

## **BAB III**

### **HTML dan CSS LANJUT**

#### **A. Tujuan**

- a. Peserta didik dapat menggunakan tag tabel dalam HTML
- b. Peserta didik dapat menggunakan tag div dalam HTML

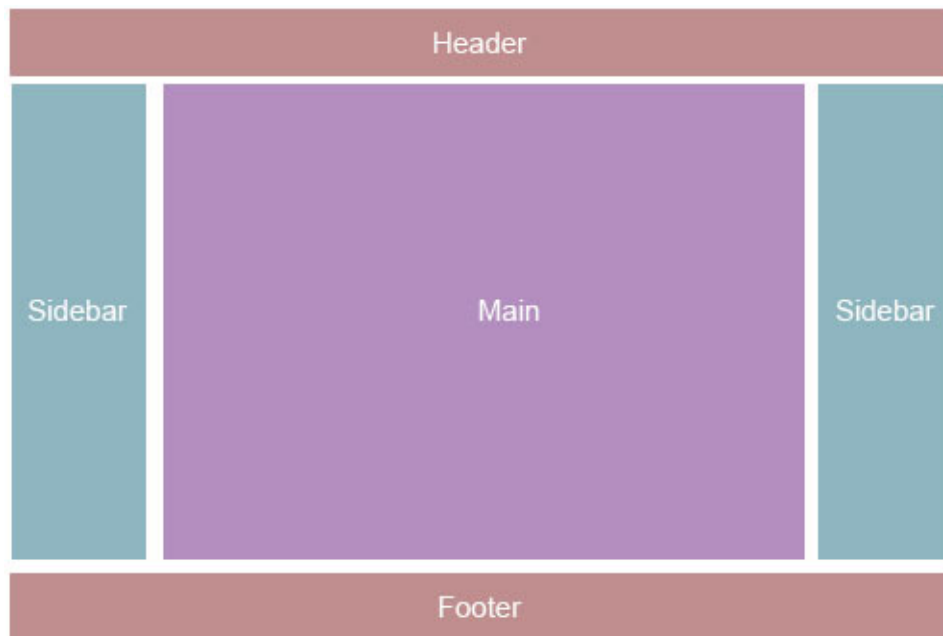
#### **B. Perlengkapan**

- a. Modul 3. HTML dan CSS Lanjut
- b. Software Text Editor atau IDE pemrograman WEB (Notepad++, Virtual Studio Code, Sublime, dsb)
- c. Komputer dengan sistem operasi Windows

#### **C. Materi**

##### **1) Tag DIV**

Dalam sebuah halaman web, terdapat section atau bagian-bagian tempat kita meletakkan konten website kita. Terdapat section yang berada di bagian atas, tengah, samping maupun di bagian bawah. Apabila divisualisasikan akan terlihat sebagai berikut:

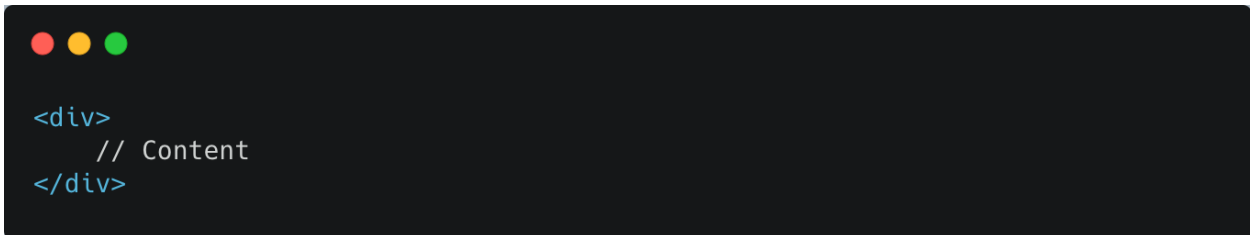


Gambar 3.1. Pembagian Section Umum Halaman Web

Pada Gambar 3.1 di atas, kita mengetahui jika secara umum bagian-bagian pada halaman web memiliki nama-nama yang sering digunakan. Header untuk bagian atas atau kepala dari halaman web, biasanya bagian ini berisi identitas web seperti logo web tersebut ataupun berisi menu-menu yang ada dalam web tersebut. Selanjutnya terdapat sidebar, sidebar dalam halaman web diletakkan di bagian samping dari halaman kita. Bisa di samping kanan maupun kiri, tergantung dari orientasi halaman web yang ingin kita buat. Kita juga bisa menggunakan dua sidebar sekaligus pada halaman web kita, seperti yang biasa digunakan pada website social media yang kita ketahui.

Setelah itu terdapat bagian main content, bagian ini adalah bagian tempat kita meletakkan konten website kita. Bagian ini yang menjadi pusat perhatian dari user website ketika mengakses halaman website kita. Yang terakhir terdapat bagian footer, bagian ini berada di bagian bawah halaman website kita. Bagian footer biasanya menampilkan informasi identitas dari website kita, seperti copyright dan beberapa menu tambahan yang ada dalam website kita.

