

Web Design

HTML, CSS & Javascript

Oleh :

Sophian

PROLOG

Buku ini untuk materi mata kuliah Web Design LP3I Kaltara, yang konsentrasi untuk mempelajari HTML, CSS dan Javascript, baik anda yang tidak tahu sama sekali apa itu HTML, CSS, dan Javascript hingga anda yang ingin mempelajari secara lanjut HTML, CSS dan Javascript.

Pembahasan dalam buku ini menitik beratkan pada study kasus, contoh penggunaan setiap kode HTML dan CSS dan tidak akan menjelaskan secara terperinci tag-tag yang terdapat/disediakan dalam HTML atau property-property yang ada dalam CSS dan Javascript yang akan membosankan anda.

Andapun akan diajak untuk mengkonversi sebuah desain web menjadi sebuah file HTML dan CSS secara utuh, setiap langkah dijelaskan sesederhana mungkin agar mudah dicerna dan dipahami.

Watch the sign!

Dalam buku ini, anda akan menemukan beberapa tanda seperti berikut



Catatan, berisi catatan singkat tentang apa yang harus anda perhatikan dan lakukan atau sebuah peringatan.



Informasi, berisi informasi singkat yang menjelaskan poin-poin tertentu dalam pembahasan setiap materi

1. Apa yang terdapat dalam box ini adalah kode dari pembahasan. Kode HTML dan CSS ditulis di dalam box ini.
2. Kadang setiap baris kode yang panjang memiliki penomoran seperti box ini.
3. Baris kode yang tidak memiliki nomor berarti ditulis satu baris dengan baris sebelumnya (seperti baris ini dan baris setelah nomor 1).

DAFTAR ISI

PROLOG	I
Watch the sign!	i
DAFTAR ISI	II
BAB 1 PERSIAPAN	2
1.1. Web Browser	2
1.2. Text Editor	5
BAB 2 PENGENALAN HTML	7
2.1. Apa itu HTML?	7
2.2. Membuat Website dalam 1 Menit	8
2.3. Pengenalan Tag HTML	9
2.4. Struktur file HTML	10
BAB 3 MEMUAT GAMBAR	12
3.1. Mengenal Atribut HTML	12
3.2. Cara penulisan lokasi file	15
BAB 4 MEMBUAT LINK	16
4.1. Link Standar	16
4.2. Link Email	20
BAB 5 HEADING/PENJUDULAN	21
BAB 6 MEMBUAT DAFTAR/LIST	23
6.1. Ordered List	23
6.2. Unordered List	24
6.3. Definition List	25
6.4. List/Daftar bersarang	25
BAB 7 MEMAHAMI TAG DIV (DIVISION)	27
7.1. Pakai ID atau Class?	28
7.2. Memahami hubungan Child, Parent dan Siblings	29
BAB 8 FORM	30
Control-control Form	30

BAB 9 TABEL	36
9.1. Struktur tabel yang dianjurkan	37
9.2. Kesalahan dalam penggunaan <table>	38
BAB 10 STUDI KASUS 1- KONVERSI DESAIN KE HTML	39
10.1. Pesiapan	39
10.2. Wrapper	40
10.3. Header	40
10.4. Content	41
10.5. Sidebar	43
10.6. Footer	44
BAB 11 VALIDASI MARKUP HTML	47
BAB 13 CASCADING STYLE SHEET (CSS)	51
13.1. Pengenalan CSS	51
13.2. Penulisan CSS	53
BAB 14 BOX-MODEL	55
14.1. Margin	55
14.2. Padding	56
14.3. CSS-Shorthand	57
14.4. Border	59
BAB 15 TYPOGRAPHY	61
Apa itu Sans-serif dan Serif	62
BAB 16 CSS-IMAGE	65
16.1. background-image	65
16.2. background-repeat	66
16.3. background-position	67
16.4. background-image merge	69
BAB 17 FLOATING	70
17.1. Permasalahan pada floating	71
17.2. CSS Reset	73
BAB 18 POSITIONING	74
18.1. Static	74
18.2. Relative	75
18.3. Absolute	76
18.4. Fixed	79

BAB 19 PSEUDO-CLASS	81
19.1. Pseudo-Class untuk Link/Anchor	81
19.2. Pseudo-Class :first-child dan :last-child	82
BAB 20 STUDIKASUS 2 – PEMBERIAN STYLE/CSS	84
20.1. CSS Reset	85
20.2. Body	85
20.3. Wrapper	85
20.4. Header	86
20.5. Menu	87
20.6. Form	88
20.7. Daftar Artikel	89
20.8. Artikel	89
20.9. Sidebar	91
20.10. Footer	92

Bagian I

HTML

Bab 1

Persiapan

Sebelum kita mempelajari HTML dan CSS, ada beberapa persiapan yang perlu dilakukan. Seperti penginstallan aplikasi-aplikasi yang dibutuhkan dan aplikasi-aplikasi pendukung dalam mempelajari HTML dan CSS.

1.1. Web Browser

Web browser adalah perangkat utama yang akan kita gunakan untuk menampilkan halaman web yang pada dasarnya terbuat dari HTML dan CSS. Saya yakin di setiap komputer telah terinstall Web Browser bawaan seperti Internet Explorer (Windows), Safari (Mac) dan Firefox (Linux Ubuntu).

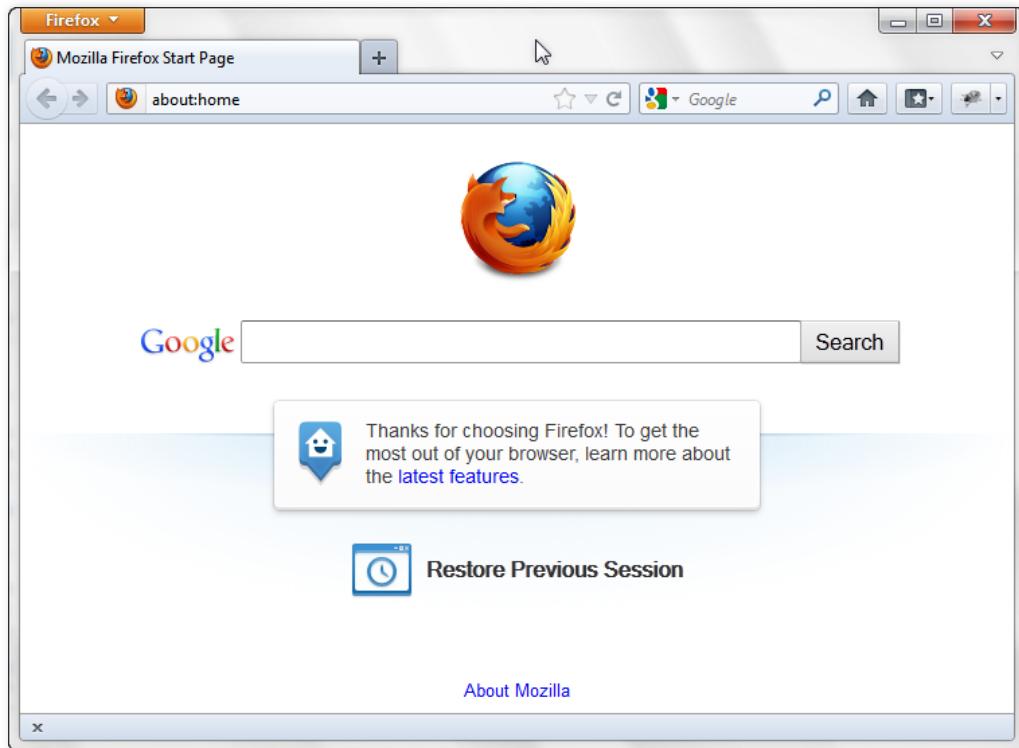
Setiap browser memiliki perbedaan dalam hal menampilkan halaman web dan fitur-fitur yang didukung dalam HTML dan CSS. Boleh jadi halaman web yang anda buat ditampilkan benar pada salah satu browser namun acak-acakan pada browser lainnya. Untuk itu perlu beberapa browser yang terinstall dalam komputer anda untuk menguji penampilan website yang dibuat.

Isu-isu kompatibilitas ini akan anda temui ketika mempelajari CSS nantinya yang berhubungan dalam penampilan halaman web.

Berikut ini adalah beberapa browser yang dapat anda download dari Internet serta review singkat.

Firefox

Web browser ini bisa dikatakan web browser yang paling disenangi oleh para developer web karena kekayaannya dalam hal Add-on¹. Anda dapat mendownloadnya di <http://firefox.com>



Gambar 1 Mozilla Firefox 9

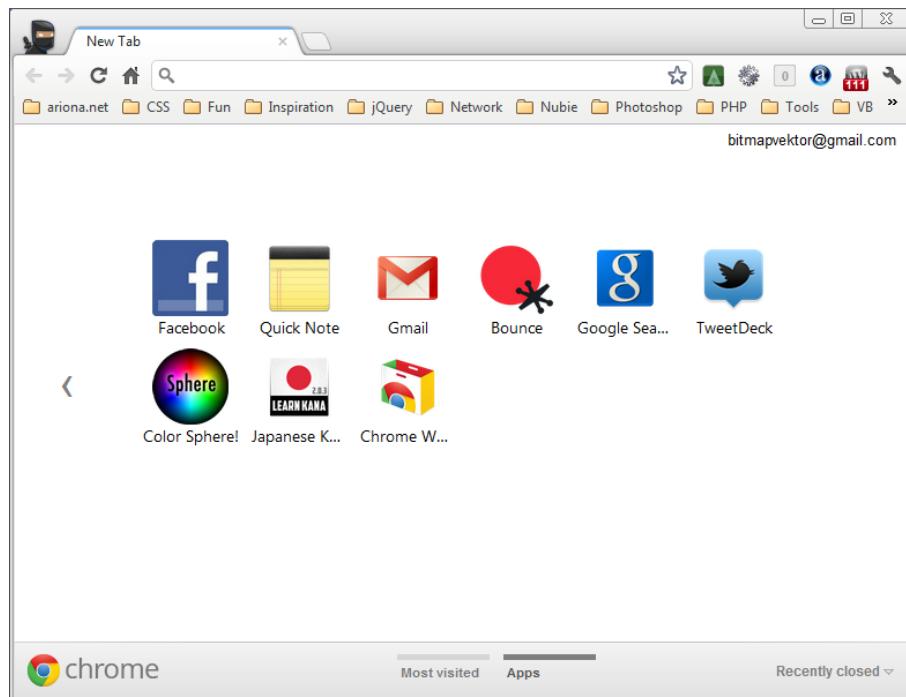
Google Chrome

Google Chrome adalah web browser besutan Google yang memiliki beberapa keunggulan. Chrome bisa dikatakan browser yang sangat ringan, cepat² dan kaya akan aplikasi/Add-on. Selain itu Chrome menggunakan engine³ Webkit, engine browser yang rata-rata mendukung fitur-fitur terbaru dari teknologi HTML dan CSS. Anda dapat mendownloadnya di <http://google.com/chrome>

¹ Add-on/Plugin adalah Aplikasi tambahan yang dapat dipasang sehingga memperkaya fitur software yang bersangkutan

² Kecepatan loading web tergantung dari kecepatan internet yang anda gunakan

³ Engine/Mesin adalah kode utama dalam aplikasi untuk menampilkan halaman web contohnya Mozilla untuk firefox, Webkit untuk Google Chrome dan Safari.

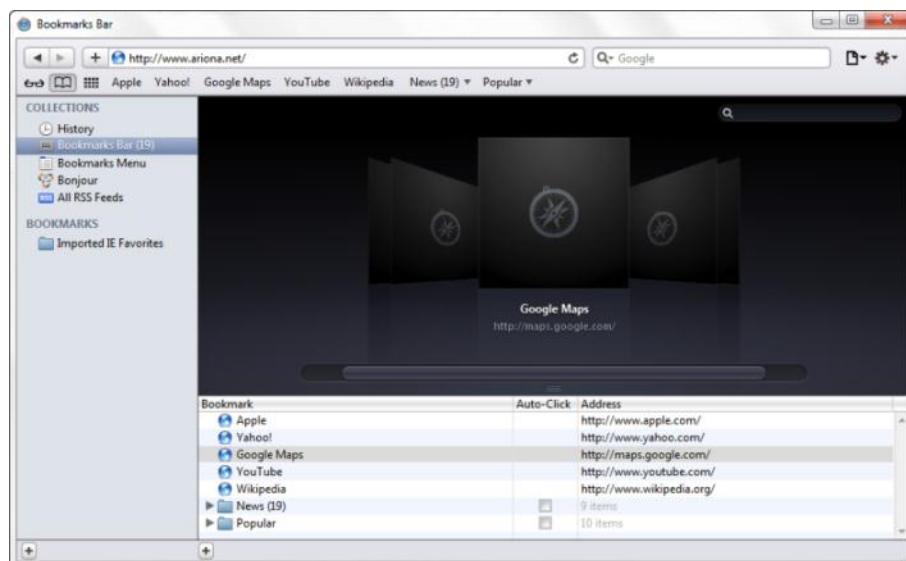


Gambar 2 Google Chrome

Safari

Safari adalah web browser dari Apple. Sama halnya dengan Chrome, safari menggunakan engine webkit sehingga mendukung fitur-fitur terbaru HTML dan CSS.

Browser ini tersedia untuk Mac dan Windows saja. <http://www.apple.com/safari>

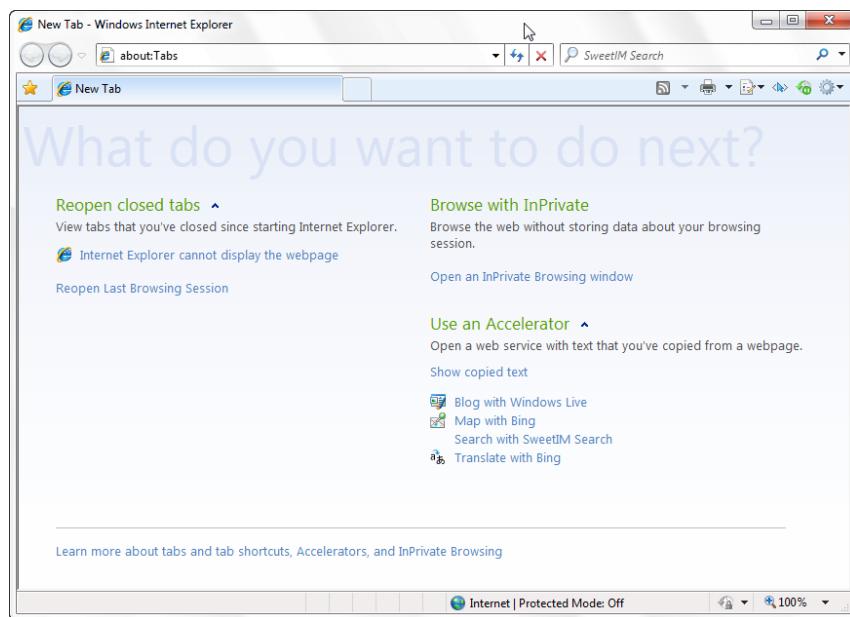


Gambar 3 Safari

Internet Explorer

Browser yang pasti sudah terinstall pada komputer dengan sistem operasi windows. Jika anda masih menggunakan Internet Explorer versi 6, 7 dan 8, saya anjurkan untuk mendownload versi terbaru yakni Internet Explorer 9/10 (untuk Windows Vista/7).

Penggunaan Internet Explorer dalam pengujian halaman web sangat tidak dianjurkan (terutama versi 7 ke bawah) karena dukungannya yang kurang serta proses penampilan halaman web yang terkadang menjadi berantakan. Namun karena pengguna Internet explorer di dunia masih mendominasi, mau tidak mau kita harus menggunakan Internet Explorer.



Gambar 4 Internet Explorer

1.2. Text Editor

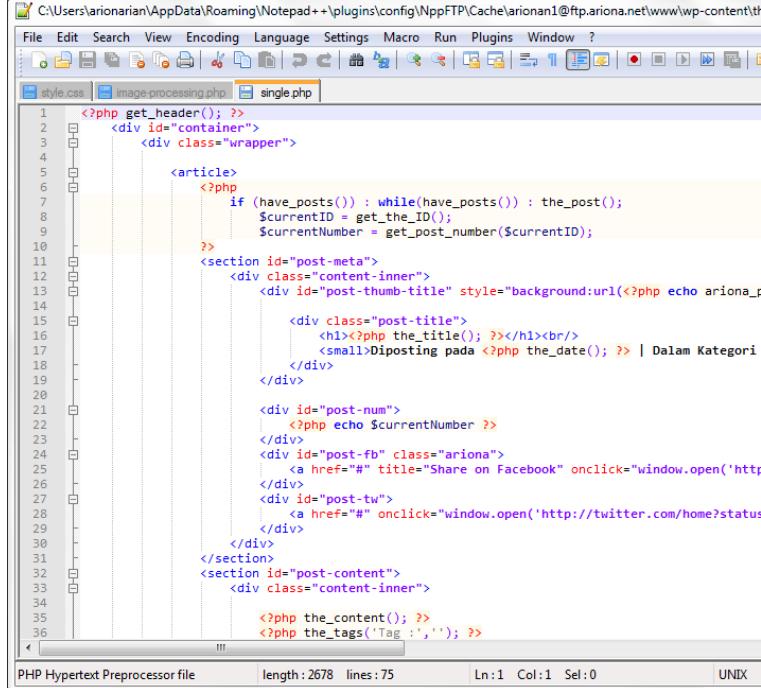
Untuk membuat halaman Web anda memerlukan sebuah text editor. Jika anda mempunyai Adobe Dreamweaver, saya sarankan untuk tidak menggunakannya, Mengingat anda masih dalam tahap belajar.

Text editor yang akan kita gunakan adalah text editor ringan namun penuh dengan fitur yang sangat membantu dalam proses *Coding(Penulisan Kode)* HTML dan CSS.

Notepad++

Tersedia untuk Windows, Notepad++ memiliki beragam fitur yang sangat mendukung para programmer. Kemampuannya untuk memperkaya diri dengan pluginpun menjadi kelebihan lainnya dengan text editor sejenisnya.

Anda dapat mendownload Notepad++ di <http://notepad-plus-plus.org>



```

1  <?php get_header(); ?>
2  <div id="container">
3  <div class="wrapper">
4
5
6  <article>
7      <?php
8          if (have_posts()) : while(have_posts()) : the_post();
9              $currentID = get_the_ID();
10             $currentNumber = get_post_number($currentID);
11         ?>
12         <section id="post-meta">
13             <div class="content-inner">
14                 <div id="post-thumb-title" style="background:url(<?php echo ariona_p
15                     <div class="post-title">
16                         <h1><?php the_title(); ?></h1><br/>
17                         <small>Diposting pada <?php the_date(); ?> | Dalam Kategori
18                     </div>
19
20                     <div id="post-num">
21                         <?php echo $currentNumber ?>
22                     </div>
23                     <div id="post-fb" class="ariona">
24                         <a href="#" title="Share on Facebook" onclick="window.open('http://www.facebook.com/sharer.php?u=' + escape(location.href))" data-fb-share=""><?php echo $currentNumber ?>
25                     </div>
26                     <div id="post-tw">
27                         <a href="#" onclick="window.open('http://twitter.com/home?status=' + escape(location.href))" data-tw-share=""><?php echo $currentNumber ?>
28                     </div>
29                 </div>
30             </section>
31             <section id="post-content">
32                 <div class="content-inner">
33                     <?php the_content(); ?>
34                     <?php the_tags('Tag : ', '') ; ?>
35
36

```

Gambar 5 Notepad++

Bab 2

Pengenalan HTML

2.1. Apa itu HTML?

HTML merupakan singkatan dari **HyperText Markup Language** Artinya adalah bahasa markup (penanda) berbasis text atau bisa juga disebut sebagai formatting language (bahasa untuk memformat), Jadi sudah jelas bahwa HTML bukanlah bahasa pemrograman, melainkan bahasa markup/formatting. Dokumen HTML disimpan dalam format teks reguler dan mengandung tag-tag yang memerintahkan web browser untuk mengeksekusi perintah-perintah yang dispesifikasikan.

Pada umumnya file HTML berekstensi **.htm** atau **.html**. Nama file/dokumen HTML boleh kombinasi antara huruf kecil dan huruf besar. Tetapi umumnya, di internet suatu file HTML ditulis dengan menggunakan huruf kecil, dikarenakan ada beberapa sistem operasi yang membedakan antara huruf kecil dan huruf besar yang bertujuan untuk mengidentifikasi antara sintaks dan suatu kalimat / statement.

