



Belajar HTML & CSS

"Tutorial Fundamental dalam Mempelajari HTML & CSS"

Rian Ariona

PROLOG

Untuk siapa buku ini ?

Buku ini cocok untuk anda yang ingin mempelajari HTML dan CSS, baik anda yang tidak tahu sama sekali apa itu HTML dan CSS hingga anda yang ingin mempelajari secara lanjut HTML dan CSS, serta mengetahui konsep fundamental dari HTML dan CSS itu sendiri.

Pembahasan dalam buku ini menitikberatkan 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 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	v
Untuk siapa buku ini ?	v
Watch the sign!	v
Apa yang disertakan dalam Paket Ebook	vi
DAFTAR ISI	VII
BAB 1 PERSIAPAN	2
1.1. Web Browser	2
1.2. Text Editor	5
1.3. Add-On	7
BAB 2 PENGENALAN HTML	10
2.1. Apa itu HTML?	10
2.2. Membuat Website dalam 1 Menit	11
2.3. Pengenalan Tag HTML	12
2.4. Struktur file HTML	13
BAB 3 MEMUAT GAMBAR	16
3.1. Mengenal Atribut HTML	16
3.2. Cara penulisan lokasi file	19
BAB 4 MEMBUAT LINK	20
4.1. Link Standar	20
4.2. Link Email	24
BAB 5 HEADING/PENJUDULAN	25
BAB 6 MEMBUAT DAFTAR/LIST	28
6.1. Ordered List	28
6.2. Unordered List	29
6.3. Definition List	30
6.4. List/Daftar bersarang	30
BAB 7 MEMAHAMI TAG DIV (DIVISION)	32
7.1. Pakai ID atau Class?	33
7.2. Memahami hubungan Child, Parent dan Siblings	34
BAB 8 FORM	35
Control-control Form	36

BAB 9 TABEL	41
9.1. Struktur tabel yang dianjurkan	42
9.2. Kesalahan dalam penggunaan <table>	43
BAB 10 STUDI KASUS 1- KONVERSI DESAIN KE HTML	44
10.1. Pesiapan	44
10.2. Wrapper	45
10.3. Header	45
10.4. Content	46
10.5. Sidebar	48
10.6. Footer	49
BAB 11 VALIDASI MARKUP HTML	52
BAB 12 SEMANTIK	55
12.1. Maksud Semantik dalam HTML	56
12.2. Pentingkah Semantik?	56
BAB 13 CASCADING STYLE SHEET (CSS)	58
13.1. Pengenalan CSS	58
13.2. Penulisan CSS	60
BAB 14 BOX-MODEL	62
14.1. Margin	62
14.2. Padding	63
14.3. CSS-Shorthand	64
14.4. Border	66
BAB 15 TYPOGRAPHY	68
Apa itu Sans-serif dan Serif	69
BAB 16 CSS-IMAGE	72
16.1. background-image	72
16.2. background-repeat	73
16.3. background-position	74
16.4. Image Sprite	76
BAB 17 FLOATING	78
17.1. Permasalahan pada floating	79
17.2. CSS Reset	81
BAB 18 POSITIONING	82
18.1. Static	82
18.2. Relative	83
18.3. Absolute	84
18.4. Fixed	87

BAB 19 PSEUDO-CLASS	89
19.1. Pseudo-Class untuk Link/Anchor	89
19.2. Pseudo-Class :first-child dan :last-child	90
BAB 20 STUDIKASUS 2 – PEMBERIAN STYLE/CSS	92
20.1. CSS Reset	93
20.2. Body	93
20.3. Wrapper	93
20.4. Header	94
20.5. Menu	95
20.6. Form	96
20.7. Daftar Artikel	97
20.8. Artikel	97
20.9. Sidebar	99
20.10. Footer	100
BAB 21 STUDI KASUS 3 – SLICING	102
21.1. Penulisan Markup HTML	104
21.2. Slicing	108
21.3. Pemberian Style	112
BAB 22 JAVASCRIPT	117
22.1. Pengenalan Javascript	117
22.2. jQuery	119
22.3. jQuery Plugin	120
BAB 23 STUDI KASUS 4 – PENAMBAHAN JQUERY SLIDER	122
Cara penggunaan basic-slider	122
Penerapan pada Studi Kasus ke-3	123
BAB 24 EXTRAS	126
24.1. Developer Tools dan Firebug	126
24.2. Zen Coding	127
24.3. Menambahkan Dummy Text	129
24.4. Buat situs anda “online”	130
24.5. Menggunakan FTP Client	134
EPILOG	136
DAFTAR PUSTAKA	137
Buku	137
Website	137

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).

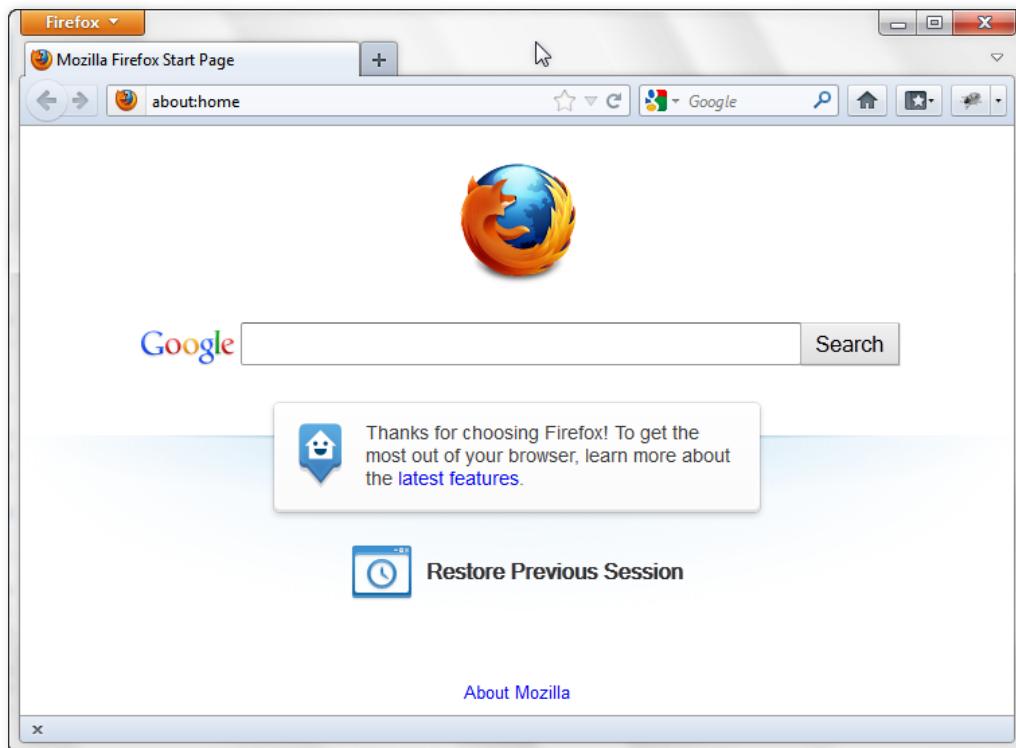
Setiap browser memiliki perbedaan dalam hal menampilkan halaman web dan fitur-fitur yang didukung dalam HTML dan CSS. Boleh jadi halaman web yang anda buat ditampilkan benar pada salah satu browser namun acak-acakan pada browser lainnya. Untuk itu perlu beberapa browser yang terinstall dalam komputer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

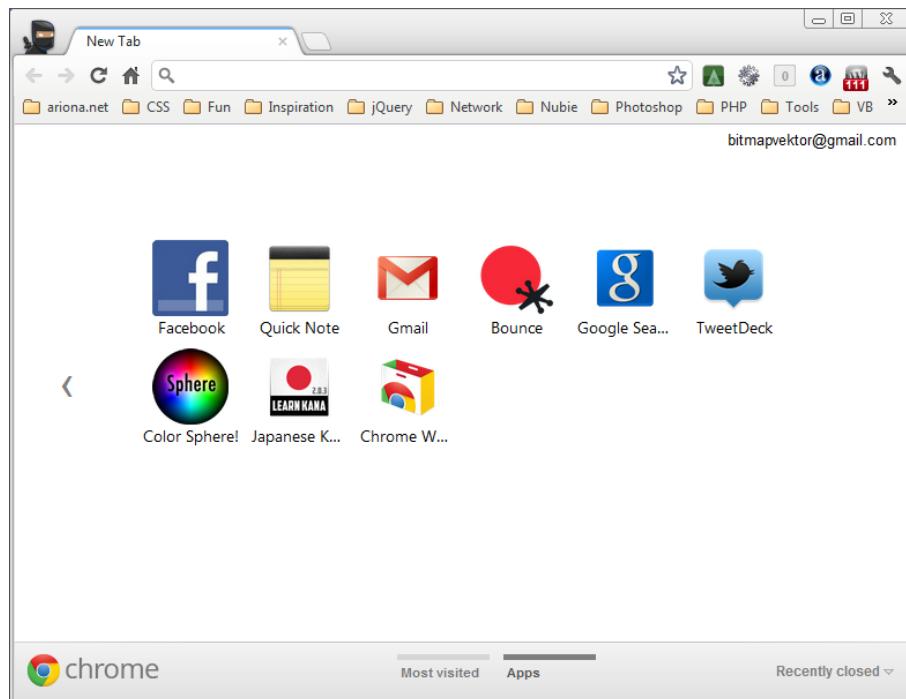
Google Chrome

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.

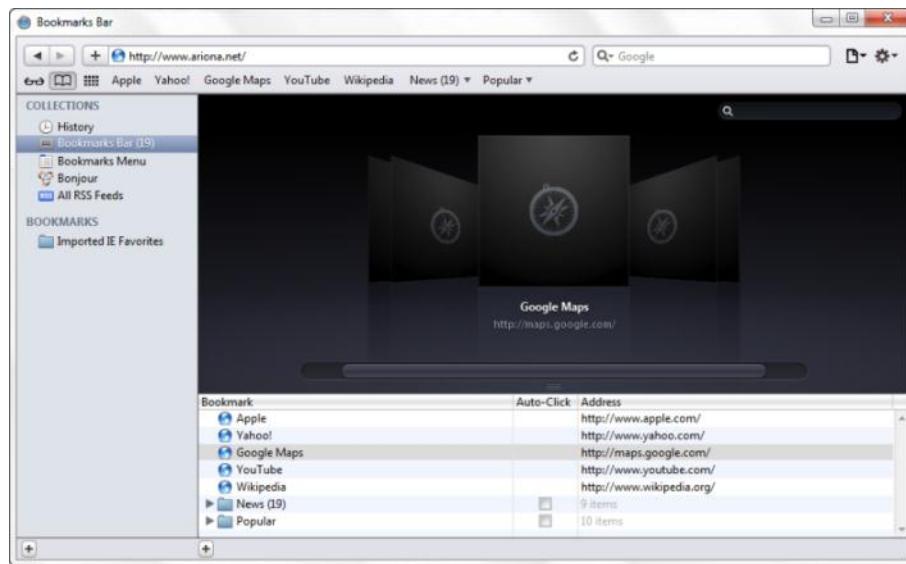


Gambar 2 Google Chrome 17

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>

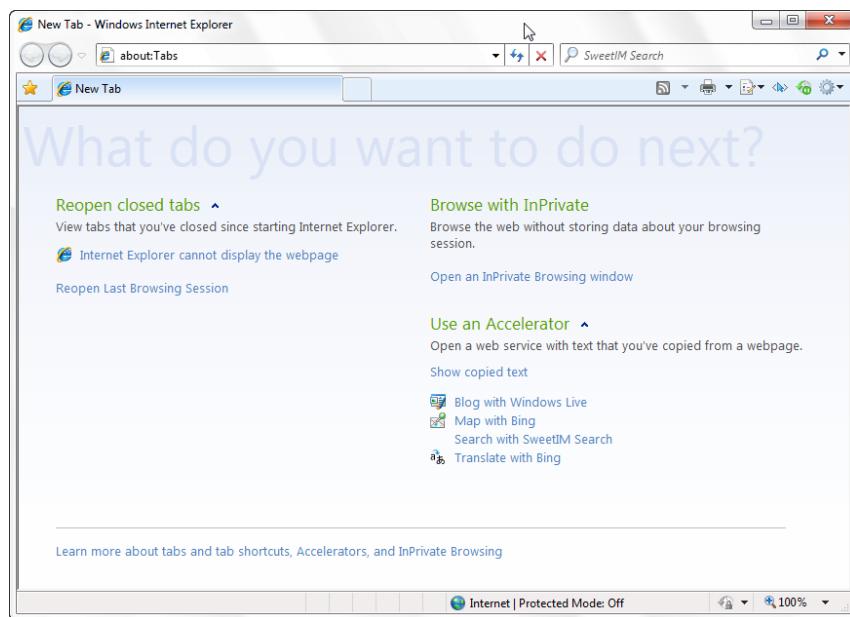


Gambar 3 Safari 5

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 8

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

```

Gambar 5 Notepad++

Sublime Text

Text editor yang terbilang masih baru yang sangat mudah digunakan, penampilan simple namun enak dipandang. Sublime text adalah aplikasi berbayar tapi anda dapat mendownload versi demo-nya (meskipun versi demo tapi tidak ada batasan dalam penggunaannya).

Sublime text tersedia untuk sistem Operasi Windows, Linux dan Mac, download di alamat berikut <http://www.sublimetext.com/2>