Untuk membuat bagian-bagian tersebut, dalam web development kita menggunakan tag `<div>`. Berikut merupakan format penulisan dalam baris kode HTML:

A dark-themed code editor window with three colored window control buttons (red, yellow, green) in the top-left corner. The code inside is written in a light blue/cyan monospace font. It shows an opening tag `<div>`, followed by a comment `// Content` on the next line, and then a closing tag `</div>`.

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

Gambar 3. 2. Format Penulisan Tag Div


Mari kita mulai untuk membuat halaman web kita, untuk membangun tampilan website kita, kita membutuhkan formula HTML dan CSS yang akan kita pelajari selanjutnya.

## 2) Membangun Layout Website

### a) Wrapper dan Container

Untuk mempermudah kita dalam membuat sebuah halaman web, kita membutuhkan sebuah kerangka. Penamaan kerangka dalam halaman web biasanya dinamakan dengan *wrapper* dan *container*. *Wrapper* merupakan pembungkus dari sebuah segment web, sedangkan *container* merupakan pembungkus dari konten website. Untuk mempelajari lebih lanjut, mari kita persiapkan apa saja keperluan kita untuk menyusun sebuah halaman website.

Pertama, buat folder baru dalam folder **jongkoding** yang telah kita buat pada bab sebelumnya. Setelah itu, buat folder **bab3** untuk menyimpan file proyek latihan kita pada bab ini. Di dalam folder tersebut buat dua file, yaitu file *index.html* dan *style.css*. Kemudian, tulis kode baris di bawah ini pada file *index.html*:



```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Wrapper dan Container</title>
  <link href="styles.css" rel="stylesheet" />
</head>
<body>
  <div class="wrapper">
    <div class="container">

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

Gambar 3.3. Baris Kode Index HTML

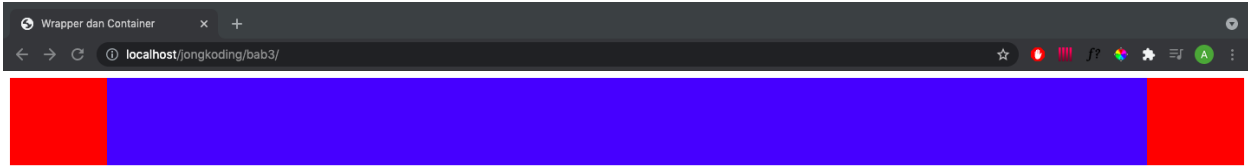
Setelah itu, tulis baris di bawah ini pada file *styles.css*:



```
.wrapper {
  background: red;
}
.container {
  width: 1200px;
  margin: auto;
  padding-right: 10px;
  padding-left: 10px;
  background: blue;
}
```

Gambar 3.4. Baris Kode Styles CSS

Setelah menuliskan dua file tersebut, maka berikut merupakan hasilnya:



Gambar 3.5. Hasil Wrapper dan Container

Berdasarkan Gambar 3.5. kita bisa mengamati bahwa *wrapper* berwarna merah dan *container* berwarna biru. *Wrapper* berfungsi sebagai pembungkus konten masing-masing segmen yang ada pada sebuah halaman web. Sedangkan *container* akan membungkus konten inti yang akan kita tampilkan pada halaman web kita. Kita sudah mengenal *wrapper* dan *container*, selanjutnya kita akan mulai membangun layout dari halaman web kita.

#### b) CSS Layout

Properti yang sering kita gunakan dalam membuat layout website kita pada umumnya adalah **float** dan **clear**. Pembahasan nya adalah sebagai berikut:

##### i) Float

Properti **float** digunakan untuk mengatur posisi dan format sebuah konten, misalnya gambar yang berada di samping kiri atau kanan dari text pada sebuah *container*. Properti **float** dapat kita gunakan untuk mengatur posisi sidebar dan konten pada halaman kita. Untuk mengaturnya mari kita ubah baris kode pada `index.html` dan `styles.css` di bawah ini:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <title>Wrapper dan Container</title>
  <link href="styles.css" rel="stylesheet" />
</head>
<body>
  <div class="wrapper">
    <div class="container">
      <div class="sidebar">

      </div>
      <div class="content">

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

```

Gambar 3.6. Baris Kode Index HTML

Selanjutnya untuk baris kode pada *styles.css* sebagai berikut:

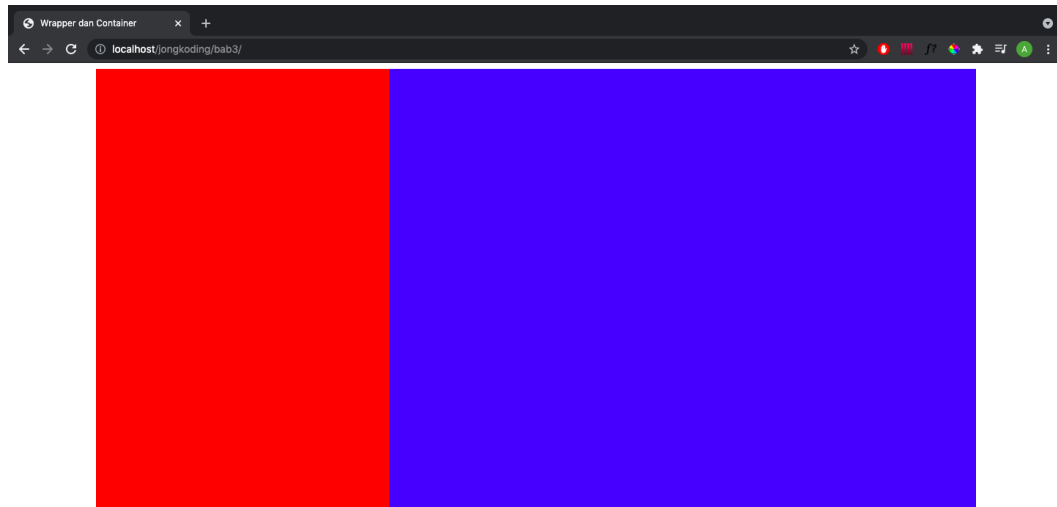
```

.wrapper {
  background: grey;
}
.container {
  width: 1200px;
  margin: auto;
}
.sidebar {
  width: 400px;
  height: 600px;
  background: red;
  float: left;
}
.content {
  width: 800px;
  height: 600px;
  background: blue;
  float: right;
}

```

Gambar 3.7. Baris Kode Styles CSS

Berikut merupakan hasil dari baris kode di atas:



Gambar 3.8. Hasil Properti Float

Apabila kita perhatikan pada baris kode dan tampilan yang dihasilkan. Kita mengubah *background wrapper* menjadi warna abu-abu, warna *content* kita hilangkan. Kemudian untuk *sidebar* kita atur berwarna merah, dan *content* kita atur berwarna biru. Terlihat bahwa *sidebar* berada di sebelah kiri dari tampilan kita, sebaliknya konten berada di sebelah kanan dari konten kita. Posisi layout tersebut diakibatkan oleh baris kode **float: left** pada *sidebar*, dan **float: right** pada *content*.

Apabila kita amati lebih lanjut terlihat bahwa kita telah mengatur *background wrapper* menjadi warna abu-abu. Akan tetapi pada hasil di atas, tidak terlihat warna *background* abu-abu dari *wrapper* kita. Mengapa demikian?

Hal ini diakibatkan ketika kita menggunakan **float**, kita harus menambahkan **clear** element setelah element **float**. Sehingga, kita harus menambahkan sebuah komponen **<div>** yang diberikan properti **clear** pada CSSnya. Ketika kita menggunakan **float** pada kiri halaman kita bisa mengatur **clear** pada bagian kiri saja, atau kita bisa mengaturnya untuk kedua sisi. Berikut merupakan contoh penggunaan *property clear* pada halaman kita, pertama tambahkan kode HTML di bawah ini pada file *index.html* kita:

```

...
<div class="container">
  <div class="sidebar">

  </div>
  <div class="content">

  </div>

  <div class="clear"></div>
</div>
...

```

Gambar 3.9. Penambahan Pada Elemen Container

Pada baris kode di atas, kita menambahkan tag `<div>` dengan class `clear` pada bagian bawah container. Kemudian, tambahkan baris kode di bawah ini pada file `styles.css`:

```

...
.clear {
  clear: both;
}

```

Gambar 3.10. Baris Kode Clear

Pada baris kode di atas, kita menambahkan CSS untuk `class clear`. Kita memberikan properti `clear` dengan nilai `both`. Maksud dari nilai `both` ini adalah `clear` akan di berikan pada kedua sisi halaman, atau kanan dan kiri.

Kita sudah mempelajari CSS layout dengan properti float dan clear. Selanjutnya kita dapat mulai mengatur tampilan apa yang kita inginkan pada website kita. Kita dapat memulainya dengan memilih warna latar belakang dan konten kita. Kalian dapat memilih warna pada palette warna yang ada di internet, salah satunya kalian dapat memilih pada [website ini](#).