2.2. Membuat Website dalam 1 Menit

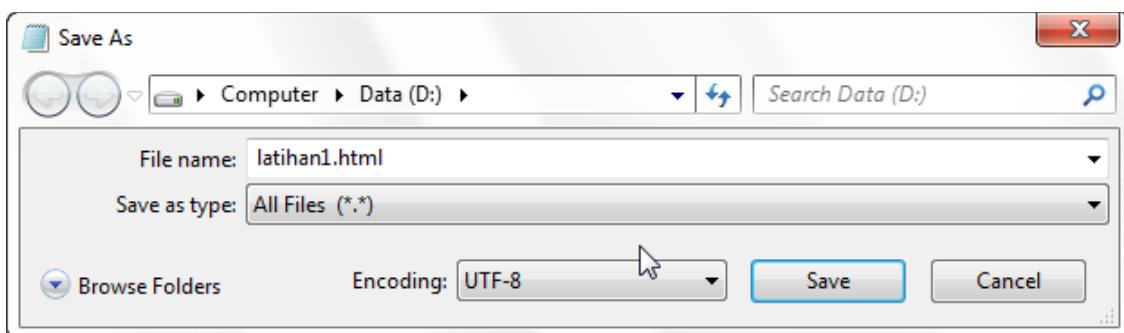
Sudah siap membuat website? Kita mulai membuat website hanya dalam 1 menit!

Bukalah Notepad (*All Programs > Accessories > Notepad*)

2. Ketikkan teks berikut :

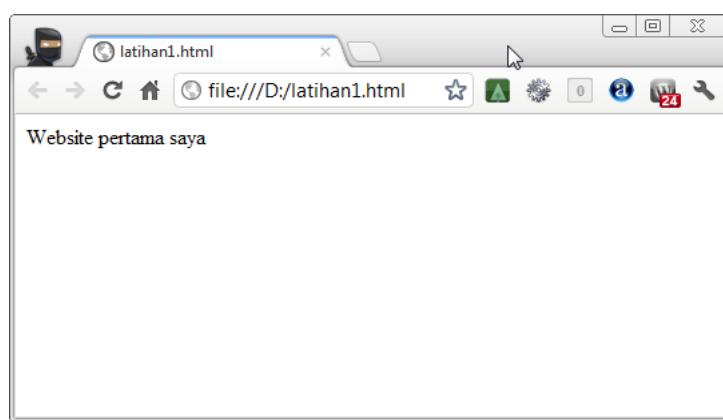
```
Website pertama saya
```

3. Pilih menu File > Save As
4. Beri nama “latihan1.html” (tanpa tanda kutip)
5. Pada bagian Save as type, pilih *All Files (*.*)* Lalu klik tombol save.



Gambar 6 Save As type

6. Sekarang klik dua kali file HTML yang telah anda buat, (atau Klik dan Drag ke Web Browser anda, disini saya menggunakan Google Chrome)



Gambar 7 Website pertama anda

Demonstrasi di atas hanyalah langkah untuk membuat file HTML, *mudah bukan?*

2.3. Pengenalan Tag HTML

Seperti yang saya sebutkan sebelumnya, bahwa HTML adalah bahasa Markup, artinya bahasa HTML hanya digunakan untuk me-markup suatu dokumen. Sebagai penyederhanaan kita analogikan dengan pembuatan dokumen di Microsoft Word atau aplikasi Word Processing lainnya.

Ketika membuat sebuah dokumen word, biasanya kita melakukan *formatting* pada teks. seperti menebalkan teks, memiringkan, memberi garis bawah, Membuat Penjedulan (Heading), membuat *list* (Daftar) dan lain sebagainya. Pada HTML, *formatting* ini bisa kita sebut dengan Proses *Markup* dan akan dilakukan dengan menggunakan tag HTML.

Misalnya, untuk mem-format suatu teks menjadimiring kita tambahkan tag `` atau `<i>`. Edit file sebelumnya, menjadi seperti berikut :

```
Website <em>pertama</em> saya
```

Dan ketika dibuka kembali pada browser, anda dapat melihat kata pertama menjadi miring.

` .. ` disebut dengan Tag HTML, yang digunakan untuk me-markup(memformat) dokumen HTML. Perhatikan bahwa setiap tag memiliki pasangan yakni tag penutupnya.

Artinya hanya teks yang berada di dalam tag tersebut yang akan diberi format miring. Tag penutup ditandai dengan penambahan tanda slash (/) sebelum nama tagnya. Berikut ini adalah anatomi dari tag HTML :

```
<namatag atribut="nilai Atribut">
    Isi atau Konten
</namatag>
```

Setiap tag HTML bisa menjadi "isi" dari tag HTML lainnya, untuk membuat suatu hierarki dari dokumen HTML.

Dalam setiap tag HTML bisa disisipkan berbagai atribut yang berfungsi untuk menampung informasi-informasi tertentu, misalnya atribut id dan class yang berfungsi untuk memberi nama suatu tag sebagai referensi CSS atau javascript nantinya.

Sekarang bagaimana jika teks yang dimiringkan tersebut ingin kita tebalkan? Untuk menebalkan teks anda dapat menggunakan tag ``atau``, menjadi seperti berikut:

```
Website <strong><em>pertama</em></strong> saya
```

2.4. Struktur file HTML

Suatu dokumen HTML juga mempunyai aturan dalam penulisannya, ada beberapa tag yang harus anda tuliskan dan sudah menjadi ketentuan. Jika kita perbaiki latihan kita sebelumnya menjadi dokumen HTML yang benar maka kodennya menjadi seperti berikut:

```
1.  <!DOCTYPE HTML>
2.  <HTML>
3.      <head>
4.          <title>Judul File HTML</title>
5.      </head>
6.
7.      <body>
8.          Website <strong><em>pertama</em></strong> saya
9.      </body>
10. </HTML>
```

Penjelasannya dari tiap baris kode tersebut.

```
<!DOCTYPE HTML>
```

Adalah Tag awal dari setiap dokumen HTML, tag ini berfungsi untuk menginformasikan pada browser bahwa dokumen yang sedang dibuka adalah dokumen HTML. Tag ini perlu dicantumkan disetiap dokumen HTML yang akan anda buat.

```
<HTML> ... </HTML>
```

Awal dari isi dokumen HTML dimulai dari sini, semua kode HTML yang akan anda buat akan ditulis di dalam tag ini, perhatikan juga bahwa setiap tag harus memiliki tag penutup.

```
<head> ... </head>
```

Tag Head akan digunakan untuk menyimpan berbagai informasi tentang dokumen HTML, lebih detailnya akan kita temui pada bab selanjutnya. Apa yang terdapat di dalam tag ini, tidak akan ditampilkan secara langsung pada web browser.

```
<title> ... </title>
```

Salah satu contoh informasi yang terdapat di dalam tag Head adalah title. Title akan menampilkan teks pada judul browser atau pada tab browser.

```
<body> ... </body>
```

Jadi, apa yang ingin anda tampilkan pada browser akan ditulis di dalam tag ini, tag body merupakan tag pembuka dari badan dokumen HTML.

Mudah dimengerti bukan? Lalu apakah penulisan tag-tag HTML harus seperti contoh di atas? Seperti penambahan tab/indentasi pada setiap tag?. Sebenarnya Anda bisa saja menuliskan kode HTML seperti ini :

```
<!DOCTYPE      HTML><HTML><head><title>Judul      File      HTML</title></head><body>Website
<i>pertama</i> saya</body></HTML>
```

Dan yang anda dapatkan dalam browserpun sama, namun anda akan **kesulitan** membaca dokumen HTML tersebut. Kalau saja dokumen HTML yang anda buat hanya 10 baris mungkin hal ini tidak akan menjadi masalah, namun bagaimana jika 1000 baris?
Repot kan ?!

Untuk itu kita menambahkan indentasi, disetiap isi (konten) dari suatu tag. Misalnya :

```
<head>
  <title>Judul File HTML</title>
</head>
```

Ini menandakan bahwa tag title berada di dalam tag head. Selain itu kita juga bisa mengontrol, apakah suatu tag sudah ditutup atau belum.

Bab 3

Memuat Gambar

Setelah anda mengetahui struktur dari sebuah file HTML dan apa saja yang harus ditulis pertama kali. Selanjutnya kita akan mempelajari cara memuat / menampilkan sebuah gambar dan memahami apa itu atribut HTML.

Untuk memuat sebuah gambar pada dokumen HTML, kita akan menggunakan tag ``. Lalu dimana kita meletakkan gambar tersebut? Di dalam tag `` kah?

3.1. Mengenal Atribut HTML

Setiap tag HTML dapat memiliki satu atau lebih atribut, atribut ini berfungsi untuk menyimpan informasi yang berkaitan dengan tag tersebut.

Pada tag `` kita akan menggunakan atribut `src` untuk menyimpan lokasi gambar :

```
<img src>
```

Selanjutnya untuk memasukkan nilai pada atribut, kita gunakan operator sama dengan (=) diikuti dengan lokasi gambar yang diapit oleh tanda kutip.

```

```

Tag img termasuk ke dalam tag spesial dan tidak memiliki tag penutup (Disebut juga sebagai *Self Closed Tag*), seperti tag-tag lainnya. Ini dikarenakan tag ini tidak memiliki konten seperti :

```
<title>Ini adalah Konten tag title</title>
```

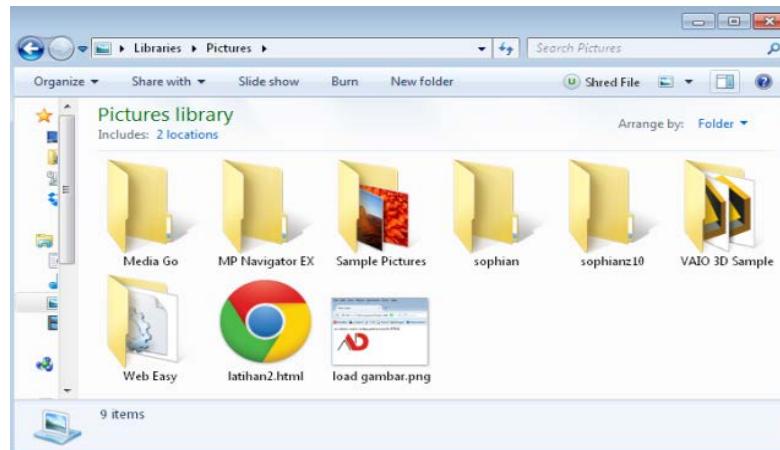
Sebagai pengganti, kita tambahkan tanda slash (/) sebelum kurung tutup :

```

```

Sekarang saatnya anda mencoba memuat gambar pada file HTML.

1. Buatlah file HTML baru dengan nama latihan2.html
2. Carilah sebuah gambar dan simpan satu folder dengan file HTML anda.



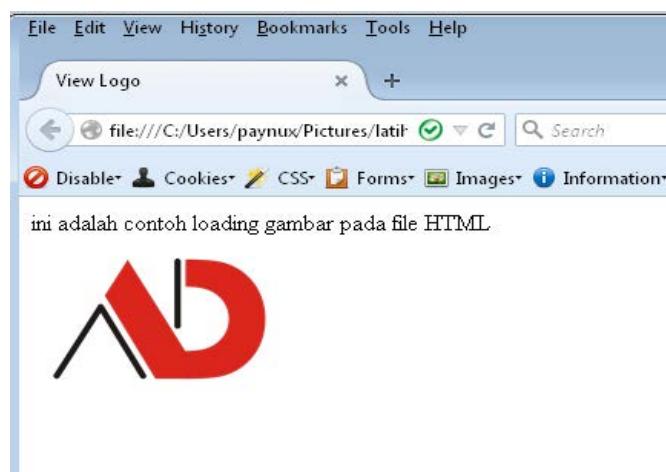
Gambar 8 Penempatan gambar sesuai dengan lokasi file HTML

3. Ketikkanlah kode HTML berikut :

```

1. <!DOCTYPE HTML>
2. <HTML>
3.   <head>
4.     <title>View Logo</title>
5.   </head>
6.
7.   <body>
8.     <p>ini adalah contoh pemuatan gambar pada file
    HTML<p>
9.     
10.   </body>
11. </HTML>
```

4. Buka pada browser, anda akan melihat gambar telah termuat pada file HTML anda.



Gambar 9 Gambar ditampilkan pada browser

Setiap tag img, selain memiliki atribut src untuk menyimpan lokasi gambar, anda juga harus menyertakan atribut alt yang akan digunakan sebagai teks alternatif ketika gambar tidak berhasil dimuat atau gambar hilang.

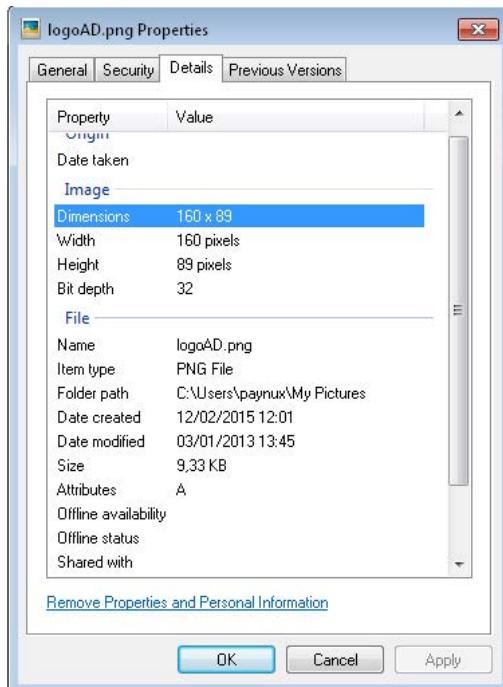
```

```

Dan atribut lainnya adalah atribut width dan height, yang berfungsi untuk menentukan lebar dan tinggi dari gambar tersebut. Memang hal ini tidak perlu dilakukan karena gambar secara otomatis akan terload dengan ukuran sebenarnya, namun hal ini sangat dianjurkan untuk mempercepat proses pemenuaan gambar.

```

```



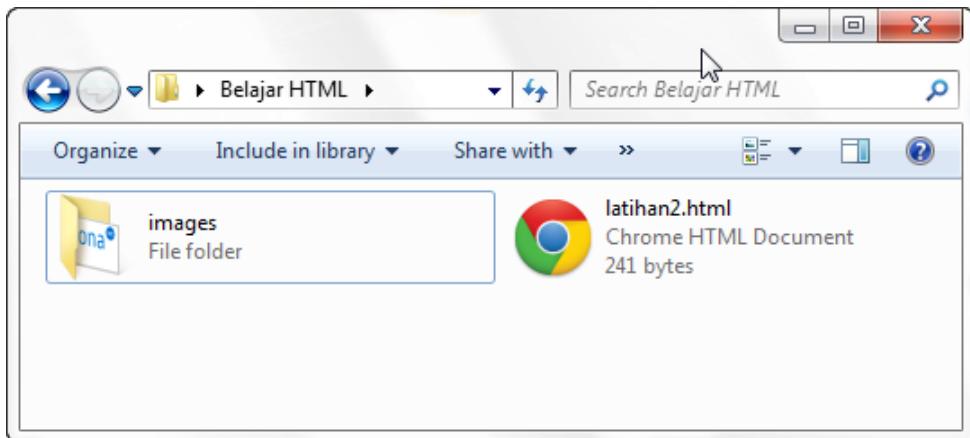
Anda tidak perlu memberi tanda kutip pada nilai lebar dan panjang. Nilai ini bisa anda dapatkan dengan mengklik kanan gambar tersebut dan pilih menu properties. Pada tab details akan anda temui ukuran gambar tersebut.

Dengan penambahan atribut alt, ketika gambar yang dimaksud hilang/tidak/gagal termuat maka akan muncul sebuah icon broken file, dan isi dari atribut alt akan ditampilkan disana.

Gambar 10 File Properties gambar

3.2. Cara penulisan lokasi file

Jika anda menyimpan gambar tersebut pada sebuah folder seperti berikut.



Gambar 11 Menyimpan gambar pada folder images

Maka yang perlu anda lakukan adalah menambahkan nama folder tersebut diikuti dengan tanda *slash* (/) dan nama file gambar yang akan dimuat:

```

```

Jika dalam folder tersebut terdapat folder lagi, dan gambar yang ingin anda muat ada di dalamnya maka penulisannya menjadi seperti berikut :

```

```

Selain penggunaan lokasi gambar seperti di atas, anda juga bisa menampilkan gambar yang sudah terdapat di internet, penulisan seperti ini disebut dengan *hotlinking* misalnya:

```

```

Penggunaan hotlinking sangat tidak dianjurkan karena gambar yang dimuat bukanlah milik kita dan tentunya juga akan merugikan si pemilik gambar.

Bab 4

Membuat Link

Yang sering anda temui dalam sebuah website adalah link. Link ini akan membuat konten atau *elemen HTML*⁴ dapat di klik dan akan mengarahkan/membawa anda ke halaman web lainnya. Biasanya suatu link ditampilkan berwarna biru dan bergaris bawah (selama belum diberi style).

4.1. Link Standar

Link atau biasa disebut dengan anchor (pengait) dapat dibuat dengan menambahkan tag `<a>` pada teks yang ingin kita buat menjadi link.

Klik `<a>disini` untuk mendownload

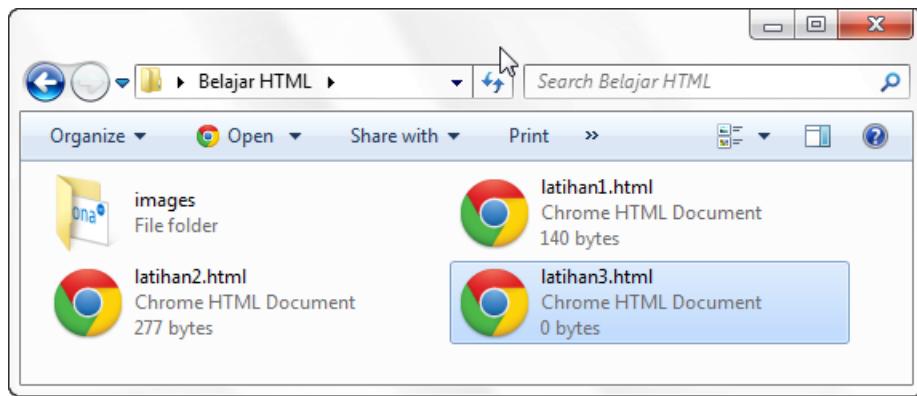
Namun anda tidak akan melihat perubahan pada teks “disini” karena kita belum “mengaitkannya” ke halaman web lain. Untuk itu kita akan menggunakan attribut `href` untuk menyimpan alamat web yang akan dituju ketika link di klik (penulisan lokasi sama halnya dengan attribut `src` pada tag `img`).

1. Klik `disini` untuk mendownload

Sebagai latihan kita akan membuat file HTML baru dan membuat link untuk mengaitkannya ke latihan-latihan sebelumnya.

1. Buatlah file baru dengan nama `latihan3.html`, simpanlah di folder yang sama dengan latihan-latihan sebelumnya.

⁴ Tag-tag HTML juga biasa sebut dengan elemen.



Gambar 12 Penempatan file pada folder yang sama

2. Ketikkan kode berikut pada file latihan3.html

```

1. <!DOCTYPE HTML>
2. <HTML>
3.   <head>
4.     <title>Judul File HTML</title>
5.   </head>
6.   <body>
7.     <h1>Selamat datang di website saya</h1>
8.     <a href="latihan1.html">buka Latihan 1</a><br>
9.     <a href="latihan2.html">buka Latihan 2</a>
10.   </body>
11. </HTML>
```

3. Buka kembali file **latihan1.html** dan **latihan2.html**, dan tambahkan sebuah link yang akan mengait ke file latihan3.html tepat sebelum penutup tag body (`</body>`).

```

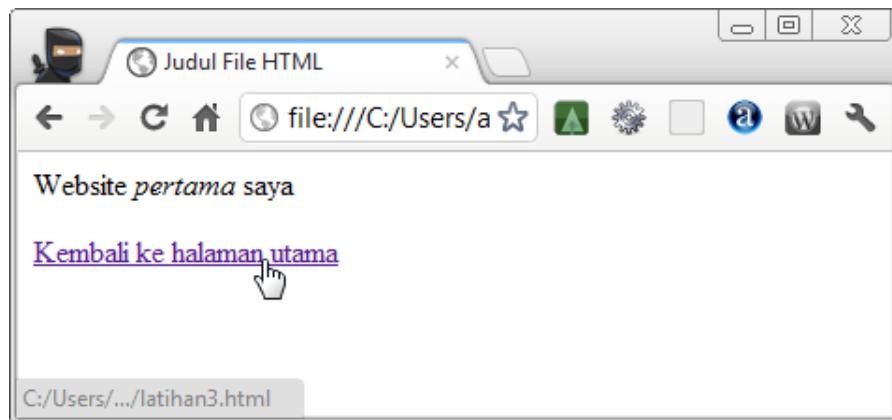
...
<a href="latihan3.html">Kembali ke halaman utama <a>
</body>
</HTML>
```

4. Save pekerjaan anda, dan bukalah file latihan3.html pada browser. Anda dapat mengklik linknya satu persatu.



Gambar 13 tampilan file Latihan3.html

Ketika link “buka latihan 1” di klik, browser akan menampilkan file latihan1.html

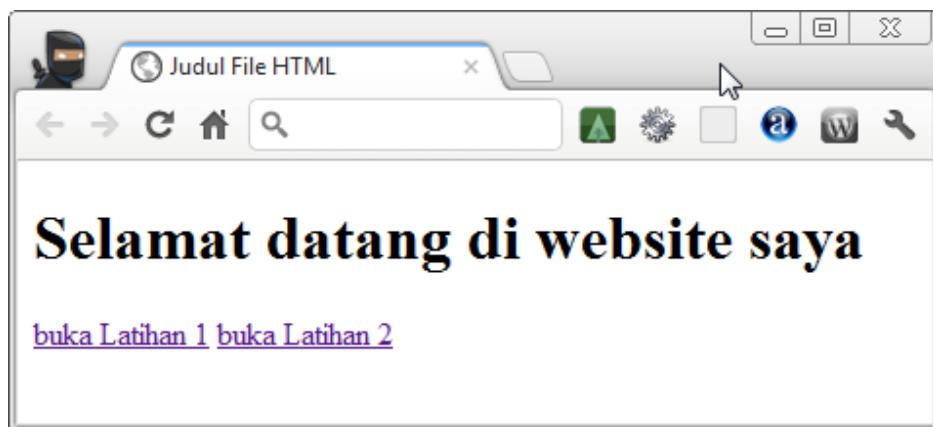


Gambar 14 Link untuk kembali ke halaman utama

Dan Klik link “Kembali ke halaman utama” untuk menampilkan latihan3.html kembali.

Jika anda perhatikan pada kode yang terdapat dalam latihan3.html, ada beberapa tag yang belum saya jelaskan, yaitu tag `<h1>` dan `
`.

Tag `
` digunakan untuk memindahkan teks kebaris baru, sehingga teks yang dipisahkan dengan tag ini akan ditampilkan di baris yang berbeda. Jika anda menghapus tag `
` ini, maka teks akan ditampilkan satu baris.

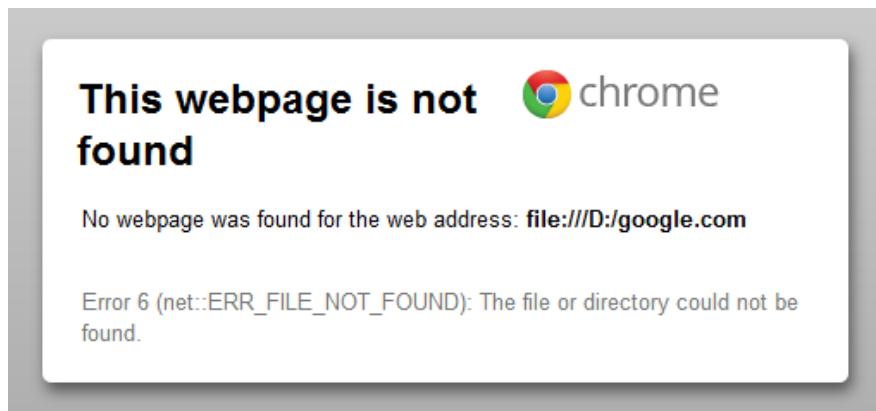


Gambar 15 link ditampilkan satu baris tanpa tag

Untuk membuat link yang mengacu ke halaman web lain, pastikan anda menyertakan http:// pada atribut href link tersebut.

```
<a href="http://www.google.com"> Buka Google </a>
```

Jika anda tidak menyertakan http:// maka link tersebut tidak akan bekerja sebagaimana mestinya.



Gambar 16 Error ketika http:// tidak ditambahkan pada alamat web lain

Setiap link yang diklik akan ditampilkan di window/tab yang sama, bagaimana jika anda ingin membuka link tersebut di tab/window baru? Tambahkan atribut target="_blank".

```
<a href=http://adcom.web.id target="_blank">adcom.web.id</a>
```

4.2. Link Email

Anda juga dapat membuat link email, link ini berisi alamat email yang ketika diklik, aplikasi untuk mengirim email akan otomatis terbuka dan tujuan email secara otomatis terisi dengan alamat yang telah ditentukan.

Untuk membuat link email, anda tinggal menambahkan mailto : alamat@email di dalam atribut href.

```
<a href="mailto:paynux@yahoo.com">Kirim Email</a>
```

Bab 5

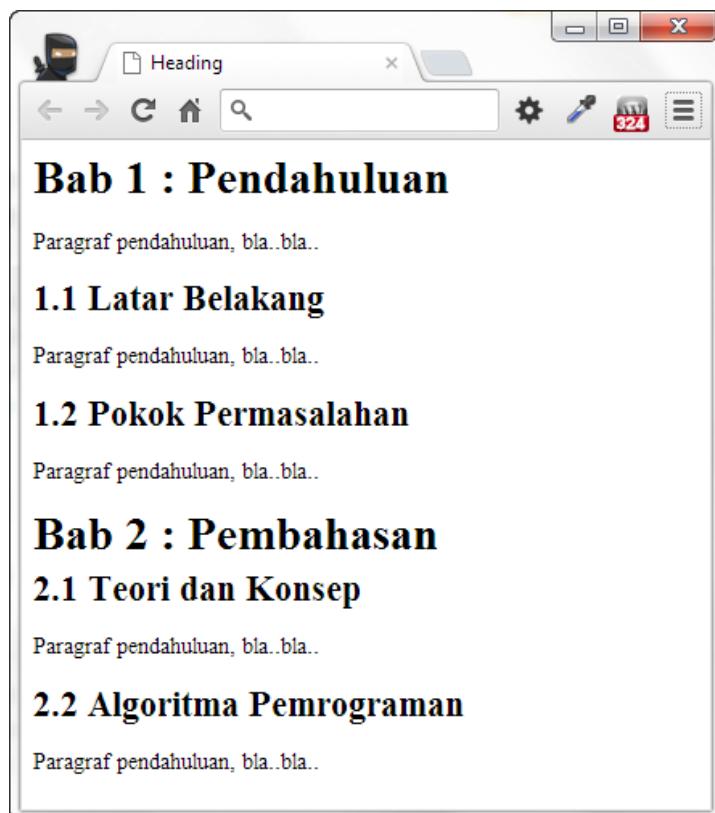
Heading/Penjudulan

Heading, digunakan untuk memberikan penjadulan pada suatu dokumen HTML. Untuk memformat penjadulan dalam HTML, kita gunakan tag `<h1>` untuk judul utama dan untuk judul sub babnya anda dapat menggunakan tag `<h2>` sampai dengan `<h6>`.

Setiap level judul memiliki ukuran huruf yang berbeda-beda (namun anda masih bisa merubah ukurannya melalui CSS).

Sebagai latihan, buatlah file HTML baru dengan nama `latihan4.html` lalu ketikkan kode HTML berikut:

```
1. <!DOCTYPE HTML>
2. <HTML lang="en-US">
3.   <head>
4.     <title>Heading</title>
5.   </head>
6.   <body>
7.     <h1>Bab 1 : Pendahuluan</h1>
8.     <h2>1.1 Latar Belakang</h2>
9.     <p>Paragraf pendahuluan, bla..bla.. </p>
10.    <h2>1.2 Pokok Permasalahan</h2>
11.    <p>Paragraf pendahuluan, bla..bla.. </p>
12.    <h1>Bab 2 : Pembahasan</h1>
13.    <h2>2.1 Teori dan Konsep</h2>
14.    <p>Paragraf pendahuluan, bla..bla.. </p>
15.    <h2>2.2 Algoritma Pemrograman</h2>
16.    <p>Paragraf pendahuluan, bla..bla.. </p>
17.  </body>
18. </HTML>
```



Gambar 17 Penjadulan pada dokumen HTML

Adanya penjadulan dimaksudkan agar suatu dokumen HTML lebih terstruktur layaknya sebuah dokumen resmi seperti skripsi /paper yang mengharuskan adanya perbedaan antara Bab utama dan sub-babnya.

Bab 6

Membuat 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*.

6.1. Ordered List

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

```
1. <h2>Ordered List</h2>
2. <ol>
3.     <li>Nyala komputer</li>
4.     <li>Buka notepad</li>
5.     <li>Mengerjakan latihan kuliah HTML</li>
6. </ol>
```

Penomoran daftar akan dilakukan secara otomatis ketika anda menambahkan daftar item.



Gunakan Ordered list, ketika sebuah list/daftar memerlukan urutan yang benar, seperti cara menyala komputer dan lain sebagainya.

Dan jika ditampilkan pada browser, maka yang akan anda dapatkan adalah seperti berikut :



Gambar 18 Daftar berurutan

6.2. Unordered List

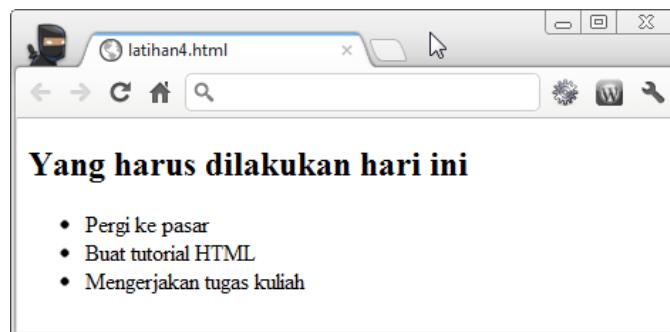
Berbeda dengan daftar berurut, *unordered list* akan menandai setiap item dengan simbol. Baik berupa lingkaran atau persegi. Untuk membuat daftar tidak berurut kita gunakan tag `` dan seperti tag ``, item yang terdapat di dalamnya harus diapit dengan tag ``.

Jika kita modifikasi contoh sebelumnya dengan merubah `` menjadi `` maka yang akan kita dapat adalah seperti berikut :

```

1. <h2>Yang harus dilakukan hari ini</h2>
2. <ul>
3.     <li>Pergi ke pasar</li>
4.     <li>Buat tutorial HTML</li>
5.     <li>Mengerjakan tugas kuliah</li>
6. </ul>

```



Gambar 19 Daftar tidak berurut

Secara default, item akan ditandai dengan lingkaran.

6.3. 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.

```
<dl>
    <dt>Test</dt>
    <dd>Test</dd>
</dl>
```

Tag `dt` (definition term) digunakan untuk menampung istilah yang akan didefinisikan, dan tag `dd` digunakan untuk menuliskan definisi dari `dt` sebelumnya. Berikut contoh penggunaan dari Definition List :

```
<dl>
    <dt>Manga</dt>
    <dd>Manga (baca: man-ga, atau ma-ng-ga) merupakan kata komik dalam bahasa Jepang; di luar Jepang, kata tersebut digunakan khusus untuk membicarakan tentang komik Jepang. </dd>

    <dt>Mangaka</dt>
    <dd>Mangaka (baca: man-ga-ka, atau ma-ng-ga-ka) adalah orang yang menggambar manga</dd>
</dl>
```

6.4. List/Daftar bersarang

Sebuah daftar bisa saja memiliki daftar lagi di dalam *itemnya*, atau biasa kita sebut dengan daftar/list bersarang (*nested list*). Contohnya seperti pada latihan yang akan kita buat berikut.

Buat file baru dengan nama file latihan5.html lalu ketikkan kode HTML berikut

```
1. <!DOCTYPE HTML>
2. <HTML>
3. <head>
4.         <title>List/Daftar</title>
5. </head>
6. <body>
7.         <h2>Yang harus dilakukan di laboratorium saat ini :</h2>
8.         <ol>
9.             <li>Nyalakan komputer</li>
10.            <ul>
11.                <li>Menyalakan power button di CPU</li>
12.                <li>Menyalakan monitor</li>
13.            <ol>
14.                <li>Buka notepad</li>
15.                <li>Menjalankan dari start program</li>
```

```
16.                                     </ol>
17.                                 </li>
18.                                 <li>Menjalankan dari menu run</li>
19.                             </ul>
20.                         </li>
21.                         <li>Buat tutorial HTML</li>
22.                         <li>Mengerjakan latihan kuliah HTML</li>
23.                     </ol>
24. </body>
25. </HTML>
```

Dan berikut tampilan pada browser.



Gambar 20 List bersarang

Yang perlu anda perhatikan adalah di mana anda meletakkan tag `` atau `` jika anda ingin menempatkannya pada suatu item, yaitu sebelum penutup tag list item (``).

Daftar seperti ini biasa digunakan untuk pembuatan menu website, atau keperluan lainnya yang memang membutuhkan penomoran atau sesuatu yang memiliki poin-poin.

Bab 7

Memahami Tag div (Division)

Anda dapat membuat suatu group dari tag-tag HTML yang anda buat dengan menggunakan tag <div>, seperti pada contoh kasus berikut :

Setiap website / weblog yang anda buka, jika diperhatikan pasti memiliki empat bagian / group. Keempat bagian / group ini antara lain :

1. Header

Bagian kepala website yang berisi logo website, nama website, slogan website, menu website dan lainnya.

2. Content

Berisi isi dari website itu sendiri, jika website berupa weblog, maka *contentnya* berisi postingan terbaru atau informasi lainnya.

3. Sidebar

Berada di samping Content dan biasanya berisi iklan, kategori artikel dan widget atau hiasan situs lainnya.

4. Footer

Bagian kaki dari website yang pada umumnya diisi dengan tag website tersebut, misalnya *copyright © 2015 by someone*.

Dalam HTML, kita dapat membagi bagian-bagian tersebut dengan menggunakan tag <div> (division/bagian) dan untuk menamai setiap div kita gunakan atribut id atau class⁵. Berikut contoh penggunaan tag div jika kita terapkan pada bagian-bagian website yang telah kita sebutkan sebelumnya:

⁵ Atribut ID dan Class tidak hanya digunakan dalam tag div saja melainkan dapat digunakan di setiap tag HTML untuk memberikan penamaan sebagai referensi jika diperlukan.

```

1. <div id="header">
2. </div>
3.
4. <div id="content">
5. </div>
6.
7. <div id="sidebar">
8. </div>
9.
10. <div id="footer">
11. </div>
```

Jika anda menampilkannya pada 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.

7.1. Pakai ID atau Class?

Bagi seorang pemula, termasuk saya ketika masih belajar HTML & CSS pasti menanyakan hal yang sama. Pasalnya atribut ID dan Class digunakan untuk menamai tag HTML. Lalu apa perbedaan dari keduanya?

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 :

```

1. <div id="menu">
2. <ul id="menu">
3.   <li>Beranda</li>
4.   <li>Tutorial</li>
5. </ul>
6. </div>
```

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

Perhatikan contoh sebelumnya, bahwa kita menggunakan ID yang berbeda untuk setiap div yakni, header, content, sidebar dan footer karena semuanya memiliki struktur dan fungsi yang berbeda dalam suatu dokumen HTML.

Class

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

```

1. <ul id="menu">
2.     <li class="merah">Beranda</li>
3.     <li>Tutorial</li>
4.     <li class="merah">Berita</li>
5.     <li>Video</li>
6. </ul>
```

Pada Kode HTML di atas, saya menggunakan class Merah 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 pengenal lebih spesifik.

7.2. Memahami hubungan Child, Parent dan Siblings

Ketika suatu tag memiliki tag/konten didalamnya maka hubungan antar tag-tag tersebut disebut dengan child-parent dan sibling. Perhatikan contoh berikut :

```

1. <div id="wrapper">
2.     <div id="content">
3.         <div id="article"></div>
4.         <div id="sidebar"></div>
5.     </div>
6. </div>
```

Dalam contoh kode HTML tersebut :

- <div id="wrapper"> disebut **Parent**, dan tag HTML yang ada di dalamnya disebut dengan **Child**(<div id="content">).
- Begitu juga dengan <div id="content"> disebut **parent** untuk <div id="article"> dan <div id="sidebar">,
- dan hubungan antara <div id="article"> dan <div id="sidebar"> disebut dengan **sibling**, atau saudara.

Bab 8

Form

Dalam sebuah website biasanya terdapat satu atau lebih form, seperti form pencarian, registrasi dan lain sebagainya. Form ini bisa digunakan untuk mengumpulkan data dari pengunjung website.

Sebuah form, boleh jadi memiliki beragam kontrol, mulai dari text input, Combo box, Button dan lain sebagainya. Kita akan mempelajari sebagian kontrol yang disebutkan diatas karena kontrol di atas adalah yang paling sering digunakan dalam halaman web.

Pertama-tama, pembuatan sebuah form diawali dengan tag <form>, dan setiap kontrol-kontrol yang dibutuhkan ditempatkan di dalam tag ini.

```
<form>
    <h1>Formulir Pendaftaran</h1>
    ...
</form>
```

Gambar 21 Formulir (FORM) pendaftaran Yahoo!

Control-control Form

Setiap control pada form dapat dibuat dengan menggunakan tag <input>. Dan yang membedakan tipe dari control tersebut berada pada atribut type (tipe kontrol). Berikut ini adalah sebagian tipe kontrol yang biasa ditemui :

Label

Label digunakan untuk memberikan keterangan pada setiap input yang ada. Jika anda perhatikan formulir pendaftaran situs yahoo, yang disebut dengan label adalah yang ditandai berikut ini :

Gambar 22 Contoh Label pada form

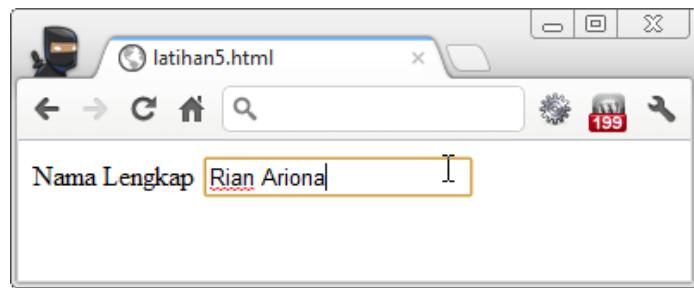
```
<label for="Name"> Keterangan Input </label>
```

Atribut for diisi dengan isi dari atribut name pada kontrol yang ingin diberi label.

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.

```
<label for="nama">Nama Lengkap</label>
<input type="text" name="nama" />
```



Gambar 23 Text Input

Jika text input yang akan ditampilkan ingin memiliki nilai, maka tuliskan nilai tersebut di dalam atribut value.

```
<input type="text" name="nama" value="Rian Ariona" />
```

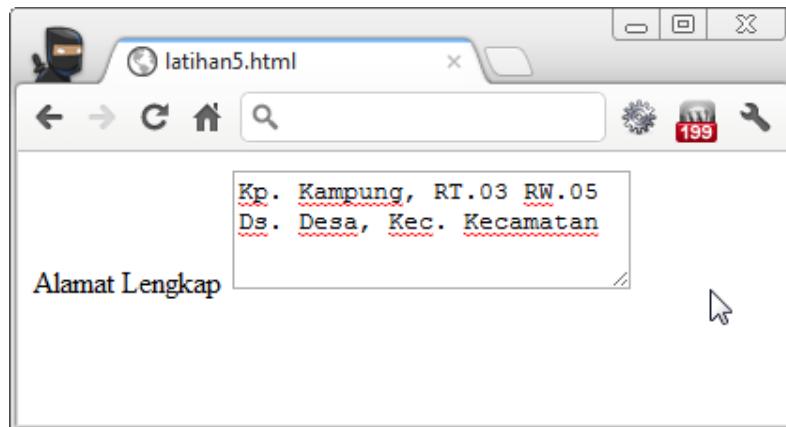
Dengan begitu, ketika halaman dibuka, text input ini secara otomatis akan terisi dengan nilai dari atribut value.

Text Area

Sama halnya dengan Input Text, namun textarea dapat diisi lebih dari satu baris, cocok digunakan untuk isian yang panjang, seperti alamat, deskripsi, atau biodata.

Berbeda dengan kontrol lainnya yang menggunakan tag <input>, text area memiliki tag sendiri yaitu <textarea></textarea>. Dan apa yang terdapat di dalam tag ini adalah value dari kontrol tersebut

```
<label for="alamat">Alamat Lengkap</label>
<textarea name="alamat"></textarea>
```

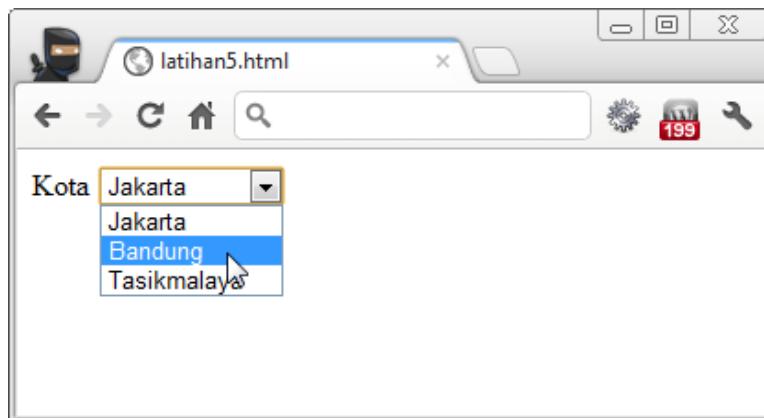


Gambar 24 Text area

Combo Box

Combo Box adalah kontrol yang memiliki pilihan ketika diklik. Pembuatannya sama seperti pembuatan Daftar/List namun dengan tag yang berbeda.

```
<label for="kota">Kota</label>
<select name="kota">
    <option>Jakarta</option>
    <option>Bandung</option>
    <option>Tasikmalaya</option>
</select>
```

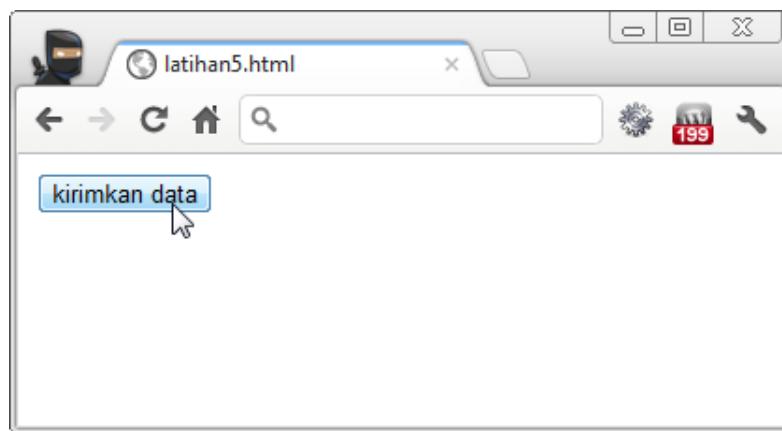


Gambar 25 Combo Box

Submit/Button

Submit atau Button, berupa tombol yang dapat di klik. Penggunaan atribut value pada kontrol ini akan merubah text yang ada di dalamnya.

```
<input type="submit" value=" kirimkan data" />
```



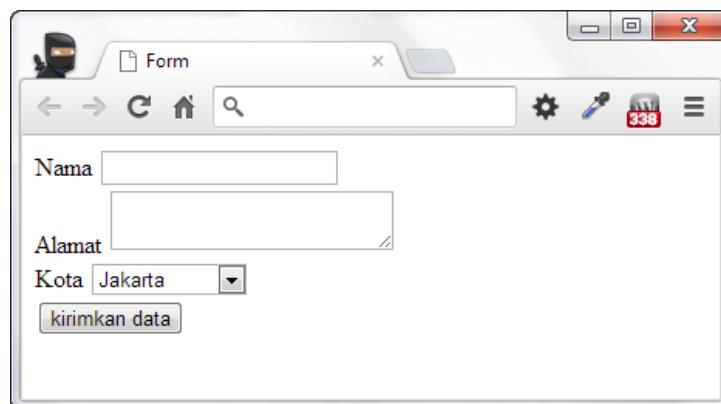
Gambar 26 Button

Sebagai latihan, kita coba menggabungkan kontrol-kontrol yang telah anda pelajari sebelumnya menjadi satu form utuh.

Buatlah file baru, dengan nama file `latihan6.html`, lalu ketikkan kode HTML berikut pada file `latihan6.html`

```
1. <!DOCTYPE HTML>
2. <HTML>
3. <head>
4.     <title>Form</title>
5. </head>
6. <body>
7.     <form>
8.         <label for="nama">Nama</label>
9.         <input type="text" name="nama"><br>
10.        <label for="alamat">Alamat</label>
11.        <textarea name="alamat"></textarea><br>
12.
13.        <label for="kota">Kota</label>
14.        <select name="kota">
15.            <option>Jakarta</option>
16.            <option>Bandung</option>
17.            <option>Tasikmalaya</option>
18.        </select><br>
19.
20.        <input type="submit" value="kirimkan data">
21.    </form>
22. </body>
23. </HTML>
```

Berikut ini adalah tampilan yang anda dapatkan jika file `latihan6.html` dibuka di dalam browser.



Gambar 27 Form utuh

Dalam buku ini, saya tidak akan membahas bagaimana cara memroses data yang dikumpulkan dari form tersebut, karena dalam HTML kita tidak bisa melakukan proses pengolahan data. Ingat bahwa HTML hanya digunakan untuk memformat dokumen, bukan untuk memroses data.

Untuk memroses data, kita bisa menggunakan bahasa pemrograman web, seperti Javascript atau PHP atau bahasa pemrograman web lainnya.

Dengan bahasa pemrograman seperti Javascript atau PHP anda dapat mengolah data-data yang dikumpulkan dari Form HTML untuk diproses dan dikeluarkan kembali menjadi suatu informasi baru.

Bab 9

Tabel

Untuk menampilkan data dengan tipe tabel dalam HTML, kita bisa menggunakan tag `<table>`.

```
<table></table>
```

Untuk membuat baris tabel, kita gunakan tag `<tr>` – singkatan dari *table row* (ditulis di dalam tag table)

```
<table>
    <tr></tr>
</table>
```

Sedangkan untuk menentukan banyaknya kolom, tergantung dari banyaknya tag `<td>` (*table data*) yang digunakan di dalam tag `<tr>`.

```
1.  <table>
2.    <tr>
3.      <td>No</td>
4.      <td>Nama</td>
5.      <td>Alamat</td>
6.    </tr>
7.  </table>
```

Untuk menyatukan kolom atau baris (*merge-cell*) kita tambahkan atribut colspan (untuk *merge horizontally*-secara horizontal) atau rowspan (untuk *merge vertically*).

```
1.  <table>
2.    <tr>
3.      <td rowspan=2>No</td>
4.      <td colspan=2>Jenis Kelamin</td>
5.      <td rowspan=2>Nama</td>
6.    </tr>
7.    <tr>
8.      <td>L</td>
9.      <td>P</td>
10.   </tr>
11.   <tr>
12.     <td>1</td>
13.     <td>L</td>
14.     <td></td>
```

```

15.          <td>Rian</td>
16.      </tr>
17.  </table>

```

Jika anda tampilkan dalam browser, maka tabel tersebut tidak akan memiliki garis, untuk memberi garis pada tabel anda dapat menambahkan atribut border=1 pada tag <table> atau anda dapat menambahkan border lewat CSS nantinya.

No	Jenis Kelamin		Nama
	L	P	
1	L		Rian

Gambar 28 Tabel setelah pemberian atribut border

Sebagai latihan, anda tulis kode tabel di atas dan tambahkan beberapa data lagi, lalu simpan dalam file latihan7.html.

9.1. Struktur tabel yang dianjurkan

Agar format tabel sesuai dengan ketentuan, Anda dapat menggunakan tag <thead> untuk baris yang menjadi Judul tabel, <tbody> untuk data/isi dari tabel dan <tfooter> untuk kaki tabel (bisa diisi sama dengan thead atau lainnya).

```

1. <table border=1>
2.   <thead>
3.     <tr>
4.       <td rowspan=2>No</td>
5.       <td colspan=2>Jenis Kelamin</td>
6.       <td rowspan=2>Nama</td>
7.     </tr>
8.     <tr>
9.       <td>L</td>
10.      <td>P</td>
11.    </tr>
12.  </thead>
13.
14.  <tbody>
15.    <tr>
16.      <td>1</td>
17.      <td>L</td>
18.      <td></td>
19.      <td>Rian</td>
20.    </tr>
21.  </tbody>
22. </table>

```

9.2. Kesalahan dalam penggunaan <table>

Tag table digunakan untuk memformat data tabular/data yang membutuhkan format tabel dalam menampilkannya, seperti contoh penggunaan tabel untuk menampilkan data nilai berikut :

No	Nama	Nilai Akhir	Index
1	Jono	90	A
2	Joni	75	B
3	Dedi	82	A

Masih banyak yang menggunakan tabel untuk keperluan layouting sebuah website, misalnya menggunakan tag table untuk membuat layout 2 kolom, 3 kolom, atau 4 kolom seperti contoh berikut :

Website Keren Saya		
- Home - Artikel - About	Selamat Datang! Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna. Nunc viverra imperdiet enim. Fusce est.	Address Jl. Sasab 34 4023XX
Copyright © 2012 by Saya		

Gambar 29 Penggunaan table untuk layout adalah menyalahi aturan

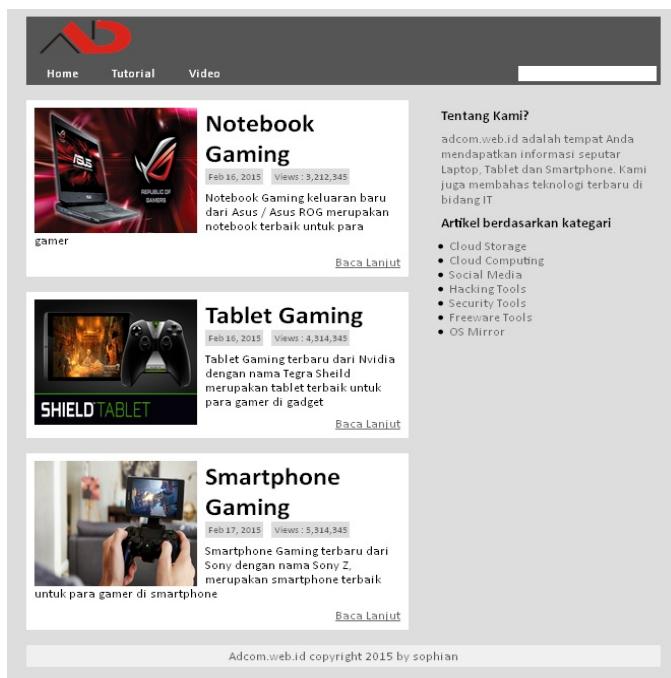
Penggunaan tag <table> untuk keperluan layout adalah menyalahi aturan/fungsi dari tag table tersebut.

Layout table biasanya dibuat secara otomatis oleh aplikasi seperti Dreamweaver atau Photoshop (di-generate oleh slicing tool).

Bab 10

Studi kasus 1- Konversi desain ke HTML

Sebagai seorang web designer, Anda harus mampu menerjemahkan suatu desain web ke dalam bahasa HTML dan CSS. Proses penerjemahan dari desain ke dalam kode HTML ini disebut dengan Pembuatan *Markup*, yang dilanjutkan dengan proses *Styling* (Penambahan CSS yang akan kita pelajari pada bab selanjutnya).



Gambar 30 Desain web yang akan kita konversi

Pada bab ini anda akan belajar membuat *markup* dari sebuah desain web yang telah disiapkan. Saya tidak akan memberikan desain web yang rumit, sebagai latihan kita gunakan desain yang sederhana dulu.

10.1. Pesiapan

Sebelum kita mulai proses pembuatan markup untuk desain web tersebut, kita buat folder baru terlebih dahulu, agar pekerjaan kita rapih dan terstruktur.

Buatlah folder baru dengan nama “**Latihan Markup**” dan buatlah file HTML baru dengan nama “**latihan-markup.html**”.

Dalam folder tersebut buat lagi satu buah folder dengan nama “**image**” untuk menyimpan seluruh gambar yang diperlukan dalam file HTML yang kita buat. Lalu copy gambar yang diperlukan ke dalam folder resources/images.

10.2. Wrapper

Kita mulai dari pembuatan sebuah div untuk menampung seluruh konten situs yang akan dibuat. Saya akan menggunakan div dengan id wrapper. Nantinya div ini akan kita buat rata-tengah, jadi setiap browser di zoom-in atau di zoom-out, konten situs tetap berada di tengah.

```

1. <!DOCTYPE HTML>
2. <HTML>
3.     <head>
4.         <title>Latihan Markup</title>
5.     </head>
6.
7.     <body>
8.         <div id="wrapper">
9.
10.            </div>
11.        </body>
12. </HTML>
```

10.3. Header

Kita mulai dari pembuatan header, perhatikan gambar berikut :



Gambar 31 Bagian header web

Sudah ada bayangan?, pertama kita buat sebuah div dengan nama id header (atau terserah anda). Dalam gambar di atas bisa anda perhatikan terdapat gambar logo, lalu Menu (kita buat dengan unordered list) dan terakhir sebuah text input untuk pencarian.

Semuanya telah anda pelajari pada bab-bab sebelumnya. Jadi, Markup untuk bagian header adalah seperti berikut :

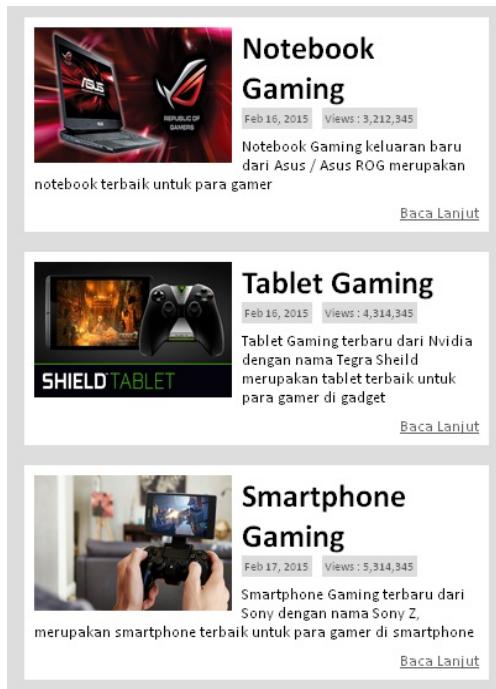
```

8. <div id="wrapper">
9.     <div id="header">
10.        
11.
12.        <ul id="menu-utama">
13.            <li><a href="">Home</a></li>
14.            <li><a href="">Tutorial</a></li>
15.            <li><a href="">Video</a></li>
16.        </ul>
17.        <form>
18.            <input type="text" name="search" />
19.        </form>
20.    </div>
21.
22.
23. </div>
```

Mudah bukan?, semua tag-tag tersebut telah saya jelaskan sebelumnya. Jika anda masih tidak mengerti atau lupa anda dapat membuka kembali bab sebelumnya.

10.4. Content

Kita lanjutkan dengan proses markup Content/isi Situs.



Gambar 32 Bagian Konten Web

Jika anda perhatikan, disana terdapat tiga Artikel yang memiliki penampilan dan struktur yang sama, untuk itu kita akan menggunakan penamaan dengan Class pada div ini. Namun sebelumnya, kita tulis sebuah div untuk menampung artikel yang akan kita buat dengan nama “**daftar-artikel**”.

```

22. <div id="daftar-artikel">
23.     <div class="artikel">
24.
25.     </div>
26. </div>
```



Gambar 33 Detail setiap post

Kita masuk ke detail setiap artikel. Di dalam artikel ini terdapat sebuah gambar, judul artikel, tanggal, ringkasan dan sebuah link “Baca Lanjut”.

Sudah terbayang bagaimana kode HTML nya?

```

22. <div id="daftar-artikel">
23.     <div class="artikel">
24.         
25.         <h1> Notebook Gaming </h1>
26.         <b>Feb 16, 2015</b><b>Views : 3,212,345</b>
27.         <p>Notebook Gaming keluaran baru dari Asus / Asus ROG merupakan
notebookterbaik untuk para gamer</p>
28.         <a href="">Baca Lanjut</a>
29.     </div>
30. </div>
```

Selanjutnya anda tinggal Copy div class **artikel** ini sebanyak tiga kali. Sehingga menjadi seperti berikut :

```

22. <div id="daftar-artikel">
23.     <div class="artikel">
24.         
25.         <h1> Notebook Gaming </h1>
26.         <b>Feb 16, 2015</b><b>Views : 3,212,345</b>
27.         <p>Notebook Gaming keluaran baru dari Asus / Asus ROG merupakan
notebookterbaik untuk para gamer</p>
28.         <a href="">Baca Lanjut</a>
29.     </div>
30.     <div class="artikel">
```

```

31.      
32.      <h1> Tablet Gaming</h1>
33.      <b>Feb 16, 2015</b><b>Views : 4,314,345</b>
34.      <p>Tablet Gaming terbaru dari Nvidia dengan nama Tegra Sheild merupakan
35.          tablet terbaik untuk para gamer di gadget</p>
36.          <a href="">Baca Lanjut</a>
37.      </div>
38.      <div class="artikel">
39.          
40.          <h1> Smartphone Gaming</h1>
41.          <b>Feb 17, 2015</b><b>Views : 5,314,345</b>
42.          <p>Smartphone Gaming terbaru dari Sony dengan nama Sony Z merupakan
43.              smartphone terbaik untuk para gamer di smartphone</p>
44.          <a href="">Baca Lanjut</a>
45.
46.

```

10.5. Sidebar

Kita lanjutkan dengan pembuatan sidebar, perhatikan apa yang terdapat di dalam sidebar.



Gambar 34 Sidebar

Setiap item dalam sidebar memiliki judul. Pertama terdapat deskripsi singkat tentang situs, dan sebuah menu navigasi berdasarkan kategori. Sama seperti menu sebelumnya, kita buat dengan menggunakan unordered list.

```

46.  <div id="sidebar">
47.    <h2>Tentang Kami?</h2>
48.    <p>adcom.web.id adalah tempat Anda mendapatkan informasi seputar Laptop, Tablet dan
49.        Smartphone. Kami juga membahas teknologi terbaru di bidang IT</p>
50.    <h2>Artikel berdasarkan kategori</h2>
51.    <ul>
52.        <li><a href="">Cloud Storage</a></li>
53.        <li><a href="">Cloud Computing</a></li>

```

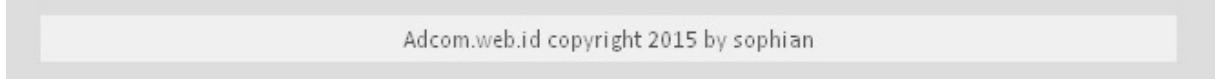
```

54.      <li><a href="">Social Media</a></li>
55.      <li><a href="">Hacking Tools</a></li>
56.      <li><a href="">Security Tools</a></li>
57.      <li><a href="">Freeware Tools</a></li>
58.      <li><a href="">OS Mirror</a></li>
59.
60.    </ul>
61.  </div>

```

10.6. Footer

Terakhir, kita buat markup untuk bagian footer.



Adcom.web.id copyright 2015 by sophian

Gambar 35 Bagian Footer Web

Bagian ini hanya mempunya text copyright, dan saya yakin anda dapat membuat markup dari bagian footer ini :

```

61. <div id="footer">
62.   <p>Adcom.web.id copyright 2015 by sophian</p>
63. </div>

```

Mudah bukan? Jika anda merasa bingung bagaimana penulisan kode selengkapnya, berikut ini kode lengkap dari proses markup kita sebelumnya.

```

1.  <!DOCTYPE HTML>
2.  <HTML>
3.    <head>
4.      <title>Latihan Markup</title>
5.    </head>
6.    <body>
7.      <div id="wrapper">
8.        <div id="header">
9.          
10.         <ul id="menu-utama">
11.           <li><a href="">Home</a></li>
12.           <li><a href="">Tutorial</a></li>
13.           <li><a href="">Video</a></li>
14.         </ul>
15.         <form>
16.           <input type="text" name="search" />
17.         </form>
18.       </div><!-- Penutup div "header" -->
19.
20.       <div id="daftar-artikel">
21.         <div class="artikel">
22.           
24.           <h1> Notebook Gaming</h1>
25.           <b>Feb 16, 2015</b><b>Views : 3,212,345</b>

```

```

25.          <p>Notebook Gaming keluaran baru dari Asus / Asus ROG
26.          merupakan notebook terbaik untuk para gamer</p>
27.          <a href="">Baca Lanjut</a>
28.          </div><!-- Penutup div "artikel" -->
29.          <div class="artikel">
30.              
32.              <h1> Tablet Gaming</h1>
33.              <b>Feb 16, 2015</b><b>Views : 4,314,345</b>
34.              <p>Tablet Gaming terbaru dari Nvidia dengan nama Tegra
35.                  Sheild merupakan tablet terbaik untuk para gamer di
36.                  gadget</p><a href="">Baca Lanjut</a>
37.          </div><!-- Penutup div "artikel" -->
38.          <div class="artikel">
39.              
41.              <h1> Smartphone Gaming</h1>
42.              <b>Feb 17, 2015</b><b>Views : 5,314,345</b>
43.              <p>Smartphone Gaming terbaru dari Sony dengan nama
44.                  Sony Z, merupakan smartphone terbaik untuk para gamer
45.                  di smartphone</p>
46.              <a href="">Baca Lanjut</a>
47.          </div><!-- Penutup div "artikel" -->
48.          </div><!-- Penutup div "daftar-artikel" -->
49.          <div id="sidebar">
50.              <h2>Tentang Kami?</h2>
51.              <p>adcom.web.id adalah tempat Anda mendapatkan informasi
52.                  seputar Laptop, Tablet dan Smartphone. Kami juga membahas
53.                  teknologi terbaru di bidang IT</p>
54.              <h2>Artikel berdasarkan kategori</h2>
55.              <ul>
56.                  <li><a href="">Cloud Storage</a></li>
57.                  <li><a href="">Cloud Computing</a></li>
58.                  <li><a href="">Social Media</a></li>
59.                  <li><a href="">Hacking Tools</a></li>
60.                  <li><a href="">Security Tools</a></li>
61.                  <li><a href="">Freeware Tools</a></li>
62.              </ul>
63.          </div><!-- Penutup div "sidebar" -->
64.          <div id="footer">
65.              <p>aadcom.web.id copyright 2015 by sophian</p>
66.          </div>
67.      </div><!-- Penutup div "wrapper" -->
68.  </body>
69. </HTML>

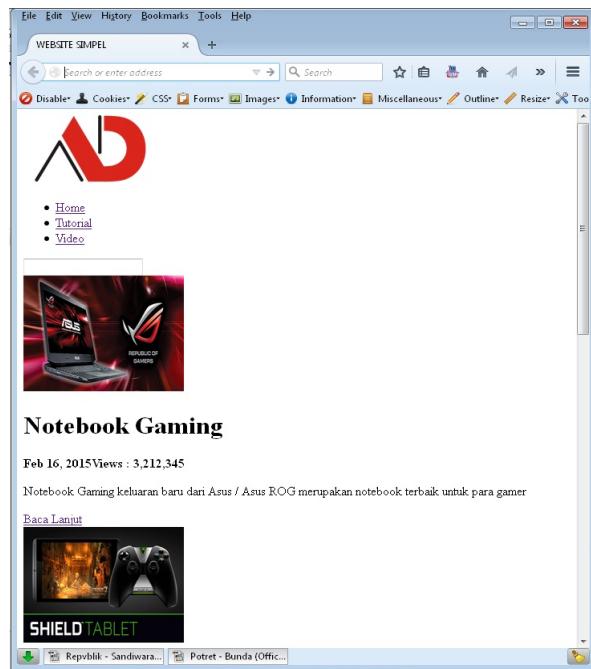
```

Agar tidak bingung membedakan penutup div yang satu dengan yang lainnya, saya menambahkan komentar di setiap penutup div. Komentar ini tidak akan ditampilkan pada browser.

Pembuatan komentar dapat dilakukan dengan menambahkan `<!--` dan ditutup dengan `-->` contohnya :

```
<!-- Komentar dituliskan di sini -->
```

Jika anda buka file HTML tersebut, maka yang akan anda lihat adalah seperti berikut :



Gambar 36 Tampilan dari Studi Kasus

File HTML yang ditampilkan akan terlihat *plain/polos*, tidak sama seperti pada desain web yang diberikan. Tapi inilah tugas dari HTML yakni *Markup/Formatting*. Adapun cara kita menampilkan bagian-bagian kode HTML agar terlihat seperti pada desain web yang diberikan, kita akan menggunakan CSS dan akan kita pelajari pada Bab CSS.

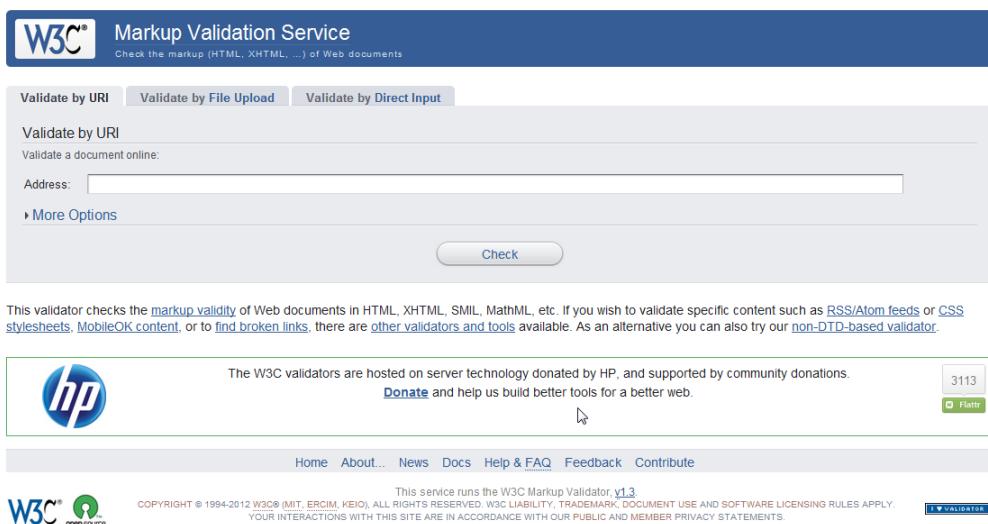
Bab 11

Validasi Markup HTML

Validasi markup adalah proses dimana kode HTML kita divalidasi apakah sudah benar atau belum oleh aplikasi validator dari W3C⁶. Validasi ini sangatlah diperlukan untuk memeriksa apakah ada yang salah dengan markup HTML yang telah kita buat atau ada beberapa bagian yang terlewat seperti penutup tag, atribut dan lain sebagainya.

Sebagian orang berpendapat bahwa markup HTML yang valid akan mempercepat proses loading website dan lebih bersahabat dengan Search Engine⁷.

Untuk melakukan validasi, silahkan buka alamat berikut: <http://validator.w3.org/>. Anda akan dibawa ke halaman validator HTML dari W3C seperti berikut :



Gambar 37 Validator HTML dari W3C

Ada tiga cara pengujian Markup HTML, yakni dengan mengetikkan alamat file HTML yang akan diuji (Validate by URL) opsi ini adalah untuk memvalidasi halaman HTML yang

⁶ W3C (World Wide Web Consortium) adalah komunitas internasional yang bekerja untuk membangun standar web seperti HTML dan CSS.<http://www.w3.org/>

⁷ Search Engine atau mesin pencari adalah aplikasi untuk mencari berbagai informasi di internet, seperti Google.com, bing.com dan lain sebagainya.

sudah “online” atau website yang sudah online. **Cara kedua** adalah dengan mengupload file HTML anda (Validate by file upload), dan validator akan memvalidasi kode HTML yang terdapat dalam file tersebut, dan **cara terakhir** adalah dengan mengetikkan langsung kode HTML dan memvalidasinya (Validate by direct input).

Sebagai contoh, kita coba memvalidasi latihan markup yang kita buat sebelumnya dengan cara “Validate by file upload”. Klik tab “Validate by file upload”, lalu klik Choose file untuk memilih file HTML dan pilihlah latihan Markup yang telah kita buat pada study kasus dan Klik tombol Check untuk mulai memvalidasi file HTML kita.

The screenshot shows the W3C Markup Validation Service interface. At the top, there are three tabs: "Validate by URI", "Validate by File Upload" (which is highlighted in blue), and "Validate by Direct Input". Below the tabs, the "Validate by File Upload" section is visible. It has a sub-header "Validate by File Upload" and a sub-instruction "Upload a document for validation:". There is a file input field with the placeholder "File: Choose File No file chosen". Below the input field is a link "More Options". At the bottom of the section is a large blue "Check" button. A note at the bottom states: "Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our [information page](#) on the W3C QA Website."

Gambar 38 Validate by File Upload

Jika pengujian sukses, maka akan muncul pesan lulus uji markup HTML seperti berikut:

The screenshot shows the validation results for a document. At the top, a green header bar says "This document was Tentatively checked as HTML5". Below it is a table with the following rows:

- Result:** Tentatively passed, 4 warning(s)
- File:** Choose File No file chosen
Use the file selection box above if you wish to re-validate the uploaded file latihan markup.html
- Encoding:** utf-8 (detect automatically)
- Doctype:** HTML5 (detect automatically)
- Root Element:** html

Gambar 39 Hasil validasi markup HTML

Perhatikan, bahwa kita mendapatkan 4 peringatan, coba scroll ke bawah dan anda akan menemukan peringatan tersebut (begitu juga dengan error – Jika ada).

Validation Output: 4 Warnings

Below is a list of the warning message(s) produced when checking your document.

⚠ The character encoding of the document was not declared.

Gambar 40 Peringatan validasi

Kesimpulan dari peringatan di atas adalah kita belum menyertakan informasi character encoding. Untuk memperbaiknya tambahkan kode berikut di dalam tag head dari file latihan markup kita.

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

Maksud dari tag tersebut adalah menentukan tipe karakter encoding yang digunakan dalam file HTML, dan kita gunakan tipe "UTF-8". Sekarang kita coba Revalidate (Memvalidasi ulang) file HTML kita.

This document was successfully checked as HTML5!	
Result:	Passed, 1 warning(s)
File:	<input type="button" value="Choose File"/> No file chosen <small>Use the file selection box above if you wish to re-validate the uploaded file latihan markup.html</small>
Encoding:	utf-8 <input type="button" value="(detect automatically)"/>
Dootype:	HTML5 <input type="button" value="(detect automatically)"/>
Root Element:	html

Gambar 41 Lulus uji validasi HTML

Sekarang kita telah lolos uji validasi, 1 peringatan yang tersisa adalah karena validator file HTML versi 5 dari W3C ini masih berupa eksperimen, so... *it's not problem and you are ready for the next level!!.*

Bagian II

CSS

Bab 13

Cascading Style Sheet (CSS)

13.1. Pengenalan CSS

CSS adalah kependekan dari Cascading Style Sheet, berfungsi untuk mempercantik penampilan HTML atau menentukan bagaimana elemen HTML ditampilkan, seperti menentukan posisi, merubah warna teks atau background dan lain sebagainya.

Perhatikan contoh kode CSS berikut :



Gambar 42 Contoh Kode CSS

Selector

Selector adalah elemen/tag HTML yang ingin diberi style. Anda dapat menuliskan langsung **nama tag** yang ingin diberi style tanpa perlu menambahkan tanda <>. Pada contoh kode CSS di atas, kita akan memberi style pada seluruh tag h1 yang terdapat dalam file HTML.

Jika tag HTML yang ingin diberi style memiliki ID, anda dapat menuliskan nama ID tersebut dengan diawali tanda kress (#).

```
#header
```

Dan jika tag yang diberi style memiliki Class, maka penulisan selector bisa dilakukan dengan tanda titik(.) diikuti dengan nama class.

.artikel

Jika anda hanya menuliskan satu selector, seperti contoh kode CSS di atas, maka seluruh tag h1 yang terdapat dalam file HTML akan memiliki style yang sama. Bagaimana jika kita hanya ingin memberi style pada tag h1 yang *hanya terdapat di dalam Class artikel*. Maka penulisan selectornya seperti berikut :

.artikel h1

Kode tersebut akan memerintahkan pada browser untuk memberi style pada tag h1 yang hanya terdapat di dalam **class artikel** (atau - h1 yang merupakan child dari class artikel).

Anda-pun dapat memilih lebih dari satu tag untuk penghematan kode CSS. Misalnya ketika anda memiliki dua atau lebih tag dengan warna background yang sama, dari pada menuliskan kode seperti ini :

```
h1{ background-color: #666666; }
P { background-color: #666666; }
a { background-color: #666666; }
```

anda dapat menggabungkan selector dengan menambahkan tanda koma pada nama tag yang ingin diberi style.

```
h1, p, a { background-color: #666666; }
```

Property dan Value

Property adalah sifat-sifat yang ingin diterapkan pada selector, seperti warna text, warna background, jarak antar elemen, garis pinggir dan lain sebagainya.

Untuk memberikan nilai/value pada property kita gunakan tanda titik dua (:). Setiap property diakhiri dengan titik koma (;), jika anda tidak mengakhirinya maka browser tidak akan mengetahui maksud dari property tersebut.

Property-property pada CSS sangat mudah dimengerti karena lebih mirip bahasa kita sehari-hari. Misalnya untuk merubah warna text kita gunakan property color, untuk

merubah warna background kita gunakan property background-color, untuk merubah ukuran huruf kita gunakan property font-size. Mudah dimengerti bukan?

```
.artikel h1 {
    color : red;
    background-color : blue;
    font-size : 20px;
}
```

Menurut saya, Kode CSS sangat mudah dimengerti, karena kata-kata yang digunakan sudah tidak asing lagi, perhatikan contoh kode CSS sebelumnya, jika kita ubah ke bahasa kita sehari-hari, kira-kira menjadi seperti berikut :

"Hey Brow (Browser) Cari tag h1 yang terdapat di dalam class artikel, jika ditemukan/ada, ubah warnanya menjadi merah, warna background menjadi biru dan ukuran huruf menjadi 20 pixel."

Cukup perkenalan dengan CSS-nya, lalu yang menjadi pertanyaan besar anda saat ini adalah :

"Dimana saya mengetikkan kode tersebut?"

13.2. Penulisan CSS

Ada tiga cara penulisan kode CSS, yaitu inline, internal dan external. Ketiganya bisa anda lakukan sesuai dengan kebutuhan. Berikut contoh penggunaan dari metode-metode tersebut:

Inline

Penulisan kode CSS dengan metode inline ini bisa dilakukan langsung pada tag yang ingin diberi style dengan menggunakan atribut style.

```
<h1 style="color : red; "> Judul Situs </h1>
```

Pada metode ini, anda tidak perlu menuliskan selector. Karena anda menuliskan CSS langsung pada tag yang ingin diberi style.

Cara ini sangat tidak dianjurkan, karena Anda akan mencampurkan antara “Format” dan “Presentasi”. Cara ini juga tidak efektif ketika anda akan melakukan perubahan pada CSS.

Internal

Metode CSS internal ditulis di dalam tag style yang ditempatkan pada tag head.

```
<HTML>
  <head>
    <title>Judul HTML</title>
    <style>
      h1 {
        color : red;
      }
    </style>
  </head>
...

```

Metode kedua ini sangat dianjurkan untuk pengujian style, atau ketika anda hanya memiliki satu halaman web.

External

Metode yang terakhir adalah dengan membuat file CSS dan dipanggil di dalam tag head. File CSS memiliki ekstensi (akhiran) .CSS misanya ***namafile.CSS***.

Pemanggilan file CSS dilakukan dengan menggunakan tag link:

```
<HTML>
  <head>
    <title>Judul HTML</title>
    <link rel="stylesheet" href="fileCSS.CSS" />
  </head>
...

```

Atribut rel adalah informasi hubungan (relationship) dari tag link tersebut, yaitu sebagai stylesheet. Href diisi dengan lokasi file CSS yang ingin dimuat. Pemanggilannya sama dengan pemanggilan gambar atau link.

Bab 14

Box-Model

14.1. Margin

Property margin digunakan untuk memberikan jarak antar elemen/tag HTML. Misalnya jarak antar <div> dan tag-tag HTML lainnya.

Sebagai contoh, buatlah file HTML baru dengan nama latihan8.html, lalu tambahkan kode berikut:

```
<div class="box">
    Box ke satu
</div>

<div class="box">
    Box ke dua
</div>
```

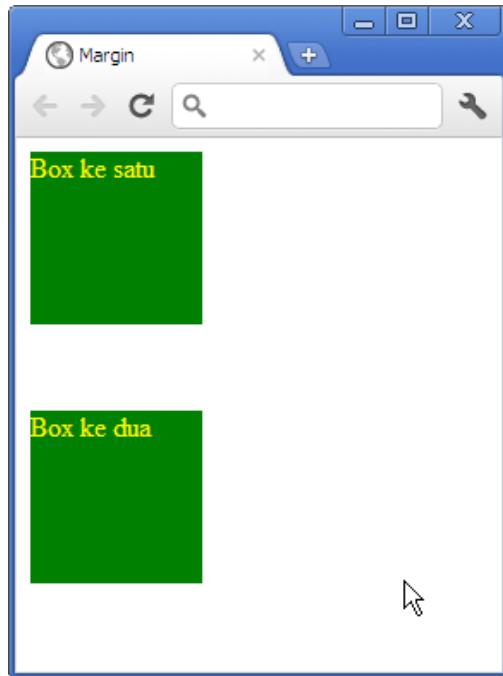


Ingat, tambahkan kode tersebut pada struktur HTML standar yang memiliki <!DOCTYPE HTML>, <HTML>, <head> dan seterusnya.

Lalu tambahkan tag style dalam tag head, dan ketikkan kode CSS berikut :

```
.box{
    background-color:green;
    color: yellow;
    width:100px;
    height:100px;
    margin-bottom:50px;
}
```

Berikut adalah tampilan yang akan anda dapatkan :



Gambar 43 Penambahan margin-bottom pada box

Jarak antar box ke satu dan box ke dua sedikit berjauhan, karena kita menambahkan margin bawah (margin-bottom) sebesar 50px.

Untuk melihat perbedaan ketika kita memberikan margin atau tidak, anda dapat menghapus property margin-bottom dan perhatikan perbedaannya.

Anda juga dapat memberikan jarak dari kanan (right), kiri (left) dan atas (top). Misalnya :

```
margin-top:20px;
margin-right:30px;
margin-bottom:50px;
margin-left:40px;
```

14.2. Padding

Pada latihan sebelumnya, kita telah memberikan jarak antar elemen HTML dalam contoh sebelumnya jarak antar <div>, namun coba perhatikan antara text (Box ke x) dengan garis pinggir kotak, tidak ada jarak sama sekali, merapat ke sisi box tersebut.

Untuk itu *property padding*lah yang akan memberikan jarak antara konten elemen dengan elemen tersebut.

Sebagai contoh tambahkan kode CSS berikut, pada class box yang kita buat sebelumnya:

```
.box{
    background-color:green;
    color: yellow;
    width:100px;
    height:100px;
    margin-bottom:50px;
    padding-left:20px;
}
```



Gambar 44 Pemberian padding pada box

Dan sekarang, anda dapat melihat jarak antara text yang berada di dalam box.

Perlu diperhatikan juga, bahwa ketika kita menambahkan padding, ukuran .box juga akan berubah.

Misalnya kita menentukan ukuran lebar dari .box sebesar 100px. Ketika kita menambahkan padding-left sebesar 20px, maka total lebar dari .box tersebut menjadi 120px.

Jadi jika anda ingin ukuran .box tetap 100px, kita harus menghitung ulang width dari box tersebut. Misalnya kita ubah menjadi 80px. Dengan begitu, 80px + 20px menjadi 100px.

14.3. CSS-Shorthand

Pada pembahasan sebelumnya, bisa kita simpulkan bahwa setiap property dapat memiliki lebih dari satu jenis, misalnya margin dapat memiliki 4 jenis. margin-top, margin-right, margin-bottom, dan margin-left.

```
margin-top:20px;
margin-right:30px;
margin-bottom:50px;
margin-left:40px;
```

Daripada menuliskan seluruh property tersebut, kita bisa menggunakan *CSS-Shorthand* yaitu menggabungkan seluruh value (top, right, bottom, dan left) ke dalam satu property.

Sebagai contoh, kita akan menggunakan *CSS-shorthand* untuk Property margin menjadi seperti berikut :

```
margin : 20px 30px 50px 40px;
```

Masing-masing dari nilai tersebut adalah :

```
margin : top right bottom left;
```

bisa kita simpulkan bahwa nilai dari CSS-shorthand margin ini memiliki urutan searah jarum jam dan dimulai dari margin-top.

Kita juga bisa bermain-main dengan property margin ini, perhatikan dengan seksama contoh berikut:

- Ketika kita memberikan **satu nilai** pada property margin, ini artinya kita memberikan nilai tersebut pada seluruh sisi.

```
margin: 20px;
```

Dengan penulisan seperti ini, maka nilai dari margin-top, right, bottom dan left adalah 20px;

- Ketika kita memberikan **dua nilai** pada property margin, berarti nilai selanjutnya mengikuti nilai sebelumnya. Contohnya seperti ini :

```
margin: 20px 30px;
```

Pada contoh di atas berarti kita memberikan nilai `margin-top` dan `right` bukan?, nah nilai dari `margin-bottom` akan disamakan dengan nilai `margin-top` dan `margin-left` sama dengan `margin-right`.

- Dan ketika kita menuliskan **tiga nilai**, maka nilai yang tidak diisi akan mengikuti nilai yang diisi, sama halnya ketika kita memberi 2 nilai.

```
margin: 20px 30px 40px;
```

Pada contoh di atas, kita memberikan nilai untuk top, right dan bottom. Sedangkan untuk left akan disamakan dengan right.

Penggunaan nilai seperti ini juga berlaku pada padding, dan property-property sejenis lainnya.

14.4. Border

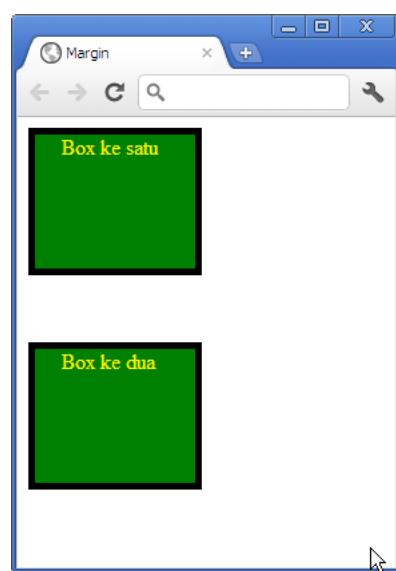
Property border digunakan untuk memberikan garis pinggir pada elemen, setidaknya ada tiga property yang harus anda set, yaitu color, style dan width.

```
border-color : black;
border-style : solid;
border-width : 5px;
```

namun anda juga dapat menggunakan *CSS shorthand* sehingga menjadi seperti berikut :

```
border : width style color;
```

jadi jika kita ubah contoh kode di atas menjadi CSS-shorthand, kodennya menjadi seperti berikut :



```
border : 5px solid black;
```

Cobalah tambahkan kode tersebut pada latihan sebelumnya dan yang akan anda dapatkan adalah seperti berikut :

Sama halnya dengan Padding, penambahan property border juga akan merubah ukuran dari .box.

Jadi, seperti yang saya jelaskan sebelumnya. Ketika anda

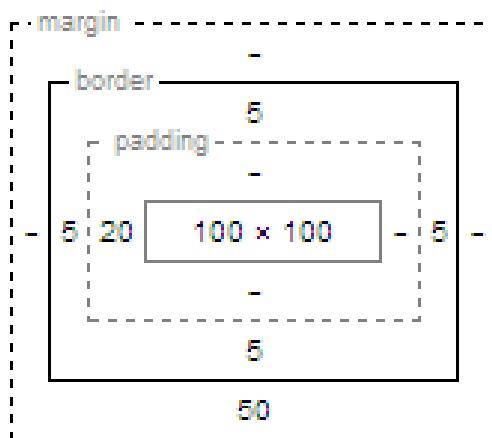
Gambar 45 Penambahan border

ingin membuat ukuran lebar .box tetap 100px, maka anda harus menghitung ulang width dari .box tersebut.

Kali ini perhatikan bahwa kita memberikan border pada seluruh sisi .box, artinya jika kita tentukan ukuran border sebesar 5px, maka 10px (untuk width, karena 5px untuk left, dan 5px untuk right) akan ditambahkan pada ukuran .box tersebut. Sehingga ukuran .box berubah kembali menjadi 110px.

Untuk itu kita kurangi width dari .box tersebut menjadi 70px sehingga : 70px (width) + 20px (padding) + 10px (border) = 100px.

Dalam dunia CSS, kasus seperti ini disebut dengan box-model. Terdapat dua property yang mempengaruhi ukuran dari sebuah box yaitu padding dan border. Perhatikan skema box model berikut untuk contoh kasus di atas :



Gambar 46 skema box-model



Tahukah anda, bahwa setiap elemen/tag HTML ditampilkan dalam bentuk box/kotak, jadi aturan box-model berlaku untuk seluruh elemen HTML

Bab 15

Typography

Secara *default* (bawaan), halaman web yang kita buat akan menggunakan font standar yaitu *Times New Roman*. Namun anda dapat merubahnya dengan property font-family. Berikut contoh penggunaan property font-family :

```
font-family: calibri;
```

Namun perlu diperhatikan bahwa tidak semua PC memiliki font yang kita tentukan, misalnya ketika kita menentukan font-family dengan calibri, maka pada PC dengan sistem operasi Linux, tidak akan ditemukan font tersebut.

Untuk itu anda dapat memberikan lebih dari satu jenis font, font-font yang disebutkan selanjutnya akan menjadi pengganti ketika font sebelumnya tidak ditemukan. Contohnya :

```
font-family: calibri, arial, tahoma, verdana, sans-serif;
```

Ketika font jenis calibri tidak ditemukan, maka yang akan digunakan adalah arial, ketika tidak ditemukan font arial maka tahoma yang akan digunakan, begitu seterusnya sampai kita tentukan jenis font yang dipakai, sans-serif atau serif dan browser akan menggunakan font yang sesuai dengan jenis yang disebutkan (serif atau sans-serif).



Ketika nama font memiliki spasi, maka anda harus memberikan tanda petik/kutip pada nama font tersebut. Misalnya "segoe ui".

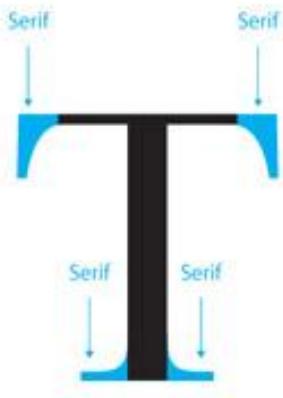
Apa itu Sans-serif dan Serif

Dalam typography jenis font dibagi menjadi dua yakni serif dan sans-serif. Bagian ini akan menjelaskan secara singkat perbedaan dari kedua jenis font ini.

Serif

Kata Serif bisa diartikan sebagai kait, jadi jenis huruf serif memiliki kait disetiap ujung hurufnya. Font jenis serif yang paling terkenal adalah Times New Roman. Perhatikan detail hurufnya.

Font jenis ini bisa kita gunakan untuk teks dengan konteks formal atau gaya klasik. Namun anda dapat memadupadankan jenis font ini sehingga memberikan kesan tertentu pada web anda.



Gambar 47 Font jenis Serif memiliki kait disetiap ujungnya

Sans-serif

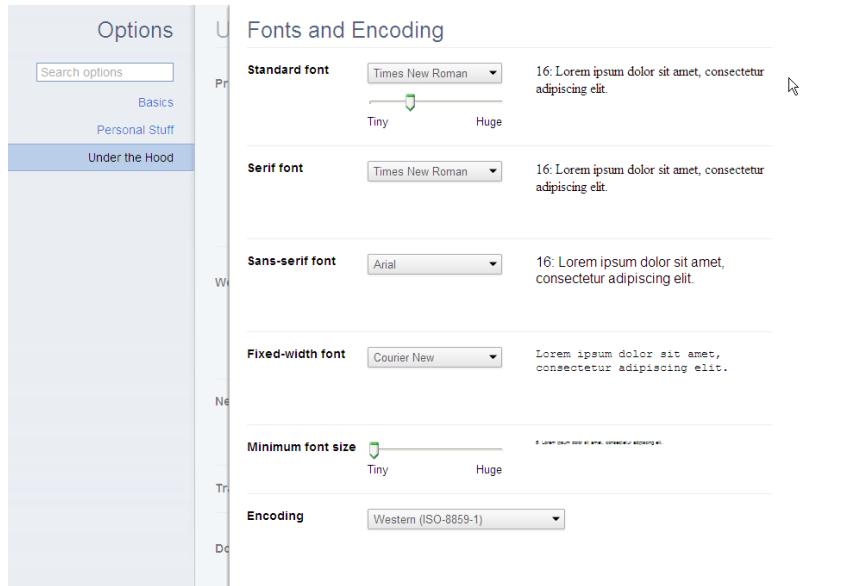
Arti dari sans, adalah "tanpa". Jadi sans-serif bisa diartikan sebagai jenis font tanpa kait. Yang paling umum digunakan untuk jenis sans-serif adalah arial.

Jenis ini memberikan kesan dynamis dan luwes. Seperti saya sebutkan sebelumnya anda dapat membuat kesan tersendiri ketika memadupadankannya.

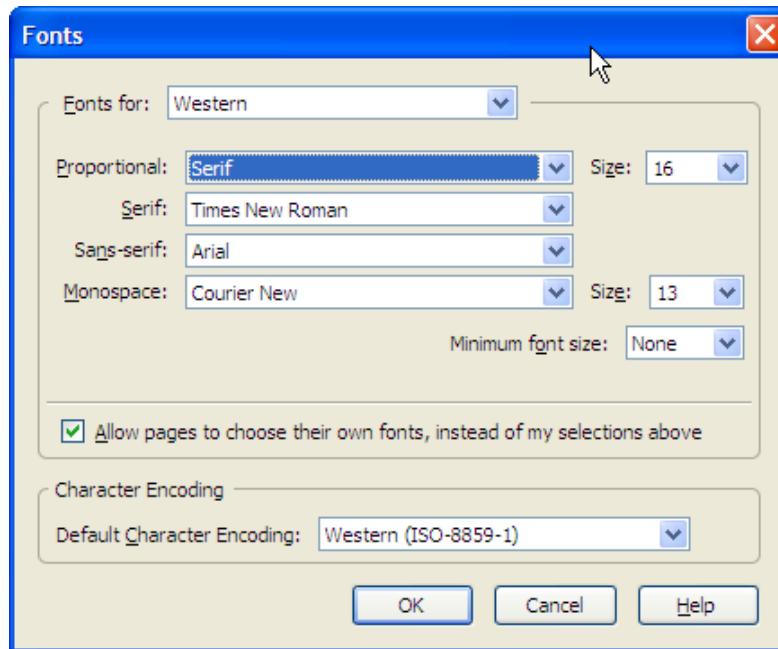


Gambar 48 Font jenis Sans Serif

Dalam pengaturan browser, kita bisa memilih font untuk serif dan sans-serif. Jadi ketika font-family menggunakan serif, maka font yang telah kita pilihlah yang akan digunakan.



Gambar 49 pengaturan font pada browser Google Chrome



Gambar 50 Pengaturan font pada firefox

Untuk menentukan **ukuran** font, kita gunakan property **font-size** :

```
font-size : 12px;
```

anda juga dapat menggunakan satuan pt (point) untuk ukuran font :

```
font-size : 12pt;
```

font yang kita gunakan pun dapat ditentukan **ketebalannya** dengan menggunakan property font-weight. Nilai yang bisa dipakai pada property ini adalah normal, bold, bolder.

```
font-weight : bold;
```

Bukan tidak mungkin untuk mengatur kerning atau jarak antar huruf. Dalam CSS kita bisa merubah jarak antar huruf dengan property letter-spacing.

```
letter-spacing : 15px;
```

Semakin besar nilai pixel yang diberikan semakin jauh jarak antar huruf. Andapun dapat menggunakan nilai negatif untuk memperdekat jarak antar huruf.

```
letter-spacing : -2px;
```

Sedangkan untuk memberikan jarak antar baris, kita gunakan property line-height.

```
line-height : 30px;
```

Dan untuk membuat text menjadi rata kiri, tengah atau kanan, kita gunakan property text-align. Nilai yang dapat digunakan adalah left, center, right, dan justify untuk rata kiri-kanan.

```
text-align : center;
```

Cobalah untuk bemain-main dengan property tersebut dan perhatikan perubahan yang didapatkan. Buatlah file `latihan9.html`, isi dengan beberapa paragraf beserta headingnya dan terapkan property-property yang telah anda pelajari di atas.

Bab 16

CSS-Image

Sebenarnya, berbicara tentang image dalam CSS, maka kita sedang membicarakan background-image. Bagaimana menggunakan sebuah gambar menjadi background untuk suatu elemen. Pada bagian ini anda akan mempelajari penggunaan `background-image`, teknik sprite, dan `background-repeat`.

16.1. `background-image`

Untuk memanggil gambar sebagai background dari elemen, kita gunakan property `background-image`. Perhatikan contoh penggunaan property ini :

```
background-image : url ("images/background.jpg");
```

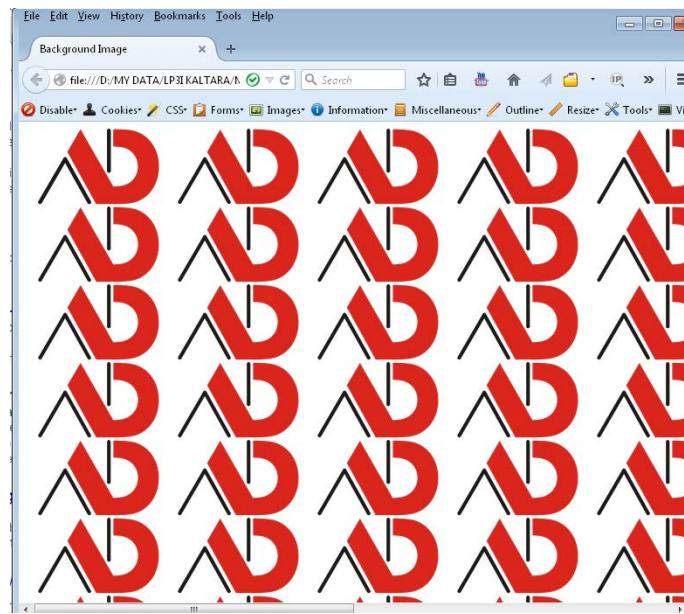
Ketika anda menggunakan gambar dengan ukuran kecil maka anda akan melihat background ditampilkan secara berulang untuk memenuhi seluruh element tersebut.

Untuk percobaan buatlah file `latihan10.html` dan selanjutnya buatlah sebuah div, dan menentukan ukuran width dan heightnya.

```
<div id = "box"></div>
```

Dan tambahkan style/CSS seperti berikut :

```
#box{  
    width : 300px;  
    height :300px;  
    background : url(logo2.png);  
}
```

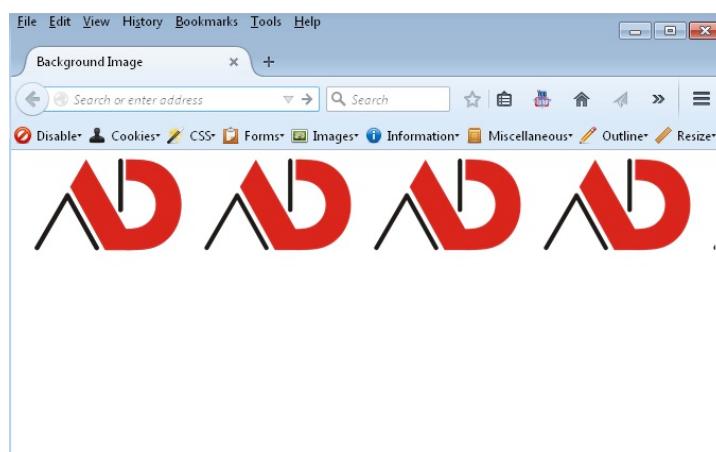


Gambar 51 Background akan terus diulang untuk memenuhi ukuran elemen

16.2. background-repeat

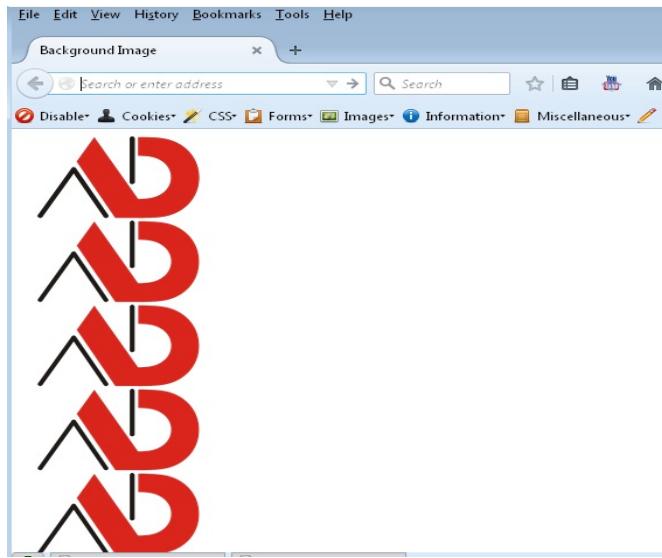
Pengulangan gambar background tersebut bisa kita atur dengan menggunakan property background-repeat. Anda bisa mengulang gambar secara horizontal (sesuai sumbu x) atau secara vertikal (y) atau anda pun dapat tidak mengulangi background tersebut.

```
background-repeat : repeat-x;
```



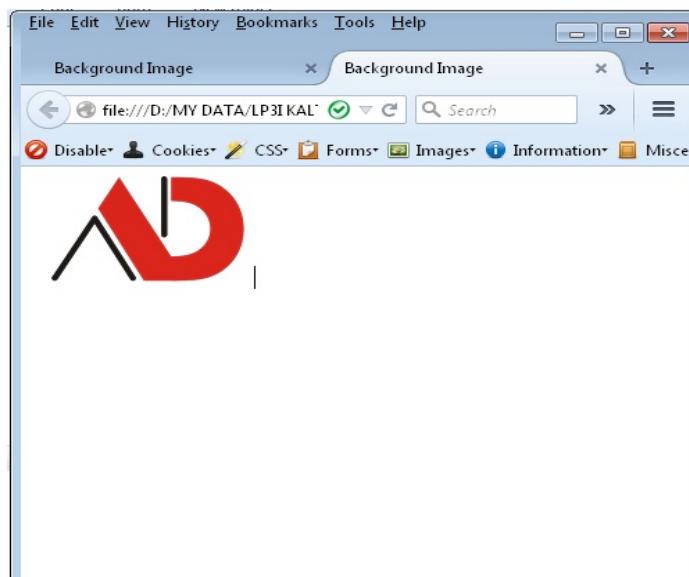
Gambar 52 Repeat X akan mengulang background secara horizontal

```
background-repeat : repeat-y;
```



Gambar 53 Pengulangan background secara vertical dengan repeat-y

```
background-repeat : no-repeat;
```



Gambar 54 Background tidak diulang

16.3. background-position

Selain memberikan kontrol terhadap perulangan gambar latar, CSS juga memberikan kontrol terhadap posisi dari gambar latar itu sendiri. Hal ini tentunya sangat berguna, terutama untuk mengatur posisi pasti dari gambar latar pada elemen yang sangat besar, seperti penampung dari seluruh elemen dalam halaman.

Posisi gambar latar dapat diatur menggunakan properti background-position, yang menerima dua nilai: koordinat sumbu x dan sumbu y tempat kita ingin meletakkan gambar latar. Koordinat yang diisikan menggunakan sistem koordinat kartesius terbalik, di mana titik 0, 0 berada pada kiri atas elemen.

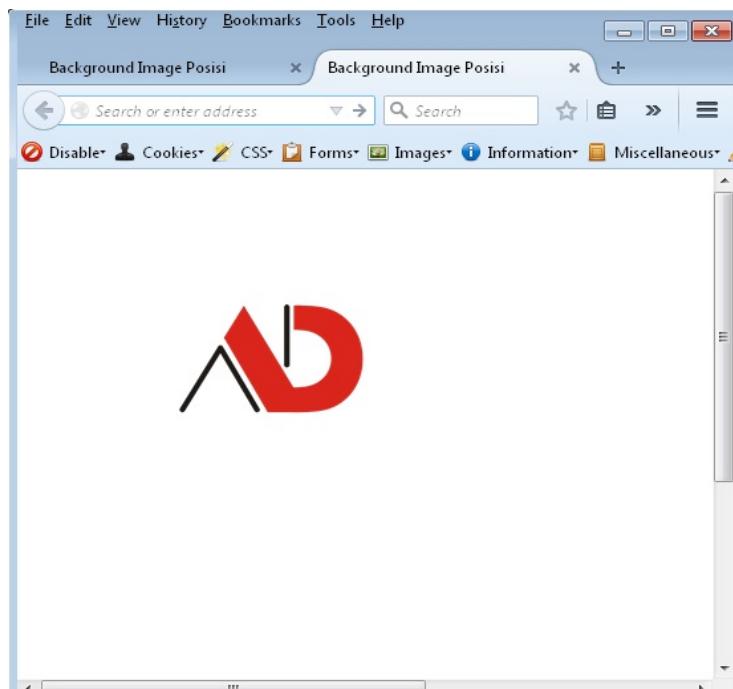
Pengisian koordinat sendiri dapat dilakukan dengan dua cara, yaitu nilai pixel atau persentasi.

Berikut adalah contoh penggunaan background-position untuk menampilkan logo pada posisi yang berbeda:

```
background-position : nilaiX nilaiY;
```

misalnya

```
background-position: 50px 50px;
```



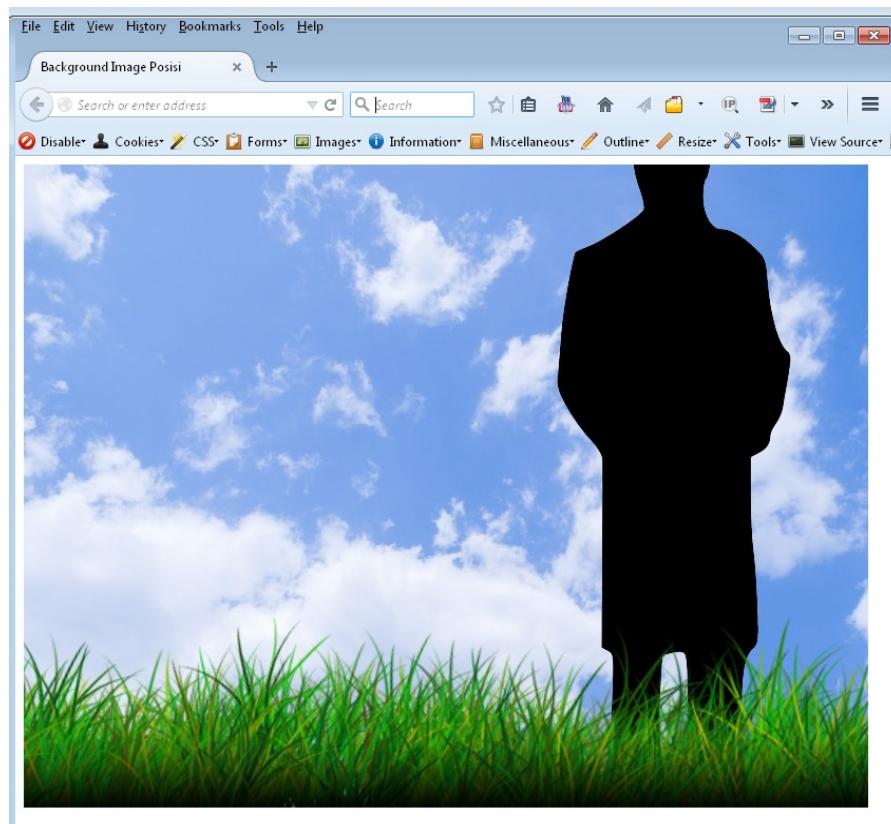
Gambar 55 posisi background ada pada 50px dari kiri dan atas

16.4. background-image merge

Selain memberikan kontrol terhadap perulangan gambar latar, CSS juga memberikan kontrol terhadap posisi dari gambar latar itu sendiri. Hal ini tentunya sangat berguna, terutama untuk mengatur posisi pasti dari gambar latar pada elemen yang sangat besar, seperti penampung dari seluruh elemen dalam halaman.

Salah satu fitur CSS yang sangat memudahkan kita dalam membuat desain yang baik ialah kemudahan yang disediakan untuk memberikan beberapa gambar latar belakang pada sebuah elemen. Cara pemberian juga sederhana, dengan hanya menambahkan daftar latar belakang yang ingin digunakan, dipisahkan dengan tanda ,

```
#main{
    width : 600px;
    height :800px;
    background : url(image/grass.png) 0 100% no-repeat,
                 url(image/man.png) 95% 100% no-repeat,
                 url(image/sky.png) 0 0 no-repeat;
}
```



Gambar 56 Pengabungan dari tiga gambar untuk background.

Bab 17

Floating

Jika sebelumnya untuk mengatur text/paragraf rata kanan atau kiri kita menggunakan property text-align, namun bagaimana jika yang ingin kita atur rata kiri atau rata kanan adalah elemen HTML?

Untuk itu kita dapat menggunakan property float. Nilai yang digunakan yaitu left dan right. Perhatikan contoh pada latihan12.html berikut dan perhatikan apa yang terjadi :

Buatlah markup HTML seperti berikut :

```

<p>Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.</p>
```

Lalu tambahkan CSS berikut :

```
img {
    float: left;
}
```

Style diatas akan merata kirikan elemen gambar dan membuat paragraf naik keatas.



Gambar 57 Penambahan Float akan mempengaruhi element disekitarnya.

17.1. Permasalahan pada floating

Ada satu permasalahan ketika anda menambahkan floating pada suatu elemen, permasalahan ini bisa anda temui pada contoh kasus dalam file latihan13.html berikut.

Buatlah markup HTML seperti berikut :

```

1. <div id="wrapper">
2.   <div id="content">
3.     <div id="artikel"></div>
4.     <div id="sidebar"></div>
5.   </div>
6.
7.   <div id="footer"></div>
8. </div>
9.

```

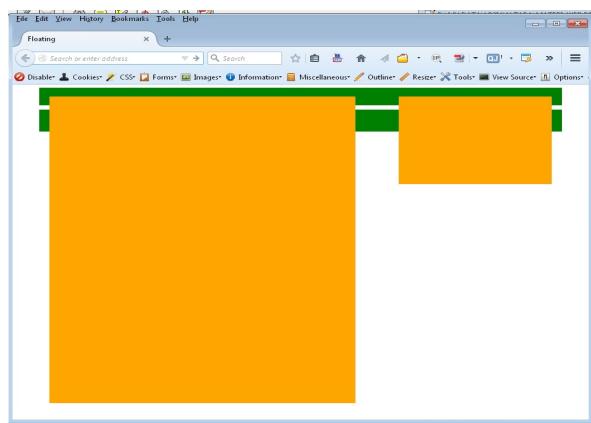
Lalu berilah style berikut :

```

1. #wrapper{
2.   width : 960px;
3.   margin : 0 auto; /*agar wrapper tetap berada ditengah */
4. }
5. #content{
6.   padding:20px;
7.   background : green;
8.   margin-bottom : 10px;
9. }
10. #artikel{
11.   width : 600px;
12.   height : 700px;
13.   float : left;
14.   background: orange;
15. }
16. #sidebar {
17.   width : 300px;
18.   height: 200px;
19.   float : right;
20.   background: orange;
21. }
22. #footer{
23.   width : 100%;
24.   height: 50px;
25.   background: green;
26. }
27.

```

Jika anda buka dalam browser, maka tampilan yang akan dapat adalah seperti ini.



Gambar 58 Permasalahan ketika anda menerapkan Floating

Ada yang janggal? Ya.. #content tidak memenuhi isi dari #artikel dan #sidebar. Seharusnya tinggi #content otomatis sama dengan tinggi dari elemen tertinggi di dalamnya, dalam kasus ini adalah artikel dengan tinggi 700px. Dengan begitu seharusnya tinggi dari #content adalah 740px (ditambah dengan padding-top dan padding-bottom). Jika anda tidak mengerti cobalah untuk membuang property float yang ada, dan perhatikan bahwa #artikel dan #sidebar berada di dalam #content.

Hal ini dikarenakan kita menambahkan floating pada #artikel dan #sidebar, sehingga #content menganggap tidak ada element didalamnya. Ada dua cara untuk menyelesaikan kasus ini :

1. Menambahkan style overflow:hidden pada #content

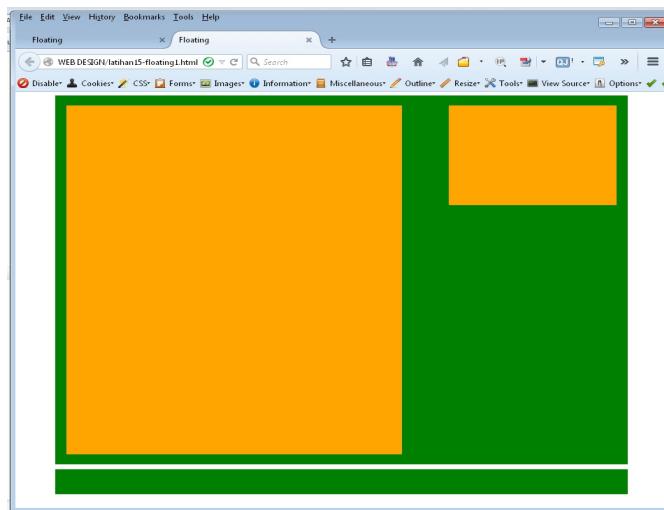
```
#content{
    ...
    overflow:hidden;
}
```

2. Menambahkan div kosong dengan style clear:both; sebelum penutup content

```
...
<div id="sidebar"></div>
<div style="clear:both;"></div>
</div>
```



Float berarti melayang. Ketika kita memberikan float pada elemen maka elemen tersebut seolah-olah melayang dan elemen di atasnya (parent) menganggap tidak ada elemen didalamnya (child).



Gambar 59 Penyelesaian masalah Float

17.2. CSS Reset

Pernahkah anda bertanya darimana warna link dari tag `<a>` berasal, atau mengapa tag `` bisa menebalkan teks, atau kenapa tag `<h1>` sampai tag `<h6>` memiliki ukuran font yang berbeda-beda dan ketebalan yang berbeda? Padahal kita sama sekali belum menambahkan CSS pada markup yang kita buat?

Setiap browser memiliki Style atau File CSS yang akan diterapkan pada setiap halaman web yang dibuka. Keberadaan style yang dimiliki browser ini sangat membantu namun terkadang juga membingungkan bagi sebagian orang.

CSS Reset berfungsi untuk mereset atau mengatur ulang seluruh property-property yang telah disetting oleh browser, seperti margin, padding, warna huruf (link) ukuran huruf heading dan lain sebagainya. Perhatikan CSS contoh berikut :

```
* {
    margin:0;
    padding:0;
}
```

Selector bintang (*) digunakan untuk memberikan style pada Seluruh elemen/Tag HTML. Dalam kasus ini, kita akan mengatur ulang margin dan padding menjadi 0 pada seluruh elemen HTML yang ada.

Bab 18

Positioning

Dalam CSS kita pun dapat memposisikan (*Positioning*) elemen sesuai kehendak kita. Hal ini membuat desain web dengan CSS lebih fleksibel karena kita bisa menempatkan elemen-elemen HTML sesuai dengan keinginan kita atau tuntutan desain web.

18.1. Static

Secara default, seluruh elemen/tag HTML memiliki positioning static. Artinya anda belum bisa/tidak dapat menentukan posisi elemen tersebut. Ketika anda ingin merubah posisi dari elemen tersebut maka anda harus memberikan `position:relative` dan selanjutnya anda dapat menggunakan property top, right, bottom, dan left untuk menentukan posisi elemen yang bersangkutan.

Sebagai contoh perhatikan kode HTML dan CSS berikut :

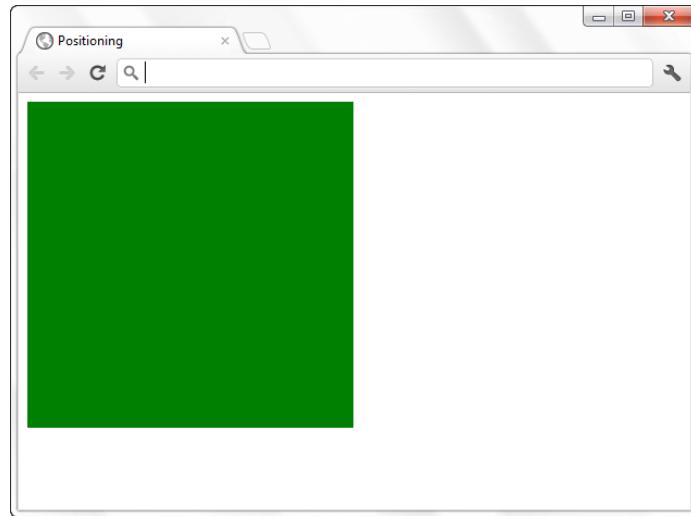
```
<div id="box"></div>
```

Berikut kode CSSnya.

```
#box{
    width : 300px;
    height : 300px;
}
```

Ketika anda tidak memberikan nilai pada property position, maka secara default, #box memiliki position:static, jadi anda tidak dapat menggunakan property top, right, bottom dan left untuk menentukan posisi #box tersebut :

```
#box{
    background : green;
    width : 300px;
    height : 300px;
    top : 100px;
    left : 200px;
}
```

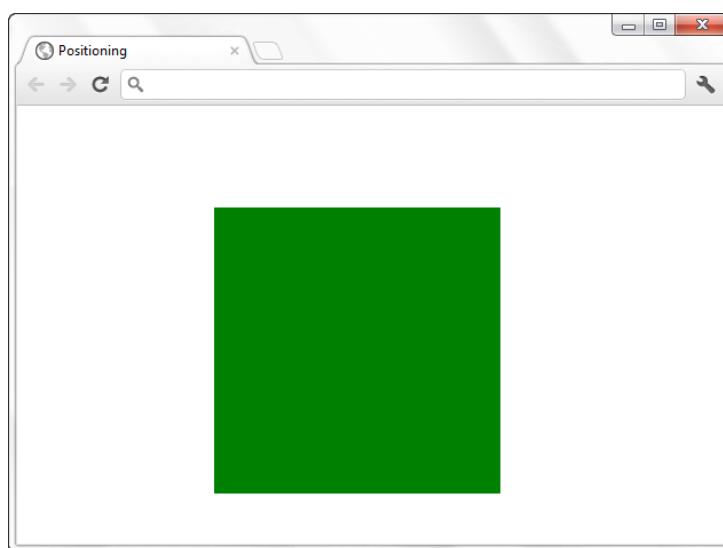


Gambar 60 Box static tidak dapat ditentukan posisinya

18.2. Relative

Sekarang cobalah untuk menambahkan `position:relative` pada `#box` dan perhatikan bahwa box tersebut bergeser 100px dari atas dan 200px dari kiri.

```
#box{  
    background : green;  
    width : 300px;  
    height : 300px;  
    top : 100px;  
    left : 200px;  
    position:relative;  
}
```



Gambar 61 Penambahan position relative memungkinkan penempatan box

18.3. Absolute

Penambahan position:absolute akan membuat elemen tersebut terlepas dari aturan penempatan elemen dan dibatasi oleh parent yang memiliki positioning relative. Sulit memahami apa yang saya tulis diatas, kita langsung ke contoh kasus.

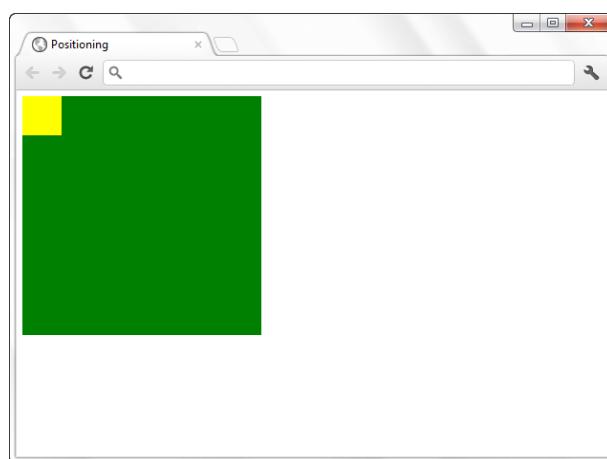
Buatlah markup HTML seperti berikut :

```
<div id="box">
    <div id="box2"></div>
</div>
```

Lalu berilah style seperti berikut :

```
#box{
    width:300px;
    height:300px;
    background:green;
}

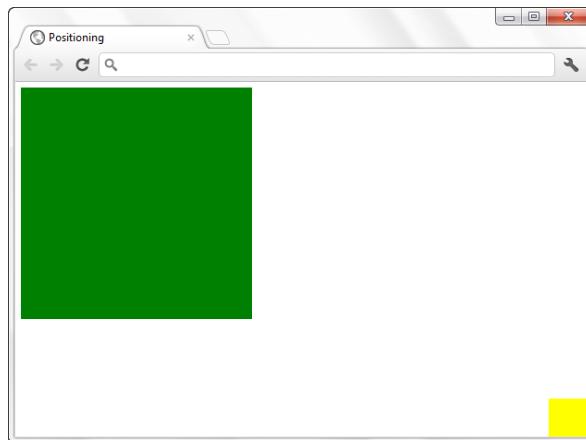
#box2{
    width:50px;
    height:50px;
    background:yellow;
}
```



Gambar 62 Child elemen akan berada di dalam parentnya

#box2 secara aturan ditampilkan berada di dalam div #box, anda dapat terlepas dari aturan tersebut dengan menambahkan positioning absolute. Sekarang tambahkan position:absolute dan tentukan property bottom dan right.

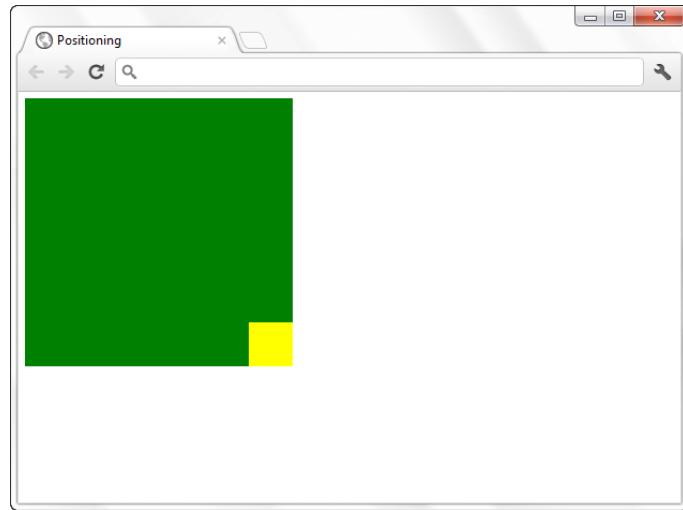
```
#box2{  
    width:50px;  
    height:50px;  
    background:yellow;  
    position:absolute;  
    bottom:0;  
    right:0;  
}
```



Gambar 63 #box2 akan terlepas dari parent ketika ditambahkan position absolute

#box2 akan terlepas dari aturan penempatan elemen (keluar dari #box) dan berada di kanan bawah browser. Untuk membatasi pergerakan dari #box2 yang memiliki positioning absolute, maka kita harus menambahkan positioning relative pada parentnya yakni #box.

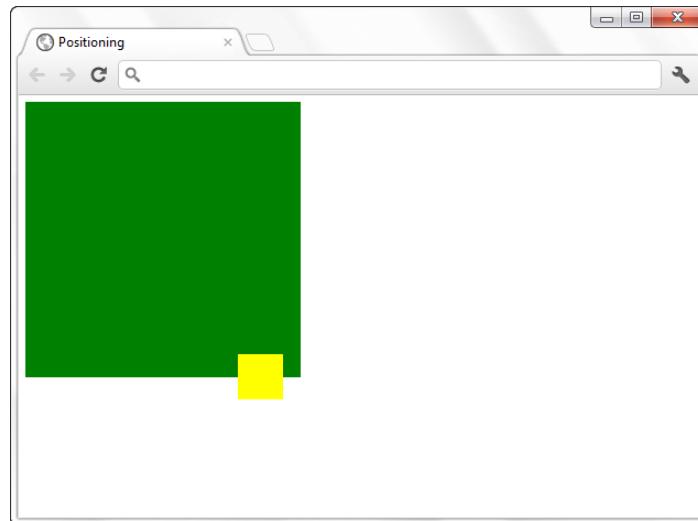
```
#box{  
    width:300px;  
    height:300px;  
    background:green;  
    position:relative;  
}
```



Gambar 64 Posisi absolute position dapat dibatasi oleh elemen dengan posisi relative

Sekarang #box2 berada di kanan bawah parentnya #box. Andapun dapat bermain dengan nilai negative untuk mendapat efek tertentu.

```
#box2{  
    width:50px;  
    height:50px;  
    background:yellow;  
    position:absolute;  
    bottom:-25px;  
    right:20px;  
}
```



Gambar 65 Pengaturan posisi dengan absolute

Perhatikan contoh penggunaan absolute position yang sering ditemukan di internet berikut :



Gambar 66 Contoh Kasus penggunaan Absolute Position

Perhatikan contoh berikut tombol untuk sharing via facebook atau twitter dan nomor posting ditampilkan berjajar se secara vertical dan setengah keluar dari kotak artikel.



*Sulit memahami Relative-Absolute? Bayangkan "Relative" sebagai **sangkar** bagi "Absolute", oleh karena itu pergerakan "Absolute" dibatasi oleh "Relative".*

18.4. Fixed

Sifat dari position fixed hampir sama dengan absolute, namun posisi dari elemen fixed akan tetap berada posisi yang ditentukan walaupun browser dizoom-in/out atau scroll ke atas atau ke bawah.

Perhatikan contoh kode HTML dan CSS berikut :

```
<div id="social"></div>
```

CSS :

```
#social{
    width:30px;
    height:100px;
    position:fixed;
    left:0;
    top:100px;
}
```

Berikut ini adalah contoh penggunaan position:fixed yang sering ditemui di internet.



Gambar 67 Contoh Kasus "Jadul" penggunaan fixed position

Bagian Sign Up di situs kaskus.co.id menggunakan position fixed sehingga ketika anda men-scroll halaman web, bagian tersebut tetap berada disana.

Bab 19

Pseudo-Class

Pseudo Class digunakan untuk memberikan efek-efek tertentu pada selector. Misalnya efek ketika cursor berada di atas suatu elemen (hover), ketika cursor meng-klik suatu elemen dan lain sebagainya.

Berikut adalah cara penulisan pseudo class :

```
selector:pseudo-class { property:value; }
```

Misalnya kita akan menambahkan pseudo-class :hover pada sebuah link

```
a:hover {
    color : red;
}
```

Style tersebut akan memberikan efek perubahan warna link menjadi merah **ketika** cursor berada di atasnya.



