

# **DIKTAT**

## **MATA KULIAH : PEMROGRAMAN WEB 1**

Dosen Pengampu :

**M. Dedy Rosyadi, S.Kom, M.Kom**  
**NIDN. 1131128902**



**FAKULTAS TEKNOLOGI INFORMASI**  
**UNIVERSITAS ISLAM KALIMANTAN (UNISKA)**  
**MUHAMMAD ARSYAD AL BANJARI**  
**BANJARMASIN**  
**2020**

## DAFTAR ISI

<b>DAFTAR ISI .....</b>	<b>1</b>
<b>DAFTAR GAMBAR.....</b>	<b>3</b>
<b>Pertemuan 1 Pendahuluan &amp; Pengenalan HTML.....</b>	<b>5</b>
1. Pengertian HTML .....	5
2. Sejarah HTML .....	6
3. Fungsi HTML .....	7
4. Apa yang baru pada HTML 5 ? .....	8
5. Kebutuhan Software .....	8
<b>Pertemuan 2 Struktur HTML 5 .....</b>	<b>10</b>
1. Struktur Dasar HTML 5 .....	10
2. Membuat File HTML Pertama.....	10
3. Pengertian Tag, Atribut dan Element.....	11
a. Tag.....	11
b. Atribut .....	11
c. Element .....	12
4. Menjalankan file HTML.....	12
5. Penjelasan Tag pada struktur HTML 5 .....	12
a. <!DOCTYPE html>.....	12
b. HTML.....	12
c. Head.....	13
d. Meta .....	13
e. Title.....	13
f. Body.....	13
6. Membuat Komentar .....	13
<b>Pertemuan 3 &amp; 4 Text Formating, Link, List, Audio, Video dan .....</b>	<b>15</b>
1. Heading.....	15
2. Paragraf.....	16
3. Anchor / Hyperlink .....	16
4. Daftar / List.....	17
a. Ordered List .....	17
b. Unordered List.....	18
c. Definition List.....	18
5. Image .....	18
6. Image dengan Link .....	19
7. Iframe .....	19
8. Audio.....	20
9. Video .....	21
10. Tag div, class dan id.....	22
<b>Pertemuan 5 HTML Table.....</b>	<b>24</b>
1. Struktur Dasar Table .....	24
2. Atribut Border .....	24
3. Atribut Cellspacing .....	25
4. Atribut Cellpadding.....	26
5. Atribut Colspan dan Rowspan .....	27
6. Thead, Tbody dan Tfoot .....	28
<b>Pertemuan 6 Form .....</b>	<b>30</b>
1. Form element.....	30

2. Form atribut .....	30
3. Label.....	31
4. Input Type .....	32
<b>Pertemuan 7 &amp; 8 CSS (cascading style sheet) .....</b>	<b>35</b>
1. Pengertian CSS .....	35
2. Fungsi CSS .....	36
3. Selector, Property dan Value.....	36
4. Jenis CSS .....	37
a. Inline CSS.....	37
b. Internal CSS .....	38
c. Eksternal CSS.....	39
<b>Pertemuan 9 Penerapan CSS sederhana pada HTML.....</b>	<b>41</b>
1. Membuat style pada anchor/link.....	41
<b>Pertemuan 10 &amp; 11 Framework Bootstrap .....</b>	<b>43</b>
1. Apa itu Bootstrap ? .....	43
2. Menggunakan Bootstrap .....	43
<b>Pertemuan 12, 13 &amp; 14 Membuat Website Statis Single Page.....</b>	<b>48</b>
<b>Daftar Pustaka .....</b>	<b>59</b>

## DAFTAR GAMBAR

Gambar 1. Tim Barnes Lee.....	6
Gambar 2. Tampilan file html di browser.....	10
Gambar 3. HTML heading.....	15
Gambar 4. Tag audio.....	21
Gambar 5. Tag video.....	21
Gambar 6. HTML table.....	24
Gambar 7. Atribut table border.....	25
Gambar 8. Atribut table cellpadding.....	26
Gambar 9. Tampilan hasil table cellpadding.....	26
Gambar 10. Hasil table colspan.....	27
Gambar 11. Hasil rowspan.....	28
Gambar 12. Hasil thead, tbody & tfoot.....	29
Gambar 13. Contoh form HTML.....	31
Gambar 14. Ilustrasi CSS.....	36
Gambar 15. Hasil inline CSS.....	38
Gambar 16. Hasil internal CSS.....	38
Gambar 17. Hasil implementasi CSS pada anchor/link (1).....	41
Gambar 18. Hasil implementasi CSS pada anchor/link (2).....	42
Gambar 19. Framework Bootstrap.....	43
Gambar 20. Versi framework Bootstrap.....	44
Gambar 21. Download Bootstrap.....	44
Gambar 22. Pilihan download Bootstrap.....	44
Gambar 23. Hasil ekstrak file download Bootstrap.....	45
Gambar 24. File HTML dengan Bootstrap.....	45
Gambar 25. Hasil Bootstrap.....	46
Gambar 26. Menu komponen Bootstrap.....	46
Gambar 27. Contoh thumbnails Bootstrap.....	47
Gambar 28. Kode untuk komponen thumbnails Bootstrap.....	47
Gambar 29. Struktur folder project.....	48
Gambar 30. Kode file index.html.....	49
Gambar 31. Script jquery.....	49
Gambar 32. File jquery.js.....	50
Gambar 33. Memanggil file jquery.....	50
Gambar 34. Navigasi dengan margin.....	51
Gambar 35. Navigasi tanpa margin.....	51
Gambar 36. Hasil website yang dibuat.....	52
Gambar 37. Kode part 1.....	53
Gambar 38. Kode part 2.....	53
Gambar 39. Kode part 3.....	54
Gambar 40. Kode part 4.....	54
Gambar 41. Kode part 5.....	54
Gambar 42. Kode part 6.....	55
Gambar 43. Kode part 7.....	55
Gambar 44. Kode part 8.....	55

Gambar 45. Kode part 9(1) .....	56
Gambar 46. Kode part 9(2) .....	56
Gambar 47. Kode part 9(3) .....	57
Gambar 48. Kode part 9(4) .....	57
Gambar 49. Hasil tampilan mobile .....	58

## Pertemuan 1 Pendahuluan & Pengenalan HTML

### 1. Pengertian HTML

HTML merupakan singkatan dari *Hypertext Markup Language*. Singkatan ini terdiri dari 3 komponen kata, yakni: Hypertext, Markup dan Language.

Kata Hypertext dari HTML menekankan pengertian: text yang lebih dari sekedar teks ('hyper'-text). Maksudnya selain berfungsi sebagai teks biasa, sebuah teks di dalam HTML juga bisa berfungsi sebagai penghubung ke halaman lain atau dikenal dengan istilah link. Nantinya kita juga akan melihat bahwa tidak hanya teks saja yang bisa digunakan sebagai link, tetapi bisa berupa gambar. Link inilah yang menjadi inti dari HTML.

Kata kedua dari singkatan HTML adalah Markup. Markup dapat diterjemahkan sebagai tanda atau penanda (bahasa inggris: mark). Di dalam HTML, kita akan menggunakan tanda-tanda khusus seperti <p>, <a>, atau <li>. Tanda ini diperlukan untuk mengatur format dan membuat struktur halaman web.

Bagian terakhir dari HTML adalah Language. Istilah language jika diterjemahkan berarti: bahasa. Khusus bagi anda yang pernah berkenalan dengan bahasa pemrograman komputer, disini HTML tidak menggunakan 'Programming Language', tetapi hanya 'Language' saja.

Hal ini secara tidak langsung menyatakan bahwa HTML bukanlah sebuah bahasa pemrograman. HTML tidak memiliki struktur dasar seperti variabel, kondisi IF, function, atau class seperti layaknya sebuah bahasa pemrograman komputer.

## 2. Sejarah HTML



Gambar 1. Tim Barnes Lee

Penggunaan HTML dikendalikan oleh W3C (*World Wide Web Consortium*). HTML dibuat & dikembangkan oleh "Tim Berners Lee" ketika mereka bekerja di CERN pada tahun 1989. Kemudian HTML dipopulerkan oleh Browser Mosaic yang dikembangkan di NSCA. Selama tahun 1990an, berbagai macam versi HTML telah muncul dengan tambahan & kelengkapan fitur-fiturnya. Versi standar HTML yang telah ada saat ini antara lain :

### a. HTML Versi 1.0

HTML versi ini merupakan versi pertama sejak lahirnya Bahasa HTML yang memiliki kemampuan untuk heading, paragraph, hypertext, manipulasi text (*bold & italic*) serta memiliki hubungan terhadap penggunaan sebuah gambar.

### b. HTML Versi 2.0

HTML versi ini lahir pada tanggal 14 Januari 1996 dengan beberapa kemampuan tambahan seperti penambahan form & hal ini menjadi cikal bakal adanya proses interaktif dengan pengguna. Mulai versi ini, HTML menjadi awal mula adanya website interaktif.

### c. HTML Versi 3.0

HTML versi ini lahir pada tanggal 18 Desember 1997 yang dikenal dengan HTML+ karena mempunyai beberapa fasilitas penambahan fitur table dalam paragraph.

**d. HTML Versi 3.2**

HTML versi ini merupakan versi terbaru penyempurnaan dari versi 3.0 pada bulan Mei 1998.

**e. HTML Versi 4.0**

HTML versi ini lahir pada tanggal 24 Desember 1999 dengan penambahan beberapa fitur lagi seperti adanya link, image & lain-lain sebagai penyempurna dari HTML versi 3.2.

**f. HTML Versi 5.0**

HTML versi ini atau lebih dikenal dengan HTML5 lahir pada tahun 2009 yang menjadi standar baru untuk *HTML*, *XHTML*, & *DOM HTML*. HTML5 ini merupakan proyek kerjasama antara W3C dengan WHATWG (*Web Hypertext Application Technology Working Group*).

### **3. Fungsi HTML**

Dalam proses web development (proses pembuatan web), HTML berfungsi untuk membuat struktur dari sebuah website. HTML digunakan untuk menandai bagian mana yang akan menjadi judul artikel, bagian mana yang berfungsi sebagai isi artikel, atau bagian mana yang butuh disajikan dalam bentuk tabel.

Mengenai tampilan dari website tersebut (misalnya apakah akan menggunakan warna background biru atau merah), diserahkan kepada teknologi web lain yang dikenal dengan CSS (*Cascading Style Sheet*).

Saat ini, untuk membuat sebuah web modern, tidak bisa hanya menggunakan HTML + CSS saja, tetapi juga memerlukan bantuan bahasa pemrograman web lain seperti JavaScript. Di sini saya menggunakan istilah ‘web modern’, karena kita memang bisa membuat website dari HTML saja (tanpa CSS dan JavaScript), tetapi web yang dihasilkan akan terkesan ‘jadul’ karena tidak memiliki tampilan yang cantik (sesuai standar saat ini).



#### **4. Apa yang baru pada HTML 5 ?**

Sebagai informasi, html5 adalah generasi berikutnya dari html 4, dimana pada html 5 ini ditambahkan fitur – fitur baru yang mengedepankan penjalanan konten yang berat pada device dengan tenaga rendah yang pastinya membuat para programmer lebih mudah dalam menerapkannya dan berakibat lebih user friendly dari sisi pengguna.

HTML5 mempunyai beberapa fitur menarik yang tidak ada pada HTML versi sebelumnya. Beberapa fitur baru pada HTML5 adalah :

- **Canvas**  
Merupakan element yang digunakan untuk menggambar grafik, object di website yg umumnya melalui script javascript. Untuk menggambar nya anda harus menggunakan script, misal javascript.
- **Header**  
Berguna untuk menyatakan suatu judul yang bisa diisi dengan logo berupa file gambar atau nama perusahaan.
- **Footer**  
Merupakan elemen yang berguna untuk meletakkan informasi pada bagian bawah halaman website.
- **Time**  
Berguna untuk menampilkan informasi tentang waktu
- **Audio**  
Digunakan untuk memainkan file audio dalam dokumen HTML5
- **Video**  
Digunakan untuk memaikan file video dengan format tertentu tanpa bantuan plugin pihak ketiga.

#### **5. Kebutuhan Software**

##### **a. Browser**

Web browser adalah perangkat utama yang akan kita gunakan untuk menampilkan halaman web yang pada dasarnya terbuat dari HTML dan CSS. Setiap browser memiliki perbedaan dalam hal menampilkan halaman web dan fitur-fitur yang didukung dalam HTML dan CSS. Boleh jadi halaman web yang

anda buat ditampilkan benar pada salah satu browser namun tidak sesuai pada browser lain yang digunakan. Untuk itu kita perlu menginstall beberapa browser dalam komputer untuk memastikan website yg dibuat sudah sesuai dengan tampilan yang di inginkan berdasarkan style CSS.