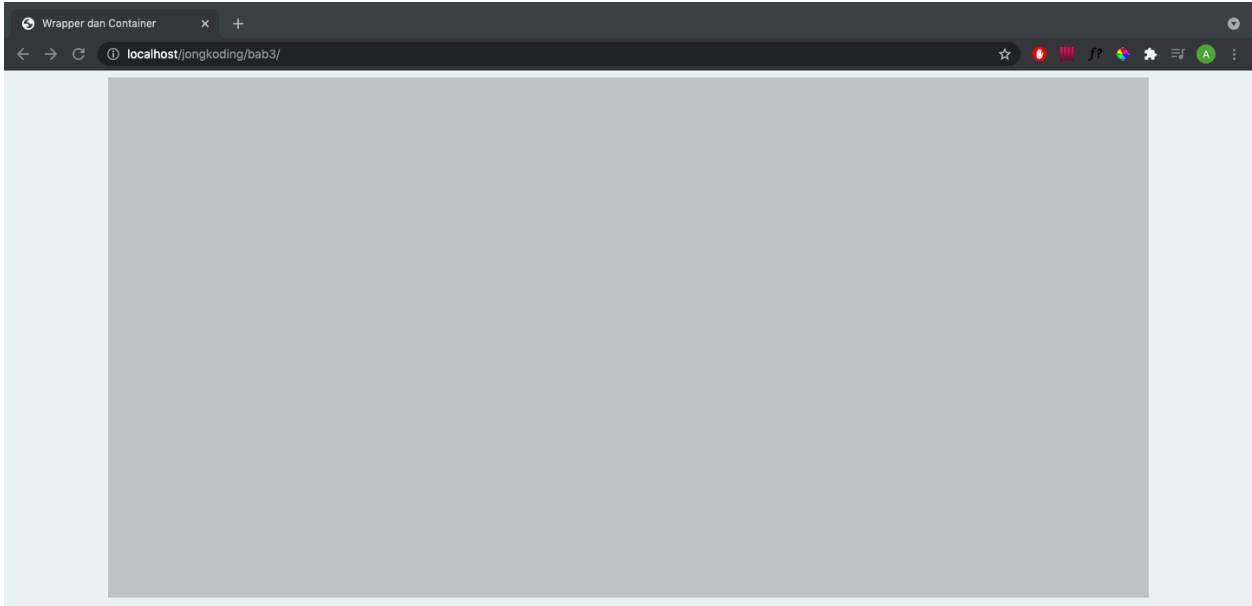
Setelah itu kalian dapat mengubah background body dan container kalian. Berikut merupakan contoh pengaturan pada *styles.css* untuk mengubah warna halaman website kita.

A code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code is written in a light-colored font and defines styles for a web page layout. It includes a `body` selector for the background color, a `.wrapper` selector, a `.container` selector for width, margin, and padding, a `.sidebar` selector for width, height, and float, a `.content` selector for width, height, and float, and a `.clear` selector for clearing floats.

```
body {  
    /* Background Halaman Web */  
    background-color: #ecf0f1;  
}  
  
.wrapper {  
}  
  
.container {  
    width: 1200px;  
    margin: auto;  
    padding-right: 10px;  
    padding-left: 10px;  
  
    /* Background Content */  
    background-color: #bdc3c7;  
}  
  
.sidebar {  
    width: 400px;  
    height: 600px;  
    float: left;  
}  
  
.content {  
    width: 800px;  
    height: 600px;  
    float: right;  
}  
  
.clear {  
    clear: both;  
}
```

Gambar 3.11. Contoh Baris Kode Style

Maka tampilan website kita menjadi sebagai berikut:



Gambar 3.12. Tampilan Hasil Perubahan Warna Website

Pada baris kode sebelumnya, kita mempelajari hal baru dimana untuk mengatur warna dalam CSS tidak harus menggunakan nama warna tersebut. Akan tetapi, kita juga dapat mengaturnya menggunakan kode warna dengan format *hexadecimal*.

#### c) Menu Layout

Menu dalam sebuah halaman web merupakan elemen yang sangat penting. Tanpa adanya menu maka user tidak bisa mengoperasikan website kita dengan optimal. Menu utama dalam sebuah website biasanya terdapat pada bagian atas dari website kita. Untuk membuatnya maka langkah pertama kita adalah menambahkan baris kode di bawah ini pada file *index.html*:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Wrapper dan Container</title>
  <link href="styles.css" rel="stylesheet" />
</head>
<body>
  <div id="header" class="wrapper">
    <div class="container">

      </div>
    </div>
    <div class="wrapper">
      <div class="container">
        <div class="sidebar">

          </div>
          <div class="content">

          </div>

          <div class="clear"></div>
        </div>
      </div>
    </body>
  </html>
```

Gambar 3.13. Penambahan Elemen Header

Pada baris kode di atas, kita menambahkan elemen header pada bagian atas dari halaman web kita. Header kita buat dengan class wrapper, dan di dalamnya kita beri elemen container seperti pada wrapper setelahnya. Setelah itu kita akan menambahkan identitas halaman website kita, kita akan memberi logo pada header kita. Tambahkan baris kode di bawah ini:

```
...  
<div id="logo">  
    
</div>  
...
```

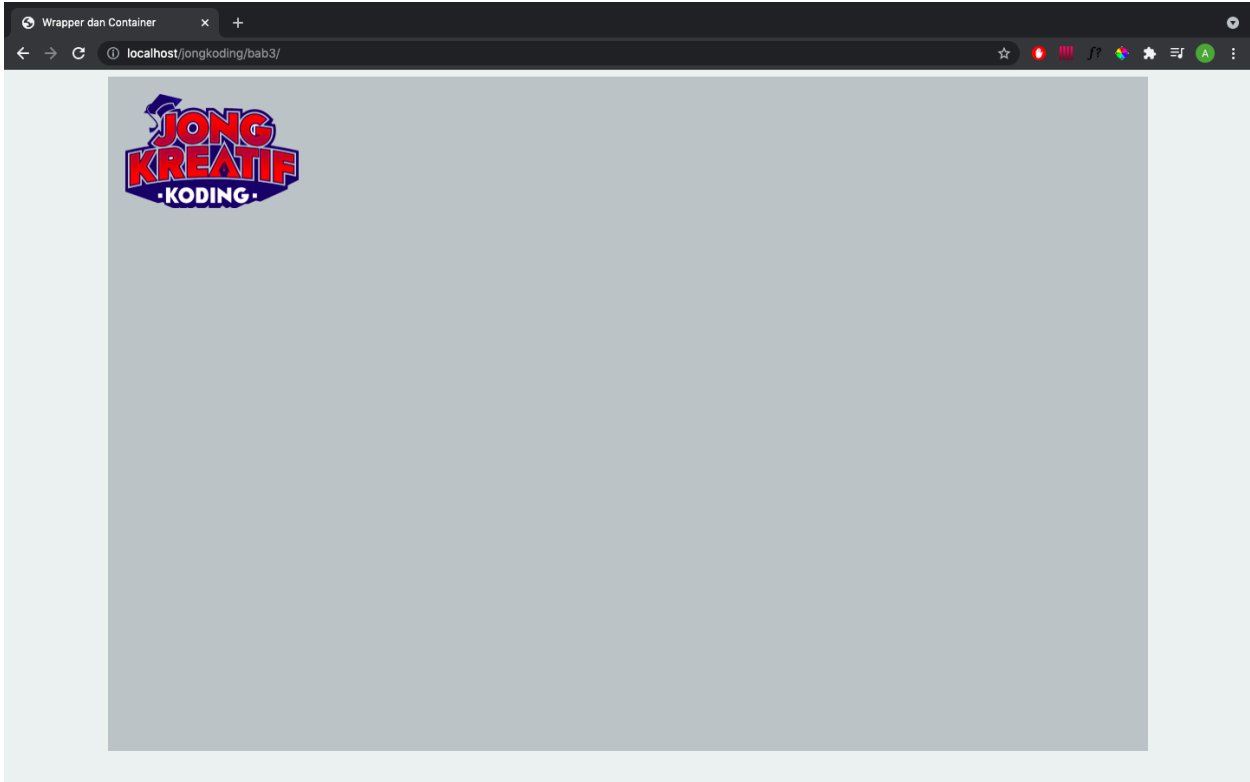
Gambar 3.14. Baris Kode Elemen Logo

Tambahkan baris kode di atas pada elemen container pada header kita. Kalian dapat menggunakan file logo kalian sendiri pada tag **<img>**. Lalu tambahkan baris kode di bawah ini pada file *style.css*:

```
#logo {  
  float: left;  
  width: 200px;  
  height: auto;  
  padding: 20px;  
}  
#logo img {  
  max-width: 100%;  
  height: auto;  
}
```

Gambar 3.15. Baris Kode File Style.css pada Header

Hasil dari baris kode di atas akan menjadi sebagai berikut:



Gambar 3.16. Hasil Tampilan Elemen Header

Selanjutnya, kita buat tampilan menu di sebelah kanan dari logo kita. Kita akan menggunakan tag `<ul>` dan `<li>` dalam menampilkan menu pada bagian kanan. Tulis baris kode di bawah ini setelah baris kode logo:

```
...
<div id="menu">
  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="tentang.html">Tentang</a></li>
    <li><a href="kontak.html">Kontak</a></li>
  </ul>
</div>
...
```

Gambar 3.17. Baris Kode Menu

Selanjutnya, tambahkan baris kode CSS berikut pada file *styles.css*:

```
...
#menu {
  padding-top: 60px;
  padding-bottom: 60px;
  float: right;
  width: 960px;
}
#menu li {
  float: left;
  list-style: none;
  margin-left: 10px;
  margin-right: 10px;
}
#menu li a {
  font-family: Verdana, Geneva, Tahoma, sans-serif;
  color: #34495e;
  font-size: 20px;
  font-weight: bold;
  text-decoration: none;
}
```

Gambar 3.18. Baris Kode CSS Menu

Berdasarkan baris kode di atas, terlihat bahwa menu kita atur berada pada sisi kanan header, sebelumnya kita juga telah mengatur elemen logo kita berada pada sisi kiri header kita. Kita juga memberikan pengaturan `float: left` pada elemen `<li>` kita, hal ini bertujuan untuk membuat elemen `<li>` agar berbaris menyamping, karena pada dasarnya komponen `<li>` akan berbaris ke bawah. Selain itu kita juga menghilangkan simbol yang dimiliki oleh elemen `<li>` dengan memberikan pengaturan `list-style: none`. Kita juga melakukan pengaturan pada link atau elemen `<a>` dengan mengganti jenis font, ukuran font, ketebalan font, warna font, dan menghilangkan garis bawah yang secara default dimiliki oleh tag `<a>`. Dikarenakan kita menggunakan pengaturan `float` pada halaman kita, maka jangan lupa menambahkan div dengan class `clear`, sehingga berikut merupakan baris kode pada elemen header kita:

```
...
<div id="header" class="wrapper">
  <div class="container">

    <div id="logo">
      
    </div>

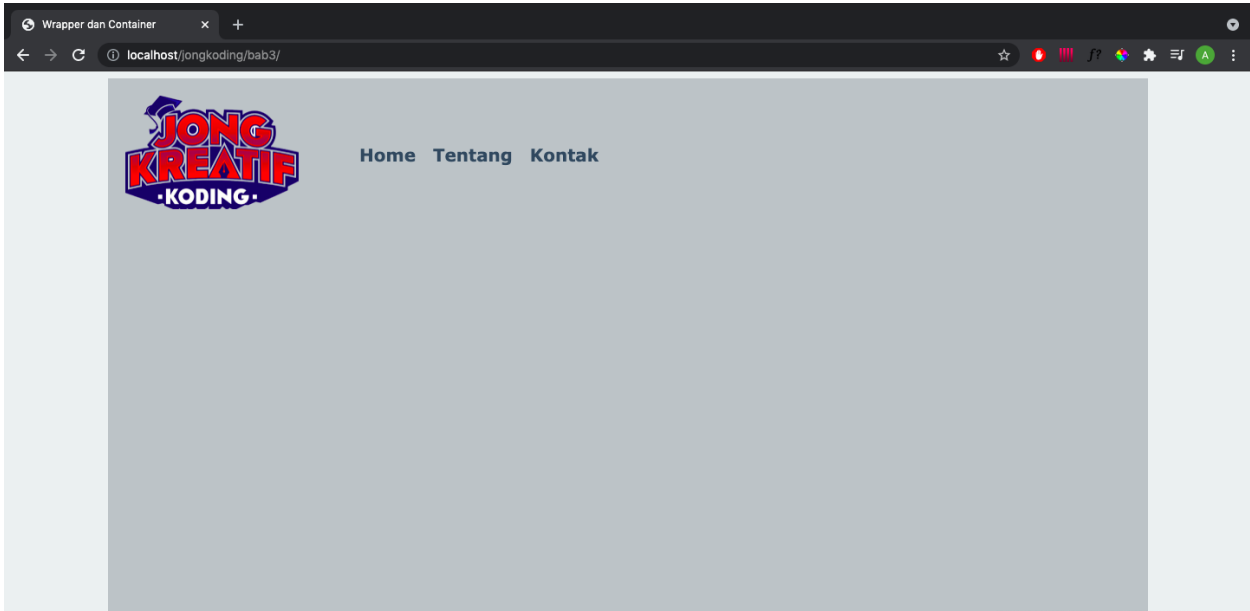
    <div id="menu">
      <ul>
        <li><a href="index.html">Home</a></li>
        <li><a href="tentang.html">Tentang</a></li>
        <li><a href="kontak.html">Kontak</a></li>
      </ul>
    </div>

    <div class="clear"></div>

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

Gambar 3.19. Baris Kode pada Elemen Header

Apabila semua kode telah kalian tulis dengan benar, maka hasilnya akan menjadi sebagai berikut:



Gambar 3.20. Hasil Tampilan Menu Layout

#### d) Konten

Terdapat beberapa macam tipe konten dalam sebuah halaman web. Konten dalam sebuah web dibuat menyesuaikan informasi apa yang ingin disampaikan dalam halaman web tersebut. Misalnya:

- *Blog* atau *Magazine*, apabila kita ingin menyampaikan informasi berita terkini dalam sebuah halaman web, maka kita menggunakan konten web dengan style blog, magazine, atau newspaper.
- *eCommerce* atau *Catalog*, ketika kita ingin membuat web yang bertujuan untuk berjualan, maka *style* konten kita menggunakan style *catalog* barang.
- *Profile*, ketika kita ingin membuat sebuah halaman untuk badan usaha, perusahaan, atau semacamnya. Maka biasanya kita menggunakan style profile, biasanya juga disebut style *corporate* atau *company profile*.
- *Admin Dashboard*, website tidak selamanya dibuat untuk akses umum, ada juga website yang dibuat untuk akses terbatas. Biasanya, website jenis ini menggunakan pembatasan akses pengguna, sehingga pengguna harus login terlebih dahulu. Website ini biasanya digunakan untuk keperluan-keperluan tertentu seperti panel monitoring transaksi, kegiatan, interaksi pengunjung dan *content management system* (CMS).