Gambar 68 Pseudo-Class :hover memberikan efek ketika cursor berada di atas elemen

19.1. Pseudo-Class untuk Link/Anchor

Ada beberapa pseudo-class untuk anchor/link yakni :

```
/* untuk link yang belum di klik */
a:link { color: blue; }

/* untuk link yang sudah di klik */
a:visited { color:green; }

/* ketika cursor berada di atas link */
a:hover { color:red; }

/* ketika link dalam keadaan aktif atau terpilih */
a:active { color:purple; }
```

```
a:active { color: orange; }
```

Perhatikan juga urutan ketika penulisan semua pseudo class tersebut (Jika anda bermaksud menuliskan seluruh pseudo class untuk link):

- :hover harus ditulis sesudah :link dan :visited
- :active harus ditulis sesudah :hover.

Jika tidak demikian maka efek tersebut tidak akan berjalan dengan baik.

Pseudo Class :hover dan :active tidak hanya bisa digunakan untuk link saja melainkan bisa diterapkan pada elemen/tag-tag HTML lainnya.

19.2. Pseudo-Class :first-child dan :last-child

Selain pseudo-class untuk link/anchor ada juga pseudo class :first-child dan :last-child. Keduanya digunakan untuk memilih elemen pertama dan terakhir dari suatu selector.

Misalnya kita mempunyai sebuah unordered list (ul), dengan beberapa list-item di dalamnya. Kita bisa memberikan style khusus untuk list-item yang pertama dan yang terakhir dengan menggunakan :first-child dan :last-child.

```
<ul>
    <li>Item ke 1</li>
    <li>Item ke 2</li>
    <li>Item ke 3</li>
    <li>Item ke 4</li>
    <li>Item ke 5</li>
</ul>
```

Dan Jika kita tambahkan style berikut :

```
ul li:first-child{
    color:green;
}

ul li:last-child{
    color:orange;
}
```

Maka yang akan kita dapatkan adalah tampilan seperti berikut :

- Item ke 1
- Item ke 2
- Item ke 3
- Item ke 4
- Item ke 5

Gambar 69 Pemberian pseudo class :first-child dan :last-child

List-item yang pertama akan memiliki warna huruf hijau dan yang terakhir memiliki warna orange. Tidak hanya pada unordered list, andapun dapat menerapkannya pada deretan elemen lainnya. Misalnya elemen p yang terdapat dalam suatu div.

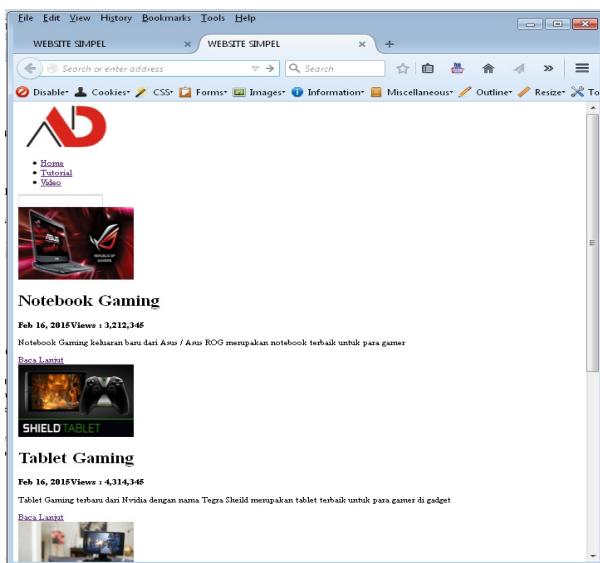
```
<div>
  <p> Paragraf pertama </p>
  <p> Paragraf kedua </p>
  <p> Paragraf ketiga </p>
  <p> Paragraf keempat </p>
</div>
```

Bab 20

StudiKasus 2 – Pemberian Style/CSS

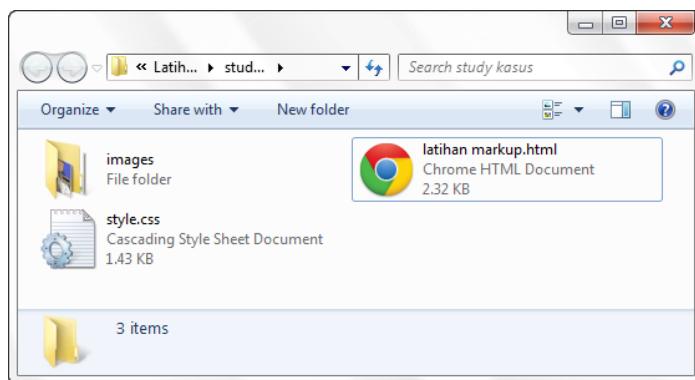
Pada studi kasus sebelumnya, anda telah membuat markup HTML dari sebuah desain web sederhana. Hasil yang didapatpun terasa polos, tidak menarik dan yang terpenting tidak sama dengan desain web yang diinginkan.

Pada bagian ini kita akan belajar menambahkan style/CSS pada file studi kasus yang kita buat sebelumnya.



Gambar 70 Markup HTML tanpa CSS

Pertama-tama, buatlah file CSS baru dengan nama style.css, simpan di folder yang sama dengan lokasi file HTML.



Gambar 71 Penempatan file latihan

Lalu tambahkan tag link untuk memanggil file style.CSS pada bagian header.

```
<head>
    <title>Latihan Markup</title>
    <link rel="stylesheet" href="style.CSS" />
</head>
```

20.1. CSS Reset

Pertama-tama kita atur ulang margin dan padding agar tidak menggunakan style bawaan dari browser, dan kita tentukan jenis font standar yang akan kita gunakan.

```
*{
    margin:0;
    padding:0;
    font-family:calibri, "segoe ui", arial, tahoma, sans-serif;
}
```

20.2. Body

Dasar background dari desain web berwarna abu-abu, jadi kita tentukan warna tersebut pada tag body:

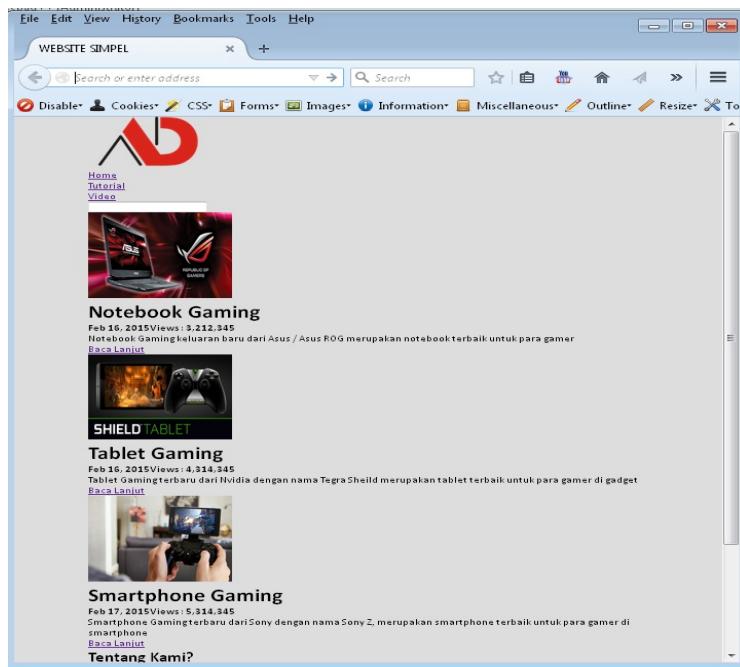
```
body{
    background-color:#DDDDDD;
}
```

20.3. Wrapper

Kita mulai dari wrapper. Seperti saya jelaskan sebelumnya, kita membutuhkan sebuah div untuk menampung seluruh isi situs karena kita akan membuatnya menjadi rata tengah. Jadi meskipun browser di zoom-in atau di zoom-out, konten tetap berada di tengah

```
#wrapper{
    width : 780px;
    margin: 0 auto;
    overflow: hidden;
}
```

Margin: 0 auto, akan memberikan jarak dari atas sebesar 0, dan untuk sisi kiri-kanannya kita beri nilai auto sehingga membuat div ini berada di tengah.



Gambar 72 Konten tetap berada ditengah meski di-zoom-in atau zoom-out

Pemberian nilai 780px untuk width bukan tanpa alasan, ini dikarenakan rata-rata pengguna internet didunia menggunakan layar dengan resolusi minimal 1024x768px jadi lebar dari situs kita berada pada ukuran aman meskipun resolusi layar yang digunakan minimal.

Sebenarnya ukuran lebar minimal standar dari sebuah desain web adalah 960px namun karena pada desain lebar dari web sebesar 780px maka kita gunakan nilai ini.

Penambahan `overflow:hidden` karena kita akan menambahkan floating pada daftar-artikel dan sidebar. (Baca kembali : Floating)

20.4. Header

 Pada bagian header, kita akan menggunakan gambar yang telah saya siapkan, ukuran lebar gambar hanya 10px namun dengan metode pengulangan/repeat-x kita akan mendapatkan tampilan yang sama dengan yang ada di dalam desain.



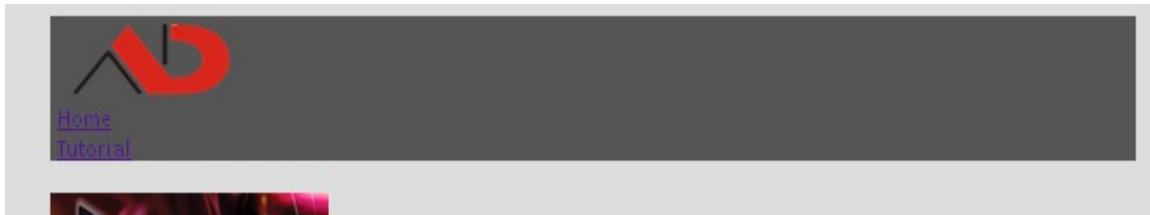
Gambar 73 Hasil akhir dari Head

```
#header{
    background: url(images/header.jpg) repeat-x #555;
    height : 70px;
    width : 100%;
    padding:5px;
    margin : 20px 0;
    overflow : hidden;
}
#header img{ width:130px ; height:35px ; }
```

Kita menggunakan nilai 100% agar lebar dari header sama dengan lebar #wrapper. Dan untuk background anda bisa tebak apa yang saya lakukan. Ya, saya menggunakan CSS-shorthand untuk memanggil gambar.

Pertama kita tentukan lokasi gambar (background-image) lalu jenis pengulangan (background-repeat) dan terakhir kita tentukan background-color sebagai fallback ketika gambar tidak berhasil dimuat.

Kita juga tambahkan `overflow : hidden` karena di dalam header nantinya akan ada elemen yang memiliki floating (menu dan form).



Gambar 74 tampilan Header setelah diberi style

20.5. Menu

Tidak ada style yang perlu anda tambahkan pada logo, jadi kita lanjut ke bagian menu. Menu yang kita buat adalah sebuah daftar/list yang secara-default ditampilkan menurun.

Kita dapat membuat daftar tersebut menjadi rata-kanan dengan teknik floating.

```
#menu-utama{
    margin-top:3px;
    list-style:none;
}
```

Kita beri jarak antara menu dengan logo sebesar 3px, dan property list-style:none berfungsi untuk menghilangkan lingkaran didepan setiap list.

Selanjutnya kita akan membuat list/daftar menjadi rata-kiri/sejajar dengan menggunakan properti display:inline :

```
#menu-utama li{
    float:left;
    padding:8px 20px;
    border-right:1px solid #555555;
}
```

Warna teks pada menu (biru) dan garis bawah berasal dari tag <a> jadi kita ubah warna dan menghilangkan garis bawahnya.

```
#menu-utama li a{
    color : #ffffff;
    text-decoration:none;
}
```



Gambar 75 Penampilan Menu utama setelah diberi style

20.6. Form

Bagian form pencarian ini sangat sederhana, kita hanya akan menentukan tinggi dan meratakan kanan form ini.

```
#header form{
    float:right;
    margin-top:7px;
    margin-right: 10px;
}
```

Selanjutnya kita hilangkan border bawaan dari tag input, dan kita tentukan tinggi dari input ini.

```
#header form input{
    border:none;
    height:20px;
}
```

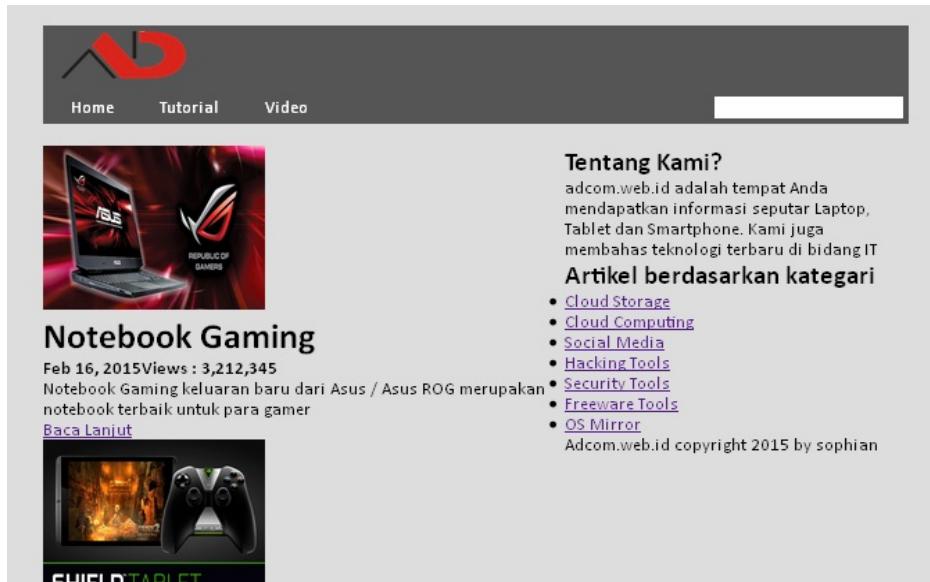


Gambar 76 Penampilan akhir header

20.7. Daftar Artikel

Selanjutnya kita beri style pada #daftar-artikel. Yang akan kita tentukan adalah lebar/width dari div ini.

```
#daftar-artikel{
    width:470px;
    float:left;
}
```



Gambar 77 Pengaturan Width dan rata kiri pada Daftar Artikel

20.8. Artikel

Jika anda perhatikan, setiap div artikel memiliki background putih dan antar artikel memiliki jarak.

```
.artikel{
    background:#ffffff;
    padding:10px;
    margin-bottom:20px;
    overflow:hidden;
}
```

Perhatikan juga, gambar dari setiap artikel bisa kita setting dengan floating agar sama dengan desain yang diberikan (karena gambar memiliki nilai float, maka kita tambahkan overflow:hidden pada .artikel)

```
.artikel img{
    float:left;
    margin-right:10px;
    width:200px;
    height:165px
}
```

Tak ada yang perlu dirubah untuk judul artikel. Yang perlu kita beri style adalah tanggal dan view. Kita akan menambahkan warna background pada kedua bagian tersebut.

```
.artikel b{
    color:#616060;
    background:#DDDDDD;
    padding:3px;
    font-size:12px;
    font-weight:bold;
    margin-right:10px;
}
```

Karena kedua elemen tersebut menggunakan tag b, maka kita tinggal menambahkan selector tersebut. Kita juga mengatur ulang ukuran font dan ketebalan dari font tersebut.

Beri jarak antar sinopsis dengan tanggal dan link di bawahnya.

```
.artikel p{
    margin:10px 0;
}
```

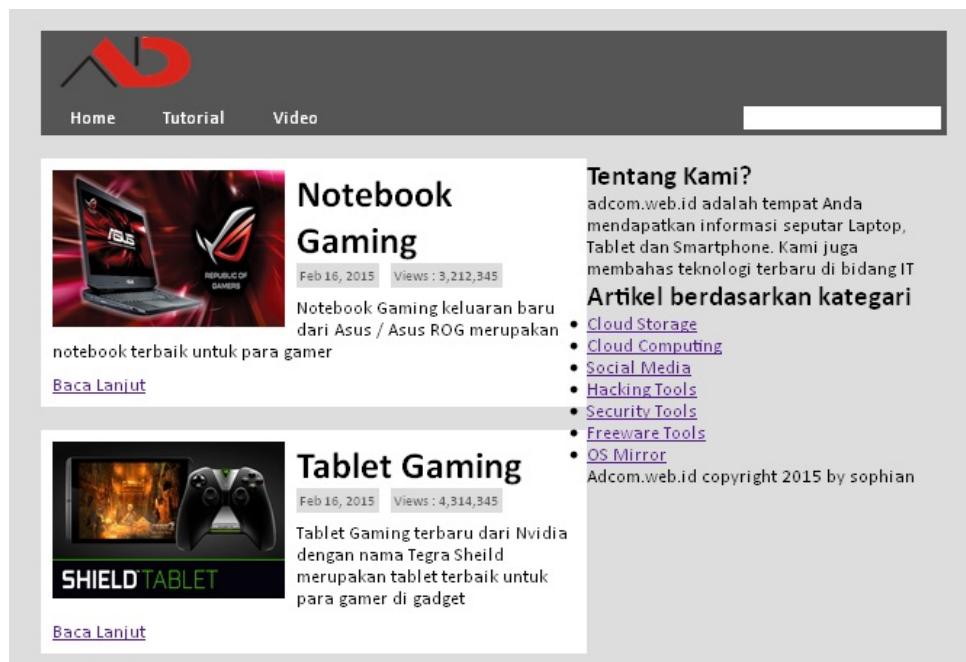
Terakhir dari bagian artikel ini, adalah penempatan link “watch this” ke sebelah kiri.

```
.artikel a{
    display:block;
    text-align:right;
    color:#636363;
}
```

Kita menggunakan metode lain, selain menggunakan floating. Yaitu dengan text-align. Namun pertama-tama kita harus mengatur elemen tersebut menjadi block.



Ukuran lebar dari elemen adalah lebar dari konten/text yang terdapat di dalamnya. Dengan menambahkan display:block maka elemen tersebut akan ditampilkan secara block/full width.



Gambar 78 Pengaturan link Watch This

20.9. Sidebar

Kita lanjut ke bagian Sidebar, pertama-tama kita tentukan lebar dari sidebar ini dan mengatur posisinya menjadi rata-kanan.

```
#sidebar{
    width:270px;
    float:right;
}
```

Isi dari sidebar sangat mudah untuk disesuaikan dengan desain, kita mulai dari judul setiap item dalam sidebar.

```
#sidebar h2{
    font-size:18px;
    margin:10px 0;
}
```

Lalu kita tentukan jarak perbaris dari text.

```
#sidebar p{
    line-height:20px;
    color : #636363;
}
```



Gambar 79 Pemberian style pada sidebar

Yang perlu kita beri style selanjutnya adalah daftar kategori yang terlalu menjorok ke kiri. Kita dapat merubahnya dengan menambahkan margin-left pada tag ul.

```
#sidebar ul{
    margin-left:10px;
}
```

Sama halnya dengan menu utama, warna biru dan garis bawah berasal dari tag link. Jadi style yang diberikan tidak jauh berbeda.

```
#sidebar ul li a{
    color:#636363;
    text-decoration:none;
}
```

20.10. Footer

Terakhir kita akan menggarap bagian footer. Perhatikan bahwa lokasi footer berada pada sidebar. Ini disebabkan oleh floating yang kita tambahkan pada daftar-artikel dan sidebar.

Untuk memperbaikinya kita bisa menggunakan clear:both; pada footer.

```
#footer{
    clear:both;
    margin:20px 0;
    text-align:center;
    background-color: #f0f0f0;
    padding:5px;
    color:#555555;
}
```

Saya yakin anda sudah mengerti mengapa saya menambahkan property-property tersebut, jika masih bingung silahkan baca kembali bab-bab sebelumnya.



Home Tutorial Video

 **Notebook Gaming**

Feb 16, 2015 Views : 3,212,345

Notebook Gaming keluaran baru dari Asus / Asus ROG merupakan notebook terbaik untuk para gamer

[Baca Lanjut](#)

 **Tablet Gaming**

Feb 16, 2015 Views : 4,314,345

Tablet Gaming terbaru dari Nvidia dengan nama Tegra Sheild merupakan tablet terbaik untuk para gamer di gadget

[Baca Lanjut](#)

 **Smartphone Gaming**

Feb 17, 2015 Views : 5,314,345

Smartphone Gaming terbaru dari Sony dengan nama Sony Z, merupakan smartphone terbaik untuk para gamer di smartphone

[Baca Lanjut](#)

Tentang Kami?

adcom.web.id adalah tempat Anda mendapatkan informasi seputar Laptop, Tablet dan Smartphone. Kami juga membahas teknologi terbaru di bidang IT

Artikel berdasarkan kategori

- Cloud Storage
- Cloud Computing
- Social Media
- Hacking Tools
- Security Tools
- Freeware Tools
- OS Mirror

Adcom.web.id copyright 2015 by sophian

Gambar 80 Hasil Akhir Latihan 1