Beberapa browser yang sering digunakan dan selalu mendapatkan update terbaru saat ini adalah :

- 1) Google Chrome
- 2) Firefox
- 3) Safari
- 4) Opera
- 5) Microsoft Edge

#### **b. Text Editor**

Untuk membuat website, kita memerlukan bantuan text editor. Setiap orang memang mempunyai selera yang berbeda terhadap text editor. Biasanya para programmer menggunakan text editor yang ringan tetapi powerfull untuk digunakan pada saat coding.

Jika anda mempunyai atau terbiasa menggunakan aplikasi Adobe Dreamweaver, saya sarankan untuk tidak menggunakannya, mengingat anda masih dalam tahap belajar dan agar kalian mengerti setiap proses yang berjalan pada pembuatan halaman website. Berikut text editor yang biasa digunakan untuk menuliskan coding :

- 1) Visual Studio Code
- 2) Sublime text
- 3) Atom
- 4) Brackets
- 5) Notepad ++
- 6) Dll (sesuai dengan selera masing-masing).

Masih banyak lagi jenis text editor yang dapat digunakan, tergantung dengan selera anda.

## Pertemuan 2 Struktur HTML 5

### 1. Struktur Dasar HTML 5

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

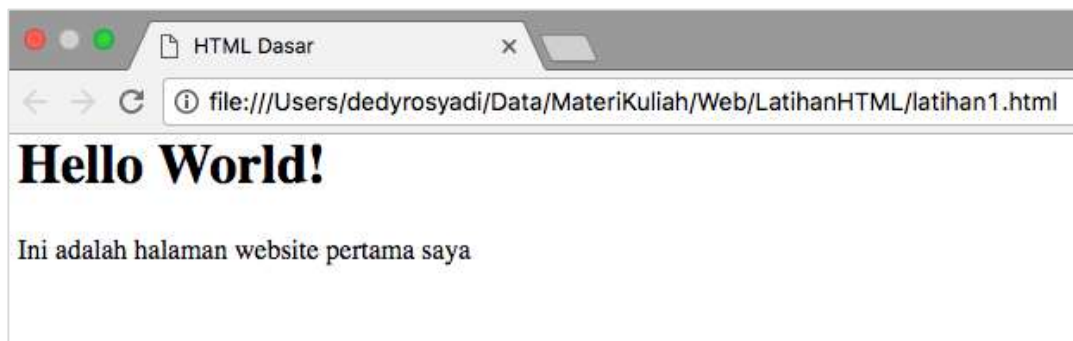
</body>
</html>
```

### 2. Membuat File HTML Pertama

Buka text editor anda kemudian masukan kode dari struktur HTML5 diatas dan tambahkan beberapa baris kode seperti contoh berikut

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title> HTML Dasar</title>
</head>
<body>
  <h1>Hello World!</h1>
  <p>Ini adalah halaman website pertama saya</p>
</body>
</html>
```

setelah menuliskan kode diatas silahkan simpan file tersebut dengan nama **latihan1.html** dan double klik pada file yang sudah tersimpan tadi. Maka browser default yang anda gunakan secara otomatis membuka file html yang telah dibuat.



Gambar 2. Tampilan file html di browser

### 3. Pengertian Tag, Atribut dan Element

#### a. Tag

Tag di dalam HTML berfungsi untuk memberi tahu web browser apa dan bagaimana sebuah teks harus ditampilkan. Sebagaimana yang telah kita pelajari, file HTML hanyalah sebuah file text biasa. Web browser lah yang akan memproses file tersebut dan menampilkan hasilnya.

Tag pada html ditulis secara berpasangan yang terdiri dari **tag pembuka** dan **tag penutup**. Tag pembuka : <tagpembuka> isi </tagpenutup>

Contoh penggunaan tag untuk paragraf :

```
<p>Ini adalah paragraf pertama</p>
<p>Ini adalah paragraf kedua</p>
```

***Keterangan :** tag penutup ditambah dengan garis miring*

#### b. Atribut

Atribut adalah informasi tambahan yang ditulis pada tag pembuka. Fungsi dari atribut ini bermacam-macam tergantung nilai dan pada tag mana ia ditempatkan. Beberapa atribut bersifat umum dan bisa digunakan dalam seluruh tag (dikenal sebagai global atribut), tetapi kebanyakan hanya berfungsi untuk tag tertentu saja.

Contoh atribut pada tag img :

```

```

Penjelasan :

**Img :**

Merupakan tag pada html yang digunakan untuk menampilkan gambar

**Src :**

Merupakan atribut untuk menuju ke sumber file gambar yang akan di tampilkan

**Alt :**

Atribut yang digunakan untuk menampilkan text apabila gambar tidak dapat di tampilkan

**Width & height :**

Merupakan atribut untuk mengatur lebar dan tinggi gambar. Tapi pengaturan ini disarankan menggunakan CSS.

c. Element

Element merupakan gabungan dari tag, atribut dan seluruh isi yang dimulai dari tag pembuka dan tag penutup.

#### 4. Menjalankan file HTML

Untuk membuka dan menjalankan file HTML diperlukan browser untuk menampilkan konten pada halaman yang telah kita buat. Kita bisa menjalankan dengan Klik dua kali pada file HTML yang ingin kita jalankan, secara otomatis browser default pada laptop/computer akan menampilkan konten pada halaman HTML yang telah kita buat.

#### 5. Penjelasan Tag pada struktur HTML 5

a. `<!DOCTYPE html>`

Tag khusus ini dikenal dengan sebutan DTD (Document Type Declaration) atau doctype” berfungsi memberikan informasi pada web browser tentang aturan penulisan dari suatu dokumen.

Penulisan DTD pada html 5 menjadi lebih singkat. Perhatikan perbedaan dengan dari penulisan DTD berikut ini

##### HTML 4

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

##### HTML 5

```
<!DOCTYPE html>
```

Posisi DTD juga harus ditulis pada baris paling awal, boleh ditulis dengan huruf kecil, seperti `<!doctype html>`. Namun anda akan sering menemukan penulisannya yang menggunakan huruf besar, seperti: `<!DOCTYPE html>`.

b. HTML

Tag HTML digunakan untuk membungkus seluruh kode HTML yang lain. Di dalam tag `<html>`, bisa diletakkan atribut lang. Atribut ini diisi dengan dua digit kode bahasa yang akan digunakan untuk isi website. Sebagai contoh, jika kita

membuat website dengan bahasa indonesia, bisa menulis tag `<html>` sebagai: `<html lang="id">`. Atau `<html lang="en">` jika akan membuat website dengan bahasa inggris.

Atribut lang sepenuhnya opsional (boleh ditulis, boleh juga tidak). Namun anda akan sering menemukan atribut ini.

c. Head

Tag `<head>` berfungsi sebagai ‘penampung’ berbagai tag yang umumnya tidak terlihat dalam web browser. Tag-tag di dalam bagian `<head>` berfungsi untuk memberikan informasi tambahan mengenai halaman yang sedang diproses, atau menyediakan referensi ke file lain (seperti file CSS atau JavaScript).

Beberapa tag HTML lain yang biasa terdapat pada bagian `<head>` adalah: `<meta>`, `<title>`, dan `<link>`.

d. Meta

Merupakan singkatan dari metadata. Metadata adalah data yang berisi informasi tentang suatu dokumen, berisi tentang informasi/instruksi mengenai halaman HTML yang sedang dijalankan.

Salah satu tag `<meta>` yang terdapat dalam struktur diatas adalah `<meta charset="UTF-8">`. Tag ini akan menginstruksikan web browser untuk menggunakan karakter set UTF-8 dalam memproses halaman HTML.

e. Title

Sesuai dengan namanya, tag ini berfungsi untuk membuat title dari sebuah halaman. Title adalah teks yang ditampilkan pada bagian atas jendela web browser. Selain itu, jika kita men-bookmark sebuah halaman, title inilah yang akan menjadi nama bookmark tersebut.

f. Body

Tag `<body>` isi `</body>` merupakan penampung dari seluruh element HTML yang akan kita tulis nantinya.

## 6. Membuat Komentar

Selain sebagai pengingat atau catatan, tag komentar juga akan berguna untuk membuat sebagian isi web tidak tampil untuk sementara. hal ini sangat berguna jika

kita ingin mencoba berbagai tampilan kode **HTML** yang akan dibuat, tetapi tidak ingin menghapus kode sebelumnya.

Contoh :

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <title>HTML Dasar</title>
</head>
<body>
  <!-- membuat file HTML pertama -->
  <h1>Hello World!</h1>
  <!-- <p>Ini adalah halaman website pertama saya</p> -->
</body>
</html>
```

text yang telah diberikan tanda *<!-- membuat file HTML pertama -->* dan *<!-- <p>Ini adalah halaman website pertama saya</p> -->* tidak akan di tampilkan di browser

## Pertemuan 3 & 4

### Text Formating, Link, List, Audio, Video, tag div, class dan id

#### 1. Heading

Heading element biasanya digunakan untuk membuat judul (heading) di dalam halaman HTML. Heading element terdiri dari 6 level atau 6 tingkatan, yakni dari `<h1>` sampai dengan `<h6>`.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <h1>Judul Artikel dengan h1</h1>
  <h2>Judul Artikel dengan h2</h2>
  <h3>Judul Artikel dengan h3</h3>
  <h4>Judul Artikel dengan h4</h4>
  <h5>Judul Artikel dengan h5</h5>
  <h6>Judul Artikel dengan h6</h6>
</body>
</html>
```

Simpan file yang telah ditambahkan seperti coding diatas dan jalankan melalui browser maka hasil yang akan tampil seperti ini.



Gambar 3. HTML heading



## 2. Paragraf

Tag `<p>` digunakan untuk membuat paragraf pada halaman website. Tag `<p>` merupakan block level element, sehingga setiap paragraph akan ditampilkan secara terpisah

Perhatikan contoh berikut :

```
<p>Ini adalah paragraf pertama</p>
<p>Ini adalah paragraf kedua</p>
```

Tag paragraf bisa kita tambahkan atribut align, contoh paragraf dengan atribut align :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <p align="center">
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quidem alias
    dolore laborum, eaque, ducimus est vitae porro, excepturi ullam
    inventore possimus. Alias voluptatum eius asperiores, nulla dicta
    repellendus ullam officia!
  </p>
  <p align="right">
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quidem alias
    dolore laborum, eaque, ducimus est vitae porro, excepturi ullam
    inventore possimus. Alias voluptatum eius asperiores, nulla dicta
    repellendus ullam officia!
  </p>
</body>
</html>
```

simpan file kemudian jalankan dan perhatikan paragraph yang telah kalian buat seperti penulisan kode diatas.

## 3. Anchor / Hyperlink

Tag `<a>` disebut dengan anchor adalah tag yang digunakan untuk membuat link menuju kehalaman yang lain. Contoh :

```
<a href="http://google.com">Menuju halaman google</a>
```

Atribut **href** digunakan untuk mendeskripsikan alamat yang akan dituju, seperti contoh diatas link yang dituju adalah <http://google.com> biasa disebut juga dengan **alamat absolut**.

Kita juga bisa mengatur atribut **href** menuju alamat local yang biasa disebut dengan istilah **alamat relative** berdasarkan tujuan struktur folder. Contoh :

```
<a href="biodata.html">Menuju halaman biodata</a>
```

### Atribut target

Fungsi dari atribut ini adalah untuk memberikan perintah kepada browser agar menampilkan halaman web ke tempat yang telah di tentukan. Secara default jika tidak menggunakan atribut target, browser akan membuka link pada window/tab yang sama. Misalnya kita menambahkan perintah target="\_blank" pada suatu link, maka jika kita klik link tersebut akan terbuka pada window.atau tab baru pada browser.

Sekarang buatlah kode HTML seperti contoh dibawah ini kemudian jalankan. Perhatikan perbedaan ketika link di klik dengan link sebelumnya yang tidak menggunakan atribut target.