- Lain-lain, banyak lagi jenis tampilan website yang digunakan menyesuaikan kebutuhan website tersebut

Sebagai contoh, kita akan membuat tampilan website dengan *style blog* atau *magazine*. Untuk memulainya, kita tambahkan baris kode di bawah ini pada file `index.html` yang telah kita buat.

```


<div class="image-box">
        
    </div>
    <div class="text-box">
        <h2>Bersepeda Di Tengah Pandemi</h2>
        <p>
            Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
            do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad
            minim veniam
        </p>
        <a href="#">Read more</a>
    </div>
</div>
<div class="blog-content">
    <div class="image-box">
        
    </div>
    <div class="text-box">
        <h2>Solusi Work From Home Tetap Produktif</h2>
        <p>
            Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
            do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad
            minim veniam
        </p>
        <a href="#">Read more</a>
    </div>
</div>
</div>


```

Gambar 3.21. Baris Kode Blok HTML

Tulis baris kode di atas dalam tag <div> yang memiliki class content. Setelah itu ubah dan tambahkan baris kode di bawah ini pada file styles.css kita:

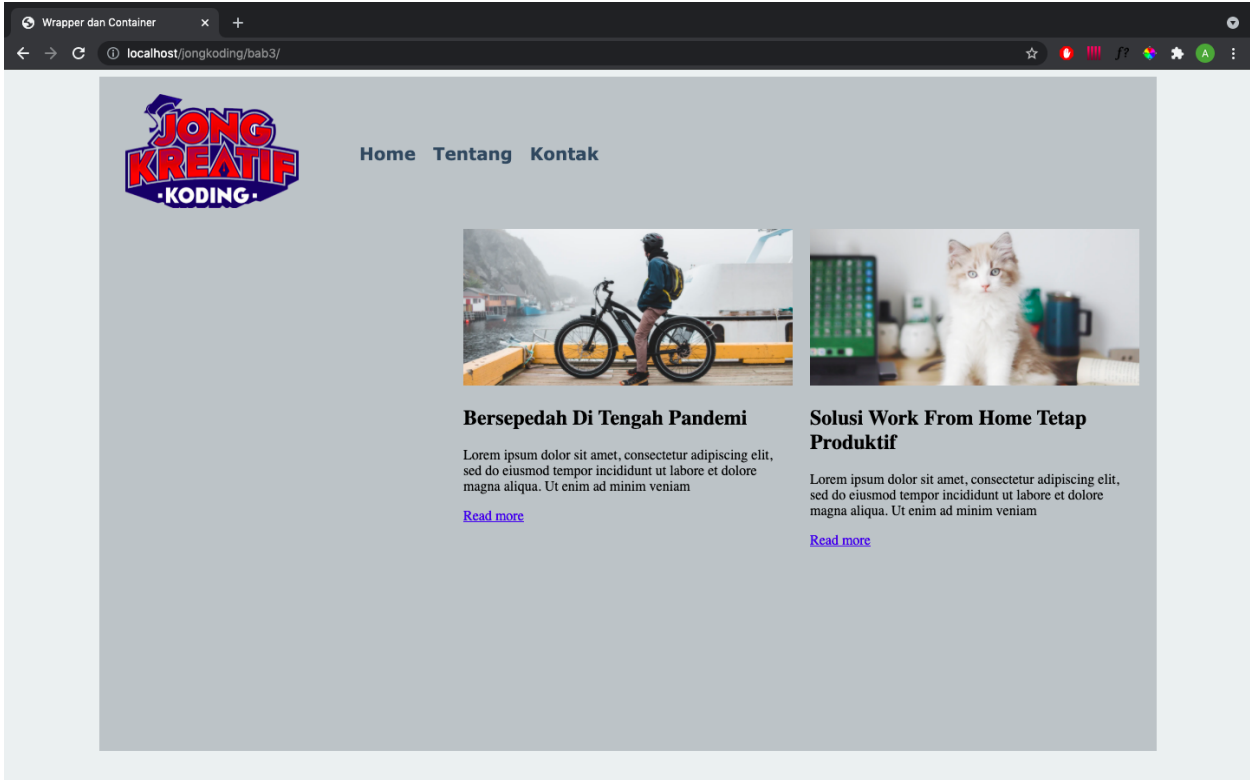
```
...  
.sidebar {  
  width: 400px;  
  float: left;  
}  
.content {  
  width: 800px;  
  float: right;  
}  
...
```

Gambar 3.22. Baris Kode CSS Sidebar dan Content

```
...  
.blog-content {  
  float: left;  
  max-width: 380px;  
  margin-right: 10px;  
  margin-left: 10px;  
  margin-bottom: 20px;  
}  
.blog-content img {  
  width: 100%;  
  height: 180px;  
  object-fit: cover;  
}
```

Gambar 3.23. Baris Kode Blog Content

Pada Gambar 3.22. kita mengubah properti CSS pada class sidebar dan content. Kita menghilangkan properti *height* pada kedua *class* tersebut. Selanjutnya pada Gambar 3.23. Kita mengatur properti untuk *class* **blog-content** dan tag **<img>** yang ada di dalamnya. Berikut merupakan hasil dari baris kode di atas:



Gambar 3.24. Tampilan Blog Content

Kalian dapat mengganti konten seperti yang kalian inginkan, kalian dapat menambah juga menjadi lebih dari 2 konten blog. Kemudian kita dapat menambahkan konten pada sidebar blog kita. Misalnya kita akan menampilkan daftar artikel terbaru pada blog kita, kita dapat menambahkan baris kode di bawah ini:

```

<div class="recent-article">

  <h2>Terbaru</h2>

  <ul>
    <li><a href="#"> Bersepeda Di Tengah Pandemi </a></li>
    <li><a href="#"> Solusi Work From Home Tetap Produktif </a></li>
  </ul>

</div>

```

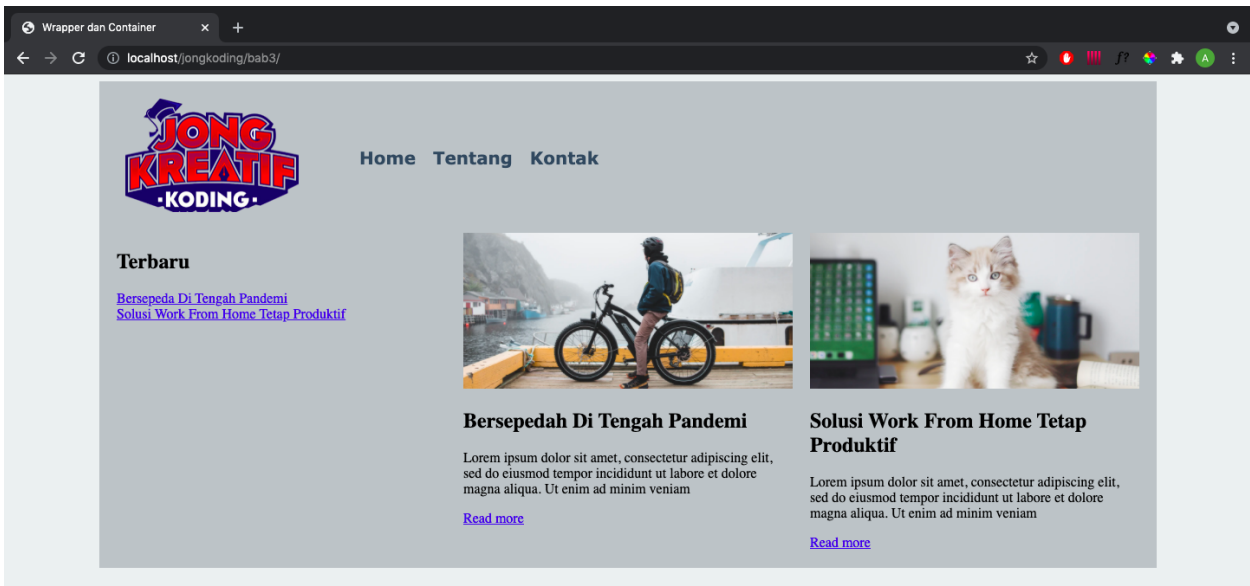
Gambar 3.25. Baris Kode Sidebar Artikel Terbaru

Tambahkan baris kode di atas pada tag `<div>` yang memiliki class `sidebar`. Setelah itu tambahkan baris kode CSS di bawah ini:

```
.recent-article {
    padding-left: 10px;
    padding-right: 10px;
}
.recent-article ul {
    padding: 0;
    margin: 0;
    list-style: none;
}
```

Gambar 3.26. Baris Kode CSS Recent Article

Berikut merupakan hasil dari baris kode di atas:



Gambar 3.27. Baris Kode Sidebar

## D. Tugas

- 1) Buatlah halaman untuk menampilkan tiap artikel yang ada pada halaman blog kalian, buat masing-masing satu halaman artikel. Lalu koneksikan link pada tag `<a>` ke masing-masing halaman yang telah kalian buat.

Petunjuk pembuatan halaman artikel:

- Buat file baru dengan nama judul artikel kalian, **tanpa spasi**, kalian dapat mengganti spasi dengan tanda *dash* (-).
- *Copy* dan *paste* semua isi file *index.html* yang telah kita buat, ke dalam file artikel kalian
- Hapus `<div>` dengan class `.blog-content` yang tidak ditampilkan
- Ubah class `.blog-content` pada tag `<div>` tersebut, ganti menjadi class **`blog-single`**
- Tambahkan baris kode CSS dengan cara mengcopy baris kode *style* pada class `.blog-content` beserta baris kode CSS untuk image di dalamnya. *Paste* baris kode tersebut dan ganti nama *class* menjadi **`.blog-single`**
- Kemudian, ganti *width* pada class **`.blog-single`** menjadi **`720px`**
- Kamu juga perlu menambahkan tinggi gambar yang ada di dalam tag `<div>` dengan class **`.blog-single`** menjadi **`400px`**
- Maka hasilnya akan menjadi sebagai berikut:

