap typography, including global settings, headings, body text, lists, and more.' Below this is a 'Global settings' section with a list of bullet points. To the right, there's a sidebar titled 'On this page' with links to various Bootstrap components like Headings, Text utilities, and Lists.

Typography

Documentation and examples for Bootstrap typography, including global settings, headings, body text, lists, and more.

Global settings #

Bootstrap sets basic global display, typography, and link styles. When more control is needed, check out the [textual utility classes](#).

- Use a [native font stack](#) that selects the best `font-family` for each OS and device.
- For a more inclusive and accessible type scale, we use the browser's default root `font-size` (typically 16px) so visitors can customize their browser defaults as needed.
- Use the `$font-family-base`, `$font-size-base`, and `$line-height-base` attributes as our typographic base applied to the `<body>`.
- Set the global link color via `$link-color`.
- Use `$body-bg` to set a `background-color` on the `<body>` (#fff by default).

5. Carilah Subbab "Alignment":

Alignment

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

A well-known quote, contained in ablockquote element.

— Someone famous in *Source Title*

HTML



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

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

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

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

7. Hasilnya akan terlihat seperti berikut:

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

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

```
<figure class="text-center">
<blockquote class="blockquote">
    <p>Sebenarnya otak kita sama aja, yang <br>
        membedakan hanya siapa yang <br>
        mulai belajar duluan dan siapa yang <br>
        belajar terus menerus </p>
</blockquote>
<figcaption class="blockquote-footer">
    Dea Afrizal Doroboka<cite title="Source Title">Doroboka</cite>
</figcaption>
</figure>
```

Hasilnya adalah sebagai berikut:

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

Keterangan :

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

Keterangan:

Untuk bagian "Dea Afrizal" adalah nama seseorang yang mengatakan quotes tersebut dan "Doroboka" adalah sumber dimana orang tersebut mengatakn kata-kata itu. Kata- kata yang dituliskan yaitu berada dalam tag `p` .

Navbar

1. Bukalah Kembali getbootstrap.com lalu ketik "navbar" dikolom "search" dan klik enter



Badge
Breadcrumb
Buttons
Button group
Card
Carousel
Close button
Collapse
Dropdowns
List group
Modal
Navbar
Navs & tabs
Offcanvas
Pagination
Placeholders
Popovers
Progress
Scrollspy
Spinners
Toasts
Toolips

Waiting for ak7kmzkzhq-2.algolianet.com...

Navbar

[View on GitHub](#)

Documentation and examples for Bootstrap's powerful, responsive navigation header, the navbar. Includes support for branding, navigation, and more, including support for our collapse plugin.

How it works

Here's what you need to know before getting started with the navbar:

- Navbars require a wrapping `.navbar` with `.navbar-expand{-sm|-md|-lg|-xl|-xxl}` for responsive collapsing and `color-scheme` classes.
- Navbars and their contents are fluid by default. Change the `container` to limit their horizontal width in different ways.
- Use our `spacing` and `flex` utility classes for controlling spacing and alignment within navbars.
- Navbars are responsive by default, but you can easily modify them to change that. Responsive behavior depends on our Collapse JavaScript plugin.
- Ensure accessibility by using a `<nav>` element or, if using a more generic element such as a `<div>`, add a `role="navigation"` to every navbar to explicitly identify it as a landmark region for users of assistive technologies.
- Indicate the current item by using `aria-current="page"` for the current page or `aria-current="true"` for the current item in a set.

New in v5.2.0: Navbars can be themed with CSS variables that are scoped to the `.navbar` base class. `.navbar-`

On this page

How it works
Supported content
Brand
Text
Image
Image and text
Nav
Forms
Text
Color schemes
Containers
Placement
Scrolling
Responsive behaviors
Toggler
External content
Offcanvas
CSS
Variables
Sass variables
Sass loops

2. Carilah jenis navbar yang diinginkan.

The screenshot shows the Bootstrap documentation page for the Navbar component. On the left, there is a sidebar with a list of navigation items. The 'Image and text' item is highlighted with a purple background. The main content area displays the 'Image and text' example. It features a dark header with a purple 'B' logo and the word 'Bootstrap'. Below the header, there is an 'HTML' code editor containing the following code:

```
<nav class="navbar bg-body-tertiary">
  <div class="container">
    <a class="navbar-brand" href="#">
      
    </a>
  </div>
</nav>
```

On this page

How it works
Supported content
Brand
Text
Image
Image and text
Nav
Forms
Text
Color schemes
Containers
Placement
Scrolling
Responsive behaviors
Toggler
External content
Offcanvas
CSS
Variables
Sass variables
Sass loops

Image and text

You can also make use of some additional utilities to add an image and text at the same time. Note the addition of `.d-inline-block` and `.align-text-top` on the ``.

The screenshot shows the Bootstrap documentation page for the Navbar component. On the left, there is a sidebar with a list of navigation items. The 'Image and text' item is highlighted with a purple background. The main content area displays the 'Image and text' example. It features a dark header with a purple 'B' logo and the word 'Bootstrap'. Below the header, there is an 'HTML' code editor containing the following code:

```
<nav class="navbar bg-body-tertiary">
  <div class="container">
    <a class="navbar-brand" href="#">
      <img alt="Bootstrap" class="d-inline-block align-text-top" width="30" height="20"/>
      Bootstrap
    </a>
  </div>
</nav>
```

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

Navbar Home Features Pricing Dropdown link ▾

HTML



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

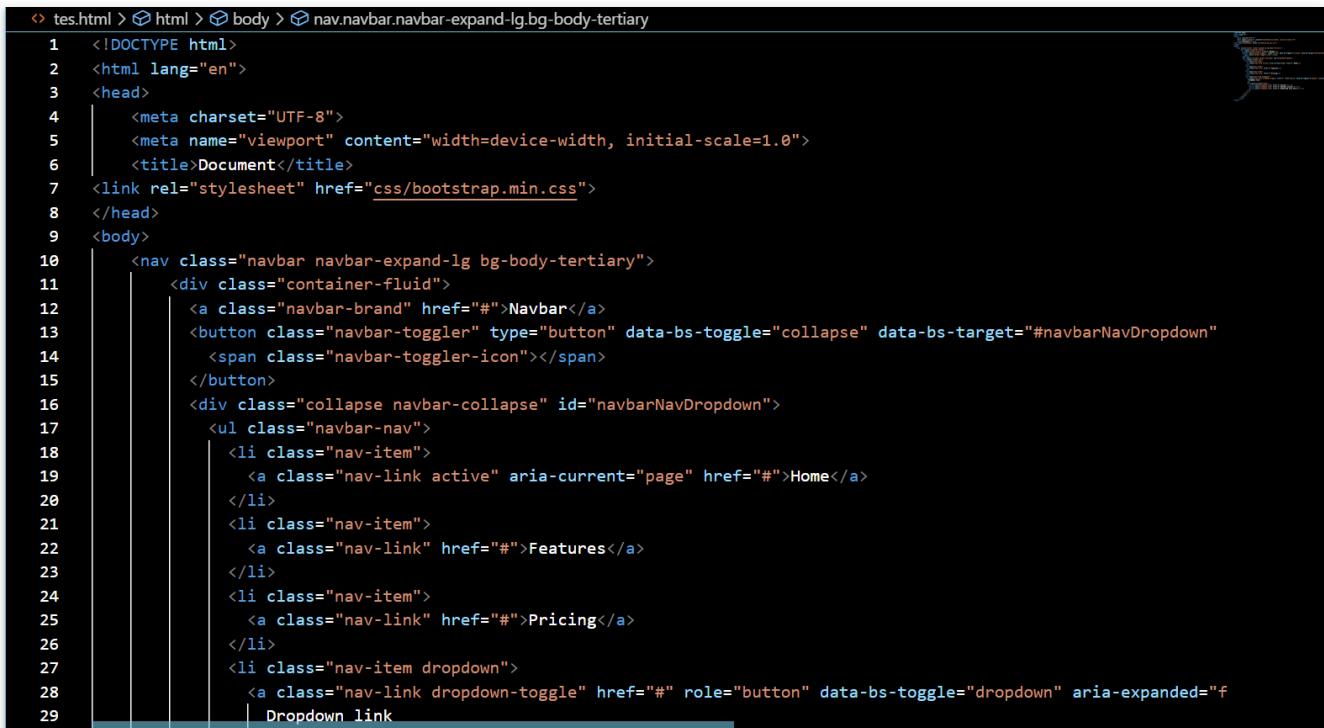
```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false"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

```

        <li><a class="dropdown-item" href="#">Action</a></li>
        <li><a class="dropdown-item" href="#">Another action</a></li>
        <li><a class="dropdown-item" href="#">Something else here</a></li>
    </ul>
</li>
</ul>
</div>
</div>
</nav>

```

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

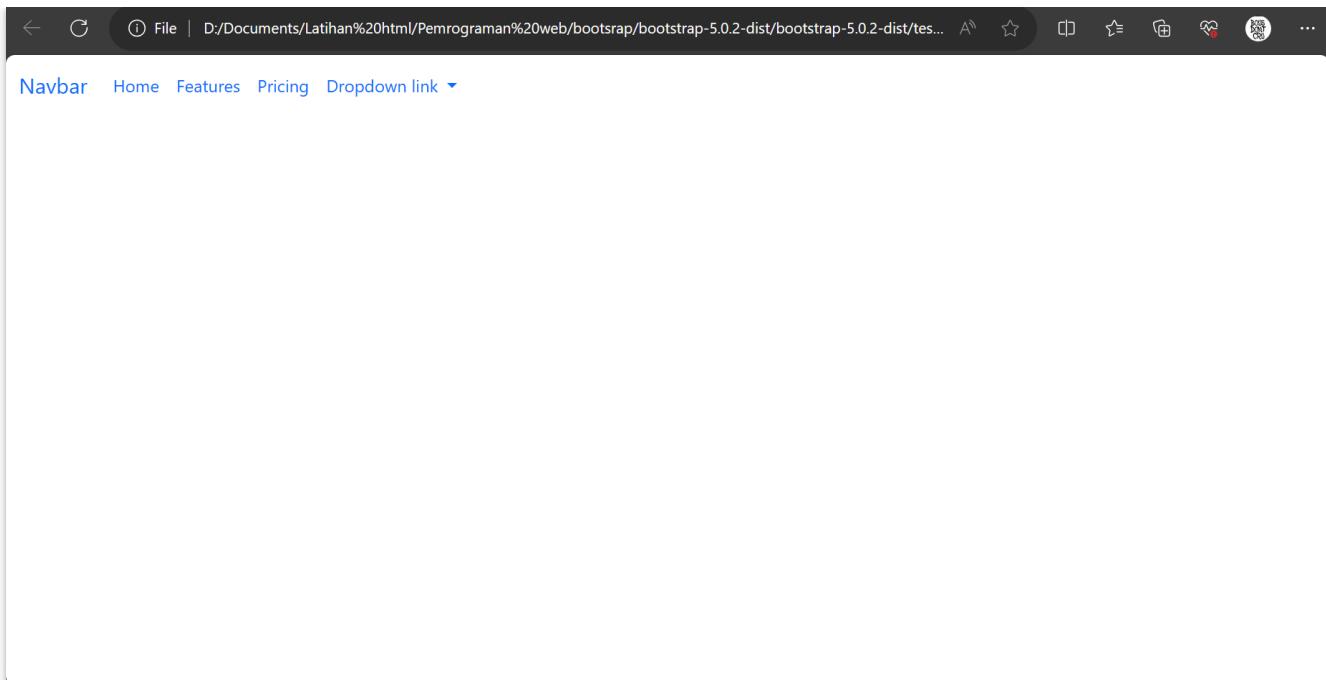