```

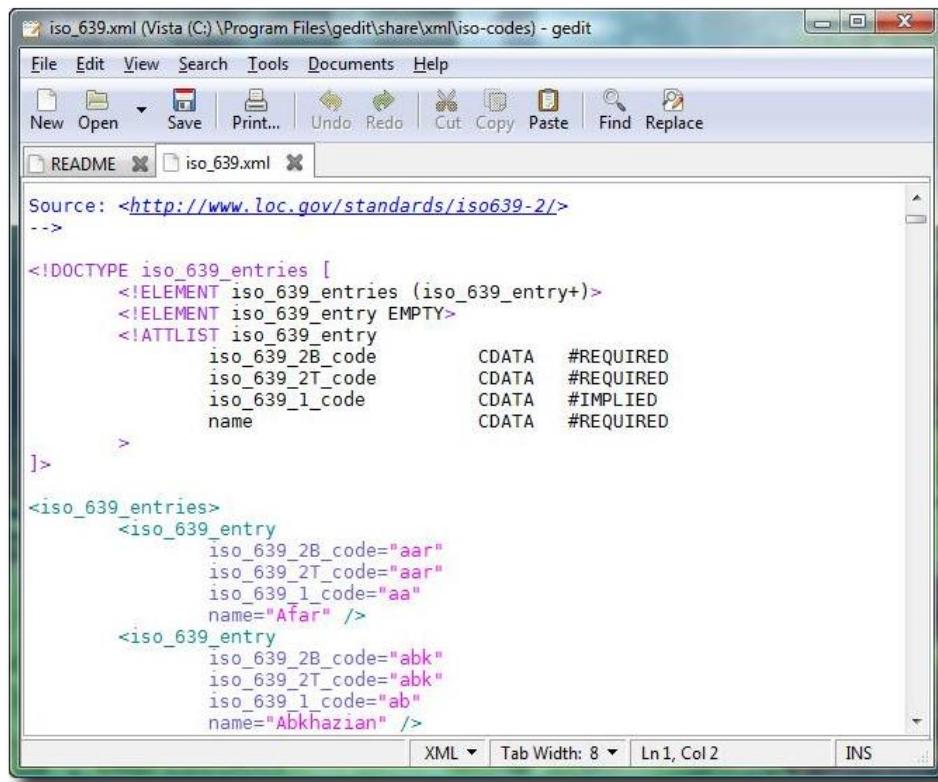
1 <?php
2 echo for Gambar 6 Sublime Text2
3 ?>
4 <fieldset>
5   <legend>Formulir Penambahan Siswa Baru</legend>
6 <div class="row">
7   <div class="span8">
8     <div class="clearfix">
9       <label for="no_induk">No Induk</label>
10      <div class="input">
11        <input type="text" name="no_induk" class="span2">
12      </div>
13    </div>
14    <div class="clearfix">
15      <label for="nisn">NISN</label>
16      <div class="input">
17        <input type="text" name="nisn">
18      </div>
19    </div>
20    <div class="clearfix">
21      <label for="nama">Nama Lengkap</label>
22      <div class="input">
23        <input type="text" name="nama" class="xlarge">

```

Gedit

Anda pengguna Linux? Khususnya Linux dengan desktop Gnome sudah memiliki text editor bawaan, GEdit. GEdit bisa digunakan untuk menuliskan berbagai macam bahasa pemrograman.

Anda pengguna Windows-pun dapat mendownload GEdit versi windows di situs resminya <http://projects.gnome.org>.

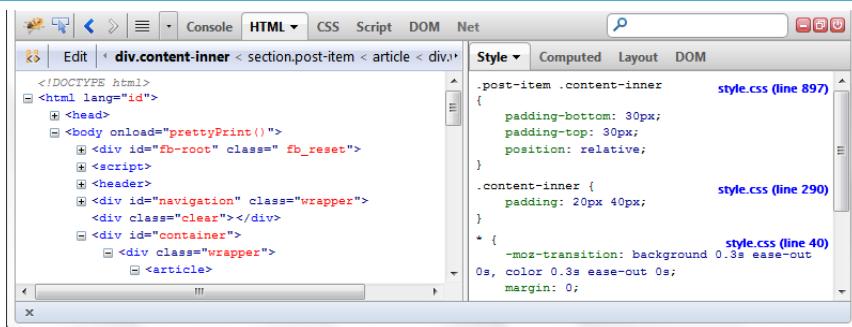


Gambar 7 GEdit untuk Windows

1.3. Add-On

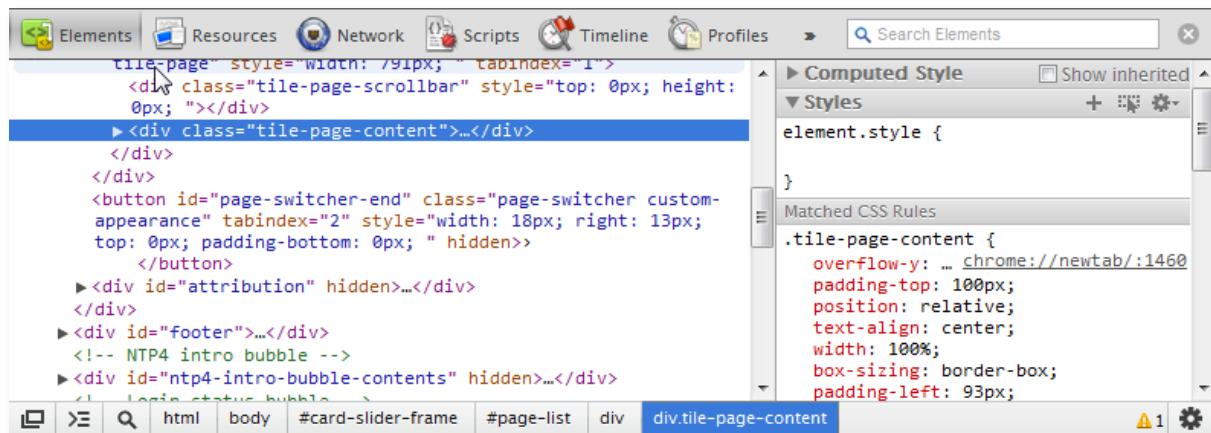
Firebug

Firebug akan kita gunakan untuk memeriksa kode HTML dari setiap halaman web, melihat CSS yang digunakan dan untuk menguji script javascript. Anda dapat menginstallnya pada browser Firefox. Untuk menginstallnya cukup masuk ke halaman Add-on dan cari add-on firebug.



Gambar 8 Tampilan Firebug pada firefox

Jika anda menggunakan Google Chrome atau Safari, tidak usah menginstall firebug karena kedua browser dengan engine webkit ini sudah memiliki tool serupa yang dinamai dengan Developer Tool.

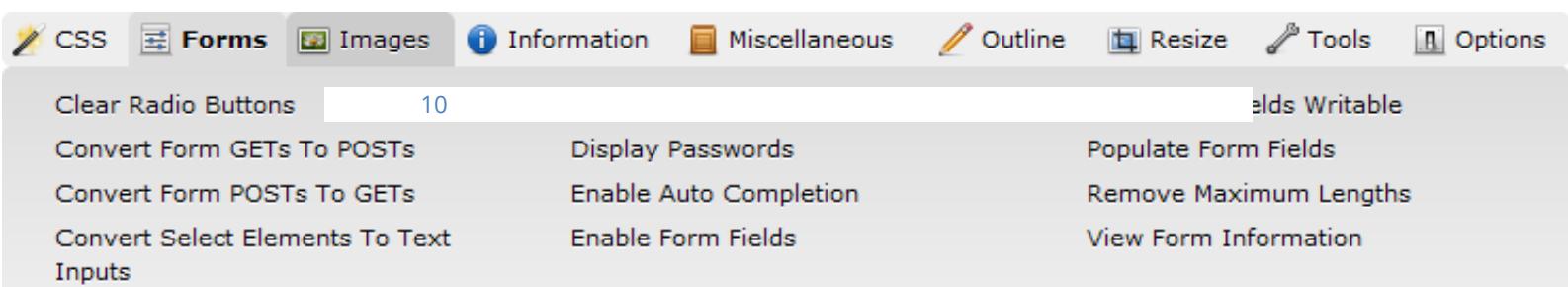


Gambar 9 Developer Tools pada Google Chrome

Web Developer Tools

Add-on ini digunakan untuk menguji halaman web dan melakukan beberapa perubahan/perbaikan sementara terhadap halaman web. Anda dapat menginstallnya dari add-on firebug atau dari Chrome Web Store.

Gambar 10 Web Developer Tools memiliki sejumlah tools yang sangat membantu



Itulah aplikasi-aplikasi yang diperlukan untuk mempelajari HTML dan CSS, tapi yang terpenting adalah kehadiran **Web Browser** dan **Text editor**, sisanya adalah *optional* yang keberadannya hanya mendukung aktifitas kita dalam pembangunan halaman web. Seperti Firebug dan Developer tools memang penting namun anda tidak harus menginstall keduanya. Keberadaan kedua tools tersebut sangat membantu untuk menguji dan memperbaiki kesalahan yang ditimbulkan baik kesalahan dalam pengetikan kode atau lainnya.

Bab 2

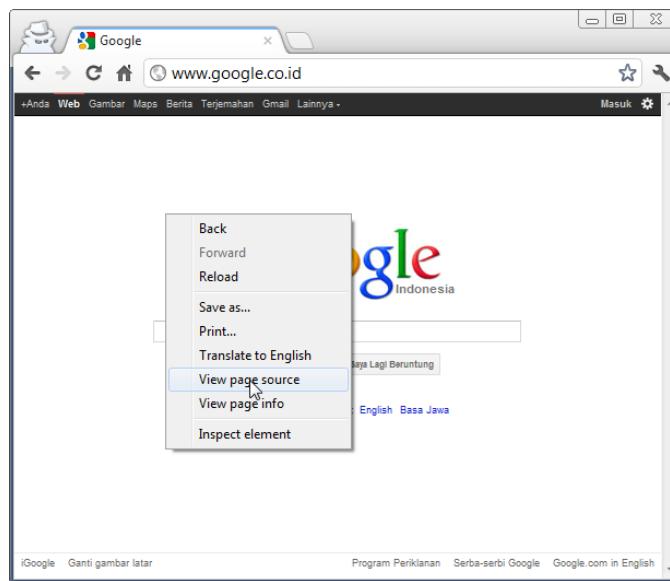
Pengenalan HTML

2.1. Apa itu HTML?

Semua halaman web yang sering anda buka, seperti facebook.com, twitter.com, google.com dan lain sebagainya ditampilkan dengan menggunakan HTML. Jadi bisa dikatakan HTML adalah bahasa dasar untuk menampilkan halaman web pada web browser.

Buka Web browser anda lalu bukalah sebuah situs, <http://www.google.com> misalnya.

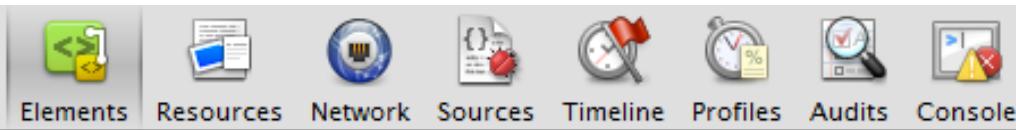
1. Lalu klik kanan di sembarang area, dan pilih "**Inspect element**" (menu paling akhir)



Gambar 11 Inspect Element

2. Perhatikan kode hasil inspect element, yang terlihat adalah sebagai berikut :

Gambar 12 Inspect Element akan menampilkan kode HTML dari situs yang bersangkutan



```
<!DOCTYPE html>
<html itemscope="itemscope" itemtype="http://schema.org/WebPage">
  <head>...</head>
  <body onload="try{if(!google.j.b){document.f&&document.f.q.focus();
  document.gbqf&&document.gbqf.q.focus();}}catch(e){}if(document.images)new Image().src='/images/
  nav_logo114.png'" dir="ltr" alink="#dd4b39" bgcolor="#fff" id="gsr" link="#12c" text="#222" vlink=
  "#61c" style cz-shortcut-listen="true" class>
    <div id="pocs" style="display: none; position: absolute; ">...</div>
    <a href="/search#q=www+http://www.google.co.id/&tbo=q&tbs=d+2114-HD&q=t+2-%2B%2B+summa-2">
```

Baris pertama pasti diawali dengan `<!DOCTYPE HTML>`, ini menandakan bahwa dokumen yang sedang anda buka saat ini adalah HTML. Begitu juga dengan baris kedua : `<HTML>`, kode tersebut menandakan bahkan kode-kode yang ditulis di dalamnya adalah kode HTML.

Jika anda ingin tahu kepanjangan HTML, HTML adalah kependekan 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.

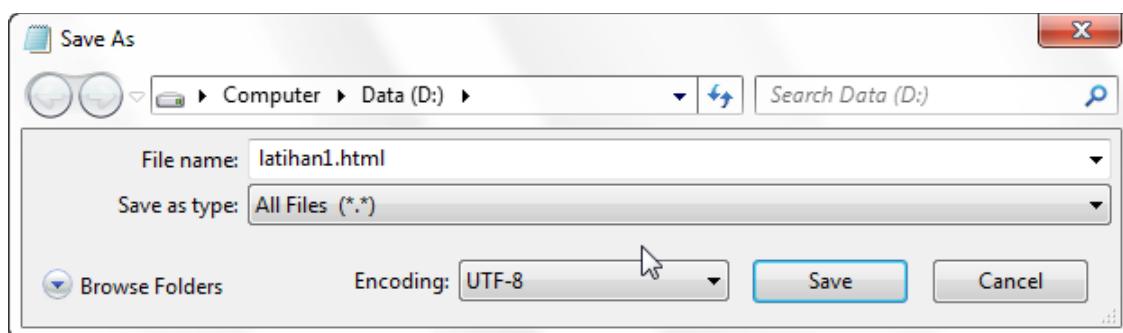
2.2. Membuat Website dalam 1 Menit

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

1. Buka Notepad (*All Programs > Accessories > Notepad*)
2. Ketikkan teks berikut :

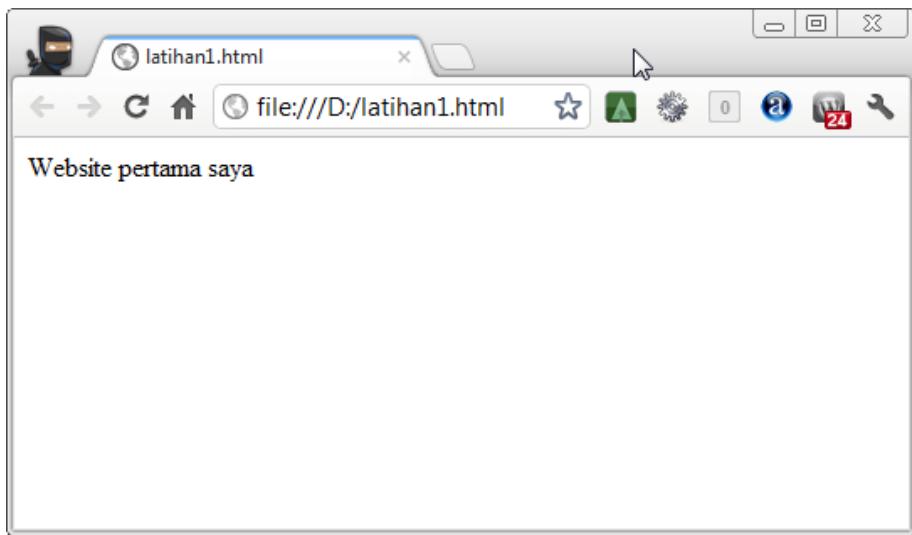
```
Website pertama saya
```

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 13 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 14 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>
```

Bingung? Tenang, saya jelaskan 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>
```

Nah, 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?

Nah loh?!

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 saya akan mengajak anda untuk 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>
```

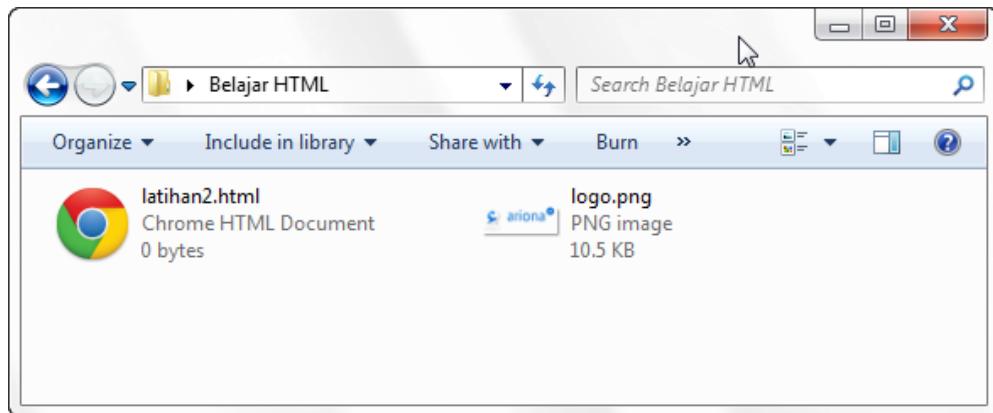
Sebagai pengganti, kita tambahkan tanda slash (/) sebelum kurung tutup :

```

```

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 15 Penempatan gambar sesuai dengan lokasi file HTML

3. Ketikkanlah kode HTML berikut :

```

1. <!DOCTYPE HTML>
2. <HTML>
3.   <head>
4.     <title>Judul File HTML</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 16 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.

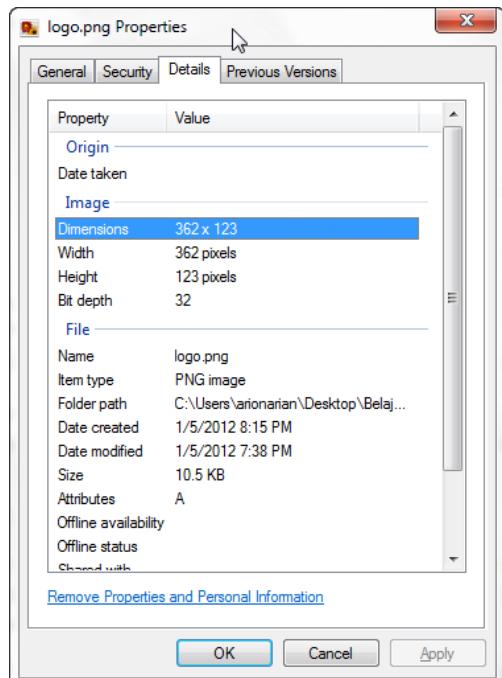
```

```

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.

```

```



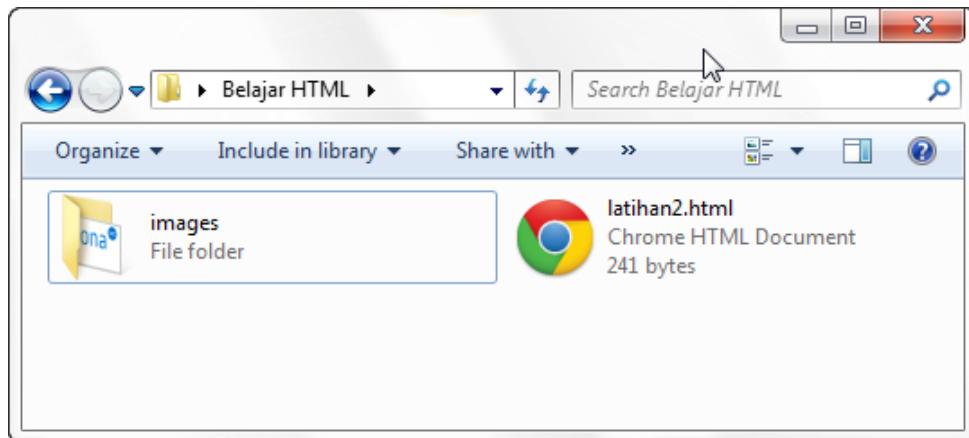
Gambar 17 File Properties gambar

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.

3.2. Cara penulisan lokasi file

Jika anda menyimpan gambar tersebut pada sebuah folder seperti berikut.



Gambar 18 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

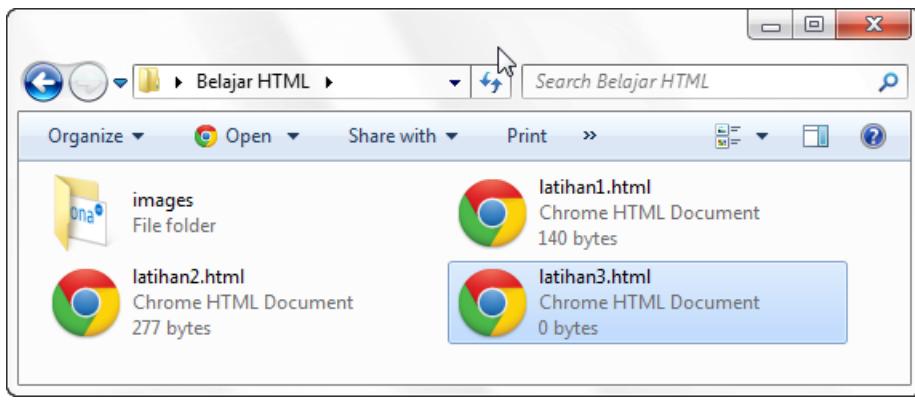
Namun anda tidak akan melihat perubahan pada teks “disini” karena kita belum “mengaitkannya” ke halaman web lain. Untuk itu kita akan menggunakan attribut `href` untuk menyimpan alamat web yang akan dituju ketika link di klik (penulisan lokasi sama halnya dengan attribut `src` pada tag `img`).

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 19 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>`).

```

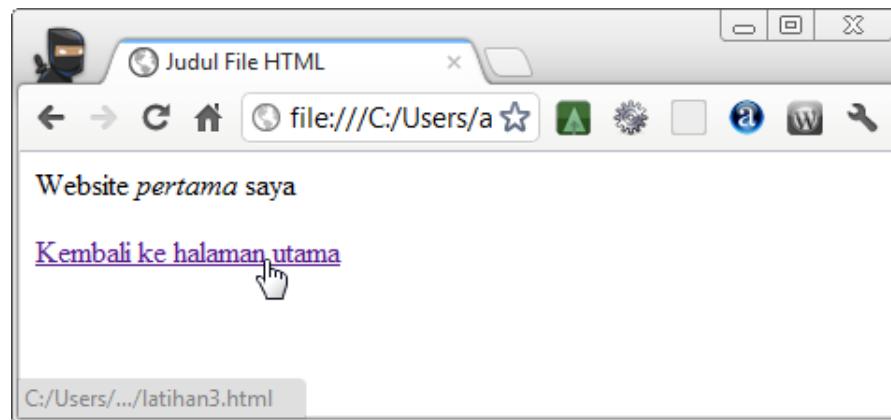
...
<a href="latihan3.html">Kembali ke halaman utama <a>
</body>
</HTML>
```

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



Gambar 20 tampilan file Latihan3.html

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

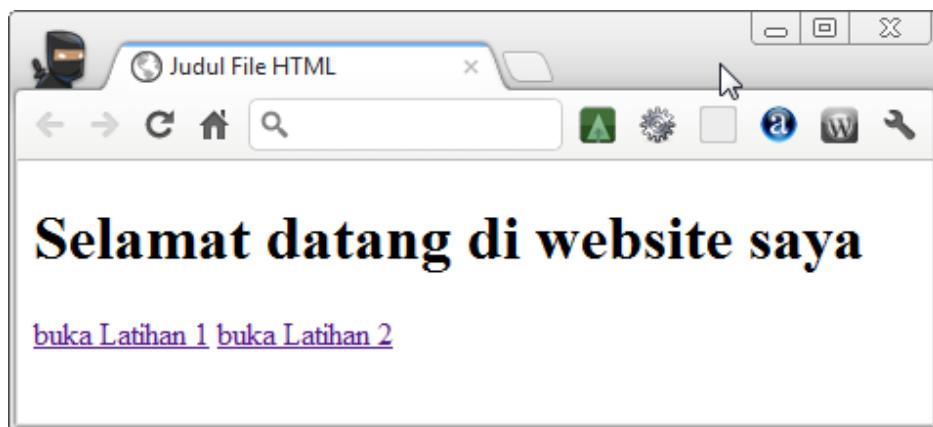


Gambar 21 Link untuk kembali ke halaman utama

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.

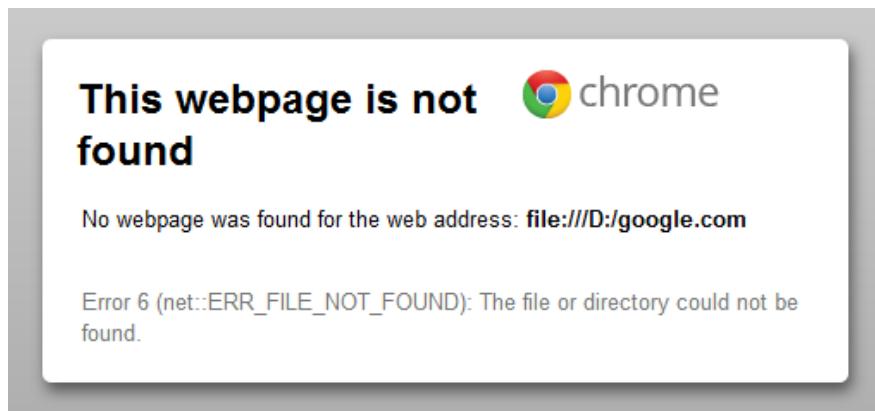


Gambar 22 link ditampilkan satu baris tanpa tag

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 23 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://ariona.net target="_blank">ariona.net</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:mrxyz@yahoo.com">Kirim Email</a>
```

Selain tipe link-link di atas, masih banyak lagi tipe link lainnya, seperti link No. Telp, Aplikasi dan lainnya.

Bab 5

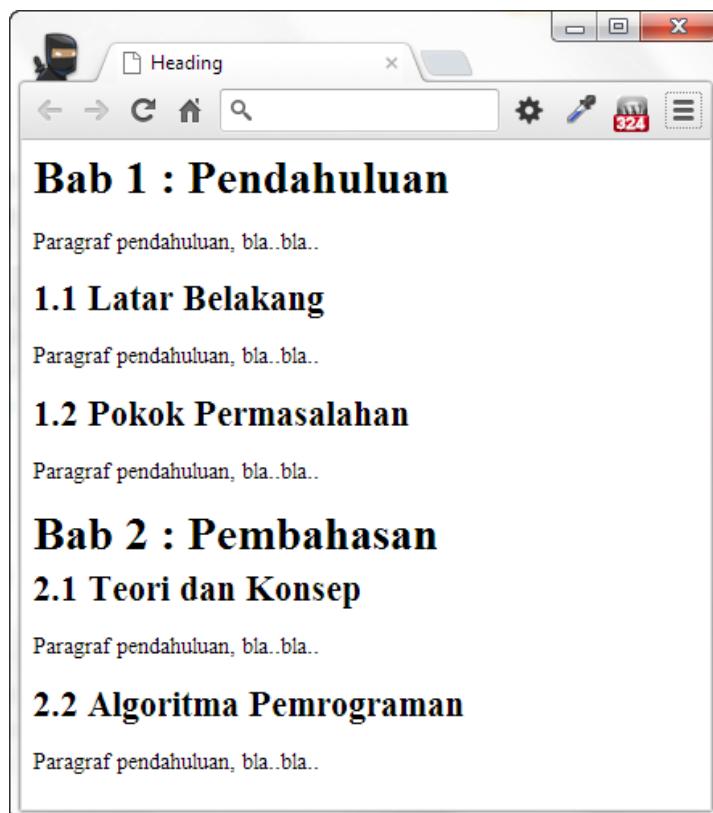
Heading/Penjudulan

Heading, digunakan untuk memberikan penjadulan pada suatu dokumen HTML. Bayangkanlah sebuah skripsi atau buku yang memiliki bab dan subbab-subbab di dalamnya. Untuk memformat penjadulan dalam HTML, kita gunakan tag `<h1>` untuk judul utama dan untuk judul subbabnya 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 24 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.

Perhatikan contoh penggunaan Heading dalam suatu website di halaman selanjutnya.

Dalam website tersebut, Logo/Judul dari website diberi Heading Level 1, untuk judul artikel diberi heading level 2 dan judul dari bagian-bagian di dalamnya diberi heading level 3 s.d 4

h1 untuk Judul Situs

h2 untuk Judul Artikel

h3 untuk Sub-judul Artikel

h4 untuk Sub-sub-judul Artikel

SUBLINE TEXT

Untuk sublime text anda dapat mendownload package Zen Coding [disini](#). Simpan file yang sudah di download di folder Installed Package dalam folder instalasi Sublime Text. Untuk instalasi text editor lainnya silahkan untuk mengunjungi situs official sublime text [disini](http://code.google.com/p/zen-coding/).

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 :

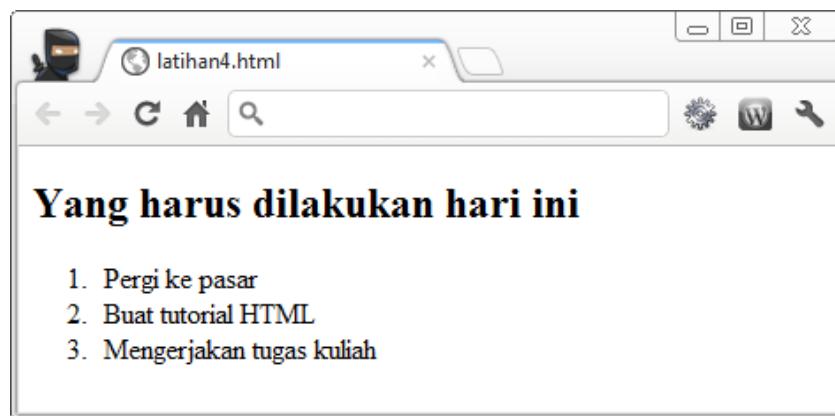
```
1. <h2>Yang harus dilakukan hari ini</h2>
2. <ol>
3.     <li>Pergi ke pasar</li>
4.     <li>Buat tutorial HTML</li>
5.     <li>Mengerjakan tugas kuliah</li>
6. </ol>
```

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



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

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



Gambar 25 Daftar berurutan

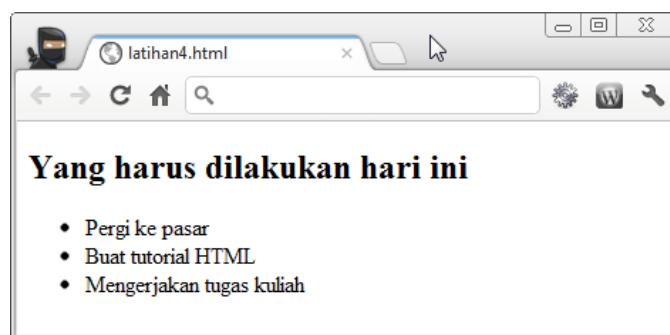
6.2. Unordered List

Berbeda dengan daftar berurut, *unordered list* akan menandai setiap item dengan simbol. Baik berupa lingkaran atau persegi (anda masih bisa merubahnya dengan CSS). 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 26 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 hari ini</h2>
8.         <ol>
9.             <li>Pergi ke pasar
10.                <ul>
11.                    <li>Beli sayuran</li>
12.                    <li>Beli kertas
13.                        <ol>
14.                            <li>A4</li>
15.                            <li>Legal</li>
```

```
16.                                     </ol>
17.                               </li>
18.                               <li>Beli tinta</li>
19.                           </ul>
20.                         </li>
21.                         <li>Buat tutorial HTML</li>
22.                         <li>Mengerjakan tugas kuliah</li>
23.                     </ol>
24. </body>
25. </HTML>
```

Dan berikut tampilan pada browser.



Gambar 27 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 © 2012 by someone*.

Lebih sederhananya, bayangkanlah Surat resmi yang memiliki Kop surat, badan surat dan penutup surat.

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>
```



Dengan Account Yahoo!, dapatkan email gratis dan layanan web terkemuka lainnya.

Nama Saya	<input type="text" value="Nama Depan"/>	<input type="text" value="Nama Belakang"/>
Jenis Kelamin	<input type="button" value="- Pilih Satu -"/>	
Tanggal Lahir	<input type="text" value="Tanggal"/>	<input type="button" value="- Pilih Bulan -"/>
Saya tinggal di	<input type="button" value="Tahun"/>	
	<input type="button" value="Indonesia"/>	

Pilih ID dan kata sandi

ID Yahoo! dan Email	<input type="text"/> @ <input type="text" value="yahoo.co.id"/>	<input type="button" value="Periksa"/>
Kata Sandi	<input type="password"/>	Kekuatan Kata Sandi <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Ketik ulang Kata Sandi	<input type="password"/>	

Jika Anda lupa kata sandi atau ID Anda...

Pertanyaan Rahasia 1	<input type="button" value="- Pilih Satu -"/>
Jawaban Anda	<input type="text"/>
Pertanyaan Rahasia 2	<input type="button" value="- Pilih Satu -"/>
Jawaban Anda	<input type="text"/>

Gambar 28 Formulir (FORM) pendaftaran Yahoo!



*Jika anda merasa asing dengan istilah **FORM**, bayangkan saja sebuah **FORMULIR** yang sering ditemui di sekitar kita seperti formulir pendaftaran siswa baru, formulir pembuatan KTP dan lain sebagainya.*

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 29 Contoh Label pada form

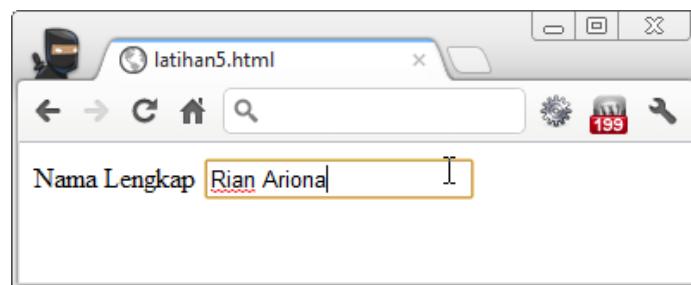
```
<label for="Name"> Keterangan Input </label>
```

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 30 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" />
```

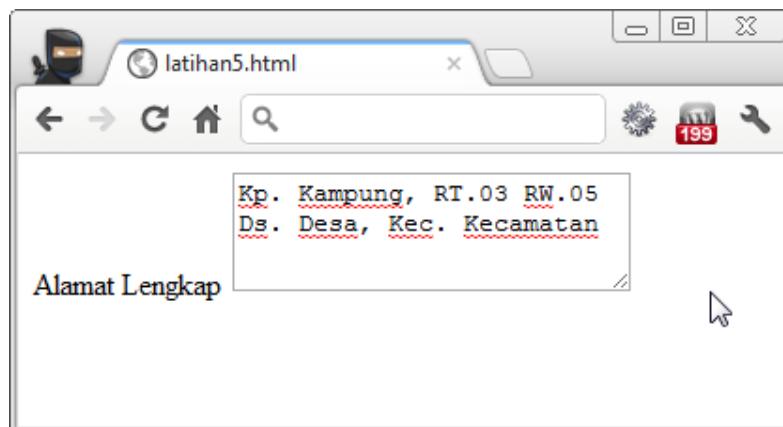
Dengan begitu, ketika halaman dibuka, text input ini secara otomatis akan terisi dengan nilai dari atribut value.

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>
```



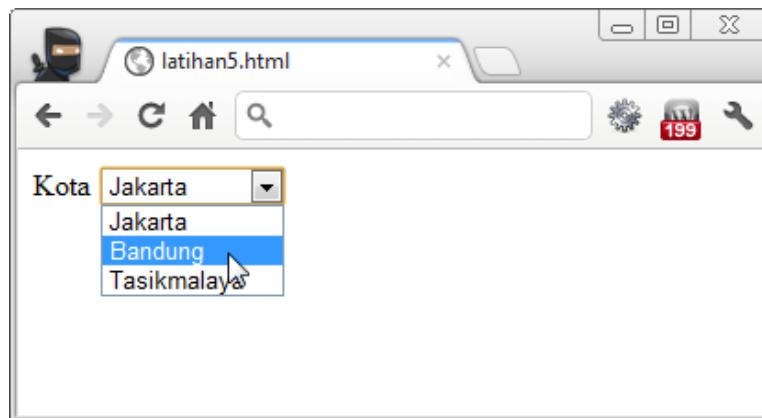
Gambar 31 Text area

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>
```

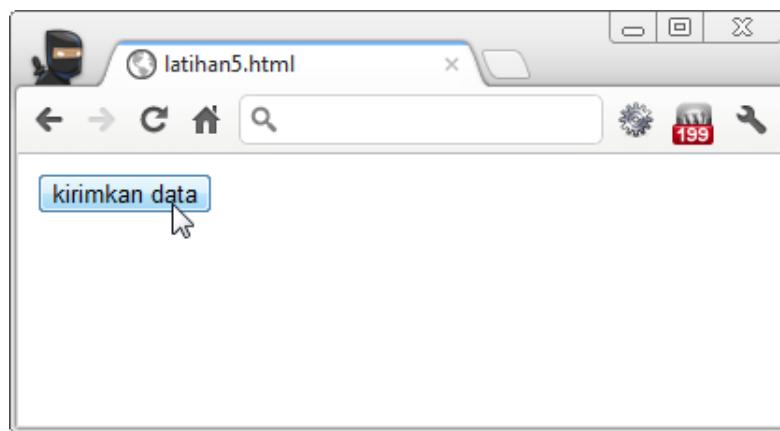


Gambar 32 Combo Box

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 33 Button

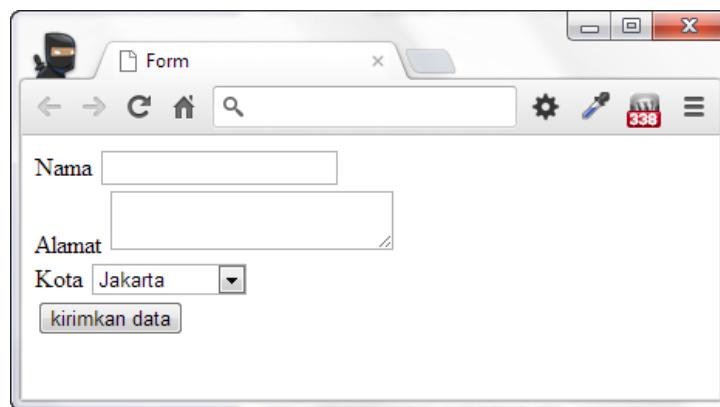
Sebagai latihan, kita coba menggabungkan kontrol-kontrol yang telah anda pelajari sebelumnya menjadi satu form utuh.

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 34 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 35 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 36 Penggunaan table untuk layout adalah menyalahi aturan

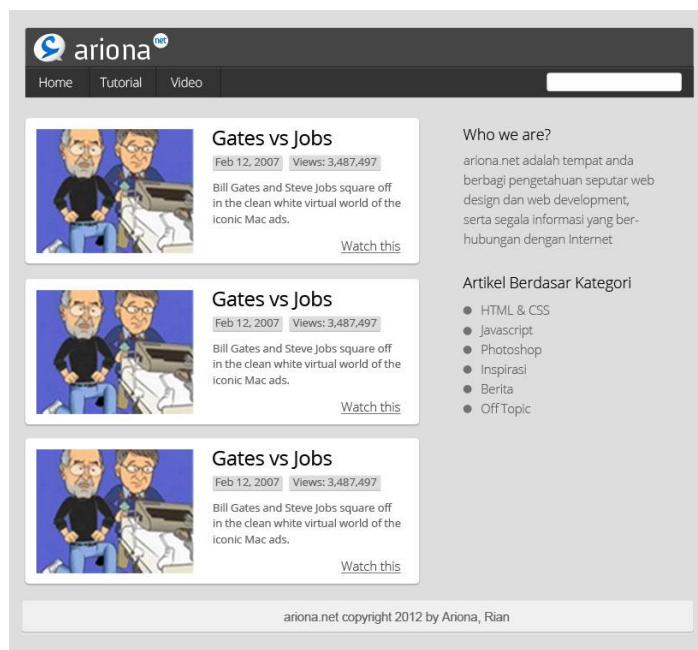
Penggunaan tag <table> untuk keperluan layout adalah menyalahi aturan/fungsi dari tag table tersebut.

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 37 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 “**images**” untuk menyimpan seluruh gambar yang diperlukan dalam file HTML yang kita buat. Lalu copy gambar yang terdapat di dalam folder resources/images yang disertakan bersama ebook ini.

10.2. Wrapper

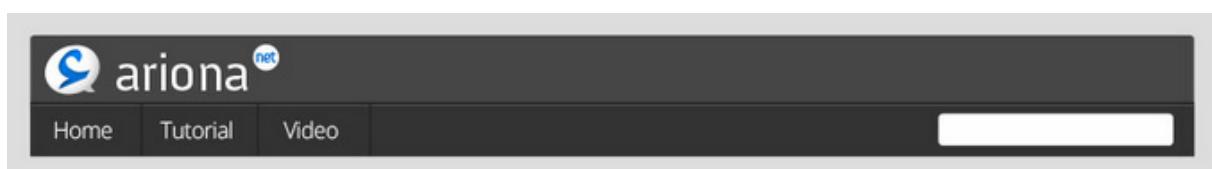
Kita mulai dari pembuatan sebuah div untuk menampung seluruh konten situs yang akan dibuat. Saya akan menggunakan div dengan id wrapper. Nantinya div ini akan kita buat rata-tengah, jadi setiap browser di zoom-in atau di zoom-out, konten situs tetap berada di tengah.

```

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 38 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 39 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 40 Detail setiap post

Kita masuk ke detail setiap artikel. Di dalam artikel ini terdapat sebuah gambar, judul artikel, tanggal, ringkasan dan sebuah link "watch this".

Sudah terbayang bagaimana kode HTML nya?

```

22. <div id="daftar-artikel">
23.     <div class="artikel">
24.         
25.         <h1> Gates vs Jobs</h1>
26.         <b>Feb 12, 2007</b><b>Views : 3,487,497</b>
27.         <p>Bill Gates and Steve Jobs square off in the clean white virtual world of
the iconic Mac ads</p>
28.         <a href="">Watch This</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> Gates vs Jobs</h1>
26.         <b>Feb 12, 2007</b><b>Views : 3,487,497</b>
27.         <p>Bill Gates and Steve Jobs square off in the clean white virtual world of
the iconic Mac ads</p>
28.         <a href="">Watch This</a>
29.     </div>
30.     <div class="artikel">
```

```

31.      
32.      <h1> Gates vs Jobs</h1>
33.      <b>Feb 12, 2007</b><b>Views : 3,487,497</b>
34.      <p>Bill Gates and Steve Jobs square off in the clean white virtual world of
35.          the iconic Mac ads</p>
36.          <a href="">Watch This</a>
37.      </div>
38.      <div class="artikel">
39.          
40.          <h1> Gates vs Jobs</h1>
41.          <b>Feb 12, 2007</b><b>Views : 3,487,497</b>
42.          <p>Bill Gates and Steve Jobs square off in the clean white virtual world of
43.              the iconic Mac ads</p>
44.          <a href="">Watch This</a>
45.
46.

```

10.5. Sidebar

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



Gambar 41 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>Who we are?</h2>
48.    <p>Ariona.net adalah tempat anda berbagi pengetahuan seputar web design dan web
49.        development serta segala informasi yang berhubungan dengan internet</p>
50.    <h2>Artikel berdasarkan kategori</h2>
51.    <ul>
52.      <li><a href="">HTML dan CSS</a></li>
53.      <li><a href="">Javascript</a></li>

```

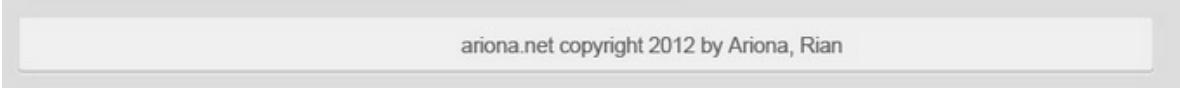
```

54.      <li><a href="">Photoshop</a></li>
55.      <li><a href="">Inspirasi</a></li>
56.      <li><a href="">Berita</a></li>
57.      <li><a href="">Off Topic</a></li>
58.    </ul>
59.  </div>
60.
61.

```

10.6. Footer

Terakhir, kita buat markup untuk bagian footer.



ariona.net copyright 2012 by Ariona, Rian

Gambar 42 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>Ariona.net copyright 2012 by Ariona, Rian</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> Gates vs Jobs</h1>
25.           <b>Feb 12, 2007</b><b>Views : 3,487,497</b>

```

```

25.          <p>Bill Gates and Steve Jobs square off in the clean
26.          white virtual world of the iconic Mac ads</p>
27.          <a href="">Watch This</a>
28.          </div><!-- Penutup div "artikel" -->
29.          <div class="artikel">
30.              
32.              <h1> Gates vs Jobs</h1>
33.              <b>Feb 12, 2007</b><b>Views : 3,487,497</b>
34.              <p>Bill Gates and Steve Jobs square off in the clean
35.              white virtual world of the iconic Mac ads</p>
36.              <a href="">Watch This</a>
37.              </div><!-- Penutup div "artikel" -->
38.              <div class="artikel">
39.                  
41.                  <h1> Gates vs Jobs</h1>
42.                  <b>Feb 12, 2007</b><b>Views : 3,487,497</b>
43.                  <p>Bill Gates and Steve Jobs square off in the clean
44.                  white virtual world of the iconic Mac ads</p>
45.                  <a href="">Watch This</a>
46.                  </div><!-- Penutup div "artikel" -->
47.                  </div><!-- Penutup div "daftar-artikel" -->
48.
49.          <div id="sidebar">
50.              <h2>Who we are?</h2>
51.              <p>Ariona.net adalah tempat anda berbagi pengetahuan seputar
52.                  webdesign dan web development serta segala informasi yang
53.                  berhubungan dengan internet</p>
54.              <h2>Artikel berdasarkan kategori</h2>
55.              <ul>
56.                  <li><a href="">HTML dan CSS</a></li>
57.                  <li><a href="">Javascript</a></li>
58.                  <li><a href="">Photoshop</a></li>
59.                  <li><a href="">Inspirasi</a></li>
60.                  <li><a href="">Berita</a></li>
61.                  <li><a href="">Off Topic</a></li>
62.              </ul>
63.          </div><!-- Penutup div "sidebar" -->
64.          <div id="footer">
65.              <p>ariona.net copyright 2012 by Ariona,Rian</p>
66.          </div>
67.      </div><!-- Penutup div "wrapper" -->
68.  </body>
69. </HTML>

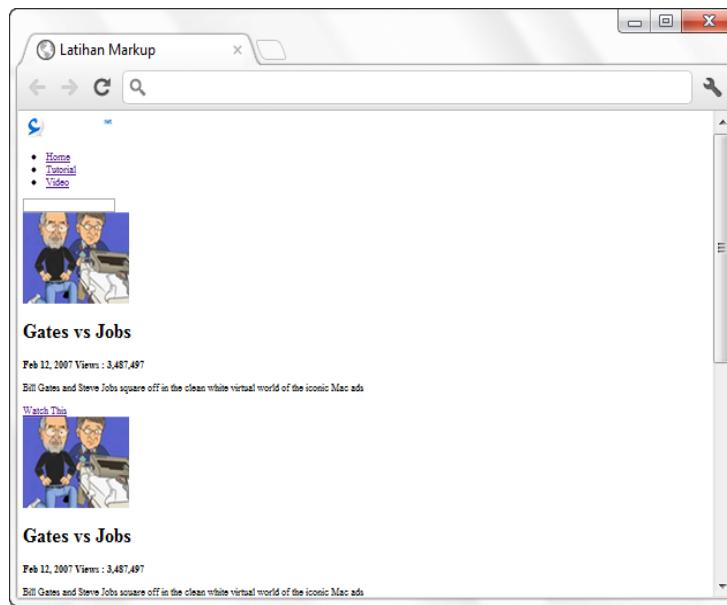
```

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 43 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.

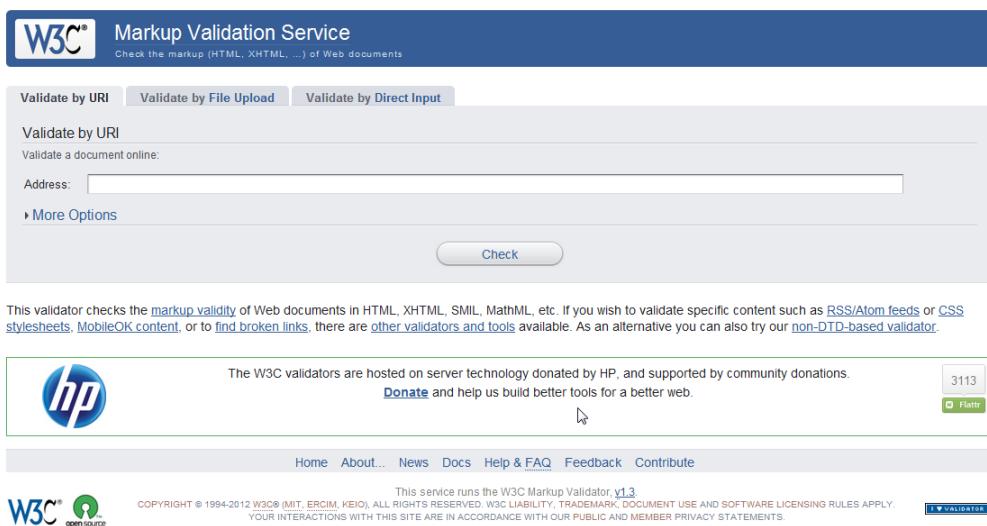
Bab 11

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 44 Validator HTML dari W3C

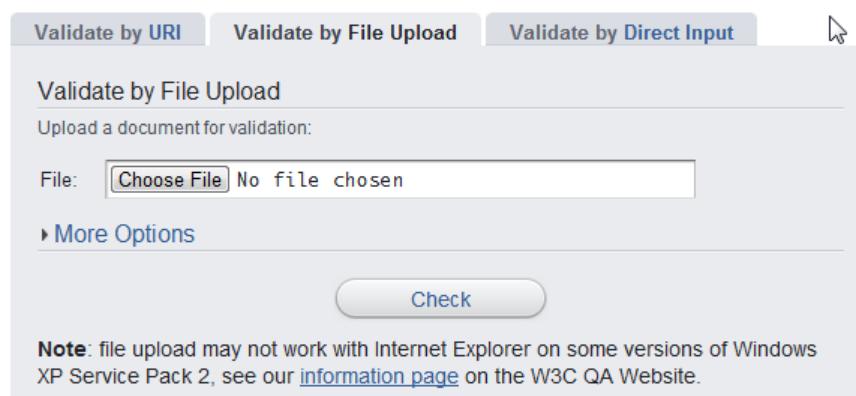
Ada tiga cara pengujian Markup HTML, yakni dengan mengetikkan alamat file HTML yang akan diuji (Validate by URL) opsi ini adalah untuk memvalidasi halaman HTML yang

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.



Gambar 45 Validate by File Upload

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

This document was <i>Tentatively checked as HTML5</i>	
Result:	Tentatively passed, 4 warning(s)
File:	<input type="button" value="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 <input type="button" value="detect automatically"/>
Datatype:	HTML5 <input type="button" value="detect automatically"/>
Root Element:	html

Gambar 46 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 47 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 48 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!.*

Bab 12

Semantik

Sebuah dokumen HTML tidak lain adalah kumpulan dari tag-tag HTML dan seperti yang telah anda ketahui bahwa HTML adalah bahasa untuk *me-markup* atau *mem-format* suatu dokumen. HTML **BUKAN** untuk membuat bagaimana dokumen tersebut ditampilkan.

Sama halnya dengan aplikasi pengolah kata, formating yang dilakukan pada suatu dokumen ditujukan untuk memberikan makna-makna tersendiri pada setiap kata-kata yang diolahnya.

Misalnya, ketika suatu teks ditampilkan miring, biasanya ada makna tertentu pada teks tersebut, seperti pada pertanyaan berikut :

“Berikut ini adalah tag-tag HTML, *kecuali* :”

Kata “kecuali” di sana diberi format miring karena penulis ingin menegaskan bahwa jawaban yang benar adalah yang bukan tag HTML. Jika kalimat pertanyaan tersebut dibaca, biasanya akan ada perbedaan intonasi pada kata “kecuali”.

Dalam HTML, terdapat dua tag yang dapat membuat suatu teks miring yakni tag ** dan *<i>*. Lalu apa perbedaan kedua tag tersebut?

Jika kita sangkutpautkan pertanyaan ini dengan contoh kasus di atas, penggunaan tag ** untuk memiringkan kata “kecuali” adalah cara yang benar. Mengingat tag ** ini memang berfungsi untuk memberikan tekanan pada suatu teks (*emphasis*). Berbeda dengan tag *<i>* yang hanya berfungsi untuk menampilkan teks menjadi miring tanpa ada makna sama sekali.

12.1. Maksud Semantik dalam HTML

Semantik berarti makna yang terkandung dari suatu kata. Dalam HTML, semantik berarti makna yang terkandung dari suatu tag HTML. Seperti pada contoh kasus sebelumnya, penggunaan tag lebih memiliki nilai semantik dari pada tag <i>.

Pada dasarnya, Gunakanlah tag-tag HTML sesuai dengan maksud/fungsi dari tag HTML tersebut, Jangan menggunakan suatu tag HTML karena tag tersebut memiliki penampilan berbeda.

Misalnya, penggunaan tag <table>. Tag ini berfungsi untuk membuat suatu tabel atau data tabular, penggunaan diluar itu adalah salah, seperti penggunaan tag table untuk pembuatan layout web.

Untuk mengetahui fungsi dari setiap tag HTML anda dapat membacanya di <http://www.w3schools.com/tags/default.asp>.

12.2. Pentingkah Semantik?

Penulisan dokumen HTML yang semantik ditujukan agar dokumen HTML dapat dipahami oleh komputer, karena pada dasarnya cara berfikir komputer berbeda dengan cara berfikir manusia.

Manusia bisa memahami suatu dokumen dengan adanya tanda baca, atau formatting (seperti teks miring tebal). Berbeda dengan komputer, dan pada kasus ini tag-tag HTML lah yang menjadi tanda baca yang akan membantu komputer untuk memahami dokumen HTML.

Kembali kepada penting tidaknya semantik, boleh jadi dokumen yang semantik dapat mempermudah sistem komputer untuk mengolah informasi dalam dokumen untuk keperluan tertentu, misalnya untuk keperluan Mesin Pencarian (Search Engine) agar hasil pencarian lebih akurat dan sesuai dengan yang diinginkan.

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 49 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>
```

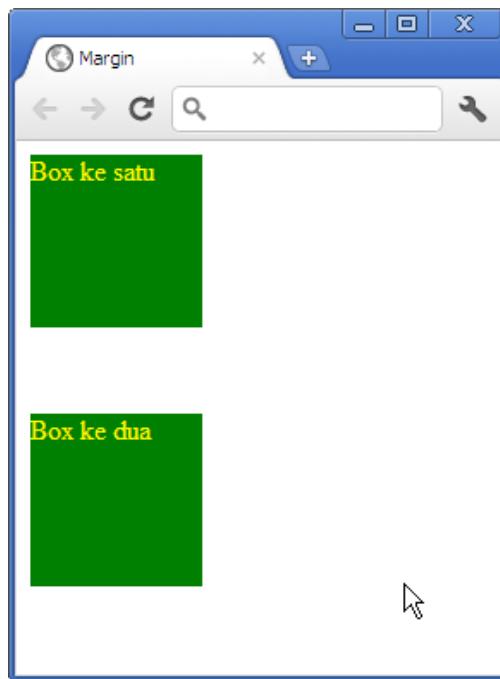


Ingat, tambahkan kode tersebut pada struktur HTML standar yang memiliki <!DOCTYPE HTML>, <HTML>, <head> dan seterusnya.

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 50Penambahan 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 51 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

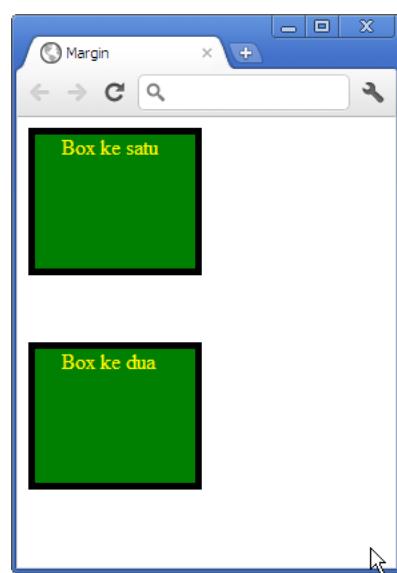
Property border digunakan untuk memberikan garis pinggir pada elemen, setidaknya ada tiga property yang harus anda set, yaitu color, style dan width.

```
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

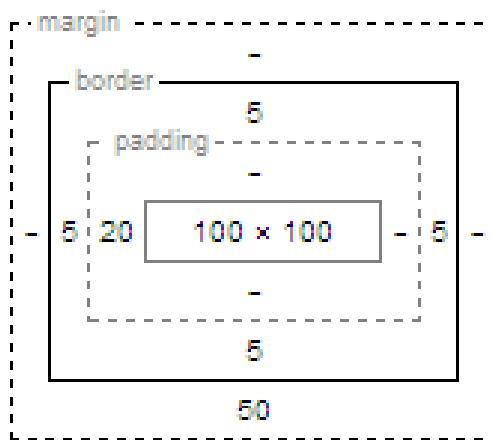
Gambar 52 Penambahan border

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 53 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".

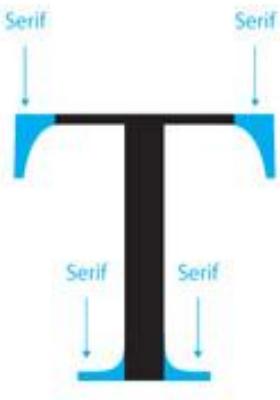
Apa itu Sans-serif dan Serif

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 54 Font jenis Serif memiliki kait disetiap ujungnya

Sans-serif

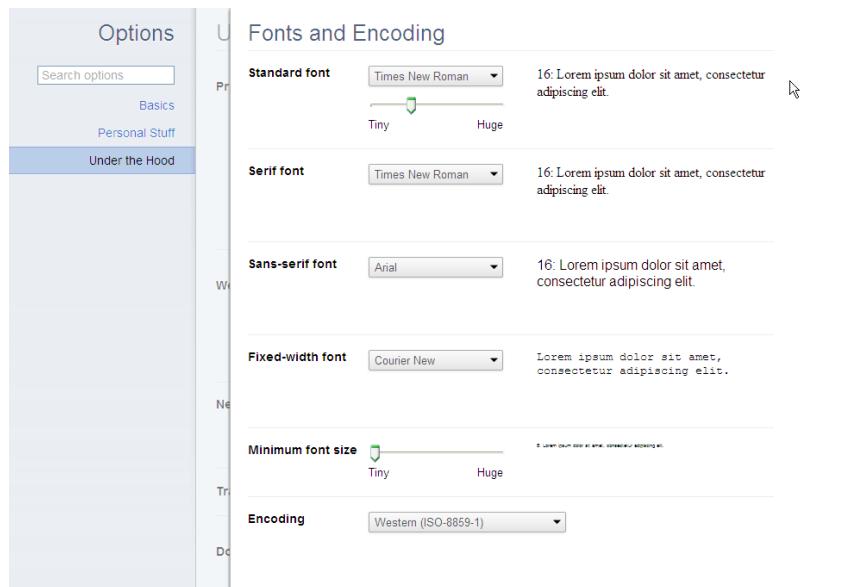
Arti dari sans, adalah "tanpa". Jadi sans-serif bisa diartikan sebagai jenis font tanpa kait. Yang paling umum digunakan untuk jenis sans-serif adalah arial.

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

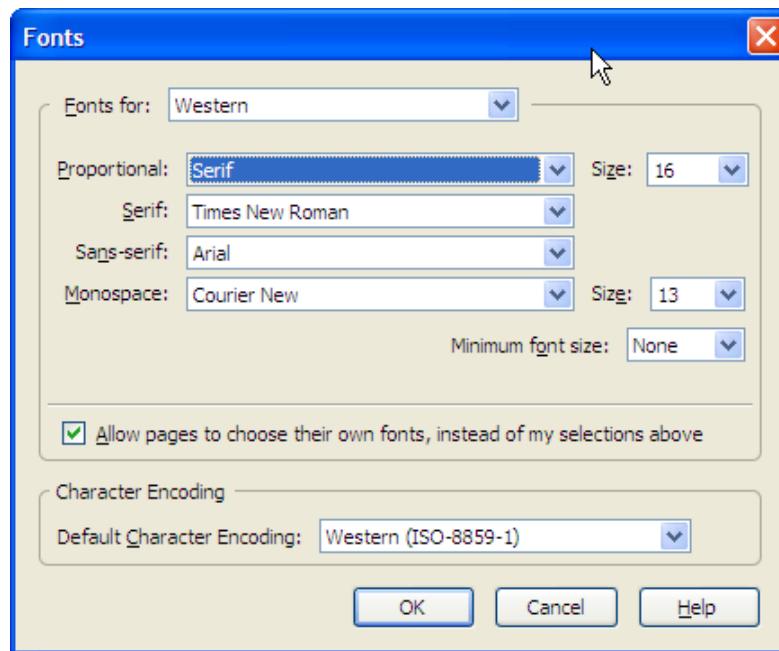


Gambar 55 Font jenis Sans Serif

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 56 pengaturan font pada browser Google Chrome



Gambar 57 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");
```

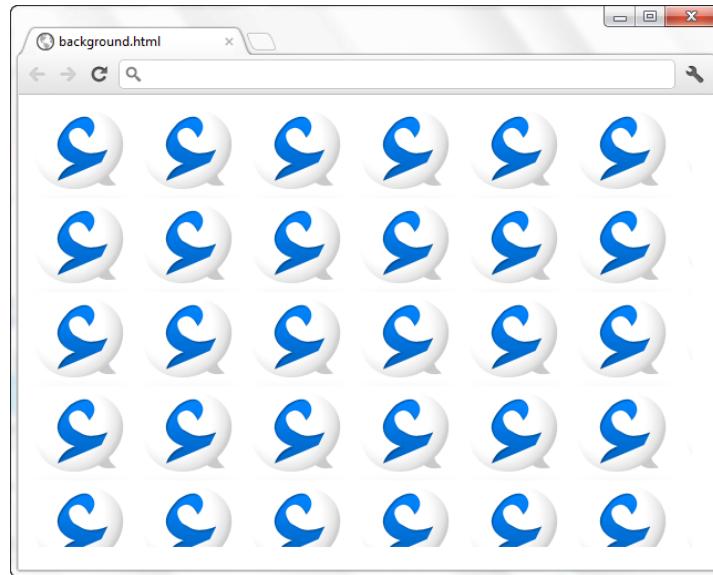
Ketika anda menggunakan gambar dengan ukuran kecil maka anda akan melihat background ditampilkan secara berulang untuk memenuhi seluruh element tersebut.

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);
}
```

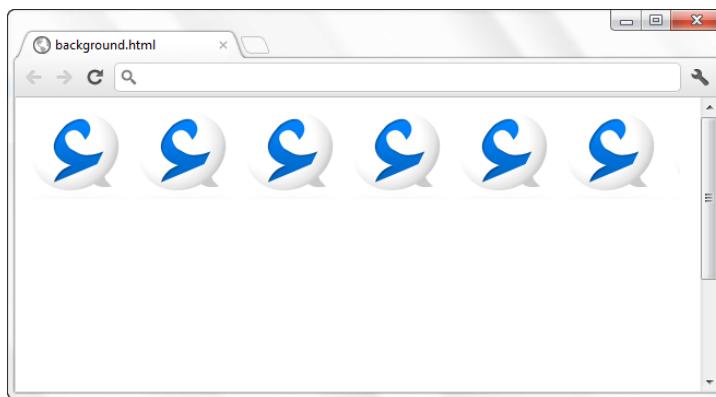


Gambar 58 Background akan terus diulang untuk memenuhi ukuran elemen

16.2. background-repeat

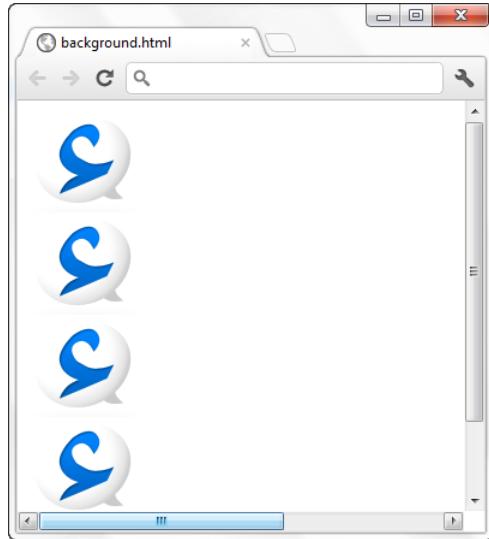
Pengulangan gambar background tersebut bisa kita atur dengan menggunakan property background-repeat. Anda bisa mengulang gambar secara horizontal (sesuai sumbu x) atau secara vertikal (y) atau anda pun dapat tidak mengulangi background tersebut.

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



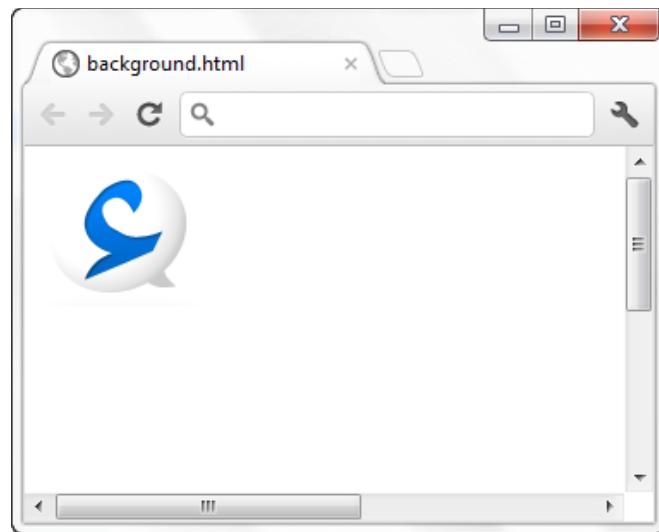
Gambar 59 Repeat X akan mengulang background secara horizontal

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



Gambar 60 Pengulangan background secara vertical dengan repeat-y

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



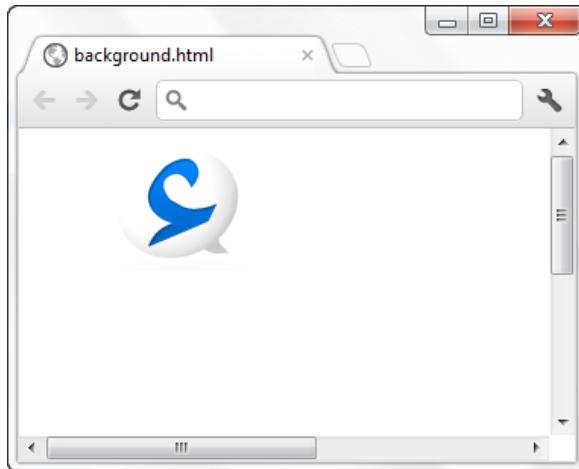
Gambar 61 Background tidak diulang

16.3. background-position

Posisi gambar background dapat kita tentukan letaknya dengan menentukan jarak x (horizontal) dan y (vertical).

Sebagai contoh, kode berikut akan menggeser background-image sebesar 50px dari kiri.

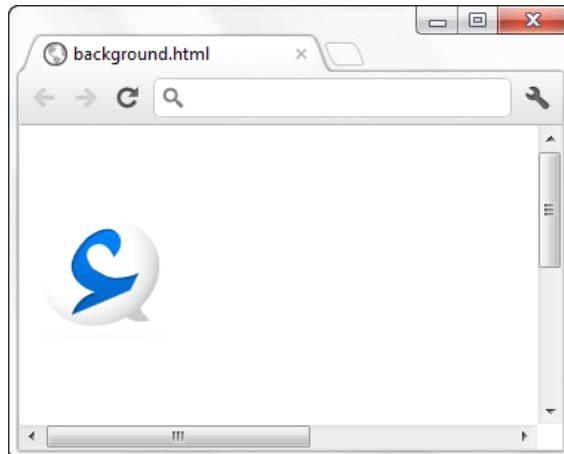
```
background-position-x : 50px;
```



Gambar 62 background bergeser selebar 50px dari kiri

Dan contoh berikut akan menggeser background-image sebesar 50px ke bawah.

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



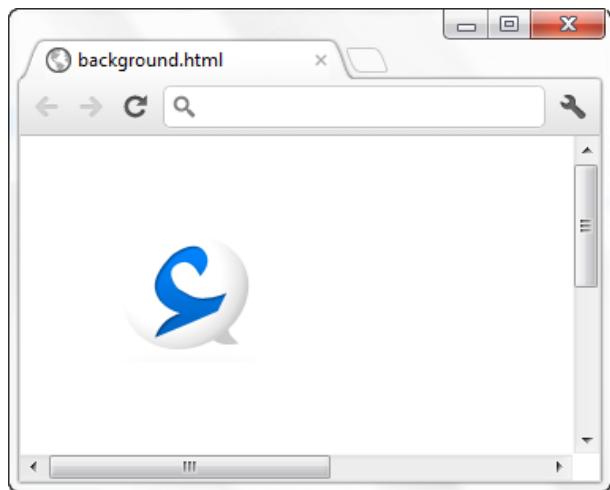
Gambar 63 Background turun 50px dari posisi sebelumnya

Andapun dapat menggunakan CSS-shorthand untuk background-position :

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

misalnya

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



Gambar 64 posisi background ada pada 50px dari kiri dan atas

16.4. Image Sprite

Dengan adanya property `background-position` untuk menentukan lokasi penempatan gambar, kita dapat bermain-main dengan image-sprite. Image Sprite adalah sekumpulan gambar berbeda yang disatukan dalam satu file gambar.



Gambar 65 contoh Image sprite

Teknik ini dilakukan untuk mempercepat loading website tanpa harus memanggil beberapa file gambar yang banyak. Cukup satu gambar dan kita tentukan lokasi gambar yang ingin ditampilkan.

Buatlah file baru dengan nama file `latihan11.html` dan ketikkan markup HTML seperti berikut :

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

Lalu tambahkan style CSS berikut :

```
#button {
    width:240px;
    height:83px;
    background-image : url(sprite.jpg);
    background-position-y : 0;
}
#button:hover{
    background-position-y : -83px;
}
```

Style tersebut akan mengambil gambar yang terdapat di posisi x, 0 dan posisi y turun 83px. Sehingga gambar yang ditampilkan adalah bagian berikut :



Gambar 66 Hanya sebagian gambar yang ditampilkan

Lalu apa maksud dari selector #button:hover? :hover adalah salah satu Pseudo Class, saya akan membahasnya di bab selanjutnya.

Contoh penggunaan Image sprite ini banyak diimplementasikan pada beberapa situs besar seperti Google atau facebook, berikut ini adalah image sprite untuk icon-icon yang ada dalam google plus.



Gambar 67 Image Sprite Google Plus

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 :

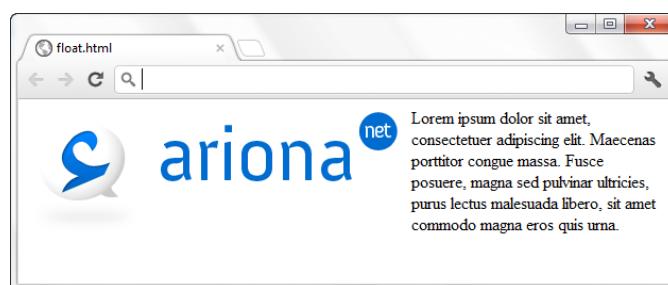
```

<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 meratakan elemen gambar dan membuat paragraf naik keatas.



Gambar 68 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 :

```

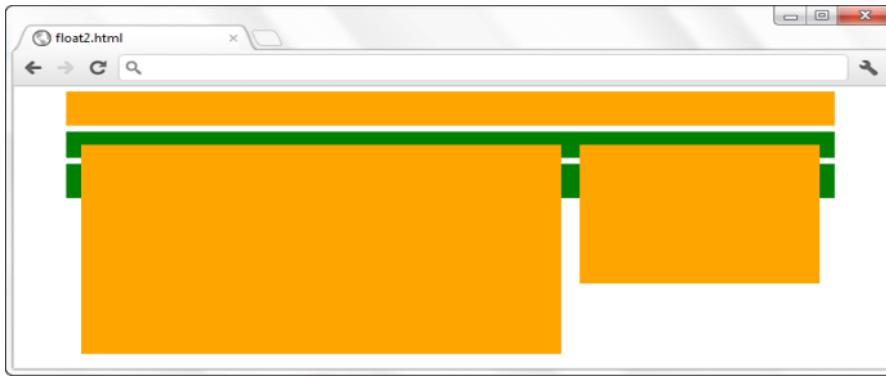
1. <div id="wrapper">
2.     <div id="header"></div>
3.
4.     <div id="content">
5.         <div id="artikel"></div>
6.         <div id="sidebar"></div>
7.     </div>
8.
9.     <div id="footer"></div>
10. </div>
```

Lalu berilah style berikut :

```

1. #wrapper{
2.     width : 960px;
3.     margin : 0 auto; /*agar wrapper tetap berada ditengah */
4. }
5. #header {
6.     width:100%;
7.     height:50px;
8.     background: orange;
9.     margin-bottom: 10px;
10. }
11. #content{
12.     padding:20px;
13.     background : green;
14.     margin-bottom : 10px;
15. }
16. #artikel{
17.     width : 600px;
18.     height : 700px;
19.     float : left;
20.     background: orange;
21. }
22. #sidebar {
23.     width : 300px;
24.     height: 200px;
25.     float : right;
26.     background: orange;
27. }
28. footer{
29.     width : 100%;
30.     height: 50px;
31.     background: green;
32. }
```

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



Gambar 69 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

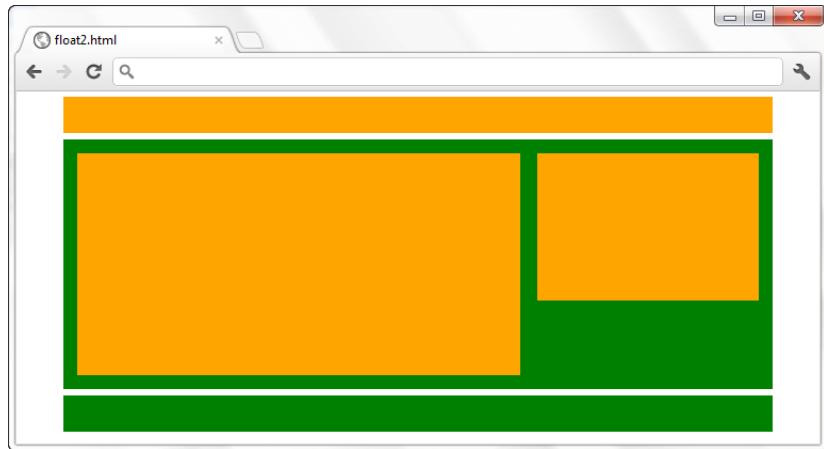
```
#content{
    ...
    overflow:hidden;
}
```

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 70 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 :

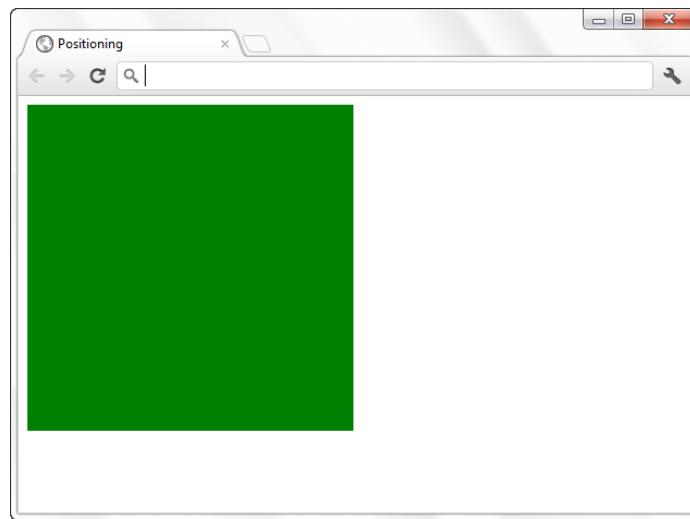
```
<div id="box"></div>
```

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;
}
```

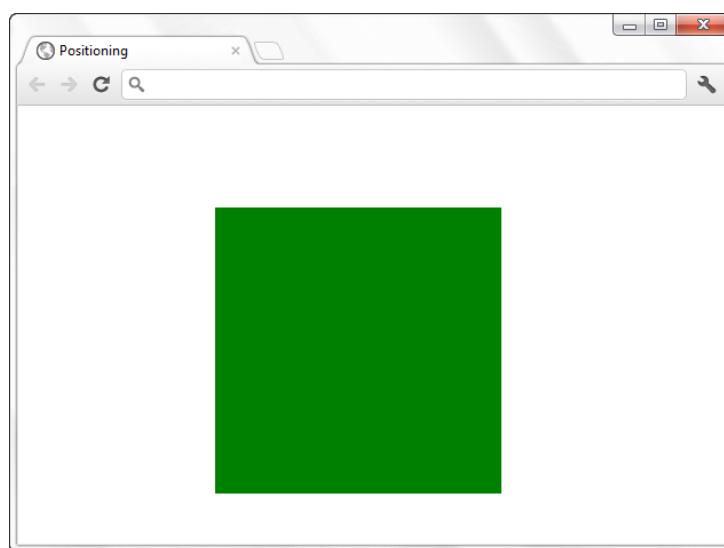


Gambar 71 Box static tidak dapat ditentukan posisinya

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 72 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.

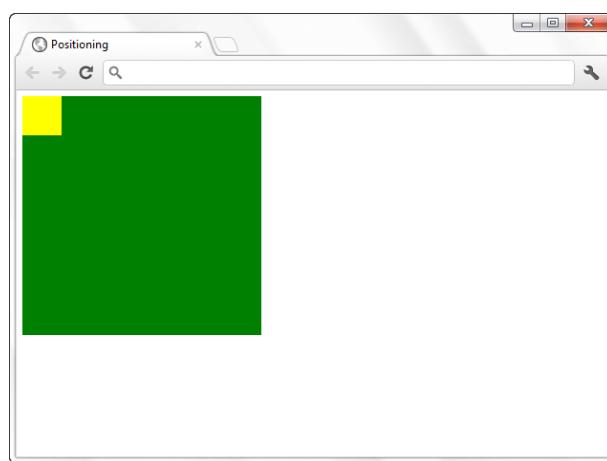
Buatlah markup HTML seperti berikut :

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

Lalu berilah style seperti berikut :

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

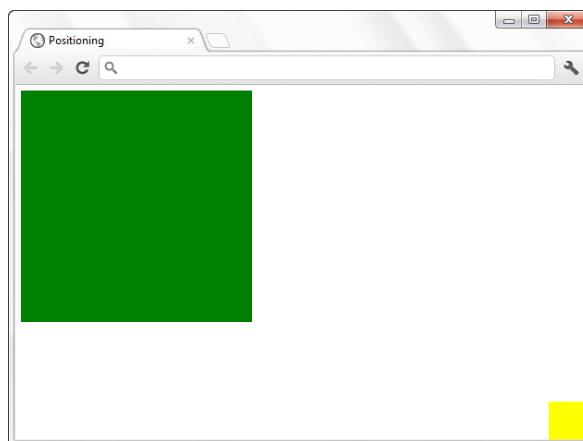
#box2{
    width:50px;
    height:50px;
    background:yellow;
}
```



Gambar 73 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.

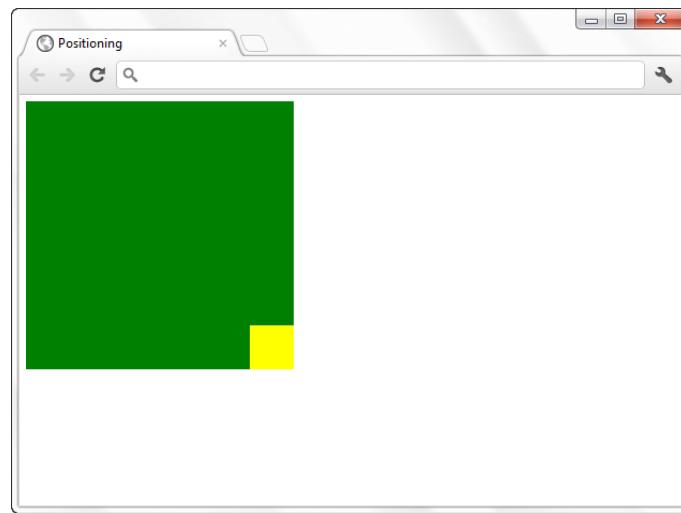
```
#box2{  
    width:50px;  
    height:50px;  
    background:yellow;  
    position:absolute;  
    bottom:0;  
    right:0;  
}
```



Gambar 74 #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.

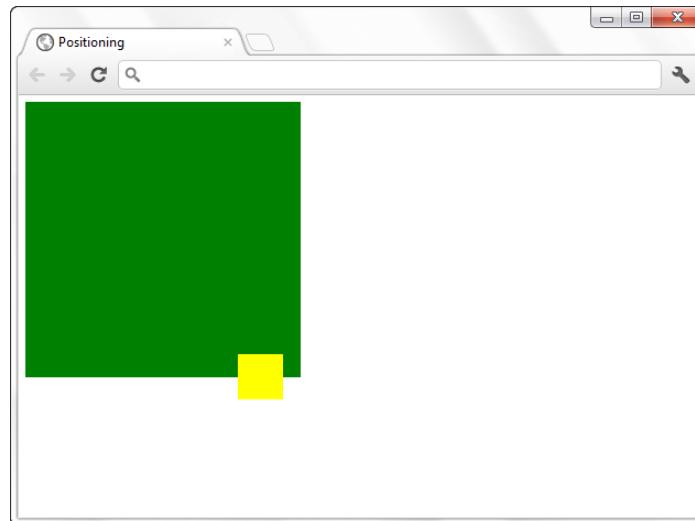
```
#box{  
    width:300px;  
    height:300px;  
    background:green;  
    position:relative;  
}
```



Gambar 75 Posisi absolute position dapat dibatasi oleh 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 76 Pengaturan posisi dengan absolute

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



Gambar 77 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 78 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 79 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. **Masih ingat dengan demonstrasi Image Sprite pada bab sebelumnya?***

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 :



Gambar 80 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.

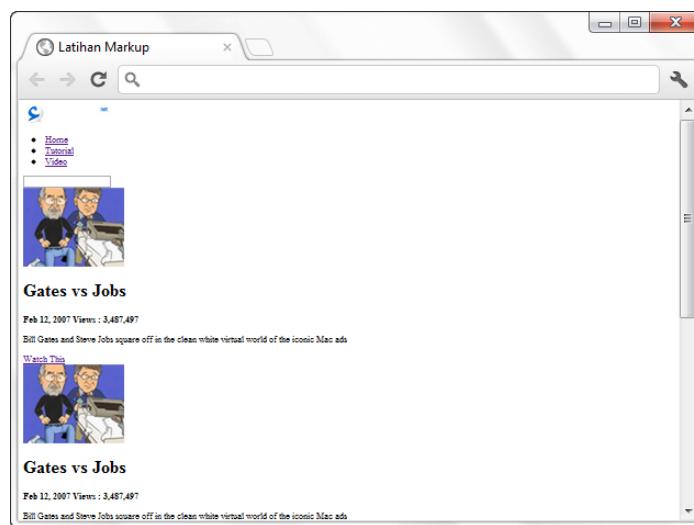
```
<div>
  <p> Paragraf pertama </p>
  <p> Paragraf kedua </p>
  <p> Paragraf ketiga </p>
  <p> Paragraf keempat </p>
</div>
```

Bab 20

StudiKasus 2 – Pemberian Style/CSS

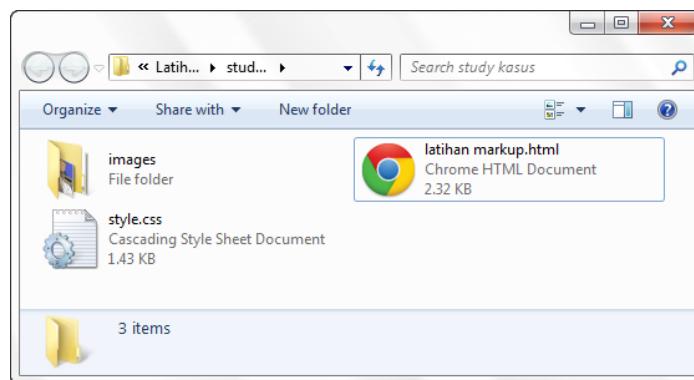
Pada studi kasus sebelumnya, anda telah membuat markup HTML dari sebuah desain web sederhana. Hasil yang didapatpun terasa polos, tidak menarik dan yang terpenting tidak sama dengan desain web yang diinginkan.

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



Gambar 81 Markup HTML tanpa CSS

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



Gambar 82 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:

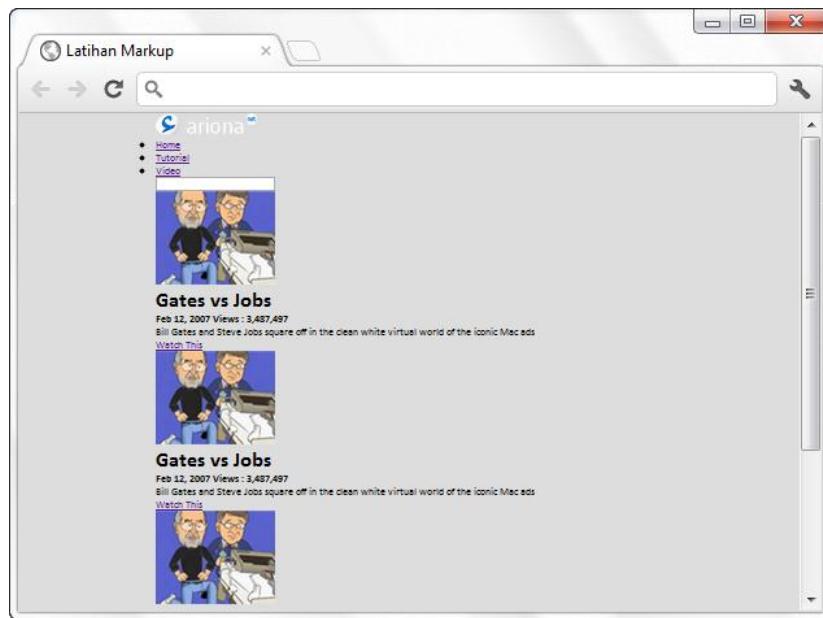
```
body{
    background-color:#DDDDDD;
}
```

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 83 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.

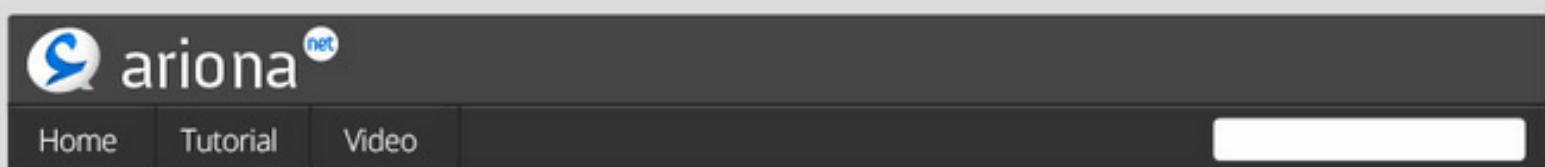
Sebenarnya ukuran lebar minimal standar dari sebuah desain web adalah 960px namun karena pada desain lebar dari web sebesar 780px maka kita gunakan nilai ini.

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 84 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 85 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;
}
```

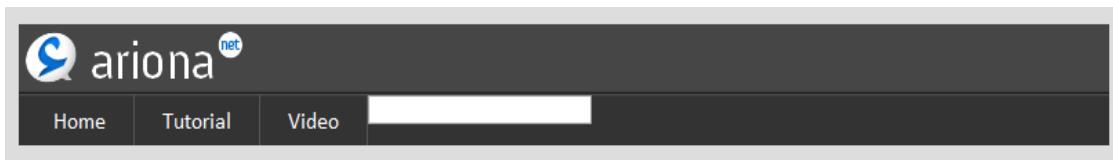
Kita beri jarak antara menu dengan logo sebesar 3px, dan property list-style:none berfungsi untuk menghilangkan lingkaran didepan setiap list.

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 86 Penampilan Menu utama setelah diberi style

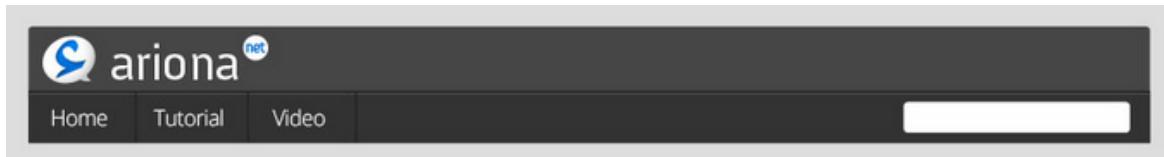
20.6. Form

Bagian form pencarian ini sangat sederhana, kita hanya akan menentukan tinggi dan meratakanankan 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 87 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 88 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;
}
```

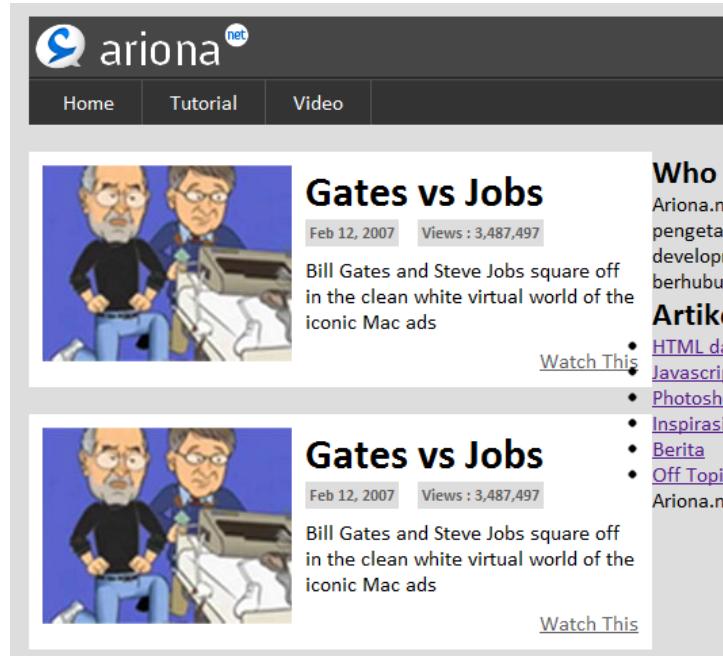
Terakhir dari bagian artikel ini, adalah penempatan link “watch this” ke sebelah kiri.

```
.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 89 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.

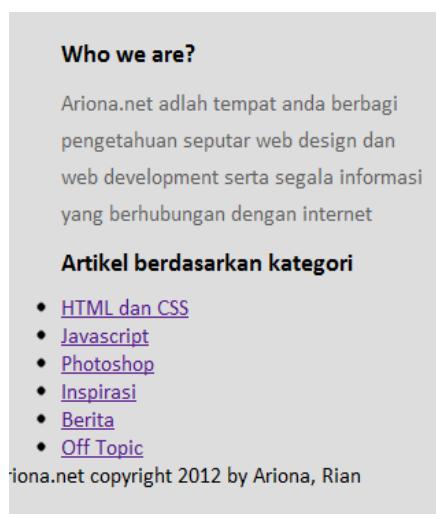
```
#sidebar{
    width:270px;
    float:right;
}
```

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 90 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.

 ariona^{net}

Home Tutorial Video



Gates vs Jobs

Feb 12, 2007 Views : 3,487,497

Bill Gates and Steve Jobs square off in the clean white virtual world of the iconic Mac ads

[Watch This](#)



Gates vs Jobs

Feb 12, 2007 Views : 3,487,497

Bill Gates and Steve Jobs square off in the clean white virtual world of the iconic Mac ads

[Watch This](#)



Gates vs Jobs

Feb 12, 2007 Views : 3,487,497

Bill Gates and Steve Jobs square off in the clean white virtual world of the iconic Mac ads

[Watch This](#)

Who we are?

Ariona.net adalah tempat anda berbagi pengetahuan seputar web design dan web development serta segala informasi yang berhubungan dengan internet

Artikel berdasarkan kategori

- HTML dan CSS
- Javascript
- Photoshop
- Inspirasi
- Berita
- Off Topic

Ariona.net copyright 2012 by Ariona, Rian

Gambar 91Hasil Akhir Latihan 1

Bab 21

Studi Kasus 3 – Slicing

Pada bab ini kita kembali melakukan latihan Konversi desain web ke dalam HTML & CSS, namun dengan desain web yang berbeda, kita juga akan mempelajari bagaimana teknik slicing gambar-gambar yang diperlukan dalam desain web tersebut untuk bagian-bagian tertentu seperti background, logo, dan slide.

Yang akan kita konversi kali ini adalah desain web bertema minimalis. Jika diperhatikan, desain kali ini lebih sederhana dibandingkan desain sebelumnya.

Gambar 92 Desain Web untuk Latihan Slicing

The screenshot shows a minimalist website template. At the top, there is a navigation bar with links for Home, Portfolio, Blog, and Contact. Below the navigation, there is a large, colorful graphic featuring abstract shapes in shades of red, yellow, and blue. Overlaid on this graphic is the title "Belajar HTML & CSS : Tutorial fundamental mempelajari HTML & CSS". The main content area below the graphic contains the question "Bagaimana cara menjadi seorang webdesigner ?". There are three sections at the bottom: "Belajar HTML", "Belajar CSS", and "Buat Website". Each section has a brief description and a downward-pointing arrow indicating a flow or next step.

Belajar HTML	Belajar CSS	Buat Website
Pelajari terlebih dahulu apa itu HTML, bagaimana cara pembuatan file HTML, Tag-tag HTML yang ada beserta fungsinya	Pelajari apa itu CSS, hubungan antara CSS dengan HTML, bagaimana cara penggunaan CSS untuk mengatur layout HTML	Setelah mempelajari HTML & CSS saatnya untuk menerapkan keduanya dalam kasus konversi design web menjadi file HTML & CSS

Agar lebih mudah dalam proses markup, saya buatkan versi wireframe untuk desain web tersebut :



Gambar 93 Wireframe dari Desain Web

Dengan begitu kita bisa menentukan bagian-bagian yang akan kita buat div. Selanjutnya saya juga sudah menambahkan beberapa keterangan tentang nama-nama div yang akan digunakan:



Gambar 94 Rencana pemberian nama elemen

Dengan begitu pasti anda sudah memiliki bayangan akan apa yang anda tulis bukan? Ya, mari kita mulai ke dalam proses markup HTML.

21.1. Penulisan Markup HTML

Sebelum memulai proses konversi, buat folder baru untuk latihan konversi kita kali ini dengan nama “Latihan Slicing” atau terserah anda. Buat lagi satu folder dengan nama “images” untuk menyimpan gambar-gambar pendukung hasil slicing nanti.

wrapper

Seperti kasus sebelumnya, seluruh konten website yang ada ditampung dalam sebuah div dengan id wrapper, jadi sebagai permulaan kita buat markup standar HTML beserta sebuah div dengan id wrapper:

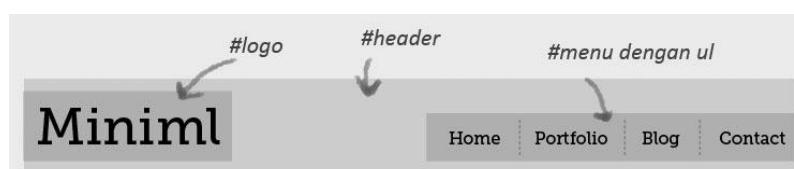
```

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

Kode-kode selanjutnya ditulis di dalam div wrapper ini.

Header

Jika anda perhatikan, bagian header memiliki dua elemen, logo dan menu. Untuk logo kita bisa gunakan heading level 1 (h1) atau dengan gambar, namun kali ini kita coba dengan menggunakan Heading level 1 saja. Dan untuk menu kita gunakan unordered list (ul).



Gambar 95 Header

```

1. <div id="header">
2.   <h1 id="logo">Miniml</h1>
3.   <ul id="menu">
4.     <li><a href="#">Home</a></li>
5.     <li><a href="#">Portfolio</a></li>
6.     <li><a href="#">Blog</a></li>
7.     <li><a href="#">Contact</a></li>
8.   </ul>
9. </div>

```

Slideshow

Bagian selanjutnya adalah slideshow, bagian ini akan menampilkan gambar-gambar secara bergantian (slideshow). Elemen di dalamnya antara lain gambar-gambar slide dan sebuah heading untuk judul setiap slide disertai dengan link.



Gambar 96 Slideshow

```

1. <div id="slideshow">
2.   
3.   <h1><a href="">Belajar HTML & CSS : Tutorial fundamental mempelajari HTML dan CSS</a></h1>
4. </div>

```

Untuk sementara kita gunakan gambar statis terlebih dahulu, untuk membuat slideshow ini berfungsi sebagai mestinya kita akan bahas di bagian jQuery selanjutnya.

Langkah



Gambar 97 Layout Kolom untuk bagian #Langkah

Bagian ini berisi langkah-langkah (sesuai dengan desain) secara konsep, bagian ini memiliki layout tiga kolom. Untuk itu kita gunakan div dengan class kolom. Jadi kira-kira kodennya menjadi seperti berikut :

```

1. <div id="langkah">
2.   <h2>Bagaimana menjadi seorang Web Designer?</h2>

3.   <div class="kolom">
4.     <h3>Belajar HTML</h3>
5.     <p>Pelajari terlebih dahulu apa itu HTML, bagaimana cara pembuatan file HTML,
       Tag-tag HTML yang ada beserta fungsinya</p>
6.   </div>

7.   <div class="kolom">
8.     <h3>Belajar CSS</h3>
9.     <p>Pelajari apa itu CSS, hubungan antara CSS dengan HTML, bagaimana cara
       penggunaan CSS untuk mengatur layout HTML</p>
10.    </div>

11.   <div class="kolom">
12.     <h3>Buat Website</h3>
13.     <p>Setelah mempelajari HTML & CSS saatnya untuk menerapkan keduanya dalam
       kasus konversi design web menjadi file HTML & CSS</p>
14.   </div>
15. </div>
```

Footer

Bagian terakhir adalah footer, pada bagian ini terdapat dua element, yang pertama adalah text copyright (seperti biasa) dan menu footer yang akan kita buat dengan unordered list – ul (lagi).



Gambar 98 Footer

```

<div id ="footer">
  <p>&copy; copyright 2012 by ariona.net</p>
  <ul id="footer-menu">
    <li><a href="#">Tentang</a></li>
    <li><a href="#">Blog</a></li>
    <li><a href="#">twitter</a></li>
    <li><a href="#">kontak</a></li>
  </ul>
</div>
```

Tidak sulit bukan untuk membuat markup HTML dari desain yang diberikan? Yang perlu anda lakukan adalah membayangkan layout dan pembagian masing-masing elemen yang akan digunakan lalu mengodekannya menjadi HTML.

Kode lengkap dari proses Markup yang telah kita lakukan adalah seperti berikut :

```

1.  <!doctypeHTML>
2.  <HTML>
3.      <head>
4.          <title>Miniml</title>
5.      </head>
6.      <body>
7.          <div id="wrapper">
8.              <div id="header">
9.                  <h1 id="logo">Miniml</h1>
10.                 <ul id="menu">
11.                     <li><a href="#">Home</a></li>
12.                     <li><a href="#">Portfolio</a></li>
13.                     <li><a href="#">Blog</a></li>
14.                     <li><a href="#">Contact</a></li>
15.                 </ul>
16.             </div>
17.             <div id="slideshow">
18.                 
19.                 <h1><a href="">Belajar HTML & CSS : Tutorial fundamental  
mempelajari HTML dan CSS</a></h1>
20.             </div>
21.             <div id="langkah">
22.                 <h2>Bagaimana menjadi seorang Web Designer?</h2>
23.                 <div class="kolom">
24.                     <h3>Belajar HTML</h3>
25.                     <p>Pelajari terlebih dahulu apa itu HTML, bagaimana  
cara pembuatan file HTML, Tag-tag HTML yang ada beserta  
fungsiya</p>
26.                 </div>
27.                 <div class="kolom">
28.                     <h3>Belajar CSS</h3>
29.                     <p>Pelajari apa itu CSS, hubungan antara CSS dengan  
HTML, bagaimana cara penggunaan CSS untuk mengatur  
layout HTML</p>
30.                 </div>
31.                 <div class="kolom">
32.                     <h3>Buat Website</h3>
33.                     <p>Setelah mempelajari HTML & CSS saatnya untuk  
menerapkan keduanya dalam kasus konversi design web  
menjadi file HTML & CSS</p>
34.                 </div>
35.             </div>
36.             <div id = "footer">
37.                 <p>&copy; copyright 2012 by ariona.net</p>
38.                 <ul id="footer-menu">
39.                     <li><a href="#">Tentang</a></li>
40.                     <li><a href="#">Blog</a></li>
41.                     <li><a href="#">twitter</a></li>
42.                     <li><a href="#">kontak</a></li>
43.                 </ul>
44.             </div>
45.         </div>
46.     </body>
47. </HTML>
```

Simpan dengan nama “index.html” di dalam folder yang kita buat sebelumnya “Latihan Slicing”.

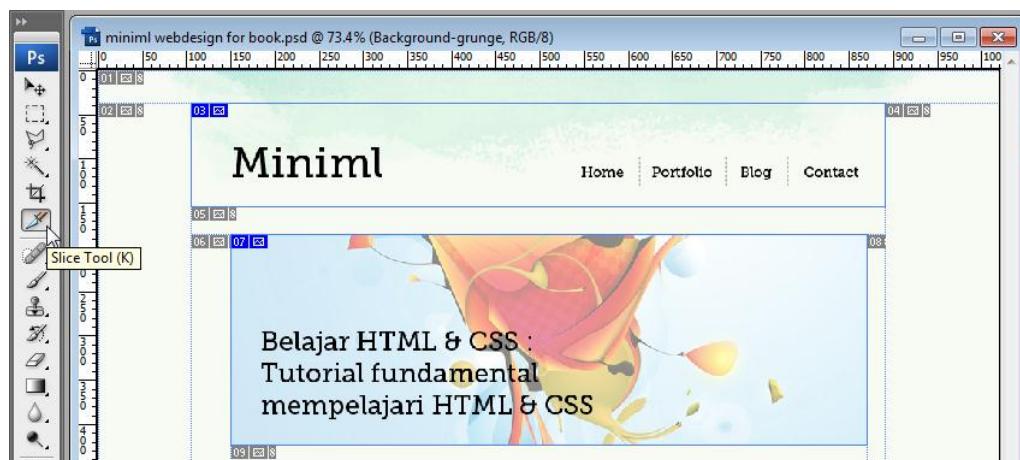
21.2. Slicing

Slicing adalah proses pemotongan gambar-gambar yang akan digunakan dalam website agar sesuai dengan desain yang diinginkan. Gambar-gambar tersebut meliputi background, logo, slide, icon dan gambar pendukung lainnya.

Desain Website tersebut saya sertakan dalam File penyerta buku ini, untuk membukanya anda membutuhkan software Adobe Photoshop. Buka file **miniml-website-design.psd** dan kita mulai proses slicing.

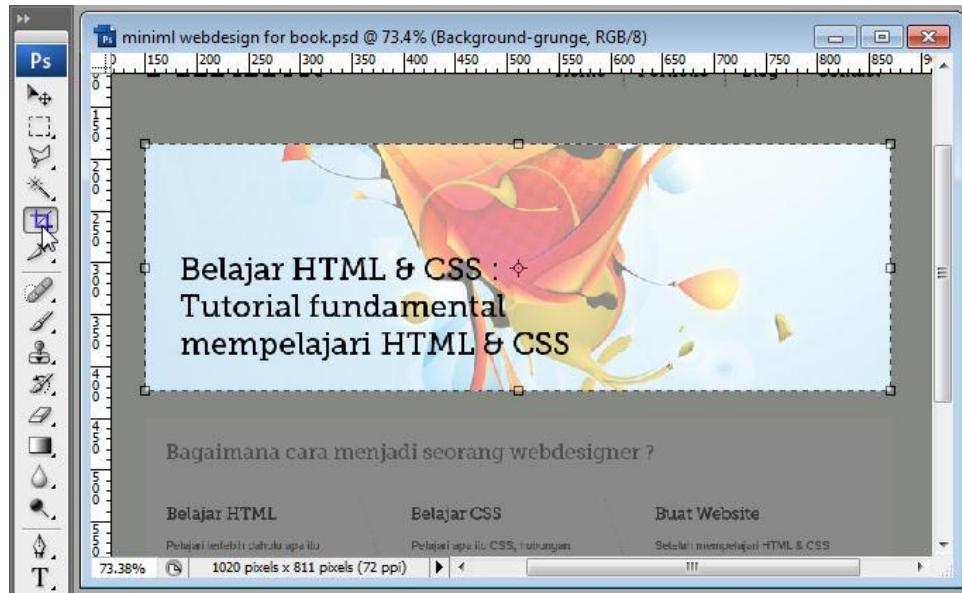
Ada tiga cara untuk melakukan slicing dengan photoshop:

1. Slice Tool : Cara ini lebih efektif untuk melakukan proses slicing secara batch/sekaligus. Kita tinggal menyeleksi bagian-bagian yang akan di slice dengan tool ini kemudian pilih menu save for web & device.



Gambar 99 Contoh Penggunaan Slice Tool

2. Cropping Tool : Buat seleksi ada bagian yang ingin diambil, crop, save menjadi gambar, lalu kembalikan ke kondisi awal dengan CTRL+ALT+Z. Cukup ribet namun terkadang menguntungkan.



Gambar 100 Slicing dengan teknik Cropping

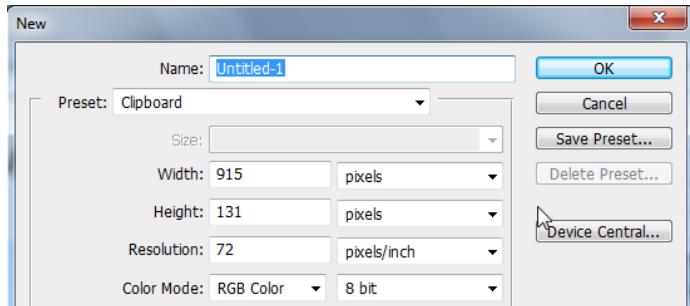
3. Manual Slicing : Cara ini adalah cara manual namun lebih efektif. Mengingat hanya bagian-bagian tertentu yang akan kita ambil, jadi kita cukup memilih layer yang dibutuhkan, copy, buat file baru, paste dan save. Dan Cara ini yang akan kita gunakan untuk melakukan slicing pada desain website yang sedang kita kerjakan.

Pertama-tama kita akan mengambil background, tepatnya di bagian atas gambar. Pada window layer pilih layer dengan nama background-grunge, lalu **tekan CTRL** dan **Klik** pada **Layer Thumbnail**. Langkah ini akan menyeleksi bagian yang terdapat dalam layer tersebut.



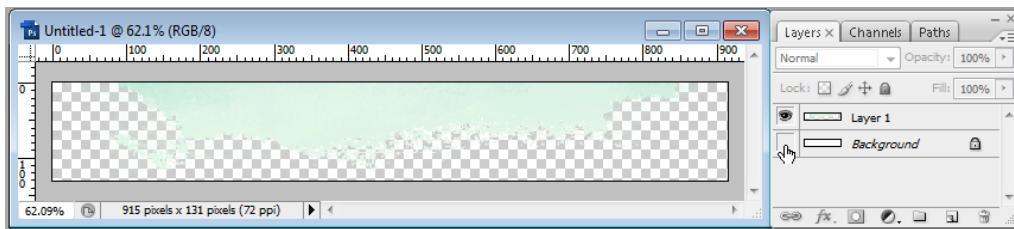
Gambar 101 Ctrl+Click untuk menyeleksi konten dari layer

Selanjutnya tekan Kombinasi tombol **CTRL+C** untuk meng-copy isi dari layer tersebut. Buat dokumen/canvas baru dengan menekan **CTRL+N (File > New)**. Ukuran dari canvas baru akan sesuai dengan isi dari seleksi yang kita copy.



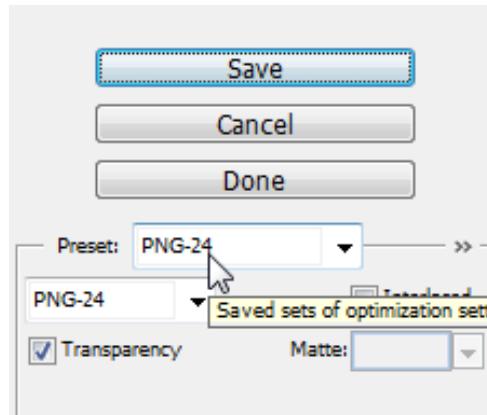
Gambar 102 Membuat file baru untuk hasil slice

Tekan OK, lalu tekan tombol **CTRL+V** untuk mem-paste apa yang kita copy sebelumnya. Hilangkan tanda “mata” pada layer background untuk menyembunyikan layer ini karena kita akan membuatnya transparan.



Gambar 103 Hilangkan icon mata untuk menyembunyikan layer

Selanjutnya kita tinggal menekan tombol **CTRL+ALT+SHIFT+S** untuk menyimpan sebagai gambar web/device atau pilih menu **“File > Save for Web & Devices”**. Pilih format PNG agar gambar yang dihasilkan dapat memiliki area transparan. Tekan Save untuk menyimpan gambar, simpan di folder Images yang ada di dalam folder latihan yang kita buat dengan nama **“background.png”** (tanpa tanda kutip).

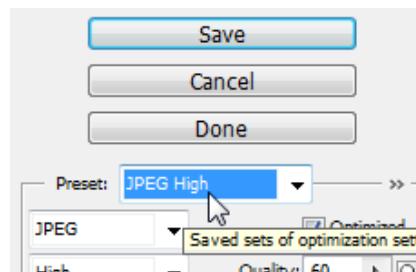


Gambar 104 PNG-24 mendukung transparansi pada gambar



*Jika gambar yang dibutuhkan memiliki area transparan, selalu pilih preset **PNG-24***

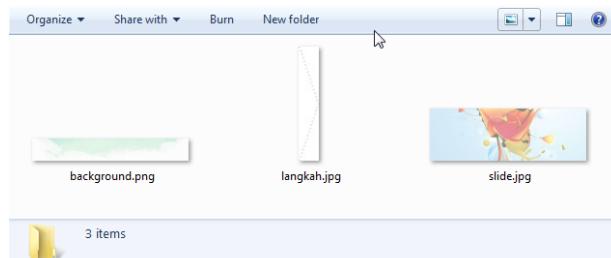
Selanjutnya kita membutuhkan gambar untuk Slideshow. Pada Window Layer, Buka folder Slideshow lalu klik layer Slide. Lakukan proses slicing seperti langkah sebelumnya, **CTRL+Klik**, **Copy**, Buat Dokumen Baru, *Paste*. Kali ini pilih *preset JPEG High*, dan simpan di folder yang sama seperti sebelumnya dengan nama “**slide.jpg**”.



Gambar 105 Preset JPG High untuk kualitas gambar tinggi

Gambar terakhir yang kita butuhkan adalah tanda panah pada bagian “Langkah”. Cari Layer “Separator” di dalam folder langkah lalu lakukan langkah slicing seperti sebelumnya. Simpan dengan nama “**langkah.jpg**”. Untuk bagian ini cukup satu saja yang di slice, OK.

Kini dalam folder images, kita mempunyai 3 gambar pendukung, background.png, slide.jpg dan langkah.jpg.



Gambar 106 File-file hasil slicing

21.3. Pemberian Style

Lanjut ke bagian paling menyenangkan (menurut saya), pemberian Style atau CSS. Buat file baru dengan nama style.CSS, simpan di folder yang sama dengan file HTML yang kita buat sebelumnya.

CSS Reset

Seperti biasa, kita buat dulu CSS Reset untuk mereset margin dan padding bawaan dari tiap browser.

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

Body

Untuk body kita akan tambahkan warna background dan gambar background yang telah kita *slice*.