```
<a href="biodata.html" target="_blank">Menuju halaman biodata</a>
```

## 4. Daftar / List

Sebuah daftar dapat memiliki penomoran atau hanya sebuah simbol berupa lingkaran atau bentuk lainnya. Dalam HTML, daftar yang menggunakan penomoran disebut dengan ordered list dan yang menggunakan simbol disebut dengan unordered list.

### a. Ordered List

Ordered list atau Daftar berurutan dapat dibuat dengan menggunakan tag <ol> (singkatan dari ordered list) dan untuk setiap listnya kita gunakan tag <li> (singkatan dari list item/item daftar). Sebagai contoh, perhatikan kode HTML berikut :

```
<h2>Pekerjaan Hari ini</h2>
<ol>
  <li>Kuliah</li>
  <li>Buat tutorial HTML</li>
  <li>Mengerjakan tugas kuliah</li>
</ol>
```

Buatlah kode html seperti contoh diatas kemudian lihat hasilnya pada browser kalian. Penomoran daftar akan dilakukan secara otomatis ketika anda menambahkan daftar item.

#### b. Unordered List

Berbeda dengan daftar berurut, unordered list akan menandai setiap item dengan simbol baik berupa lingkaran atau persegi dan kita masih bias merubahnya menggunakan CSS. Untuk membuat daftar tidak berurut kita gunakan tag `<ul>` dan seperti tag `<ol>`, item yang terdapat di dalamnya harus diapit dengan tag `<li>`. Sebagai contoh, perhatikan kode HTML berikut :

```
<h2>Pekerjaan Hari ini</h2>
<ul>
  <li>Kuliah</li>
  <li>Buat tutorial HTML</li>
  <li>Mengerjakan tugas kuliah</li>
</ul>
```

Buatlah kode html seperti contoh diatas kemudian lihat hasilnya pada browser kalian. Perhatikan perbedaannya dengan ordered list.

#### c. Definition List

Berbeda dengan Ordered List dan Unordered List yang memiliki struktur sama, Data list memiliki struktur tersendiri. Biasanya data list ini digunakan untuk membuat daftar istilah beserta definisinya seperti halnya dalam kamus. Perhatikan Contoh Berikut :

```
<h1>Definition List</h1>
<h3>Pemrograman Web 1</h3>
<dl>
  <dt><b>HTML</b></dt>
  <dd>Hypertext Markup Language</dd>

  <dt><b>CSS</b></dt>
  <dd>Cascading Style Sheet</dd>
</dl>
```

### 5. Image

Untuk menampilkan gambar ke dalam web browser, tag `<img>` membutuhkan beberapa atribut yaitu atribut **src**. Atribut src (singkatan dari source) berfungsi untuk menulis alamat gambar yang akan ditampilkan. Alamat ini bisa berupa alamat absolut atau alamat relative. Perhatikan kode HTML berikut :

```

```

pada contoh kode HTML diatas alamat yang digunakan untuk membuka gambar adalah alamat absolut. Jika kita ingin membuka gambar yang ada pada folder local

kita maka bias dituliskan nama filenya beserta ekstensi file. Perhatikan contoh berikut :

```

```

### Atribut alt

(singkatan dari alternate text) berfungsi untuk memberikan keterangan mengenai gambar. Isi dari atribut ini adalah pesan teks yang akan ditampilkan pada kondisi-kondisi berikut:

- Gambar tidak dapat ditampilkan (misalkan karena gangguan jaringan internet).
- Web browser tidak mendukung format gambar.
- Web browser sedang dalam proses menampilkan gambar.
- Web browser diset agar tidak menampilkan gambar.
- Pengguna menggunakan browser khusus seperti “screen reader” yang tidak akan menampilkan gambar.

Perhatikan kode HTML berikut :

```

```

### Atribut Height dan Width

Atribut height dan width digunakan untuk mengatur ukuran tinggi dan lebar gambar. Nilai dari kedua atribut ini berupa angka dalam satuan pixel maupun persen (%).

```

```

sebenarnya masih banyak atribut lain yang bias digunakan. Tetapi pembahasan ini akan kita bahas lebih lengkap ketika mempelajari CSS.

## 6. Image dengan Link

Pada halaman web, kita bisa menyisipkan link pada image/gambar yang biasa digunakan sebagai tombol dengan menggabungkan hyperlink dan image. Contoh kode HTML image sebagai link :

```
<a href="http://facebook.com/mdedyrosyadi" target="_blank">  
    
</a>
```

## 7. Iframe

Iframe ini digunakan untuk menampilkan halaman web dalam halaman web.

Contoh penggunaan iframe :

```
<iframe src="url"></iframe>
```

sekarang kita akan mencoba menampilkan halaman google di dalam halaman web kita. Perhatikan kode HTML berikut ini :

```
<iframe src="http://www.w3schools.com"></iframe>
```

setelah membuat kode diatas silahkan simpan dan jalankan melalui browser. Lebar dan tinggi **iframe** bisa di atur menggunakan atribut **width** dan **height** seperti mengatur ukuran gambar.

## 8. Audio

HTML5 menyediakan tag <audio> untuk menjalankan file audio. Berikut adalah cara penggunaan tag untuk membuka audio :

```
<audio src="musik.mp3"></audio>
```

Tuliskan kode html seperti diatas dan jalankan melalui browser. Ketika dijalankan tidak akan terlihat tampilan apa-apa pada browser. Kita perlu menambahkan beberapa atribut.

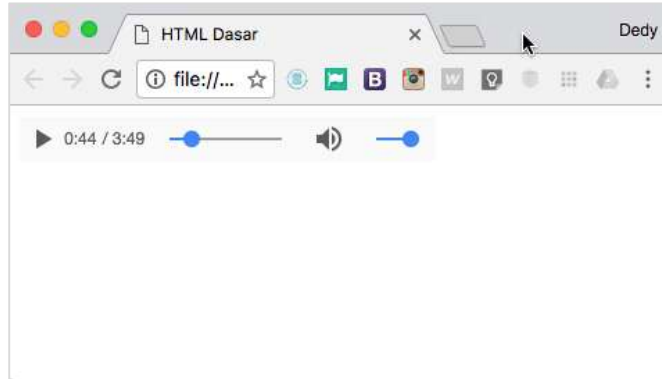
### Atribut src

Atribut src (source) digunakan sebagai alamat file audio yang akan dibuka. Dalam contoh diatas dengan nama file musik.mp3 yang ada didalam folder yang sama pada file html.

### Atribut controls

Atribut controls berfungsi untuk menampilkan tombol control pada audio player, yakni tombol play/stop, slider, volume, timer, dll. Jika tag <audio> ditulis tanpa menggunakan atribut controls, tidak akan terlihat tampilan apa-apa. perhatikan kode html berikut :

```
<audio src="musik.mp3" controls ></audio>
```



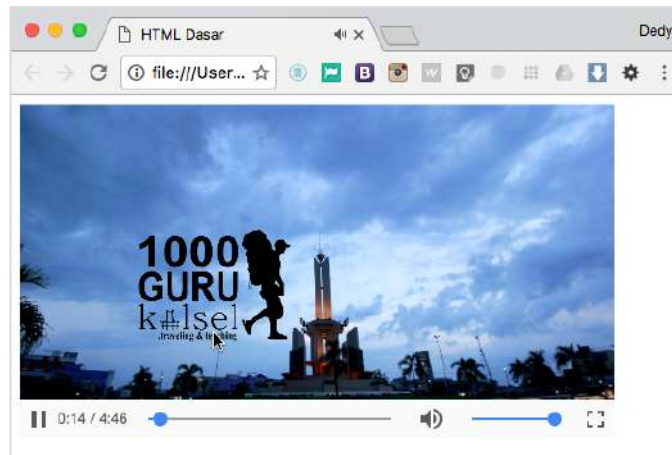
Gambar 4. Tag audio

Masih banyak lagi atribut yang digunakan untuk tag audio.

## 9. Video

HTML5 juga menyediakan tag <video> untuk menjalankan video. Tag <video> memiliki atribut-atribut yang sama dengan tag <audio>, seperti src, controls, autoplay, loop, muted dan preload.

```
<video src="video.mp4" controls ></video>
```



Gambar 5. Tag video

## 10. Tag div, class dan id

### a. Div

Dalam HTML, kita dapat membuat suatu group dari tag HTML yang telah dibuat dengan menggunakan tag <div> (division/bagian) dan untuk menamai setiap div kita gunakan atribut id atau class. Contoh :

```
<div id="header"> </div>
<div id="content"> </div>
<div id="sidebar"> </div>
<div id="footer"> </div>
```

setelah di jalankan di browser anda tidak akan melihat apa-apa karena tag div tidak akan menampilkan efek visual. Tag ini hanya digunakan untuk membuat dokumen HTML lebih terstruktur dengan membagi-bagi dokumen ke dalam bagian- bagian yang lebih spesifik.

### b. Id

Atribut ID digunakan untuk penamaan elemen HTML yang memiliki karakteristik unik/berbeda. Tidak boleh ada dua atau lebih elemen yang mempunyai ID yang sama. Perhatikan contoh berikut :

```
<div id="menu">
  <ul id="menu">
    <li>Beranda</li>
    <li>Tutorial</li>
  </ul>
</div>
```

Penggunaan atribut ID pada contoh di atas adalah **SALAH**, karena terdapat dua ID yang sama, yaitu “menu” pada tag <div> dan <ul>.

### c. Class

Class digunakan untuk penamaan elemen yang memiliki karakteristik/struktur sama dan dapat digunakan berulang kali dalam Kode HTML. Sebagai contoh, perhatikan kode HTML berikut :

```
<ul id="menu">
  <li class="merah">Beranda</li>
  <li>Tutorial</li>
  <li class="merah">Berita</li>
  <li>Video</li>
</ul>
```

Pada Kode HTML di atas, class merah digunakan pada beberapa list item, karena nantinya list item yang memiliki class merah akan diberi warna background merah.

Kesimpulannya adalah, ketika anda memiliki beberapa elemen dengan karakter/format yang sama, gunakan Class sebagai penamaannya dan Gunakan ID untuk elemen yang berbeda dan membutuhkan tanda pengenalan lebih spesifik.



## Pertemuan 5 HTML Table

### 1. Struktur Dasar Table

Struktur dasar dari sebuah tabel HTML terdiri dari tag berikut :

- **Table** : `<table> </table>`
- **Row** : `<tr> </tr>`
- **Sel/Data** : `<td> </td>`

Contoh penulisan table sederhana pada dokumen HTML :

```
RRR<table>
  <tr>
    <td>Baris 1 Kolom 1</td>
    <td>Baris 1 Kolom 2</td>
  </tr>
  <tr>
    <td>Baris 2 Kolom 1</td>
    <td>Baris 2 Kolom 2</td>
  </tr>
  <tr>
    <td>Baris 3 Kolom 1</td>
    <td>Baris 3 Kolom 2</td>
  </tr>
</table>
```

Hasil Table dari kode diatas :

Baris 1 Kolom 1	Baris 1 Kolom 2
Baris 2 Kolom 1	Baris 2 Kolom 2
Baris 3 Kolom 1	Baris 3 Kolom 2

Gambar 6. HTML table

Sekilas terlihat hanya beberapa baris tulisan dan tidak terlihat seperti table, tetapi sebenarnya struktur table sudah kita buat hanya saja kita perlu mengatur properties border untuk mendapatkan tampilan yg lebih bagus. Pada pembahasan selanjutnya akan kita bahas tentang atribut border.

### 2. Atribut Border

Atribut border digunakan untuk membuat garis pada pinggiran isi table. Sekarang tambahkan atribut border pada kode sebelumnya seperti contoh berikut.

```
<table border="1">
```

Sekarang tampilan table akan berubah seperti gambar dibawah ini.

Baris 1 Kolom 1	Baris 1 Kolom 2
Baris 2 Kolom 1	Baris 2 Kolom 2
Baris 3 Kolom 1	Baris 3 Kolom 2

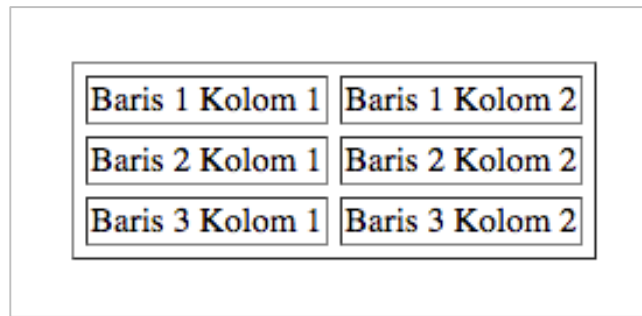
Gambar 7. Atribut table border

### 3. Atribut Cellspacing

Atribut cellspacing digunakan untuk mengatur jarak antara 1 sel dengan sel yang lainnya. Nilai cellspacing berupa angka dalam satuan pixel. Perhatikan contoh kode berikut ini dan tulis pada text editor kalian.

```
<table border="1" cellspacing="5">
  <tr>
    <td>Baris 1 Kolom 1</td>
    <td>Baris 1 Kolom 2</td>
  </tr>
  <tr>
    <td>Baris 2 Kolom 1</td>
    <td>Baris 2 Kolom 2</td>
  </tr>
  <tr>
    <td>Baris 3 Kolom 1</td>
    <td>Baris 3 Kolom 2</td>
  </tr>
</table>
```

Tampilan table :



Baris 1 Kolom 1	Baris 1 Kolom 2
Baris 2 Kolom 1	Baris 2 Kolom 2
Baris 3 Kolom 1	Baris 3 Kolom 2

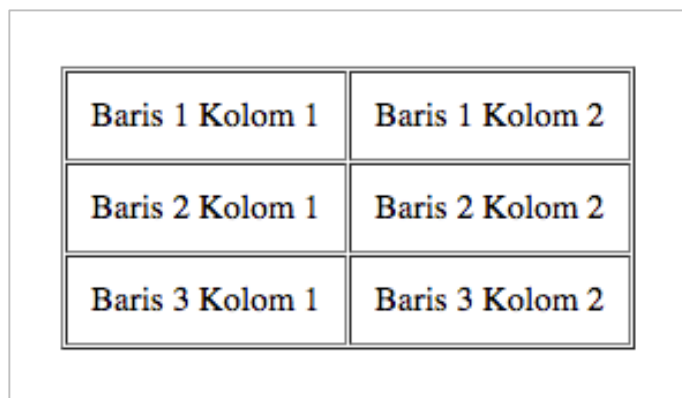
Gambar 8. Atribut table cellpadding

#### 4. Atribut Cellpadding

Atribut cellpadding digunakan untuk mengatur jarak antara garis border cell dengan teks, nilai dari attribute cellpadding berupa angka dalam satuan pixel. Perhatikan kode berikut dan tulis pada text editor kalian.

```
<table border="1" cellspacing="1" cellpadding="10" >
  <tr>
    <td>Baris 1 Kolom 1</td>
    <td>Baris 1 Kolom 2</td>
  </tr>
  <tr>
    <td>Baris 2 Kolom 1</td>
    <td>Baris 2 Kolom 2</td>
  </tr>
  <tr>
    <td>Baris 3 Kolom 1</td>
    <td>Baris 3 Kolom 2</td>
  </tr>
</table>
```

Table yang dihasilkan :



Baris 1 Kolom 1	Baris 1 Kolom 2
Baris 2 Kolom 1	Baris 2 Kolom 2
Baris 3 Kolom 1	Baris 3 Kolom 2

Gambar 9. Tampilan hasil table cellpadding

## 5. Atribut Colspan dan Rowspan

**Colspan(column span)** adalah atribut yang digunakan untuk menggabungkan kolom pada table secara horizontal, sedangkan **rowspan** merupakan atribut yang digunakan untuk menggabungkan beberapa baris secara vertical. Perhatikan contoh kode dibawah ini.

### a. Contoh Colspan

kode :

```
<table border="1">
  <tr>
    <td colspan="5">Gabungan Kotak 1-5</td>
  </tr>
  <tr>
    <td>Kotak 1</td>
    <td>Kotak 2</td>
    <td>Kotak 3</td>
    <td>Kotak 4</td>
    <td>Kotak 5</td>
  </tr>
</table>
```

hasil :

Gabungan Kotak 1-5				
Kotak 1	Kotak 2	Kotak 3	Kotak 4	Kotak 5

Gambar 10. Hasil table colspan

### b. Contoh Rowspan

kode :

```
<table border="1">
  <tr>
    <td rowspan="3">Kotak 1-3</td>
    <td>Kotak 1</td>
  </tr>
  <tr>
    <td>Kotak 2</td>
  </tr>
  <tr>
    <td>Kotak 3</td>
  </tr>
</table>
```

Hasil :

Kotak 1-3	Kotak 1
	Kotak 2
	Kotak 3

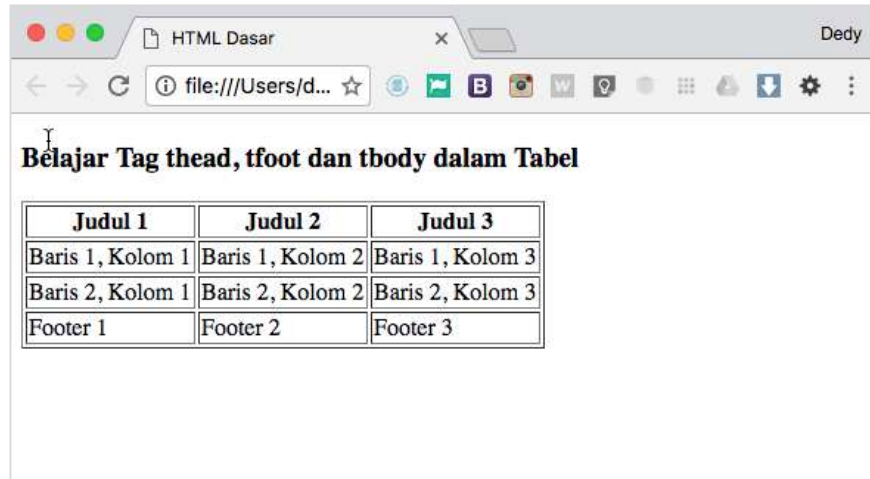
Gambar 11. Hasil rowspan

## 6. Thead, Tbody dan Tfoot

Untuk tabel yang lebih kompleks, HTML menyediakan tag yang dapat digunakan untuk ‘membagi’ tabel menjadi bagian-bagian yang terpisah, misalnya bagian judul (**head**), bagian penutup (**foot**), dan bagian isi tabel (**body**).

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>HTML Dasar</title>
  </head>
  <body>

    <h3>Belajar Tag thead, tfoot dan tbody dalam Tabel</h3>
    <table border="1">
      <thead>
        <tr>
          <th>Judul 1</th>
          <th>Judul 2</th>
          <th>Judul 3</th>
        </tr>
      </thead>
      <tfoot>
        <tr>
          <td>Footer 1</td>
          <td>Footer 2</td>
          <td>Footer 3</td>
        </tr>
      </tfoot>
      <tbody>
        <tr>
          <td>Baris 1, Kolom 1</td>
          <td>Baris 1, Kolom 2</td>
          <td>Baris 1, Kolom 3</td>
        </tr>
        <tr>
          <td>Baris 2, Kolom 1</td>
          <td>Baris 2, Kolom 2</td>
          <td>Baris 2, Kolom 3</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```



Gambar 12. Hasil thead, tbody & tfoot

## Pertemuan 6

### Form

#### 1. Form element

Form merupakan salah satu aspek terpenting di dalam sebuah website. Hampir setiap web memiliki form sebagai sarana interaksi dengan pengunjung. Apakah itu digunakan untuk register, login, polling, atau untuk pencarian (search box).

Proses pembuatan form sebenarnya cukup kompleks, karena kita tidak hanya berurusan dengan HTML saja, tetapi juga dengan teknologi web lain seperti PHP (untuk memproses form), MySQL (untuk menyimpan hasil form), dan JavaScript (membuat validasi).

Seluruh objek form (text box, radio button, checkbox, dll) harus berada diantara tag pembuka `<form>` dan tag penutup `</form>`. Tag ini berguna sebagai container atau penampung dari form.

```
<form>
...
...
...
</form>
```

#### 2. Form atribut

Terdapat beberapa atribut pada form yang dapat digunakan tetapi pada materi ini hanya menjelaskan beberapa atribut yang sering digunakan, yaitu :

##### a. Action

Atribut action digunakan untuk menulis alamat file yang menjadi tujuan pengiriman data. Biasanya nilai dari atribut action berupa alamat sebuah halaman PHP (atau bahasa pemrograman server lain) yang digunakan untuk memproses isi form.

##### b. Method

Nilai dari atribut method ini adalah get atau post. Perbedaannya adalah, jika kita mengisi atribut method dengan get maka isian form akan terlihat pada url browser. Method ini biasanya digunakan untuk query pencarian.

Method post biasanya digunakan untuk data yang lebih sensitif seperti yang berisi password, atau registrasi user. Data hasil form tidak akan terlihat pada browser.

c. Name

Setiap tag inputan di dalam form harus ditambahkan atribut name agar dapat diproses oleh web server nantinya. Di dalam halaman proses (yang biasanya berupa bahasa PHP atau ASP), nilai dari atribut name inilah yang akan menjadi variabel form.

d. Value

Atribut value berguna untuk memberi nilai pada sebuah objek form. Khusus untuk text box, nilai dari atribut value akan ditampilkan sebagai nilai awal.

Semua atribut pada form akan kita gunakan pada saat pembuatan website. Pada penjelasan dan contoh selanjutnya kita akan menggunakan atribut tersebut.

### 3. Label

Label digunakan untuk memberikan keterangan pada setiap inputan yang ada. Contoh label pada form sign up linkedin :

The image shows a portion of the LinkedIn website's sign-up interface. It features a blue header with the LinkedIn logo and the text "Make the most of your professional life". Below this is a white form box containing three input fields. Each field is preceded by a label: "First name", "Last name", and "Email". The input fields are empty text boxes.

Gambar 13. Contoh form HTML

Pada contoh diatas terdapat label yang memberikan keterangan tentang inputan yang ada (first name, last name, dan email).



#### 4. Input Type

Berikut ini adalah beberapa contoh input type yang biasa digunakan pada form HTML. Sebenarnya masih banyak jenis input pada HTML, kalian bisa mendapatkan referensi di [www.w3schools.com](http://www.w3schools.com).

Siapkan text editor kalian dan tuliskan contoh kode dibawah ini dan lihat pada browser kalian seperti apa hasilnya.

##### a. Text

Control input ini dapat diisi dengan teks yang memiliki kata yang tidak terlalu panjang/hanya satu baris, biasa digunakan dalam form pencarian, nama dan lain sebagainya.

```
<form>
  First name:<br>
  <input type="text" name="firstname"><br>
  Last name:<br>
  <input type="text" name="lastname">
</form>
```

##### b. Text Area

Mirip seperti text box, tetapi text area bisa menampung lebih banyak text.

```
<form>
  Alamat:<br>
  <textarea name="alamat" cols="100" rows="5"></textarea>
</form>
```

##### c. Radio

Tampilan pilihan dengan bentuk bulat dan hanya bisa dipilih salah satu saja, misalnya pilihan untuk memilih jenis kelamin laki-laki atau perempuan pada form inputan data.

```
<form>
  <input type="radio" name="gender" value="male" checked> Male<br>
  <input type="radio" name="gender" value="female"> Female<br>
</form>
```

##### d. Checkbox

Objek form yang tampilannya berupa kotak persegi yang digunakan untuk memilih ceklist dengan mengklik kotak pilihan tersebut.

```
<form>
  <input type="checkbox" name="vehicle1" value="Bike"> I have a bike<br>
  <input type="checkbox" name="vehicle2" value="Car"> I have a car
</form>
```

e. Select

Objek form yang digunakan untuk memilih dalam bentuk list dropdown.

```
<select>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="mercedes">Mercedes</option>
  <option value="audi">Audi</option>
</select>
```

f. Button

“Umumnya digunakan untuk membuat tombol yang berada di luar form. Tombol yang terletak bukan di dalam form tidak memiliki fungsi bawaan, sehingga cocok digunakan sebagai ‘pemanggil’ kode JavaScript.”

```
<form>
  <button type="button">Click Me!</button>
</form>
```

g. Date

Digunakan untuk memilih tanggal pada form HTML.

```
<form>
  Birthday:
  <input type="date" name="bday">
</form>
```

h. Email

Digunakan untuk input data text dengan isi text berupa email.

```
<form>
  E-mail:
  <input type="email" name="email">
</form>
```

i. Password

Untuk inputan password, HTML menyediakan objek form khusus, yakni input type password. Untuk menggunakannya, kita menambahkan atribut type="password" pada tag <input>.”

```
<form>
  User name:<br>
  <input type="text" name="username"><br>
  User password:<br>
  <input type="password" name="password">
</form>
```

j. Number

Digunakan untuk input berupa data angka.

```
<form>
  Quantity (between 1 and 5):
  <input type="number" name="quantity" min="1" max="5">
</form>
```

k. Submit

Digunakan untuk mengirimkan data pada form ke server.