```

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

```

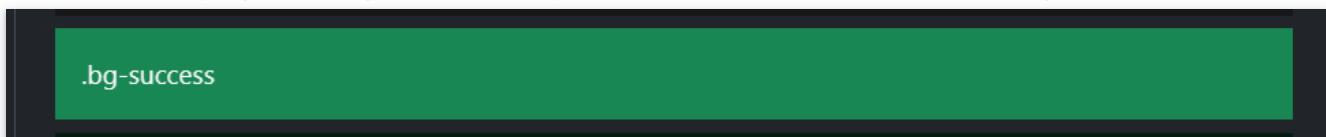
5. Jalankanlah file html yang sudah ditempelki kode program bootstrap tadi melalui web browser. Maka lihatlah hasilnya

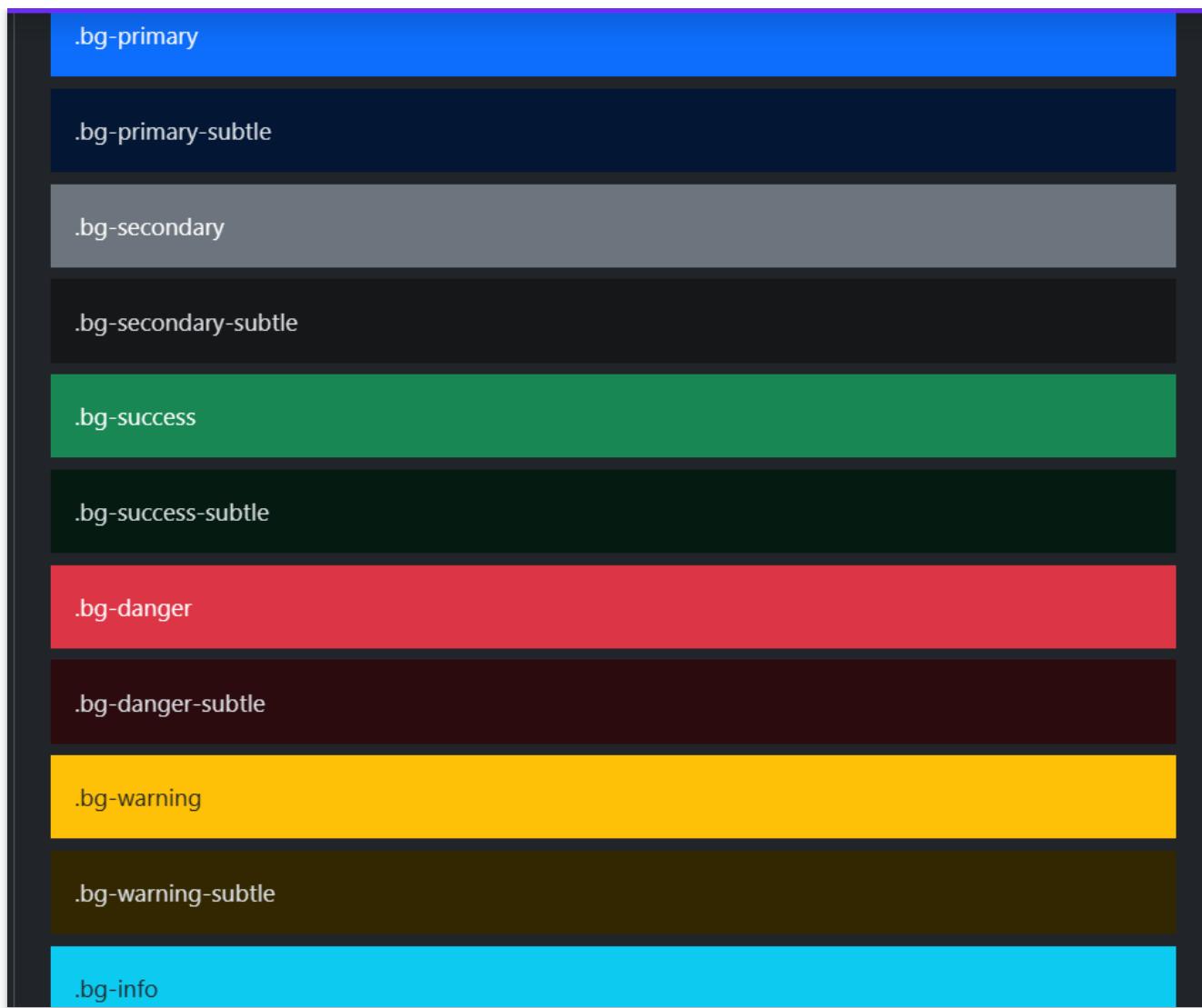


6. Misalnya kita ingin memberikan warna hijau pada background navbar kita. Pertama ketiklah "background" di kolom search

The screenshot shows the Bootstrap documentation for version 5.3. The top navigation bar includes links for Docs, Examples, Icons, Themes, and Blog, along with a search bar and social media icons. The left sidebar has a tree view of utility classes, with 'Background' selected. The main content area features a large heading 'Background' and a sub-section titled 'Background color'. It includes an accessibility tip and a code snippet for the '.bg-primary' class.

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





Terdapat banyak jenis warna dalam bootstrap.

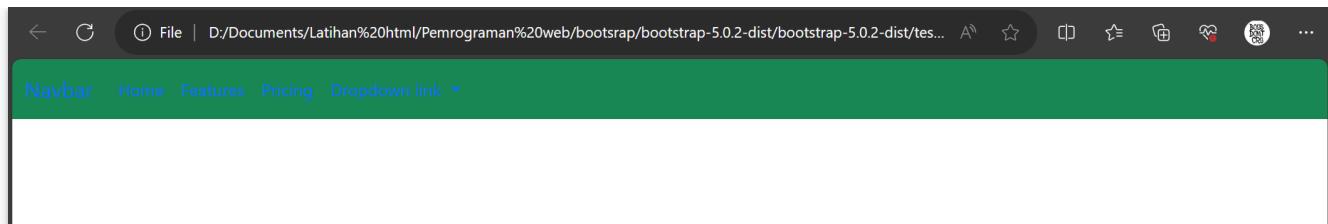
8. Karena kita ingin mengkostumisasi warna background kita menjadi hijau maka kita akan memakai bg-success . ketikkan bg-success pada class tag pembuka seperti di bawah ini

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

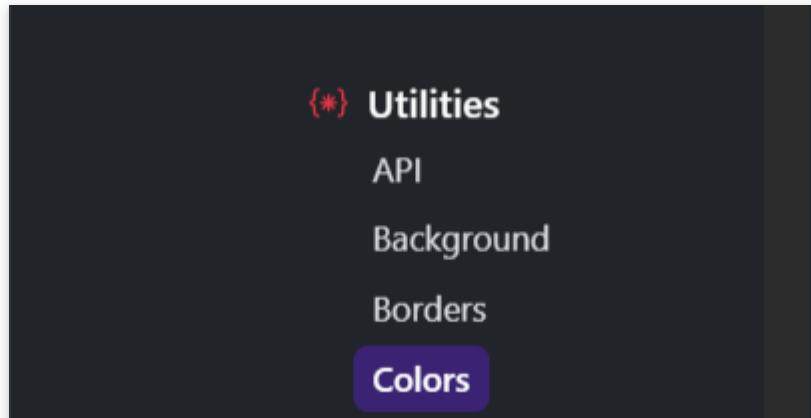
Keterangan:

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

9. Jika sudah maka hasilnya akan seperti ini:



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



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

.text-secondary-emphasis

.text-success

.text-success-emphasis

.text-danger

.text-danger-emphasis

.text-warning

.text-warning-emphasis

.text-info

.text-info-emphasis

.text-light

.text-light-emphasis

.text-dark

.text-dark-emphasis

.text-body

.text-body-emphasis

.text-body-secondary

.text-body-tertiary

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

```
.text-black
```

```
.text-white
```

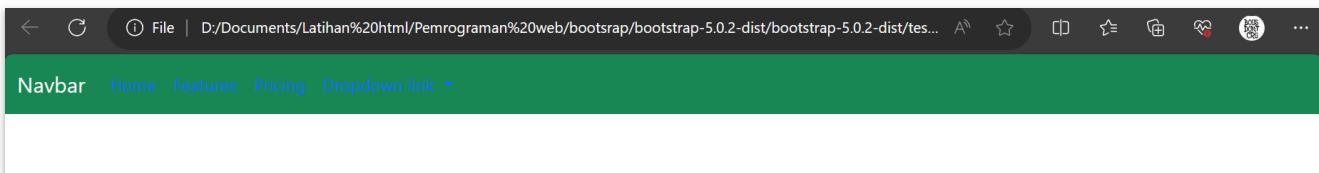
```
.text-black-50
```

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

13. Karena kita ingin menggunakan warna font putih maka perintah yang digunakan adalah "text-white"
14. Masuklah kembali ke dalam file html kita
15. pada bagian class sebuah tag yang membungkus text yang akan kita ganti warna, ketiklah text-white . Contohnya ada pada gambar berikut:

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

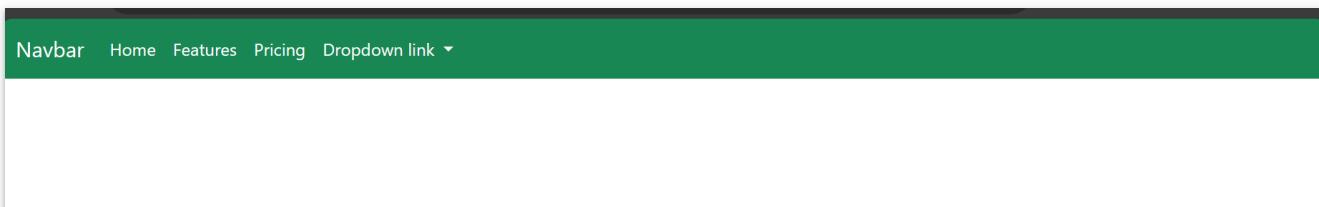
16. Hasilnya akan terlihat seperti di gambar ini:



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

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

18. Jika file html dijalankan, hasilnya adalah sebagai berikut



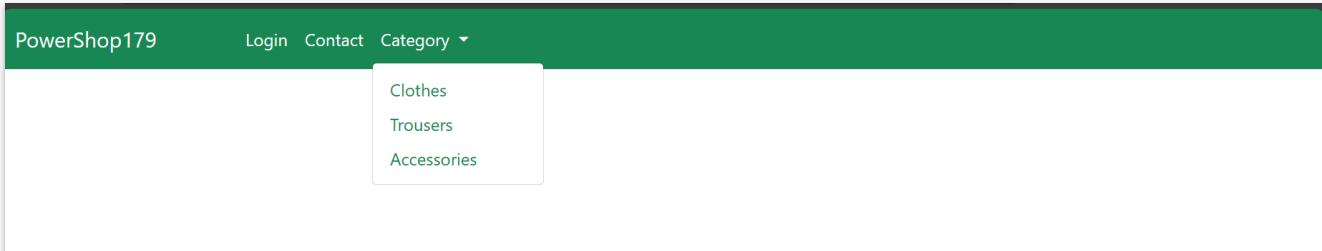
19. Sekarang kita akan mengganti text yang ada di navbar tersebut. Caranya sangat mudah, yaitu cukup kita ganti text-text tadi dengan kata-kata yang kita inginkan. Misalnya adalah sebagai berikut:

```

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

```

20. Maka hasilnya adalah sebagai berikut:



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

```

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

```

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

```

<nav class="navbar bg-body-tertiary">
  <div class="container-fluid">
    <form class="d-flex" role="search">

```

```

<input class="form-control me-2" type="search" placeholder="Search" aria-
label="Search">
    <button class="btn btn-outline-success" type="submit">Search</button>
</form>
</div>
</nav>

```

```

<!-- tes.html -->
<html>
    <body>
        <nav.navbar.bg-body-tertiary>
            25 |             <a class="nav-link text-white" href="#">Contact</a>
            26 |         </li>
            27 |         <li class="nav-item dropdown">
            28 |             <a class="nav-link dropdown-toggle text-white" href="#" role="button" data-bs-toggle="dropdown" aria-
            29 |                 Category
            30 |             </a>
            31 |             <ul class="dropdown-menu">
            32 |                 <li><a class="dropdown-item text-success" href="#">Clothes</a></li>
            33 |                 <li><a class="dropdown-item text-success" href="#">Trousers</a></li>
            34 |                 <li><a class="dropdown-item text-success" href="#">Accessories</a></li>
            35 |             </ul>
            36 |         </li>
            37 |     </ul>
            38 | </div>
            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.
- me-2** : Ini adalah kelas Bootstrap yang memberikan margin kanan sebesar 2 pada elemen yang memilikiinya. Dalam codingan ini, kelas ini diterapkan pada elemen input untuk memberikan jarak margin kanan.
- 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).

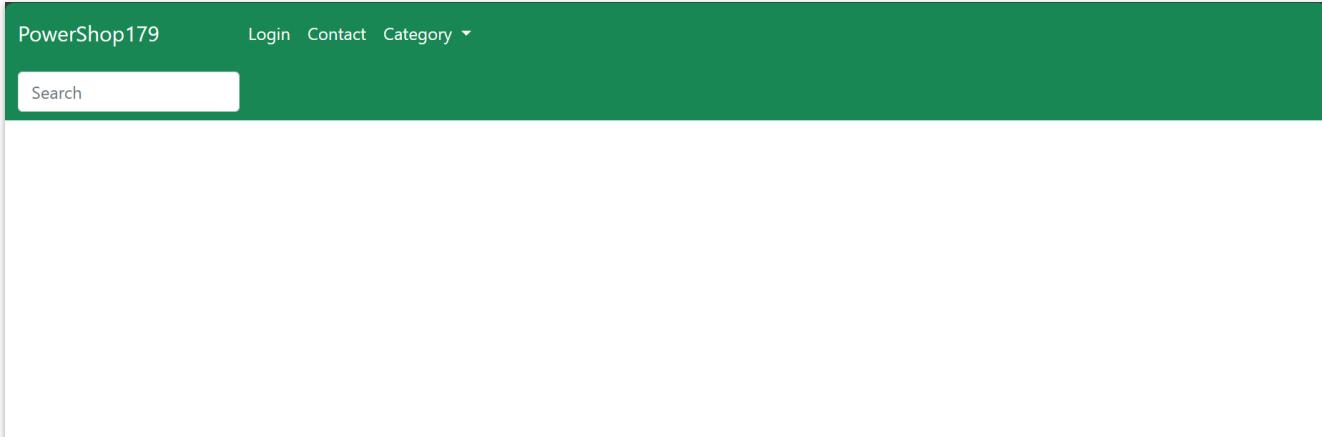
8. `type="submit"` pada tombol: Ini menentukan bahwa tombol tersebut bertindak sebagai tombol submit dalam formulir. Ketika ditekan, formulir akan dikirim.
9. Maka hasilnya adalah seperti berikut:



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>
```

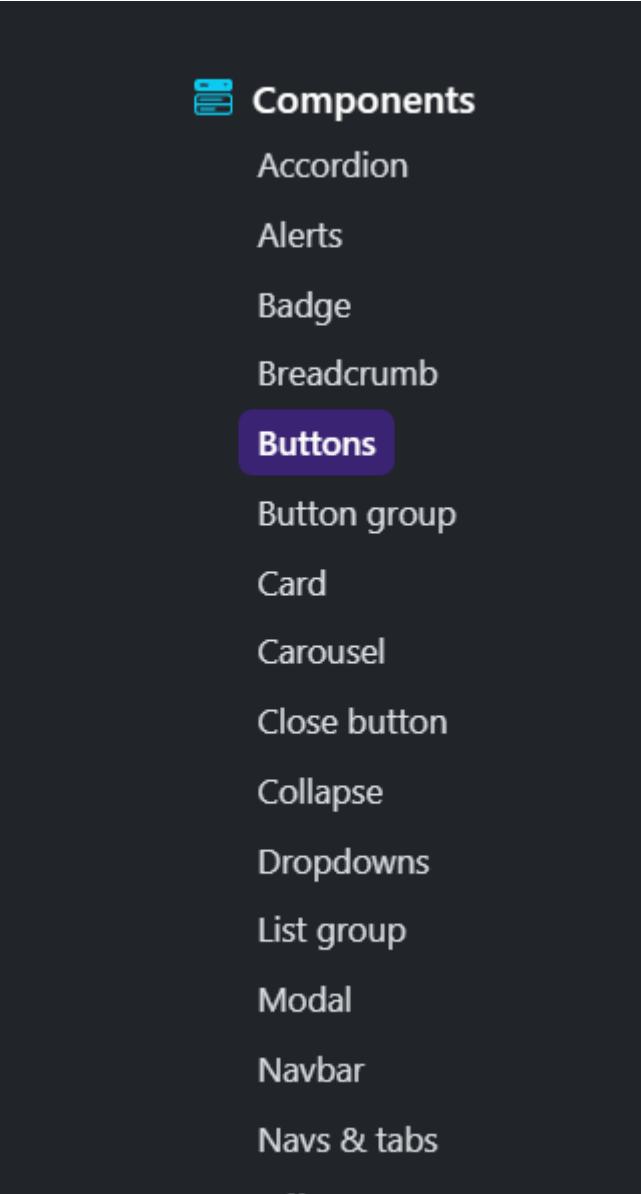
11. Maka hasil yang muncul akan seperti ini



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

[View on GitHub](#)

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



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

Made by me!

ads via Carbon

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](#)

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](#)

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

ads via Carbon

About

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

Example

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

2. Scroll 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>
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
</div>
```

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

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

```

    Featured

```

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



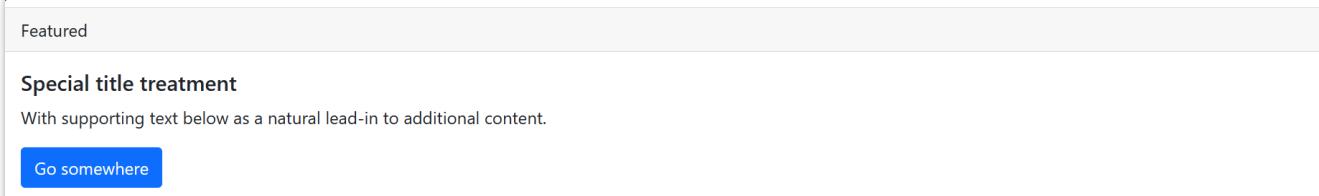
6. karena jarak atas antara card dengan batas website terlalu dekat maka pada class card tambahkan mt-2 .

```

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

```

7. Maka hasilnya adalah seperti ini:



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

```

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

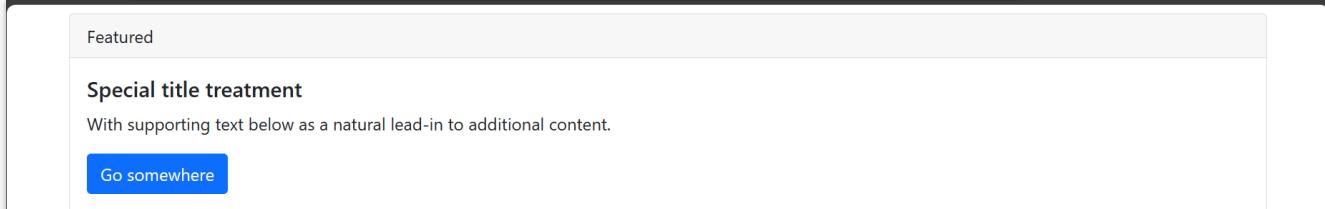
```

```

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

```

9. Berikut hasil dari di bungkusnya cards tadi.



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

```

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

```

```

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

```

Keterangan:

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.

- `Go somewhere / a>` : Ini adalah elemen anchor (tautan) yang merupakan tombol dengan kelas Bootstrap "btn" dan "btn-primary", memberikan tampilan dan warna tertentu sesuai dengan desain Bootstrap.

Perbedaan Container dengan Container-fluid:

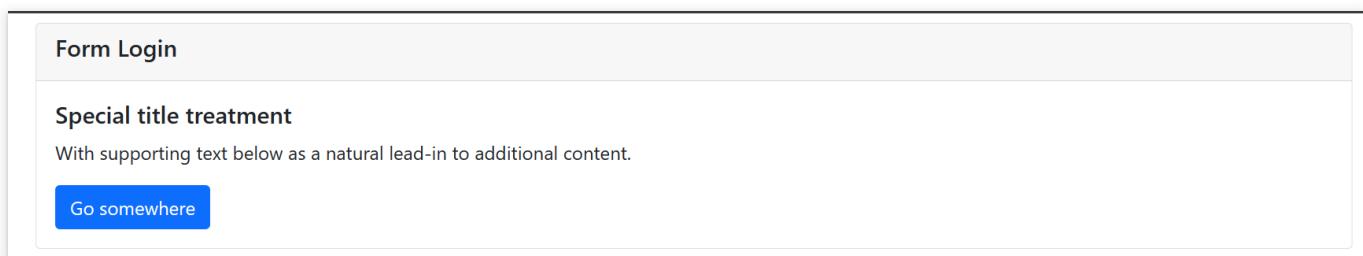
`container`:

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

`container-fluid`:

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

11. Hasilnya adalah sebagai berikut



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

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

13. Hasilnya akan seperti ini

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:

The screenshot shows a dark-themed form builder interface. At the top, there's a placeholder text "Email address" above a text input field. Below it is a note: "We'll never share your email with anyone else." Underneath is a "Password" label with a text input field. A checkbox labeled "Check me out" is present. At the bottom is a blue "Submit" button. To the right of the "Submit" button is a tooltip-like box containing the text "Copy to clipboard". Below the form template, the raw HTML code is displayed:

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

```

17. Maka hasilnya adalah seperti ini:

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 .colsm- untuk ukuran layar kecil, .col-md- untuk ukuran layar medium, dan sebagainya. Dengan cara ini, Kita dapat membuat tata letak yang responsif untuk berbagai perangkat.

6. NESTING

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

7. ORDER

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

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

Contoh Penggunaan dan Penjelasannya

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

1. Masuklah kembali ke web getbootstrap.com.

The screenshot shows the official Bootstrap website. At the top, there's a purple header bar with the Bootstrap logo, navigation links for 'Docs', 'Examples', 'Icons', 'Themes', and 'Blog', a search bar, and social media icons. A yellow button in the top right corner says 'New in v5.3.' followed by 'Color mode support, expanded color palette, and more!'. Below the header is a large purple banner featuring a white stylized 'B' icon inside a bracket shape. The main title 'Build fast, responsive sites with Bootstrap' is displayed in large white font. A subtitle below it reads: 'Powerful, extensible, and feature-packed frontend toolkit. Build and customize with Sass, utilize prebuilt grid system and components, and bring projects to life with powerful JavaScript plugins.' There are two buttons at the bottom of the banner: '\$ npm i bootstrap@5.3.2' and a purple 'Read the docs' button. Below the banner, the text 'Currently v5.3.2 · Download · All releases' is visible.

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

The screenshot shows the 'Grid system' documentation page from the Bootstrap website. On the left, a dark sidebar menu lists various Bootstrap components: Sass, Options, Color, Color modes, Components, CSS variables, Optimize, Layout (with 'Breakpoints', 'Containers', and 'Grid' selected), Columns, Gutters, Utilities, Z-index, CSS Grid, Content (with 'Reboot', 'Typography', 'Images', and 'Tables'), and a 'Grid' section. The main content area has a dark background with a light-colored grid system example. The title 'Grid system' is in large white font. Below it, a subtext explains: 'Use our powerful mobile-first flexbox grid to build layouts of all shapes and sizes thanks to a twelve column system, six default responsive tiers, Sass variables and mixins, and dozens of predefined classes.' An advertisement for Adobe Stock is visible. To the right, a sidebar titled 'On this page' lists related topics like 'Example', 'How it works', 'Grid options', 'Auto-layout columns', etc. The overall layout is clean and modern, using a dark theme with white and light gray text.

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

Row columns

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

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

Column	Column
Column	Column

HTML



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

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

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

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

```

<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">Search</button>
    </form>
  </div>
</nav>

<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>
</body>
</html>

```

6. Maka hasilnya akan seperti berikut:



Keterangan:

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

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

```

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

```

```

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

```

8. Hasil nya akan menjadi seperti berikut:

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

Keterangan:

Jadi Grid tadi mengatur Row dan Column dimana terdapat 2 baris berisi 2 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

PowerShop179

Search


HTML

HTML

LOREM IPSUM

Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.


CSS

CSS

LOREM IPSUM

Nobis totam assumenda unde quod et vitae minus beatae, quibusdam impedit laboriosam earum, possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.

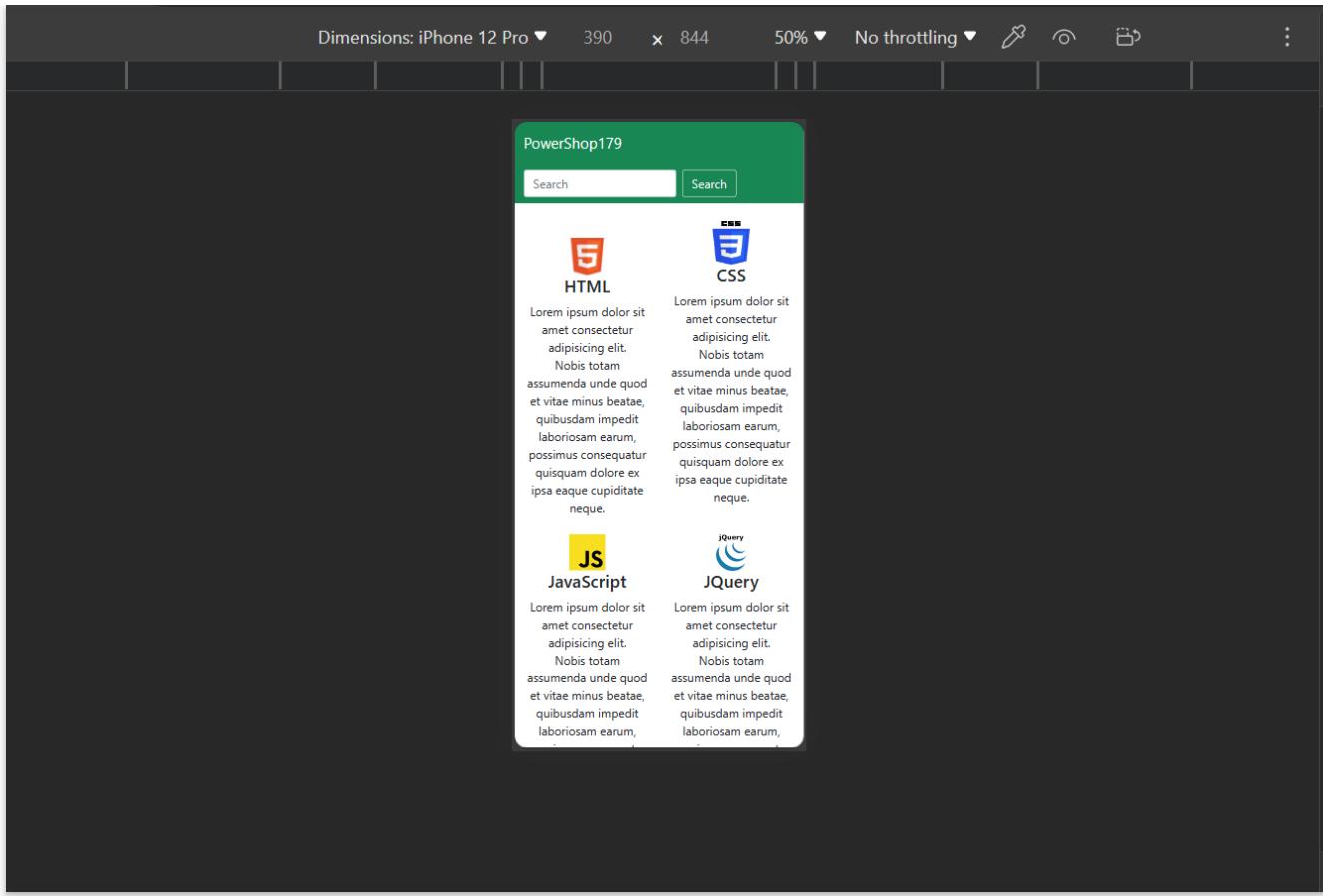

jQuery

JQuery

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 documentation homepage. A search bar at the top contains the query 'grid'. Below it, a sidebar on the left lists various documentation sections like Webpack, Parcel, Vite, Accessibility, RFS, RTL, Contribute, Customize, Layout, Breakpoints, Containers, Grid, Columns, and Grid Options. The main content area features a large heading 'Get started with Bootstrap' and a video thumbnail. A search result card for 'grid' is displayed, titled 'Example' with a purple header. It contains four items: 'How it works', 'Grid options', 'Auto-layout columns', and 'Responsive classes'. Below this is another card titled 'CSS Grid' with 'How it works' and 'Key differences'. At the bottom of the search results, there's a code snippet editor showing the HTML structure of a grid row.

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

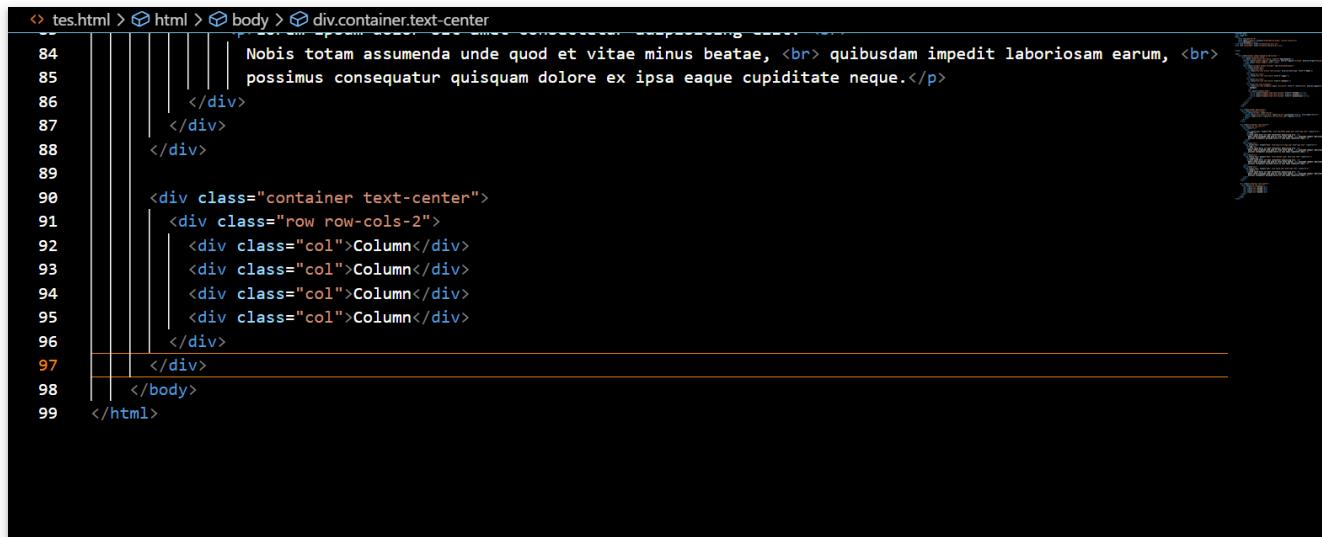
This screenshot shows the 'Row columns' documentation page. It features a visual representation of a grid with two rows and four columns, each labeled 'Column'. Below this is an 'HTML' section containing the following code:

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

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

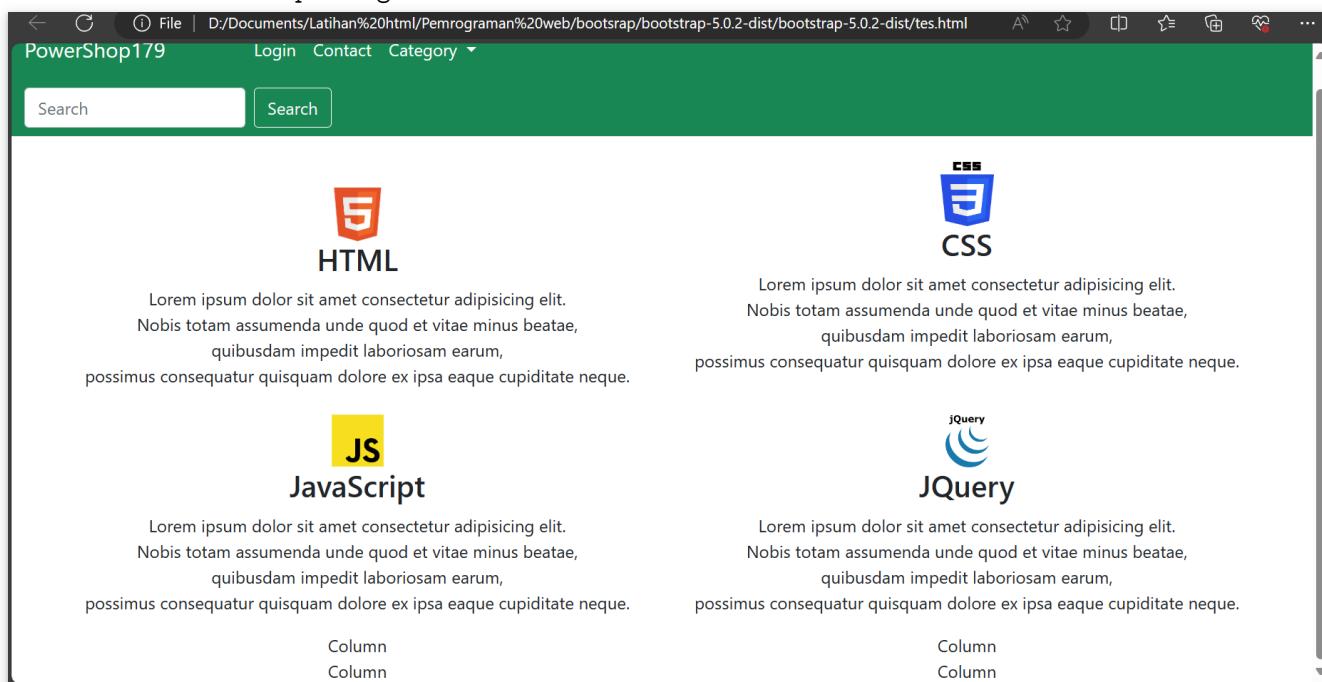
```
</div>  
</div>
```

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



```
84     |     |     | Nobis totam assumenda unde quod et vitae minus beatae, <br> quibusdam impedit laboriosam earum, <br>  
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>

```

PowerShop179 Login Contact Category ▾

Search Search



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

Lore 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

Lore 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.

Column
Column



JQuery

Lore 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.

Column
Column

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

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

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

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

Screenshot of a code editor showing the structure of a Bootstrap-based HTML file. The file includes sections for CSS, JavaScript, and jQuery, each with its own logo and placeholder text.

```

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

```

8. Maka hasilnya akan seperti ini:



Keterangan:

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

Penggunaan Tema

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

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

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

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

Search

Search



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.



JavaScript

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.



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.

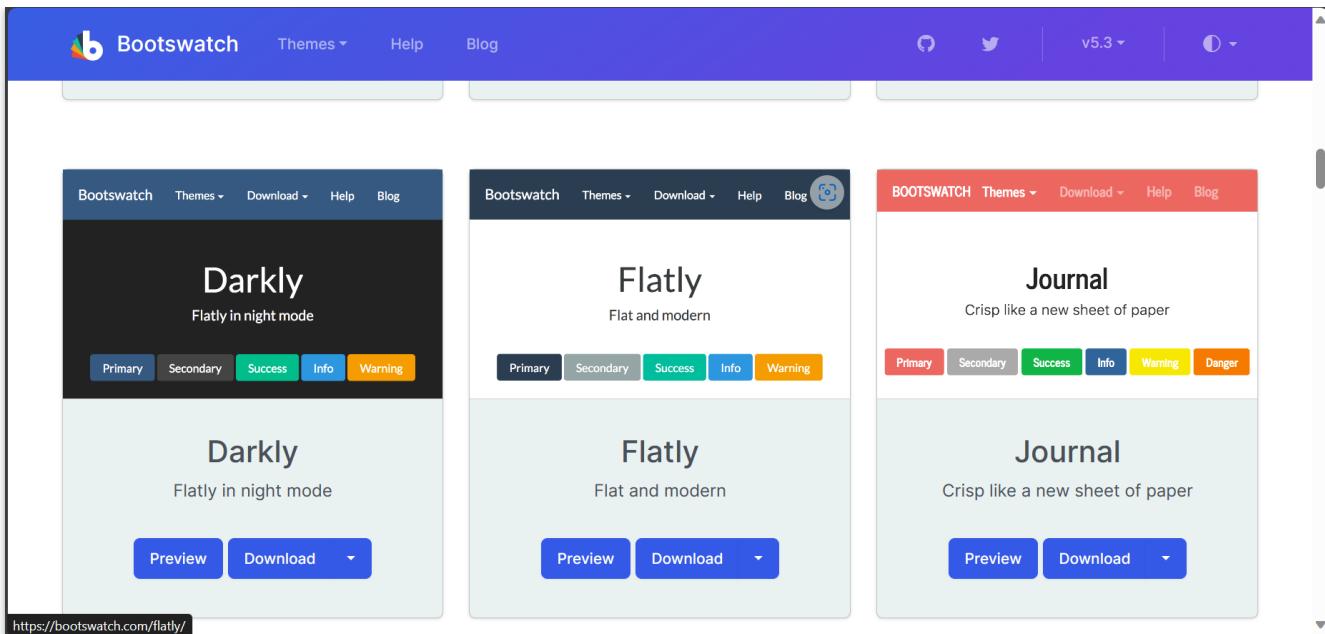


JQuery
 Lorem ipsum dolor sit amet consectetur adipisicing elit.
 Nobis totam assumenda unde quod et vitae minus beatae,
 quibusdam impedit laboriosam earum,
 possimus consequatur quisquam dolore ex ipsa eaque cupiditate neque.

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

The screenshot shows the homepage of Bootswatch, a website that provides free Bootstrap themes. The header includes the Bootswatch logo, navigation links for Themes, Help, and Blog, and social media icons for GitHub, Twitter, and LinkedIn. A prominent title "Free themes for Bootstrap" is displayed in large white font. Below it, there are two buttons: "Fork 3,723" and "Star 14,308". A small advertisement for Adobe Creative Cloud is visible at the bottom.

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.

This screenshot shows the same Bootswatch website as above, but with a context menu open over the 'Darkly' theme card. The menu is titled 'Downloads' and lists several files:

- # bootstrap.min.css (highlighted with a blue border)
- # bootstrap.min.css (Removed)
- # bootstrap.min.css (Removed)
- # bootstrap.min.css (Removed)
- # bootstrap.min.css (Removed)

 A 'See more' link is also visible at the bottom of the menu. The rest of the page content remains the same, showing the 'Darkly', 'Flatly', and 'Journal' theme cards.

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

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

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

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

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

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

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

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

Modifikasi Template Bootstrap

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

Template dalam Bootstrap adalah struktur dasar atau kerangka kerja yang telah dirancang sebelumnya dan dapat digunakan sebagai dasar untuk membangun halaman web. Bootstrap adalah sebuah framework

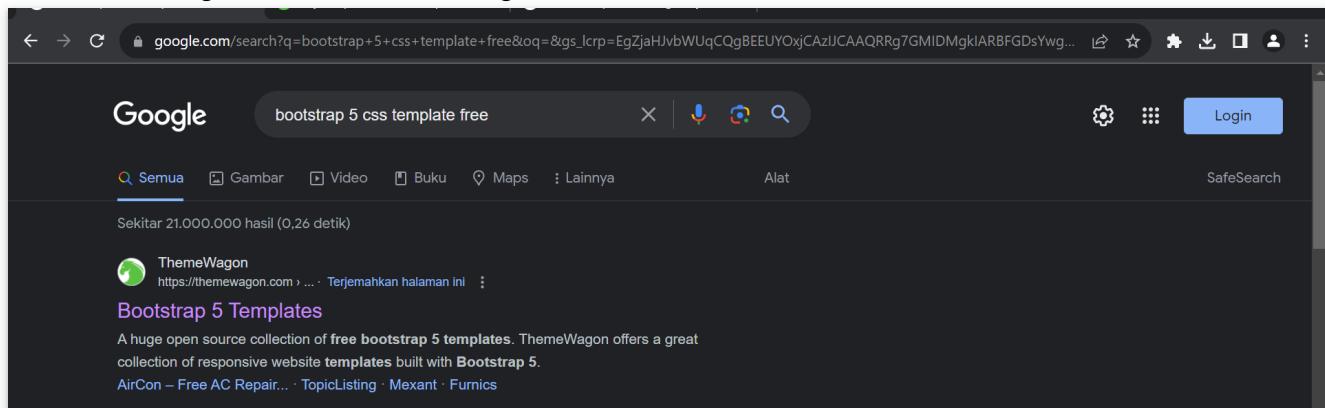
front-end yang populer digunakan untuk pengembangan web responsif. Dalam konteks Bootstrap, template menyediakan struktur HTML dan gaya CSS dasar yang dapat digunakan sebagai dasar untuk membangun halaman web yang responsif dan mudah diatur.

Dengan menggunakan template Bootstrap, kita dapat menghemat waktu dan usaha dalam merancang tata letak dan gaya dasar dari suatu proyek web. Template ini biasanya sudah mencakup elemen-elemen umum seperti navigasi, grid system, tombol, formulir, dan banyak lagi. Dengan cara ini, kita dapat fokus pada konten dan fitur unik dari situs web kita tanpa harus memulai dari awal.

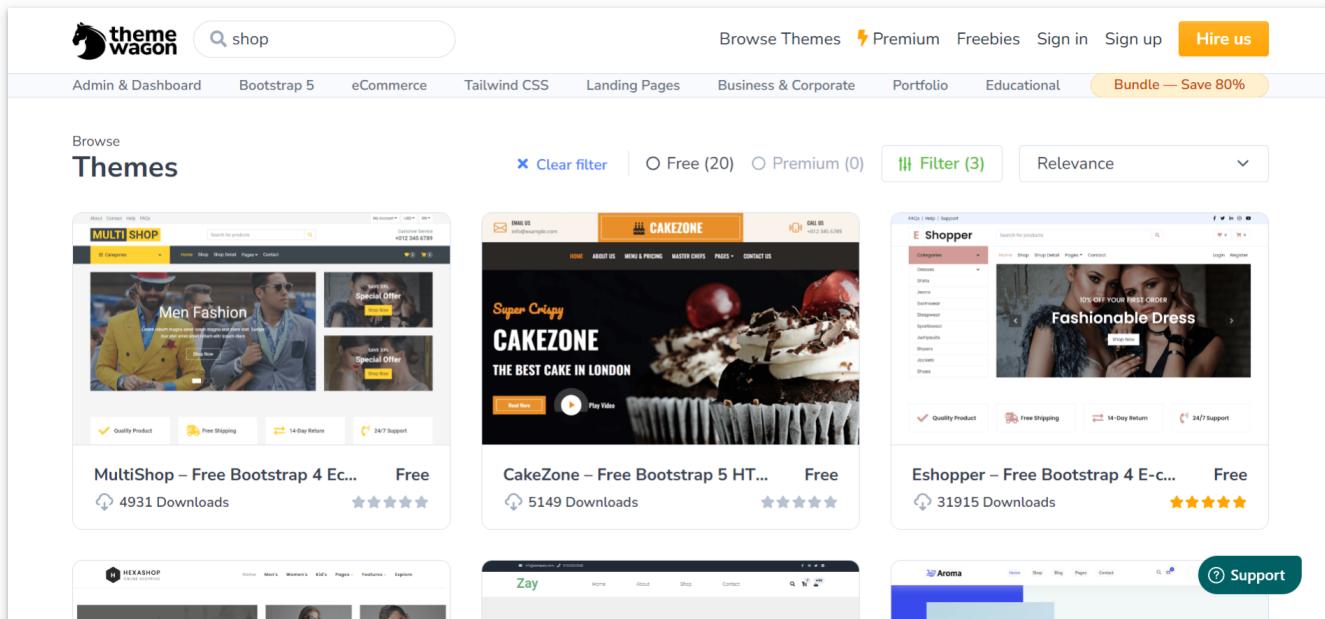
Template dalam Bootstrap juga mendukung konsep responsivitas, yang berarti halaman web atau aplikasi yang dibangun dengan menggunakan template ini dapat menyesuaikan diri dengan berbagai ukuran layar, mulai dari perangkat seluler hingga desktop, tanpa perlu penyesuaian tambahan. Ini membuat pengembangan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

Berikut cara mengkostumisasi template bootstrap:

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



2. Masuklah ke dalam website tersebut



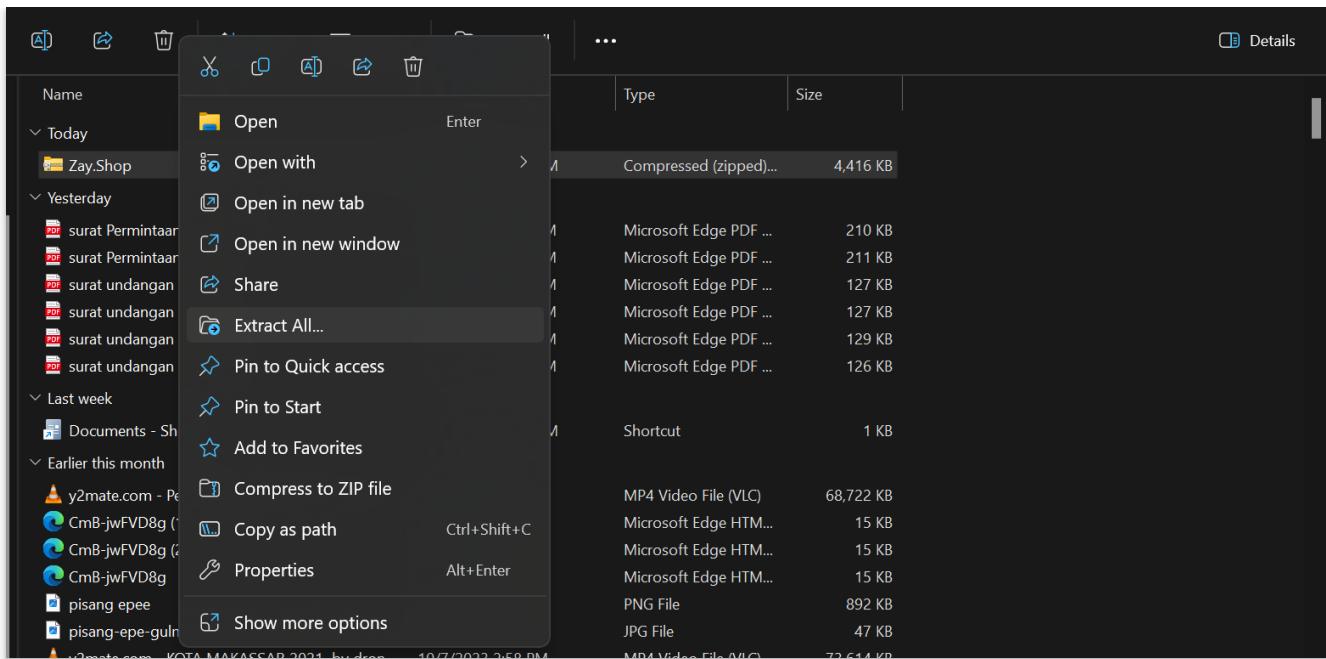
The screenshot shows the homepage of themewagon.com. At the top, there's a navigation bar with links like 'Admin & Dashboard', 'Bootstrap 5', 'eCommerce', 'Tailwind CSS', 'Landing Pages', 'Business & Corporate', 'Portfolio', 'Educational', and a prominent 'Bundle — Save 80%' button. Below the navigation, there's a section titled 'Browse Themes' with a search bar. A filter bar allows users to clear filters, choose between free or premium themes, and apply specific filters. The main content area displays four theme cards: 'MultiShop' (Free, 4931 Downloads), 'CakeZone' (Free, 5149 Downloads), 'Eshopper' (Free, 31915 Downloads), and 'Aroma' (Free). Each card includes a preview image, a download link, and a star rating.

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".



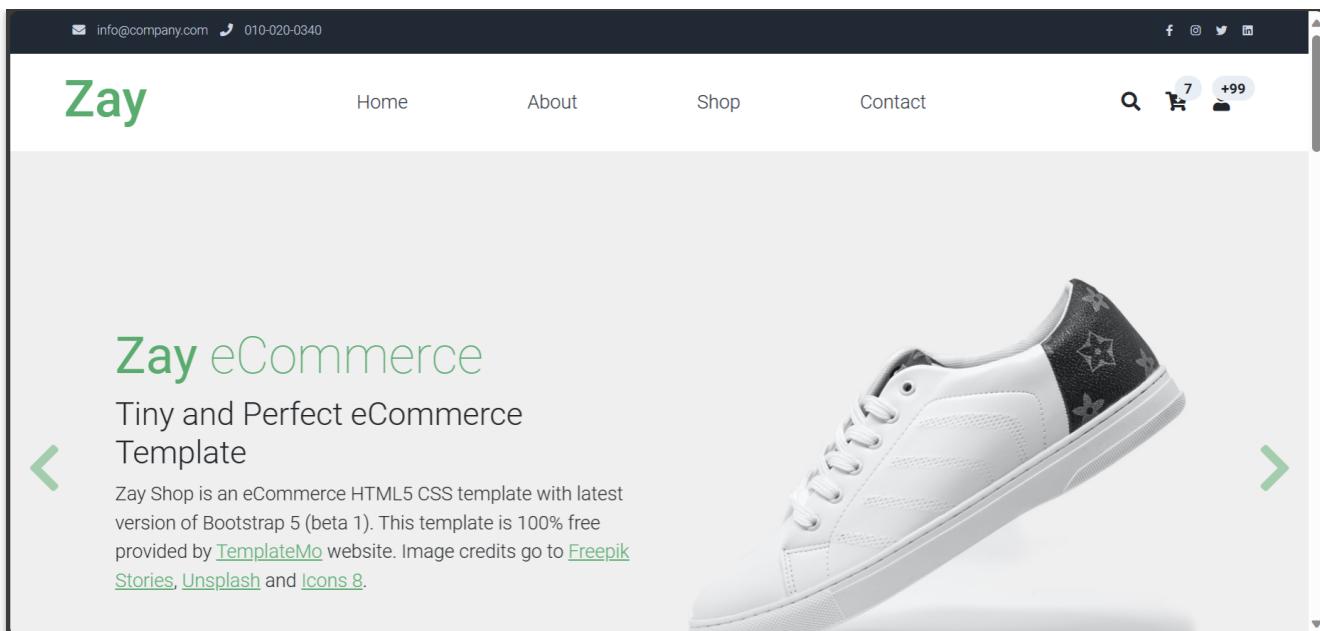
7. Jika proses ekstrak sudah selesai maka kita tela mendapatkan folder template tersebut dalam bentuk folder biasa



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

```

<!-- Start Top Nav -->
<nav class="navbar navbar-expand-lg bg-dark navbar-light d-none d-lg-block" id="templatemo_nav_top">
    <div class="container text-light">
        <div class="w-100 d-flex justify-content-between">
            <div>
                <i class="fa fa-envelope mx-2"></i>
                <a class="navbar-sm-brand text-light text-decoration-none" href="mailto:info@company.com">info@company.com</a>
                <i class="fa fa-phone mx-2"></i>
                <a class="navbar-sm-brand text-light text-decoration-none" href="tel:010-020-0340">010-020-0340</a>
            </div>
            <div>
                <a class="text-light" href="https://fb.com/templatemo" target="_blank" rel="sponsored"><i class="fab fa-facebook-f mx-2"></i></a>
                <a class="text-light" href="https://www.instagram.com/" target="_blank"><i class="fab fa-instagram mx-2"></i></a>
                <a class="text-light" href="https://twitter.com/" target="_blank"><i class="fab fa-twitter fa-sm mx-2"></i></a>
                <a class="text-light" href="https://www.linkedin.com/" target="_blank"><i class="fab fa-linkedin mx-2"></i></a>
            </div>
        </div>
    </div>
<!-- Close Top Nav -->

```

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

✉ info@company.com ☎ 010-020-0340

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

```

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

```

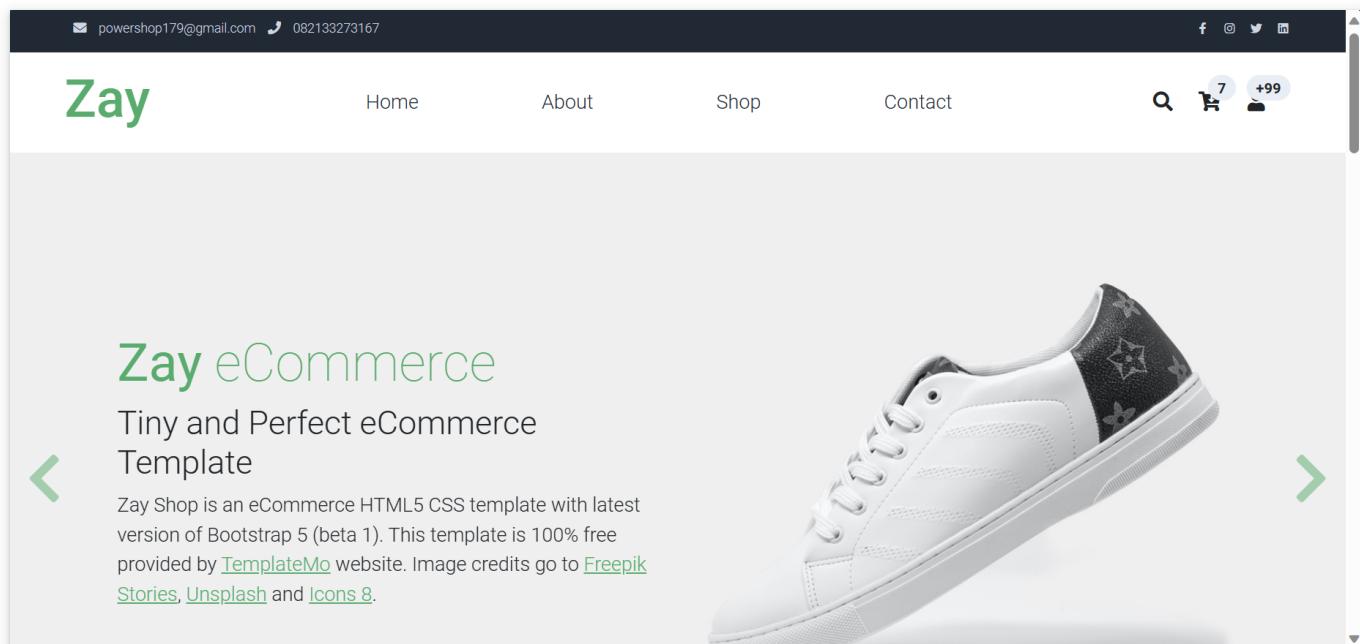
13. Gantilah kalimat dan tujuan email dan nomor telp itu dengan yang kita inginkan

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

Maka hasilnya akan terlihat seperti dibawah ini:

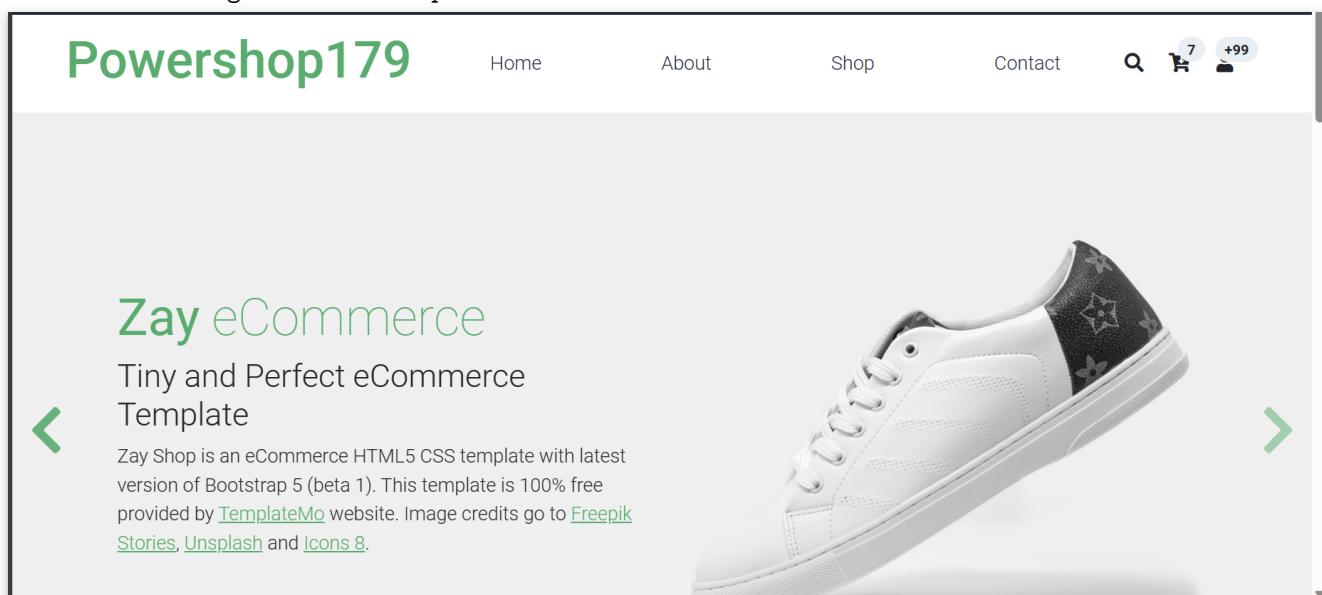


14. Mulailah mengganti elemen-elemen lainnya sesuai keinginan kita. Misalnya kita ingin juga mengganti logo yang ada di website ini



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

15. Hasil setelah diganti adalah seperti ini



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



Zay eCommerce
Tiny and Perfect eCommerce Template

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

< >

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

Hasilnya akan menjadi seperti ini:



Powershop179

Home About Shop Contact

< >

Powershow179
eCommerce

Tiny and Perfect eCommerce 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](#).

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







★★★★★ ★★★★★ \$240.00

Gym Weight
Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sunt in culpa qui officia deserunt.

Reviews (24)

★★★★★ ★★★★★ \$480.00

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

Reviews (48)

★★★★★ ★★★★★ \$360.00

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

Reviews (74)

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

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

Maka hasilnya akan menjadi seperti ini:







★★★★★ ★★★★★ \$240.00

Gym Weight
Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sunt in culpa qui officia deserunt.

Reviews (24)

★★★★★ ★★★★★ \$480.00

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

Reviews (48)

★★★★★ ★★★★★ \$360.00

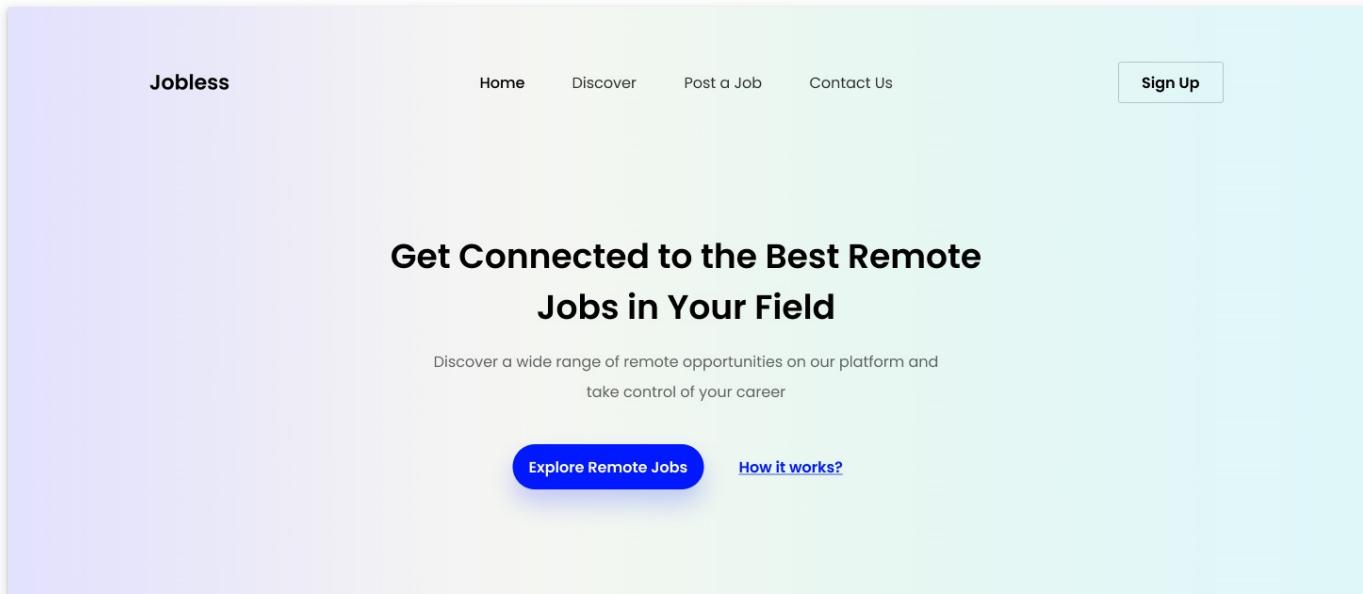
Summer Addides Shoes
Curabitur ac mi sit amet diam luctus porta. Phasellus pulvinar sagittis diam, et scelerisque ipsum lobortis nec.

Reviews (74)

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

Studi Kasus Pengaplikasian Bootstrap dalam Desain Web

Contoh Desain Web Yang Akan di Implementasikan:



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

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

Keterangan:

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

The screenshot shows the Bootstrap documentation homepage. A search bar at the top has 'Navbar' typed into it. Below the search bar, a sidebar on the left lists various Bootstrap components like Accordion, Alerts, Badge, Breadcrumb, Buttons, etc. The main content area features a large heading 'Get started with Bootstrap' and a section titled 'Quick start' with a video thumbnail. A modal window is open over the content, titled 'Navbar', containing a 'How it works' section with a sub-section 'Placement'. The right sidebar contains links to 'On this page' sections such as 'Quick start', 'CDN links', 'Next steps', and 'Community'.

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

The screenshot shows the Bootstrap documentation for the 'Navbar' component. The sidebar on the left has 'Navbar' selected. The main content area displays the 'HTML' representation of the navbar, which includes a brand logo, a toggler button, and a dropdown menu. To the right, there is a 'Search' bar and a 'Search by algolia' link. The right sidebar lists 'On this page' sections related to the Navbar component.

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

Keterangan:

1. **navbar** : Kelas ini menunjukkan bahwa elemen `<nav>` adalah bagian dari komponen navbar. Navbar adalah bagian dari Bootstrap yang menyediakan navigasi di bagian atas halaman web. Dengan

memberikan kelas navbar , kita memberitahu Bootstrap untuk menerapkan gaya dan tata letak khusus yang terkait dengan navigasi.

2. **bg-body-tertiary** : Kelas ini memberikan warna latar belakang pada navbar. Dalam hal ini, warna latar belakangnya diberi warna sesuai dengan warna "hijau" Warna latar belakang ini dapat disesuaikan sesuai kebutuhan desain.

3. **container-fluid** : Kelas ini diterapkan pada elemen `<div>` yang membungkus elemen-elemen dalam navbar. Kelas ini memberikan padding yang sesuai dan membuat elemen-elemen di dalamnya menjaga lebar penuh dari layar (menggunakan grid system Bootstrap). Dengan kata lain, kontennya akan merespons secara baik pada berbagai lebar perangkat.

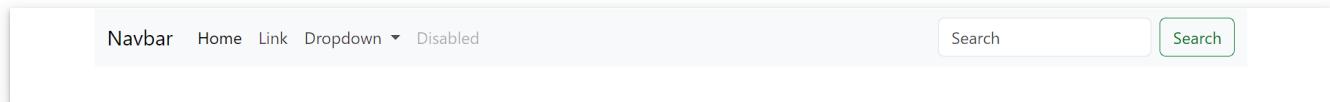
4. **d-flex** : Kelas ini mengubah elemen `<form>` menjadi kontainer flex. Flexbox adalah teknik tata letak yang kuat di CSS yang memudahkan pengaturan dan penataan elemen dalam satu atau dua dimensi. Dengan memberikan kelas d-flex , elemen form dan anak-anaknya dapat disusun secara fleksibel.

5. **me-2** : Ini adalah kelas Bootstrap yang memberikan margin kanan sebesar 2 pada elemen yang memilikiinya. Dalam codingan ini, kelas ini diterapkan pada elemen input untuk memberikan jarak margin kanan.

5. Tempelkanlah kode program itu didalam class **container** sebelumnya pada halaman html terkait.

Tentu saja dengan bootstrap yang sudah ditautkan, baik offline, maupun online.

6. Maka hasilnya akan seperti ini:



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

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

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

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

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

Keterangan:

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

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

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

Keterangan:

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

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

```

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

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

```

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

Jobless Home Discover Post a Job Contact Us Sign Up

sebelum

```

<div class="container">

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

```

sesudah

```

<div class="container">

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

</div>

```

13. Hasilnya akan seperti ini:

Jobless Home Discover Post a Job Contact Us Sign Up

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

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

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

```

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

```

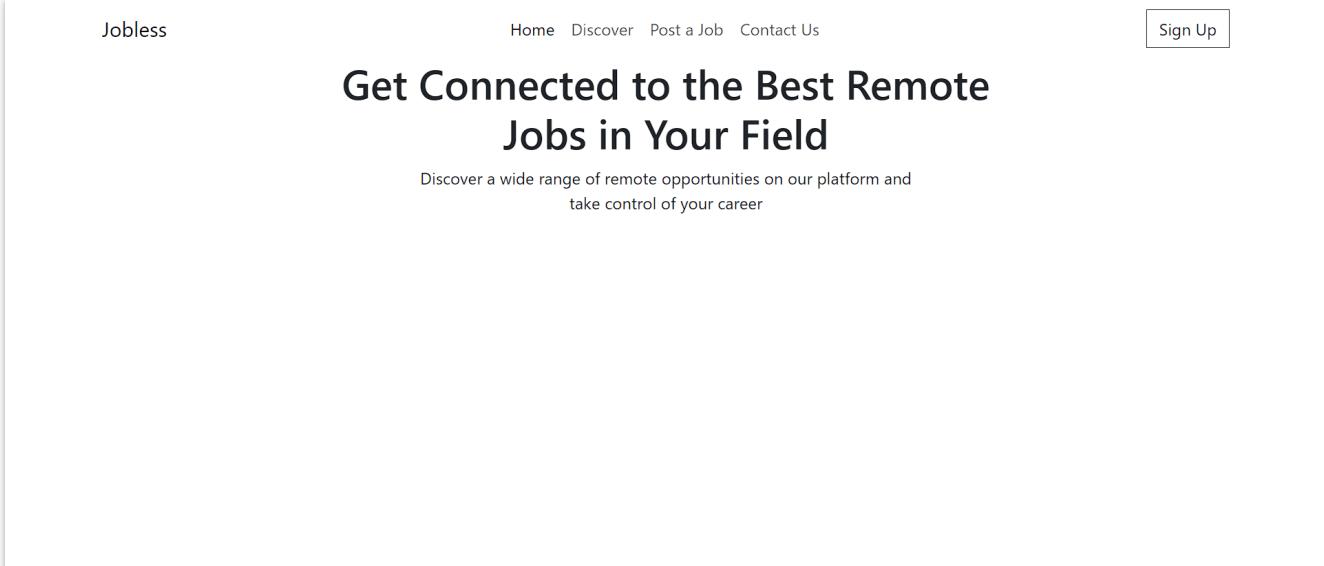
Keterangan:

1. `row` digunakan untuk mengatur tata letak konten dalam satu baris. Class ini menambahkan beberapa stil ke tata letak konten, seperti mengatur margin dan padding untuk konten yang berada di

sisi.

2. `h-100` adalah class Bootstrap yang digunakan untuk mengatur tinggi konten dalam sebuah halaman web. Class ini menambahkan stil yang memastikan konten berada pada tinggi penuh dari bagian atas ke bawah.d
3. `justify-content-center` adalah class Bootstrap yang digunakan untuk mengatur posisinya konten di tengah bagian horizontal.
4. class `align-items-center` adalah class Bootstrap yang digunakan untuk mengatur posisinya konten di tengah bagian vertikal.
5. class `text-center` adalah class Bootstrap yang digunakan untuk mengatur posisi teks di tengah bagian konten.
6. class `col-12` adalah class Bootstrap yang digunakan untuk mengatur lebar konten. Class ini menambahkan stil yang memastikan konten berada pada lebar penuh dari bagian atas ke bawah.

16. Hasilnya akan seperti ini:



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

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

```

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

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

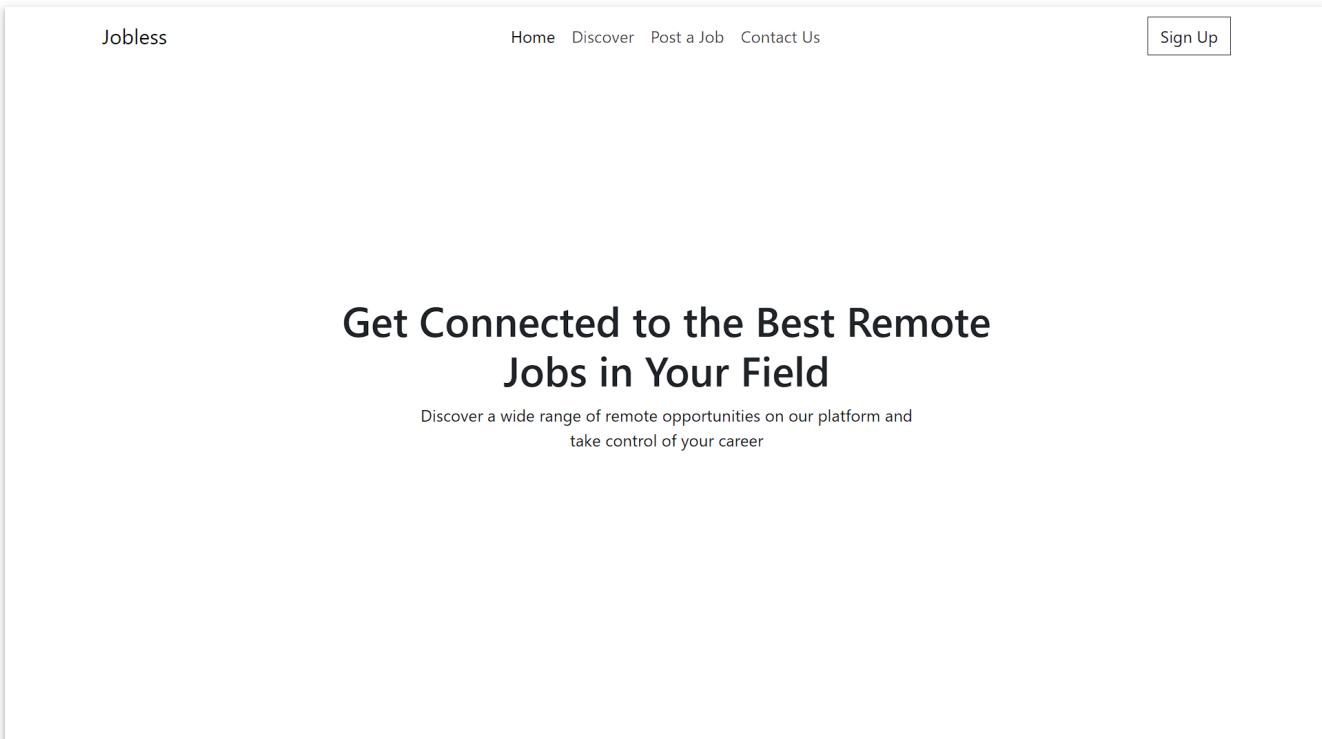
    <style>
        #hero {
            height: 85vh;
        }
    </style>
</head>

```

Keterangan:

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

18. Hasilnya akan seperti ini:



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

```

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

```

Keterangan:

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

The screenshot shows the homepage of a job search platform named 'Jobless'. At the top, there is a navigation bar with links for 'Home', 'Discover', 'Post a Job', and 'Contact Us'. To the right of these links is a 'Sign Up' button. Below the navigation bar, the main content area features a large, bold title: 'Get Connected to the Best Remote Jobs in Your Field'. Underneath the title, a subtitle reads: 'Discover a wide range of remote opportunities on our platform and take control of your career'. At the bottom of the main content area are two blue links: 'Explore Remote Jobs' and 'How it works?'. The overall design is clean and professional.

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

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

Keterangan:

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

Get Connected to the Best Remote Jobs in Your Field

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

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

21. Sekarang kita akan mengganti background body nya

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

```
<style>
  body {
    background-color: #f6fafe;
    background-image: linear-gradient(120deg, #0500FF 0%, #F8FB76 35%, #4EFF75 64%, #00FFD1 100%);
    background-blend-mode: overlay;
  }

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

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.

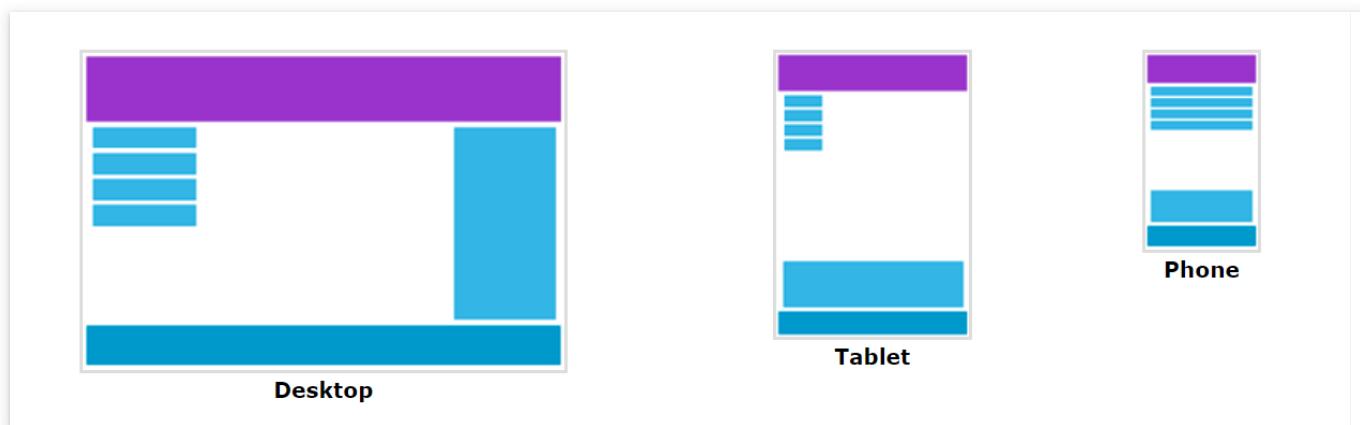
Uji pada Berbagai Browser:

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

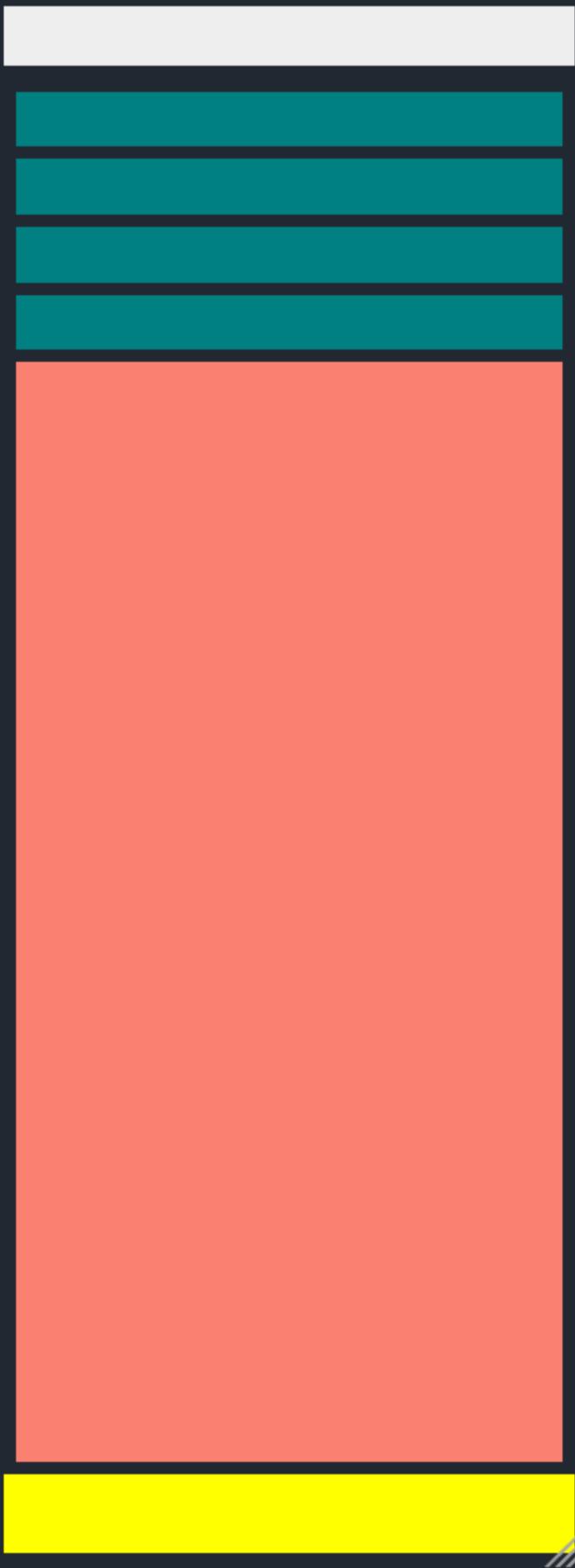
Saat mendesain website, mulailah dengan mempertimbangkan tampilan website pada perangkat mobile. Hal ini akan membantu Anda membuat website yang lebih ringkas dan mudah digunakan di semua perangkat.

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 # EEEEEEE.

```
.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.