```
<form action="">
  First name:<br>
  <input type="text" name="firstname" value="Mickey"><br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse"><br><br>
  <input type="submit" value="Submit">
</form>
```

l. Reset

“HTML juga menyediakan tombol reset untuk mengosongkan isi form.

Penggunaannya mirip dengan tombol submit”

```
<form action="">
  First name:<br>
  <input type="text" name="firstname" value="Mickey"><br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse"><br><br>
  <input type="submit" value="Submit">
  <input type="reset">
</form>
```

## Pertemuan 7 & 8 CSS (cascading style sheet)

### 1. Pengertian CSS

CSS adalah singkatan dari **Cascading Style Sheet**. Menurut Wikipedia, Cascading Style Sheet (CSS) merupakan aturan untuk mengatur beberapa komponen dalam sebuah web sehingga akan lebih terstruktur. Perlu diketahui untuk saat ini terdapat tiga versi CSS, yaitu :

#### a. CSS 1

Fokus mengatur pemformatan dokumen HTML, seperti :

- 1) Font (Jenis ketebalan).
- 2) Warna, teks, background dan elemen lainnya.
- 3) Text attributes, misalnya spasi antar baris, kata dan huruf.
- 4) Posisi teks, gambar, table dan elemen lainnya.
- 5) Margin, border dan padding.

#### b. CSS 2

Kehadiran CSS 2 diharapkan lebih baik dari versi sebelumnya. Versi dikembangkan untuk memenuhi kebutuhan terhadap format dokumen agar bisa di tampilkan di printer. CSS 2 mendukung penentuan posisi konten, downloadable, huruf (font), tampilan pada table/table layout dan media tipe untuk printer.

#### c. CSS 3

CSS3 juga dapat melakukan animasi pada halaman website, di antaranya animasi warna hingga animasi 3D. Dengan CSS3 desainer lebih dimudahkan dalam hal kompatibilitas websitenya pada smartphone dengan dukungan fitur baru yakni media query. Selain itu, banyak fitur baru pada CSS3 seperti: multiple background, border-radius, drop-shadow, border-image, CSS Math, dan CSS Object Model.

Masih banyak definisi tentang CSS dan kemungkinan CSS akan terus dikembangkan seiring berkembangnya teknologi.

## 2. Fungsi CSS

Berikut ilustrasi tentang fungsi dari CSS.



Gambar 14. Ilustrasi CSS

Pasti faham kan maksud dari ilustrasi diatas ? Ya! Pada intinya CSS digunakan untuk mengatur halaman website kita menjadi lebih bagus dan lebih baik, menarik dan lebih terstruktur. Istilah lain menyebutkan CSS berfungsi untuk mempercantik tampilan website.

## 3. Selector, Property dan Value

Selector dasar dari CSS adalah tag dari HTML itu sendiri yang biasa kita gunakan pada saat menulis kode HTML, misalnya tag h1, p, a, img, dan masih banyak tag yang lain pada kode HTML. Berikut ini saya berikan contoh penulisan kode CSS sederhana untuk merubah ukuran font dan warna pada tag h1.

```
h1 {  
    font-size: 20px;  
    color: red;  
}
```

Maksud dari kode CSS yang dituliskan diatas adalah, kita memberikan perintah melalui kode CSS untuk merubah **ukuran(font-size)** menjadi 20px dan merubah

**warna(color)** menjadi warna merah pada tag **h1**. Setelah ini kita akan melihat beberapa bagian pada penulisan CSS yang saya bedakan dengan warna background

```
h1 {  
  font-size: 20px;  
  color: red;  
}
```

Keterangan warna :

Selector

Property

Value

Cara penulisan kode CSS :

- Diawali dengan penulisan **selector**, pada contoh kode diatas “**h1**” merupakan selector.
- Setelah penulisan selector, terdapat bagian declaration (property & value) yang berada di dalam tanda kurung kurawal dan penutupnya. Pada contoh kode CSS diatas terdapat dua declaration yaitu **font-size:20px;** dan **color:red;**
- Penulisan peroperty dan value dipisah menggunakan tanda titik dua “:”
- Property yang satu dengan yang lainnya dipisahkan menggunakan tanda titik koma “;”.

#### 4. Jenis CSS

##### a. Inline CSS

Metode inline CSS akan menyisipkan kode CSS langsung kedalam tag HTML. Ini dilakukan menggunakan atribut style. Cara ini kurang di rekomendasikan karena setiap tag HTML harus kita beri style secara berulang-ulang, kecuali hanya merubah style untuk 1 element saja.

Kode :

```

<!DOCTYPE html>
<head>
  <title>Latihan CSS</title>
</head>
<body>
  <h1 style="color:red;">Fakultas Teknologi Informasi</h1>
  <p style="color:green;">Teknik Informatika</p>
</body>
</html>

```

Hasil :



Gambar 15. Hasil inline CSS

b. Internal CSS

Metode internal CSS dilakukan dengan menginput kode CSS pada bagian `<head>` dari sebuah halaman HTML. Kita menggunakan tag `<style>` untuk keperluan ini.

Kode :

```

<!DOCTYPE html>
<head>
  <title>Latihan CSS</title>
  <style>
    h1 {
      color: red;
    }
    p {
      color: green;
    }
  </style>
</head>
<body>
  <h1>Fakultas Teknologi Informasi</h1>
  <p>Teknik Informatika</p>
</body>
</html>

```

hasil :



Gambar 16. Hasil internal CSS

c. Eksternal CSS

Metode external CSS adalah menginput file CSS pada bagian <head> dengan menggunakan tag <link>. Dengan cara ini, sebuah file CSS bisa digunakan oleh banyak halaman HTML. Cara ini paling di rekomendasikan karena saat kita merubah kode css maka semua halaman akan mengikuti perubahan tersebut tanpa menuliskan secara berulang-ulang pada setiap halaman seperti **internal CSS**.

Sekarang tugas kalian ikuti langkah berikut ini :

1. Buatlah file baru dan masukan kode CSS yang sebelumnya sudah dituliskan pada CSS Internal kemudial simpan file tersebut dengan nama **style.css** didalam folder yang sama.
2. Hapus CSS internal pada file HTML kalian yang sebelumnya sudah dibuat.
3. Yang perlu diperhatikan disini ada baris kode yang harus ditambahkan didalam tag <head> pada dokumen HTML kita untuk memanggil file CSS.

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

tuliskan kode diatas didalam bagian tag <head>

4. Simpan file kalian dan lihat pada browser kalian. Jangan lupa refresh browser untuk melihat hasil perubahan pada kode terbaru.

Berikut kode lengkap untuk memanggil CSS Eksternal

Kode CSS :

```
h1 {  
    color: red;  
}  
p {  
    color: green;  
}
```

Kode HTML :



```
<!DOCTYPE html>
<head>
  <title>Latihan CSS</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
  <h1>Fakultas Teknologi Informasi</h1>
  <p>Teknik Informatika</p>
</body>
</html>
```

## Pertemuan 9

### Penerapan CSS sederhana pada HTML

#### 1. Membuat style pada anchor/link

Pada latihan ini saya akan menuliskan kode CSS eksternal yang nantinya akan saya panggil pada file HTML sesuai dengan contoh pada materi sebelumnya. Kita akan membuat style sederhana pada anchor seperti merubah warna background, warna font, padding, dll.

Buat file HTML dan tuliskan kode berikut ini.

Kode HTML :

```
<!DOCTYPE html>
<head>
  <title>Latihan CSS</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
  <h1>Social Media</h1>
  <a href="http://facebook.com" target="_blank">Facebook</a>
</body>
</html>
```

Kode CSS :

```
a {
  padding: 15px 30px;
  background-color: #516EC3;
  text-decoration: none;
  color: #fff;
}
```

Hasil :



Gambar 17. Hasil implementasi CSS pada anchor/link (1)

Sekarang kita akan membuat beberapa anchor dengan warna yang berbeda berdasarkan nama Class.

Kode HTML :

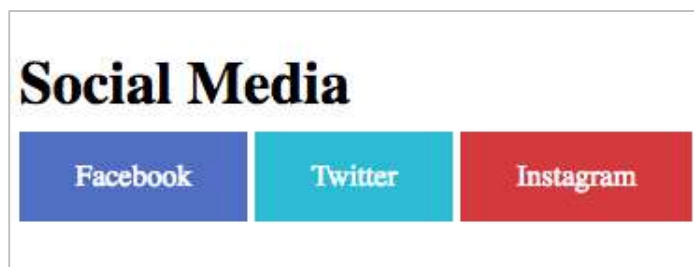
```
<!DOCTYPE html>
<head>
  <title>Latihan CSS</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
  <h1>Social Media</h1>
  <a href="http://facebook.com" target="_blank" class="fb">Facebook</a>
  <a href="http://twitter.com" target="_blank" class="tw">Twitter</a>
  <a href="http://instagram.com" target="_blank" class="ig">Instagram</a>
</body>
</html>
```

Kode CSS :

```
a {
  padding: 15px 30px;
  text-decoration: none;
  color: #fff;
}

.fb {
  background-color: #516EC3;
}
.tw {
  background-color: #1FBBD4;
}
.ig {
  background-color: #D2393D;
}
```

Hasil :



Gambar 18. Hasil implementasi CSS pada anchor/link (2)

## Pertemuan 10 & 11 Framework Bootstrap

### 1. Apa itu Bootstrap ?



Gambar 19. Framework Bootstrap

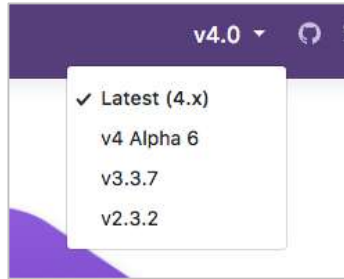
Bootstrap merupakan library framework CSS yang dibuat untuk bagian pengembangan front-end. Bootstrap juga merupakan salah satu framework yang populer di kalangan web developer yang sering digunakan untuk mengembangkan sebuah website yang responsive. Saat ini teknologi responsive merupakan bagian yang wajib ada pada sebuah halaman website, artinya sebuah website dapat menyesuaikan sesuai dengan ukuran device (desktop, tablet, ponsel).

Framework Bootstrap ini sangat membantu kita dalam membuat dan mengembangkan website kita karena bootstrap sudah menyediakan berbagai macam Class CSS yang dapat kita gunakan untuk membuat tampilan website kita menjadi lebih baik. Misalnya seperti contoh membuat style tombol pada anchor yang kita buat sebelumnya, kita harus membuat Class untuk membedakan warna pada tombol tetapi dengan Bootstrap kita dapat langsung menggunakan Class yang sudah tersedia. Pada latihan selanjutnya kita akan membuat website dengan menggunakan bantuan Framework Bootstrap.

### 2. Menggunakan Bootstrap

Untuk menggunakan Framework Bootstrap ada beberapa langkah yang harus diikuti berdasarkan Documentation dari website Bootstrap itu sendiri.

- a. Download File Bootstrap melalui halaman web [www.getbootstrap.com](http://www.getbootstrap.com), disini kita akan menggunakan versi 3.3.7, jadi silahkan pilih versi tersebut pada pilihan di bagian atas sebelah kanan.



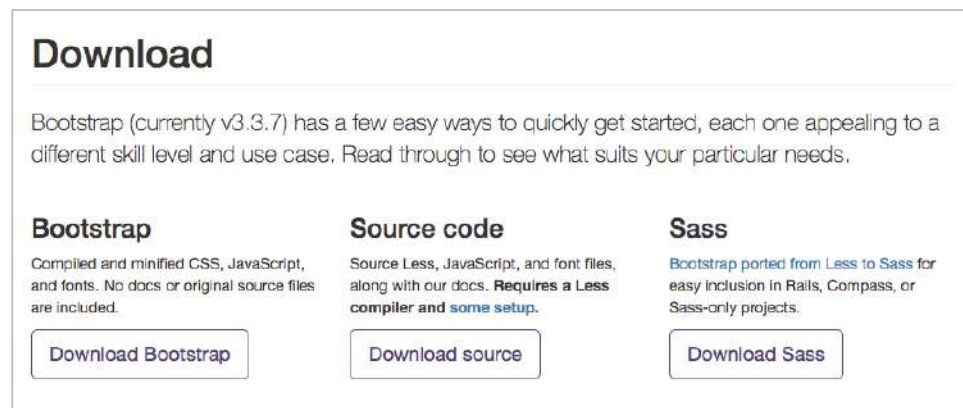
Gambar 20. Versi framework Bootstrap

Kemudian klik tombol Download Bootstrap



Gambar 21. Download Bootstrap

Setelah itu klik lagi tombol Download Bootstrap.



Gambar 22. Pilihan download Bootstrap

- b. Setelah proses download selesai ekstrak File Bootstrap tersebut dan rename folder menjadi nama project kalian. Pada file bootstrap terdapat beberapa folder yang telah tersedia



Gambar 23. Hasil ekstrak file download Bootstrap

- c. Untuk percobaan buatlah sebuah file dengan nama **index.html** di dalam folder Bootstrap yang telah di rename tadi dan tuliskan kode berikut:

```

1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="utf-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1">
7      <!-- The above 3 meta tags *must* come first in the head; any other head content must
8      <!-- come *after* these tags -->
9      <title>Bootstrap 101 Template</title>
10
11     <!-- Bootstrap -->
12     <link href="css/bootstrap.min.css" rel="stylesheet">
13
14     <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->
15     <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
16     <!--[if lt IE 9]>
17       <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
18       <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
19     <![endif]-->
20   </head>
21   <body>
22     <div class="container">
23       <h1>Hello, world!</h1>
24       <button class="btn btn-success">Contoh Tombol Bootstrap</button>
25     </div>
26     <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
27     <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
28     <!-- Include all compiled plugins (below), or include individual files as needed -->
29     <script src="js/bootstrap.min.js"></script>
30   </body>
31 </html>

```

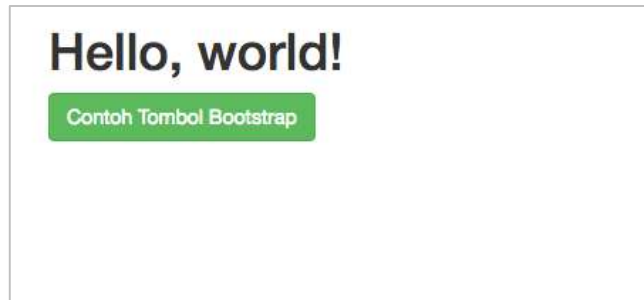
Gambar 24. File HTML dengan Bootstrap

Kode diatas adalah penulisan format penulisan standard dari Bootstrap, beberapa baris kode yang harus diperhatikan adalah :

1. Link stylesheet Bootstrap harus sesuai dengan letak file CSS yang ada pada folder project.
2. Pemanggilan file Javascript Bootstrap sudah benar sesuai dengan letak file pada folder project.

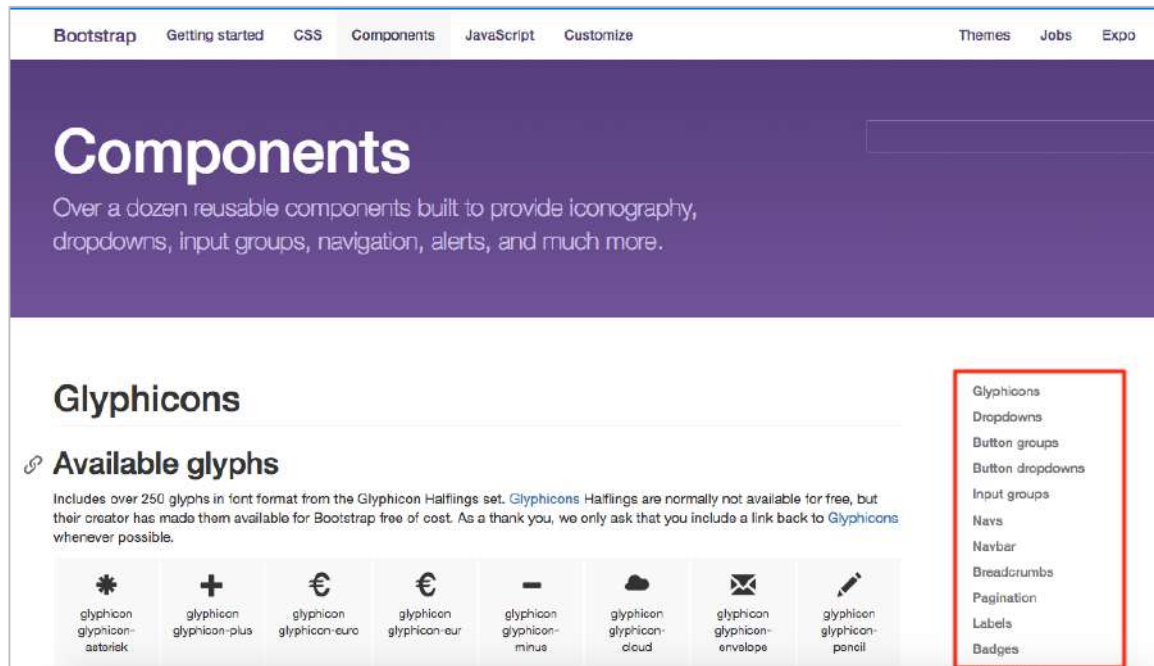
3. JQuery harus berhasil dipanggil, dalam penulisan format standard, Bootstrap memanggil jquery langsung dari library google, artinya kita harus bekerja secara online. Untuk memanggil JQuery secara offline kita harus mendownload file jquery tersebut dan diletakan pada folder project kita. Biasanya JQuery diletakan di dalam folder js.

Apabila sudah selesai melalui tahapan diatas, sekarang kita coba lihar pada browser hasil dari kode yang telah kita buat tadi.



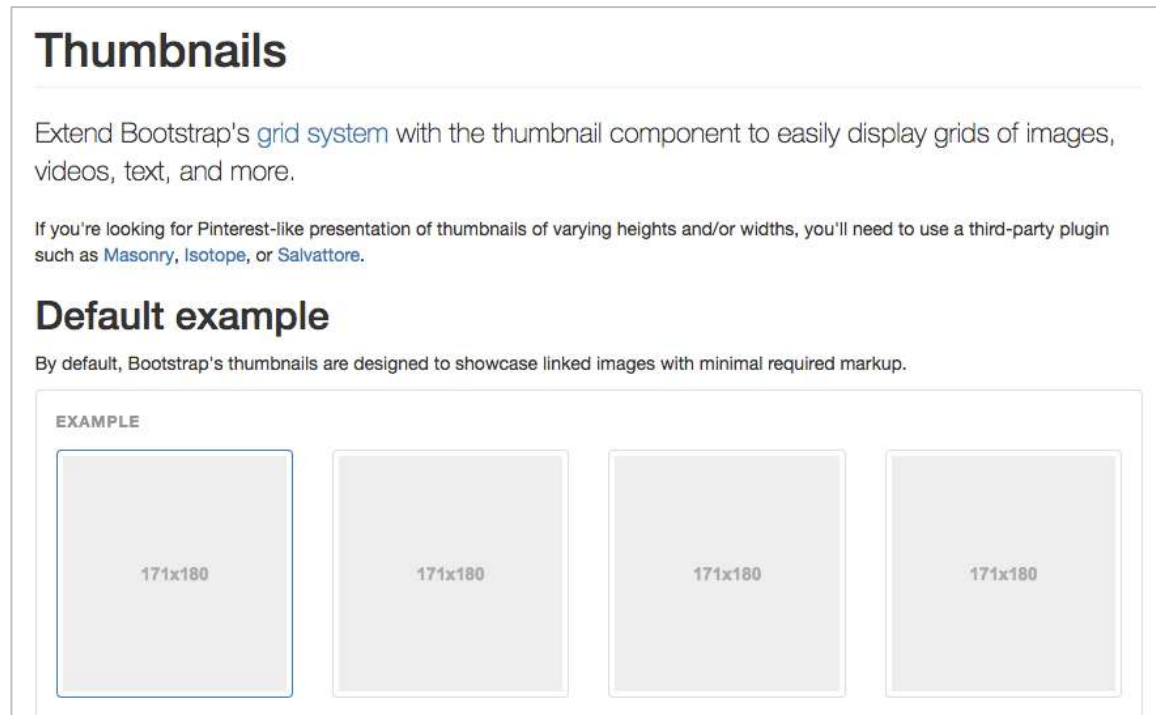
Gambar 25. Hasil Bootstrap

Komponen apa saja yang yang bisa digunakan pada Bootstrap ? Buka link ini <https://getbootstrap.com/docs/3.3/components/> dan perhatikan menu disamping kanan. Buka semua menu disamping kanan tersebut dan lihat komponen apa saja yang dapat digunakan. Selamat Mencoba!



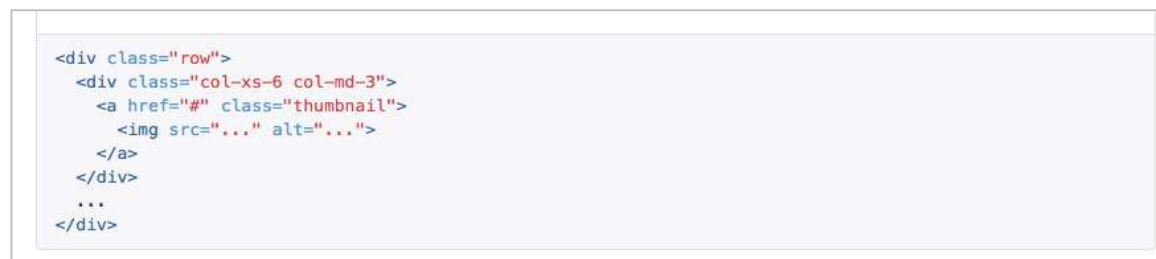
Gambar 26. Menu komponen Bootstrap

Sebagai contoh, saya membuka menu Thumbnails :



Gambar 27. Contoh thumbnails Bootstrap

Gambar diatas merupakan fasilitas dari Bootstrap untuk membuat gambar dalam bentuk thumbnails. Bootstrap tidak hanya menampilkan hasilnya saja, tetapi juga memberi class yang digunakan untuk membuat tampilan Thumbnails tersebut. Lihat coding dibawah ini.



Gambar 28. Kode untuk komponen thumbnails Bootstrap

Sudah cukup jelas? Semoga saja iya ☺. Mari kita lanjutkan pembahasan selanjutnya



## Pertemuan 12, 13 & 14

### Membuat Website Statis Single Page

Pada pembahasan ini kita akan membuat website sederhana yang menerapkan HTML, CSS dan JS/jQuery pada website yang kita buat. Untuk mengatur layout website kita akan memanfaatkan fasilitas yang ada pada framework Bootstrap. Sudah siap ?!

Buatlah folder project kalian dan copy semua file bootstrap seperti pada pembahasan pertemuan 10 dan 11. Pastikan semua file Bootstrap tercopy dengan sempurna dan tidak ada file yang kurang. Perhatikan struktur folder berikut ini.

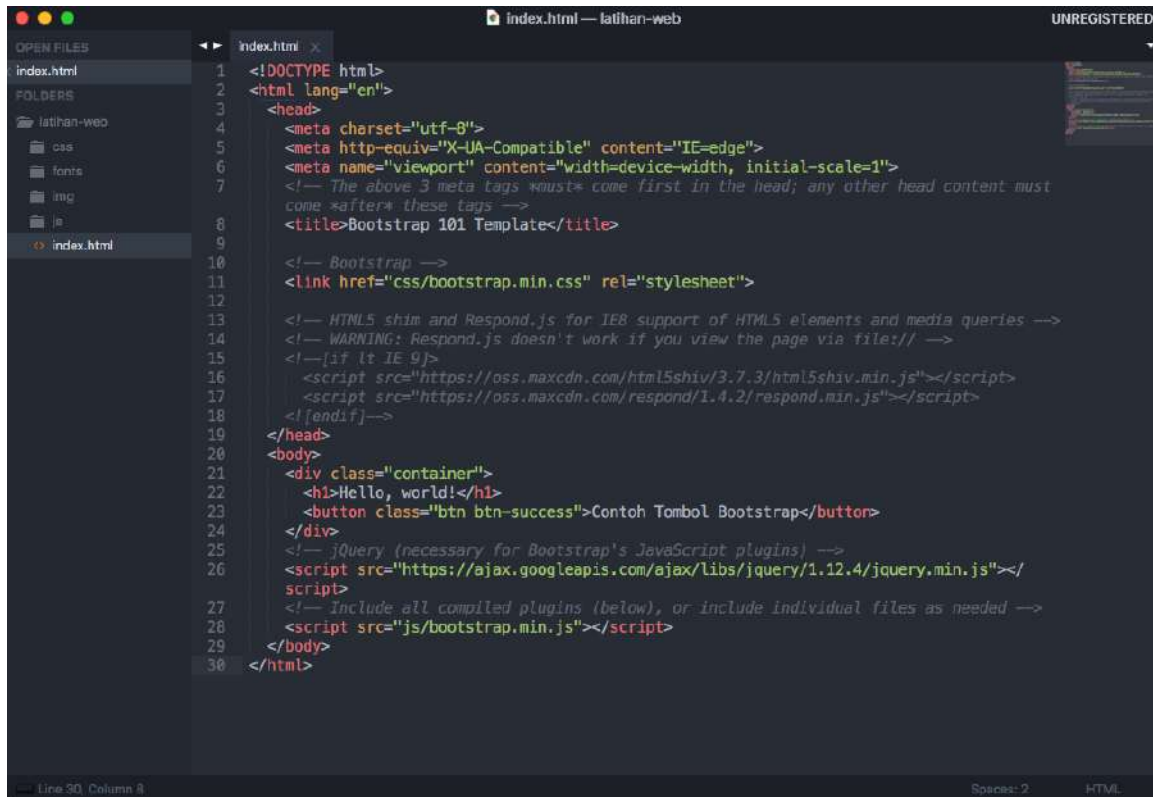


Gambar 29. Struktur folder project

Tambahkan satu folder yaitu "img" digunakan untuk menyimpan gambar yang diperlukan pada pembuatan website.

Kalau sudah selesai membuat folder project, sekarang siapkan text editor favorit kalian, disini saya menggunakan Sublime text 3 yang sudah terinstall beberapa plugin seperti Emmet dan Color picker untuk memudahkan kita dalam pembuatan website.

Buka file index.html folder project kalian di text editor.



Gambar 30. Kode file index.html

Pada coding default index.html memanggil jquery melalui tag <script> secara online, artinya jquery hanya dapat digunakan ketika mendapatkan koneksi internet saja. Agar kita dapat bekerja secara offline copy link berikut ini dan paste ke browser.

<https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js>

Maka terdapat script jquery pada halaman browser kurang lebih dengan tampilan seperti gambar berikut ini.



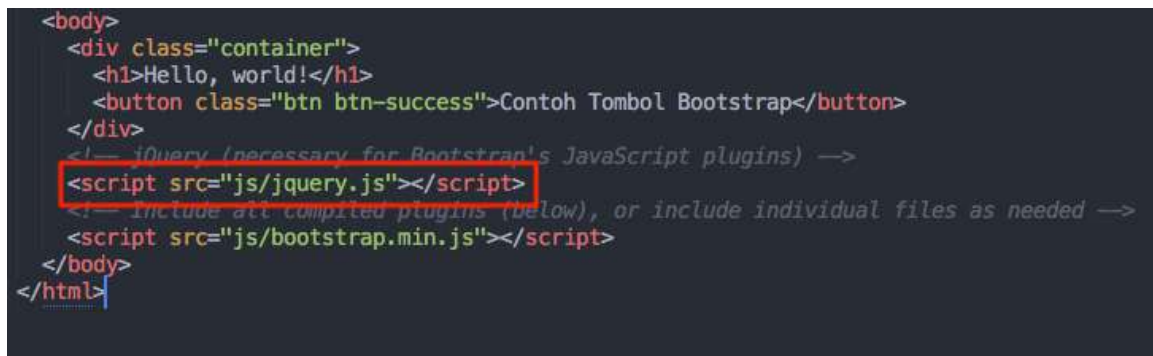
Gambar 31. Script jquery

Buat file baru dengan nama jquery.js didalam folder js pada project kalian, kemudian copy semua coding jquery tersebut dan paste kedalam file jquery.js yang telah dibuat tadi.



Gambar 32. File jquery.js

Sekarang ganti url pada tag script tersebut menuju ke file jquery.js yang telah dibuat tadi.



Gambar 33. Memanggil file jquery

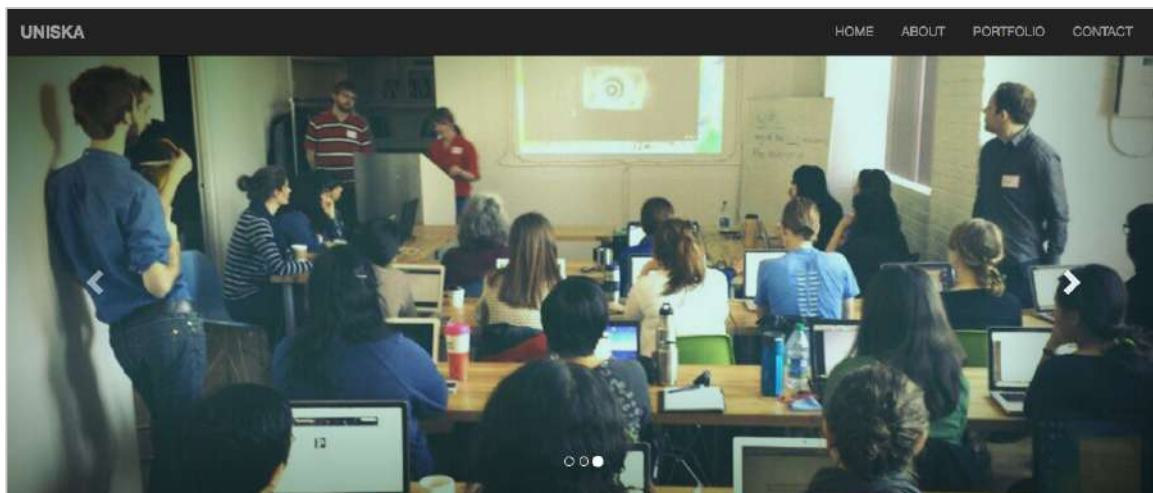
Sekarang kita dapat bekerja secara offline, jquery tetap dapat digunakan tanpa koneksi ke jaringan internet.

Langkah selanjutnya adalah membuat file dengan nama custom.css. Dimana file custom.css di letakan ? karena file ini adalah CSS maka akan kita meletakan pada folder css didalam folder project kita.

Apa fungsi dari file custom.css? tidak semua style pada Bootstrap sesuai dengan yang kita kehendaki, misalnya saya tidak ingin ada margin pada navbar yang membuat jarak terhadap navigasi dan slider/carousel. Maka untuk menghilangkan margin-bottom pada css Bootstrap tersebut saya harus memberikan style ulang pada navbar untuk menghilangkan margin-bottom tersebut. Perhatikan contoh berikut ini!



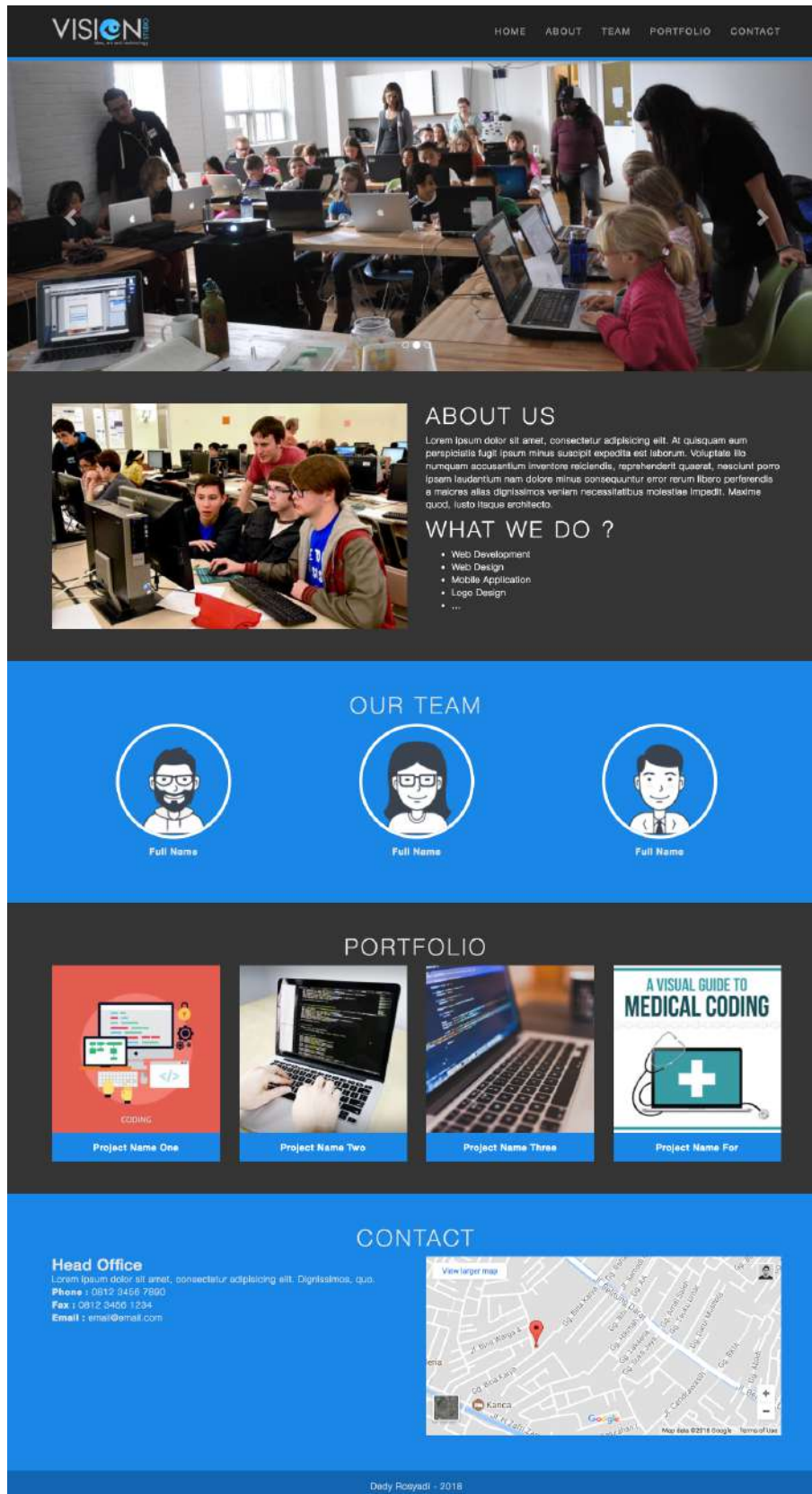
Gambar 34. Navigasi dengan margin



Gambar 35. Navigasi tanpa margin

Cukup jelas ? semoga iya! ☺. Mari lanjutkan coding lagi. Nanti akan saya lampirkan semua coding pembuatan web ini.





Gambar 36. Hasil website yang dibuat

## Coding Bagian Part 1 (Pembuka, memanggil file CSS eksternal)

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <!-- The above 3 meta tags *must* come first in the head; any other head content must come *after* these tags -->
8     <title>Bootstrap 101 Template</title>
9
10    <!-- Bootstrap -->
11    <link href="css/bootstrap.min.css" rel="stylesheet">
12    <link href="css/custom.css" rel="stylesheet">
13
14    <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->
15    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
16    <!--[if lt IE 9]>
17      <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
18      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
19    <![endif]-->
20  </head>
21  <body>
```

Gambar 37. Kode part 1

## Coding Part 2 (Header)

```
22   <header>
23     <nav class="navbar navbar-default navbar-inverse navbar-fixed-top">
24       <div class="container">
25         <!-- Brand and toggle get grouped for better mobile display -->
26         <div class="navbar-header">
27           <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
28             aria-expanded="false">
29             <span class="sr-only">Toggle navigation</span>
30             <span class="icon-bar"></span>
31             <span class="icon-bar"></span>
32             <span class="icon-bar"></span>
33           </button>
34           <a class="navbar-brand" href="#">
35             
36           </a>
37         </div>
38         <!-- Collect the nav links, forms, and other content for toggling -->
39         <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
40           <ul class="nav navbar-nav navbar-right">
41             <li><a href="index.html">HOME</a></li>
42             <li><a href="#">ABOUT</a></li>
43             <li><a href="#">TEAM</a></li>
44             <li><a href="#">PORTFOLIO</a></li>
45             <li><a href="#">CONTACT</a></li>
46           </ul>
47         </div><!-- /.navbar-collapse -->
48       </div><!-- /.container-fluid -->
49     </nav>
50   </header>
```

Gambar 38. Kode part 2

## Coding Part 3 (Carousel / Slider)

```

52 <div id="carousel-example-generic" class="carousel slide" data-ride="carousel">
53 <!-- Indicators -->
54 <ol class="carousel-indicators">
55 <li data-target="#carousel-example-generic" data-slide-to="0" class="active"></li>
56 <li data-target="#carousel-example-generic" data-slide-to="1"></li>
57 <li data-target="#carousel-example-generic" data-slide-to="2"></li>
58 </ol>
59
60 <!-- Wrapper for slides -->
61 <div class="carousel-inner" role="listbox">
62 <div class="item active">
63 
64 </div>
65 <div class="item">
66 
67 </div>
68 <div class="item">
69 
70 </div>
71 </div>
72
73 <!-- Controls -->
74 <a class="left carousel-control" href="#carousel-example-generic" role="button" data-slide="prev">
75 <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
76 <span class="sr-only">Previous</span>
77 </a>
78 <a class="right carousel-control" href="#carousel-example-generic" role="button" data-slide="next">
79 <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
80 <span class="sr-only">Next</span>
81 </a>
82 </div>

```

Gambar 39. Kode part 3

## Coding Part 4 (about)

```

84 <div id="about">
85 <div class="container">
86 <div class="row">
87 <div class="col-lg-6">
88 <!-- 
89 
90 </div>
91
92 <div class="col-lg-6">
93 <h1 class="title">ABOUT US</h1>
94 <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. At quisque eum perspiciatis fugit ipsum minus suscipit
expedita est laborum. Voluptate illo nunquam accusantium inventore reiciendis, reprehenderit queraat, nesciunt porro ipsan
laudantium nam dolore minus consequuntur error rerum libero perferendis a maiores alias dignissimos veniam necessitatibus
molestiae inpedit. Maxime quod, iusto itaque architecto.</p>
95
96 <h1 class="title">WHAT WE DO ?</h1>
97 <ul>
98 <li>Web Development</li>
99 <li>Web Design</li>
100 <li>Mobile Applications</li>
101 <li>Logo Design</li>
102 <li>...</li>
103 </ul>
104 </div>
105 </div>
106 </div>
107 </div>

```

Gambar 40. Kode part 4

## Coding Part 5 (team)

```

109 <div id="team">
110 <div class="container">
111 <h1 class="title text-center color-white">OUR TEAM</h1>
112 <div class="col-lg-4">
113 
114 <div class="team-name">
115 <p class="text-center"><strong>Full Name</strong></p>
116 </div>
117 </div>
118 <div class="col-lg-4">
119 
120 <div class="team-name">
121 <p class="text-center"><strong>Full Name</strong></p>
122 </div>
123 </div>
124 <div class="col-lg-4">
125 
126 <div class="team-name">
127 <p class="text-center"><strong>Full Name</strong></p>
128 </div>
129 </div>
130 </div>
131 </div>
132

```

Gambar 41. Kode part 5

## Coding Part 6 (portfolio)

```

135     <div id="portfolio">
136       <div class="container">
137         <h1 class="title text-center color-white">PORTFOLIO</h1>
138
139         <div class="row">
140           <div class="col-lg-3 col-sm-6 box-portfolio">
141             <a href="#"></a>
142             <div class="portfolio-title">
143               <h3 class="text-center">Project Name One</h3>
144             </div>
145           </div>
146           <div class="col-lg-3 col-sm-6 box-portfolio">
147             <a href="#"></a>
148             <div class="portfolio-title">
149               <h3 class="text-center">Project Name Two</h3>
150             </div>
151           </div>
152           <div class="col-lg-3 col-sm-6 box-portfolio">
153             <a href="#"></a>
154             <div class="portfolio-title">
155               <h3 class="text-center">Project Name Three</h3>
156             </div>
157           </div>
158           <div class="col-lg-3 col-sm-6 box-portfolio">
159             <a href="#"></a>
160             <div class="portfolio-title">
161               <h3 class="text-center">Project Name Four</h3>
162             </div>
163           </div>
164         </div>
165       </div>
166     </div>

```

Gambar 42. Kode part 6

## Coding Part 7 (Contact & Footer)

```

168     <div id="contact">
169       <div class="container">
170         <h1 class="title text-center">CONTACT</h1>
171
172         <div class="row">
173           <div class="col-md-6">
174             <h3>Head Office</h3>
175             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Dignissimos, quo. <br>
176             <strong>Phone :</strong> 0812 3456 7890 <br>
177             <strong>Fax :</strong> 0812 3456 1234 <br>
178             <strong>Email :</strong> email@email.com
179           </div>
180           <div class="col-md-6">
181             <!-- <h3>Head Office</h3> -->
182             <div class="col-md-6">
183               <iframe src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3983.176946897008!2d114.57339461475779!3d-3.3063610975929
184                 85!2m3!1f0!3f0!3f0!3m2!1!1024!2!768!4f13.1!3m3!1m2!1s0x0a340x0!2zM0KwMTgnMjIuOSJITDEuNKGWzQmMzIuNSJF!5e0!3m2!1sen!2sid!4v15
185                 02117678536" width="100%" height="280" frameborder="0" style="border:0" allowfullscreen=""></iframe>
186             </div>
187           </div>
188         </div>
189
190       <div class="container">
191         <div class="text-center">Dedy Rosyadi - 2018</div>
192       </div>
193     </div>
194
195

```

Gambar 43. Kode part 7

## Coding Part 8 (Penutup, memanggil file JS)

```

196
197
198     <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
199     <script src="js/jquery.js"></script>
200     <!-- Include all compiled plugins (below), or include individual files as needed -->
201     <script src="js/bootstrap.min.js"></script>
202
203   </body>
204 </html>

```

Gambar 44. Kode part 8



## Coding Part 9 (file custom.css)

```
1
2
3 * {
4   font-weight: 200;
5   letter-spacing: 1px;
6 }
7
8 body {
9   padding-top: 80px;
10 }
11
12 h3 {
13   margin: 0;
14 }
15
16 .color-white {
17   color: #fff;
18 }
19
20 .img-center {
21   margin: 0 auto;
22 }
23
24 .navbar {
25   min-height: 80px;
26   border-radius: 0;
27   margin-bottom: 0;
28   border-bottom: solid 6px #0007c0;
29   box-shadow: 0 2px 10px 0 #1a1a1a;
30 }
31
32 .navbar-brand {
33   padding: 0 15px;
34   height: 80px;
35   line-height: 80px;
36 }
37
38 .navbar-brand-img {
39   margin-top: 20px;
40   max-width: 150px;
41 }
42
```

Gambar 45. Kode part 9(1)

```
41
42 .navbar-toggle {
43   margin-top: 23px;
44   padding: 9px 10px !important;
45 }
46
47 @media (min-width: 768px) {
48   .navbar-nav > li > a {
49     padding-top: 26.5px;
50     padding-bottom: 26.5px;
51     line-height: 27px;
52     font-weight: 400;
53     letter-spacing: 2px;
54   }
55 }
56
57 h1.title {
58   font-size: 2.5em;
59   font-weight: 300;
60   margin: 0;
61   margin-bottom: 10px;
62   letter-spacing: 5px;
63 }
64
65 #about {
66   background: #353535;
67   color: #fff;
68 }
69
70 #about img, #portfolio img {
71   width: 100%;
72 }
73
```

Gambar 46. Kode part 9(2)

```

73
74
75 #about, #portfolio, #team, #contact {
76   padding: 50px 0;
77 }
78
79 #team {
80   background: #0087e6;
81 }
82
83 }
84 #team_img {
85   width: 180px;
86   border: solid 5px #fff;
87 }
88
89 }
90 .team-name {
91   padding: 10px;
92   color: #e6e6e6;
93 }
94
95 #portfolio {
96   background: #353535;
97 }
98
99 .portfolio-title {
100   padding: 15px;
101   background: #0087e6;
102   height: 100%;
103 }
104
105 .portfolio-title h3 {
106   font-size: 1em;
107   color: #fff;
108 }

```

Gambar 47. Kode part 9(3)

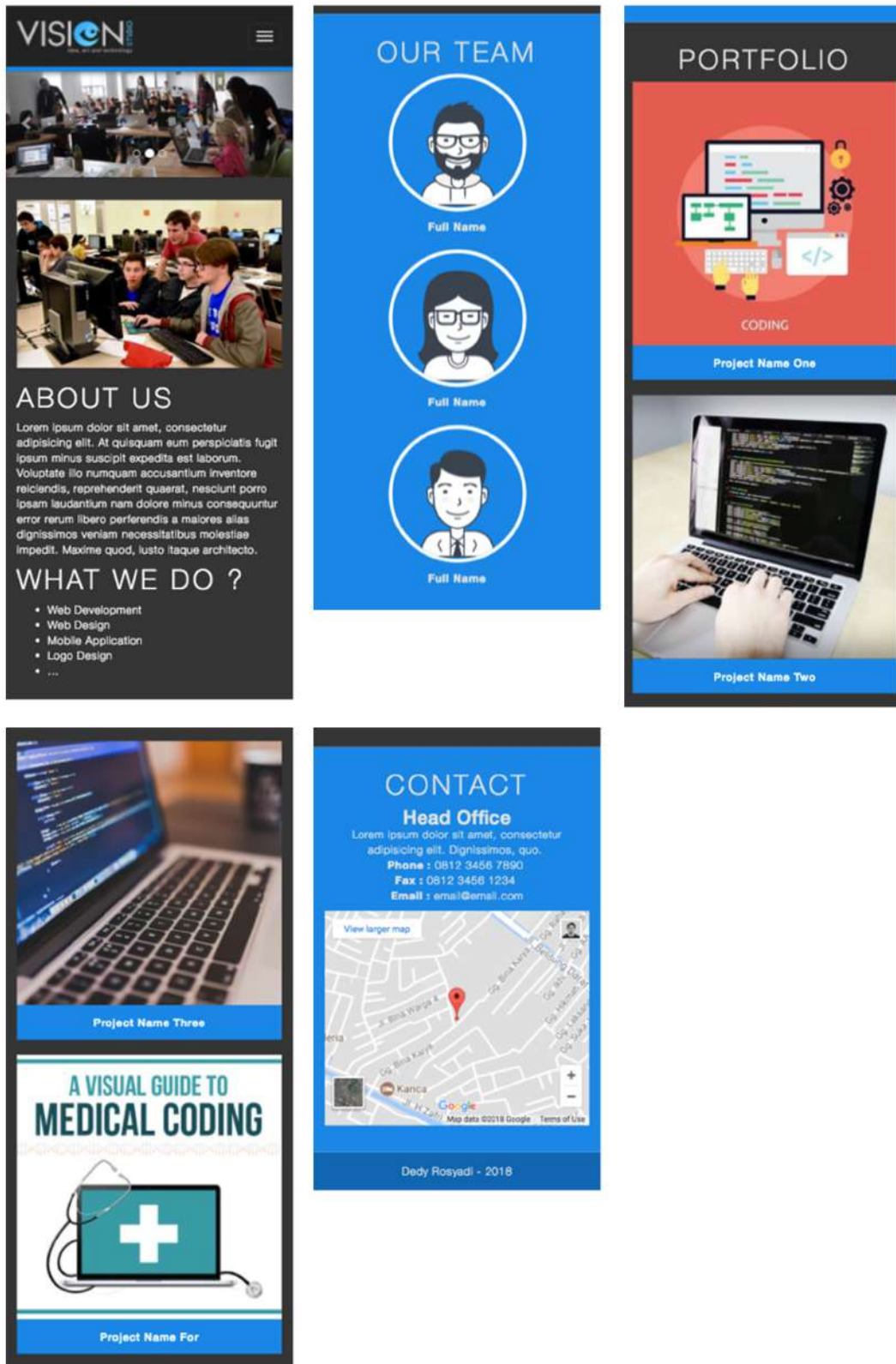
```

109
110 .portfolio h1 {
111   font-size: 1em;
112   color: #fff;
113 }
114
115 #contact {
116   color: #e6e6e6;
117   background: #0087e6;
118 }
119
120 footer {
121   padding: 15px 0;
122   background: #0087e6;
123   color: #e6e6e6;
124   font-weight: 500;
125 }
126
127
128 @media only screen and (max-width : 640px) {
129   .box-portfolio {
130     margin-bottom: 20px;
131   }
132
133   #about_img {
134     margin-bottom: 20px;
135   }
136
137   #about, #portfolio, #team, #contact {
138     padding: 30px 0;
139   }
140
141
142   #contact {
143     text-align: center;
144   }
145 }

```

Gambar 48. Kode part 9(4)

Kode diatas merupakan kode yang digunakan untuk membuat website sederhana yang dibuat hanya dengan menggunakan HTML, CSS dan JS/JQUERY yang kemudian di tampilkan penuh melalui browser. Sekarang kita akan mencoba membuka tampilan website yang telah kita buat tadi menggunakan tools dari Google Chrome. Berikut ini adalah hasil tampilan website yang telah berhasil dibuat.



Gambar 49. Hasil tampilan mobile

## **Daftar Pustaka**

Abdul Kadir (2013). From Zero To a Pro HTML5. Yogyakarta : Andi Offset

Andre Pratama (2016). HTML Uncover – Panduan Belajar HTML untuk Pemula

Andre Pratama (2016). CSS Uncover – Panduan Belajar CSS untuk Pemula

<https://www.w3schools.com/>

<http://www.html5doctor.com/>

<http://www.getbootstrap.com/>

<http://www.duniailkom.com/>