```
body{
    background-color:#f8fbf4;
    background-image: url("images/background.png");
    background-repeat: no-repeat;
    background-position: center top;
    font-size: 14px;
}
```

Ingat, kode diatas bisa anda singkat dengan menggunakan CSS-shorthand untuk background seperti berikut :

```
background:url("images/background.png") center top no-repeat #f8fbf4;
```

Saya uraikan agar anda mengerti masing-masing bagiannya.

Heading

Untuk heading, kita akan menggunakan jenis font yang berbeda dengan font keseluruhan. Font yang digunakan adalah museo slab. Anda dapat menginstallnya dari folder resources yang disertakan dalam buku ini.

```
h1,h2,h3,h4,h5,h6{
    font-family:"Museo Slab 500", impact,arial,sans-serif;
    font-weight: normal;
}
```

Wrapper

Dari desain, ukuran wrapper adalah 720px. Anda bisa menggunakan Ruler Tool untuk mengukur lebar wrapper dengan patokan gambar slide.

```
#wrapper{
    width:720px;
    margin:0 auto;
}
```

Masih ingat fungsi `margin:0 auto;` ? Ya, property ini berfungsi untuk membuat #wrapper menjadi rata tengah meski di zoom-in atau di zoom-out.

Header

Di dalam header kita akan ada 2 elemen yang memiliki floating. Jadi gunakan solusi floating pada div header.

```
#header{
    overflow: hidden;
    margin-bottom: 20px;
    padding:30px 0
}
```

Untuk Logo, karena kita menggunakan h1 maka yang kita perlukan adalah merubah font sesuai dengan yang ada dalam desain. Font yang digunakan adalah **Museo Slab 500**(font disertakan di dalam File penyerta buku ini).

```
#logo{
    font-family:"Museo Slab 500", impact, sans-serif;
    float:left;
    font-size:40px;
}
```

Kemudian untuk menu, kita buat rata kanan, lalu membuat list-item menjadi satu baris serta memberikan sentuhan border sebagai separator :

```
#menu{
    float:right;
    list-style:none;
    margin-top:10px;
}
#menu li{ display: inline }
#menu li a{
    padding:5px 10px;
    border-right: 1px dashed #aeb59c;
    text-decoration:none;
    color:black;
}
#menu li:last-child a{ border-right:none }
```

Slideshow

Untuk slideshow kita akan menerapkan positioning pada elemen #slideshow dan h1.

```
#slideshow{
    position:relative;
}
```

Kita set position menjadi relative karena h1 akan kita buat menjadi absolute. Sehingga pergerakan posisi h1 tergantung/relative pada #slideshow.

```
#slideshow h1{
    position:absolute;
    bottom:20px;
    left:20px;
    font-weight:normal;
}
```

Karena di dalam heading kita menambahkan link/anchor kita juga harus memberikan style pada link tersebut agar tidak ditampilkan dengan style bawaan.

```
#slideshow h1 a{
    color:black;
    text-decoration:none;
    font-family:"Museo Slab 500",impact,sans-serif;
    font-weight: normal;
}
```

Layout Kolom untuk #langkah

Seperti yang telah saya singgung, pada bagian #langkah kita akan menerapkan layout kolom. Caranya sangatlah mudah. Kita hanya akan menambahkan floating pada .kolom.

```
#langkah{
    overflow:hidden;
    background:white;
    margin:20px 0;
}
#langkah h2{ padding:10px }
#langkah .kolom{
    width:220px;
    margin:10px;
    padding:0 30px 0 0;
    float:left;
    background:url("images/langkah.jpg") right top no-repeat;
}
#langkah .kolom:last-child{background:none;}
```

Pertama-tama kita tambahkan overflow:hidden pada #langkah, karena di dalamnya akan ada elemen yang memiliki property float (baca lagi Masalah Floating).

Selanjutnya kita tentukan width/lebar dari .kolom, tentukan padding-right untuk background (tanda siku).

Terakhir kita hilangkan background pada kolom paling akhir dengan menggunakan pseudo-class (.kolom:last-child)

Footer

Untuk bagian #footer kita hanya memerlukan border-top dan membuat menu ditampilkan satu baris serta merata kirikan elemen tersebut.

```
#footer{
    border-top:1px dashed #aeb59c;
    padding:10px;
}
#footer p{ float:left }
#footer-menu{
    float:left;
    margin-left:30px;
    list-style:none;
}
#footer-menu li{
    display:inline;
    margin-right:20px;
}
#footer-menu li a{
    color:#777;
```

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

Akhirnya proses Konversi telah selesai, Jika anda membukanya di browser anda akan mendapatkan tampilan yang sama dengan desain. Namun masih ada yang kurang, Kita belum membuat Slideshow berfungsi sebagaimana mestinya.

Mengingat HTML & CSS tidak memiliki fungsionalitas untuk membuat sebuah slideshow⁸, maka bersama dengan ini saya akan memperkenalkan Javascript pada Anda.

⁸ Pada kenyataannya kita bisa membuat slideshow dengan menggunakan teknologi CSS3, namun mengingat teknik tersebut adalah teknik lanjut.

Bab 22

Javascript

22.1. Pengenalan Javascript

Javascript adalah bahasa pemrograman web yang berjalan disisi Client/Browser. Javascript biasa digunakan untuk memanipulasi element-element HTML dan menambahkan Style secara otomatis atau lebih sederhananya membuat dokumen HTML menjadi lebih Interaktif.

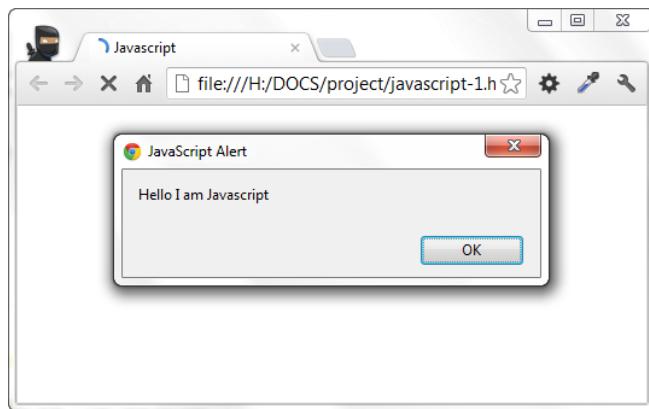
Untuk menambahkan Script javascript pada dokumen HTML, anda dapat menggunakan tag script dan diletakkan di dalam head atau di area manapun di dalam tag body.

```
...
<head>
    <title>Javascript</title>
    <script type="text/javascript">
        ... Script javascript diketik disini...
    </script>
</head>
...
...
```

Saya akan memberikan contoh sederhana dari javascript ini, tambahkan script berikut pada latihan sebelumnya atau anda dapat membuat file HTML baru dengan nama file `latihan14.html`.

```
...
<head>
    <title>Javascript</title>
    <script type="text/javascript">
        alert("Hello I am Javascript");
    </script>
</head>
...
...
```

Script tersebut akan menampilkan MessageBox dengan teks "Hello I am Javascript" ketika file HTML dibuka.



Gambar 107 Kotak pesan dengan javascript

Berikut contoh lain dari penggunaan Javascript yang bias anda lihat dalam file latihan15.html:

```

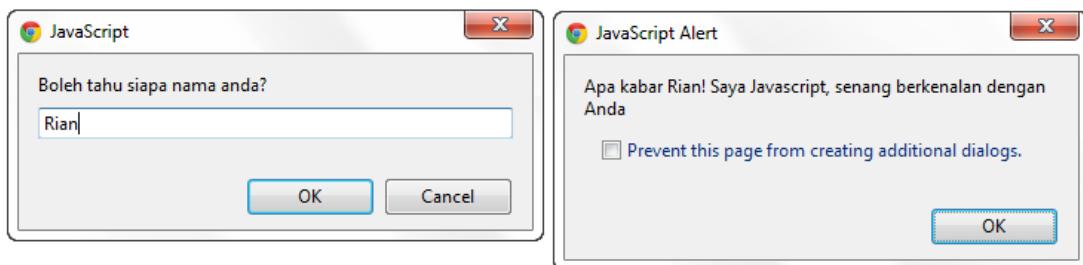
1. <script type="text/javascript">
2.     var nama;
3.     nama=prompt('Boleh tahu siapa nama anda?');
4.     if(nama){
5.         alert('Apa kabar '+nama+'! Saya Javascript, senang berkenalan dengan Anda');
6.     }else{
7.         alert('Hhmm sepertinya anda tidak mau berkenalan dengan saya.');
8.     }
9. </script>
```

Penjelasan Script :

Pertama, kita buat sebuah variabel nama. Kemudian kita meminta Javascript untuk menanyakan nama pengunjung lalu menyimpannya di dalam variabel nama.

Selanjutnya, Kita memilih, apakah variabel nama terisi (pengunjung mengetikkan nama), maka akan ditampilkan Message Box dengan teks "Apa kabar (Nama yang dimasukkan)!, Senang berkenalan dengan Anda".

Jika pengunjung tidak memasukkan nama (langsung mengklik ok atau cancel) maka akan ditampilkan message box "Hhmm sepertinya anda tidak mau berkenalan dengan saya.".



Gambar 108 Contoh penggunaan kondisi

Script tersebut adalah contoh sederhana dari penggunaan Javascript, Anda dapat mempelajarinya secara lanjut di <http://www.w3schools.com/js/>.

22.2. jQuery

jQuery, adalah library atau sekumpulan kode javascript siap pakai untuk memanipulasi elemen HTML dengan mudah dan cepat. Anda dapat mendownload jQuery di situs resminya <http://www.jquery.com>.

Untuk menggunakan jQuery, Anda diharuskan menyertakan jQuery dengan menggunakan tag Script.

```
<script type="text/javascript" src="jquery.js"></script>
```

Berikut ini adalah contoh sederhana penggunaan jQuery untuk memanipulasi HTML, tambahkan script berikut pada file latihan anda.

```
<script type="text/javascript" src="jquery.js"></script>
<script>
    $(document).ready(function(){
        $("body").CSS('background-color','red');
    });
</script>
```

Penjelasan Script :

```
$(document).ready(function(){
    ...
});
```

Block kode ini memastikan script dijalankan setelah file HTML selesai dimuat.

```
$(body).CSS('background-color','red');
```

Script tersebut akan merubah warna body dengan menambahkan atribut style yang memiliki nilai "background-color:red".

22.3. jQuery Plugin

jQuery Plugin adalah plugin/suplemen untuk menambah fungsionalitas dari jQuery. Plugin ini biasanya dibuat untuk memudahkan pengguna dalam membuat suatu aplikasi. Salah satunya adalah membuat Slider, Slideshow dan lain sebagainya.

Anda tinggal memanggil library plugin jquery tersebut seperti anda memanggil file javascript, selanjutnya tinggal memanggil fungsi yang disediakan oleh library tersebut.

Sebagai contoh, ada sebuah plugin jQuery Slider dan kita akan menerapkannya pada elemen HTML yang memiliki id slider, Plugin tersebut menyediakan fungsi "slider()" untuk membuat markup HTML kita menjadi sebuah slider. jadi anda tinggal mengetikkan script berikut :

```
$("#slider").slider();
```

Anda dapat mencari plugin-plugin jQuery yang anda butuhkan di internet atau anda dapat mencari sebagian plugin jquery di <http://www.jquery4u.com/plugins/>.



Gambar 109 jQuery4u telah merangkum sebagian plugin jQuery dari A-Z

Bab 23

Studi Kasus 4 – Penambahan jQuery

Slider

Terkait dengan Latihan Slicing sebelumnya kita akan menambahkan fungsionalitas pada Slider agar berjalan sebagaimana mestinya. Dalam kasus ini kita akan menggunakan Plugin Basic jQuery Slider yang bisa anda download di <http://basic-slider.com/>.



Sebelum menggunakan plugin jQuery, sebagiknya anda Baca Documentation/Panduannya terlebih dahulu

Cara penggunaan basic-slider

Dari Dokumentasi basic-slider yang bisa anda baca di <http://basic-slider.com/documentation/>. Berikut ini adalah cara penggunaan dari plugin tersebut :

1. Download pluginnya lalu sertakan dalam file HTML anda.

```
<!-- Sertakan file jQuery -->
<script src="js/libs/jquery-1.6.2.min.js"></script>

<!-- Sertakan plugin basic-slider SETELAH menyertakan jquery -->
<script src="js/basic-jquery-slider.js"></script>

<!-- Include the basic styles -->
<link type="text/CSS" rel="stylesheet" href="CSS/basic-jquery-slider.CSS"/>
```

2. Markup HTML untuk slider mengikuti struktur yang ditentukan oleh basic-slider seperti berikut :

```
<div id="id-slideshow-anda">
    <ul class="bjqs">
        <li><!-- Isi Slide Anda --></li>
        <li><!-- Dapat diisi oleh apapun --></li>
    </ul>
</div>
```

3. Jika sudah melakukan poin 1 dan 2 di atas, baru kita panggil fungsinya seperti berikut :

```
$(document).ready(function(){
    $("#id-slideshow-anda").bjqs({
        'width': 940,
        'height': 340,
        'showMarkers': true,
        'showControls': true,
        'centerMarkers': false
    });
});
```

Di dalam fungsi .bjqs terdapat beberapa parameter untuk pengaturan slider, Anda dapat membaca penjelasan tiap parameternya di link yang saya sebutkan di atas.

Penerapan pada Studi Kasus ke-3

Pada Studi kasus ke-4 ini, kita akan membuat slider yang terdapat pada studi kasus ke-3 berfungsi sebagaimana mestinya yakni sebagai slider.

Perubahan Markup HTML

Sebelum kita mulai membuat slider berfungsi dengan baik, rubah markup HTML berikut:

```
1. <div id="slideshow">
2.   
3.   <h1><a href="">Belajar HTML & CSS : Tutorial fundamental mempelajari HTML dan
   CSS</a></h1>
4. </div>
```

Menjadi seperti di bawah ini :

```
1. <div id="slideshow">
2.   <ul class="bjqs">
3.     <li>
4.       
5.       <h1><a href="">Belajar HTML & CSS, :
   Tutorial fundamental mempelajariHTML dan
   CSS</a></h1>
6.     </li>
7.     <li>
8.       
9.       <h1><a href="">Belajar HTML & CSS : Tutorial fundamental
   mempelajariHTML dan CSS</a></h1>
```

```

10.          </li>
11.          <li>
12.              
13.              <h1><a href="">Belajar HTML &amp; CSS : Tutorial fundamental
14.                  mempelajariHTML dan CSS</a></h1>
15.          </li>
16.      </ul>

```

Pada dasarkan kita akan menampilkan 3 buah slide/gambar. Kita juga akan merubah dan menambah style untuk bagian slideshow ini. Gambar-gambar tambahan bisa anda buat sendiri atau copy gambar yang terdapat di dalam folder "miniml - with slider/images".

Class bjqs yang terdapat pada unordered list adalah ketentuan yang diberikan oleh library jquery slider yang akan kita gunakan (sudah dijelaskan sebelumnya).

Pemanggilan Library

Selanjutnya kita panggil beberapa file javascript yakni file jQuery, plugin jQuery basic-slider, dan file CSS bawaan dari jquery-basic slider.

Ubah bagian header dan tambahkan kode yang ditebalkan berikut :

```

<!DOCTYPE HTML>
<HTML>
    <head>
        <title>Miniml</title>
        <link rel="stylesheet" href="style.css">
        <link rel="stylesheet" href="basic-jquery-slider.css">
        <script type="text/javascript" src="jquery.js"></script>
        <script type="text/javascript" src="basic-jquery-slider.js"></script>

```

Tambahkan juga style berikut ke dalam file style.css yang kita buat sebelumnya :

```

.bjqs-next a, .bjqs-prev a {
    color: orange;
    text-decoration: none;
    background: white;
    padding: 5px 10px;
}

```

Aktivasi Plugin

Langkah terakhir adalah pemanggilan fungsi basic slider yang akan merubah markup HTML kita menjadi Slider “beneran”. Tambahkan kode berikut setelah kode pemanggilan library `basic-jquery-slider.js` di atas.

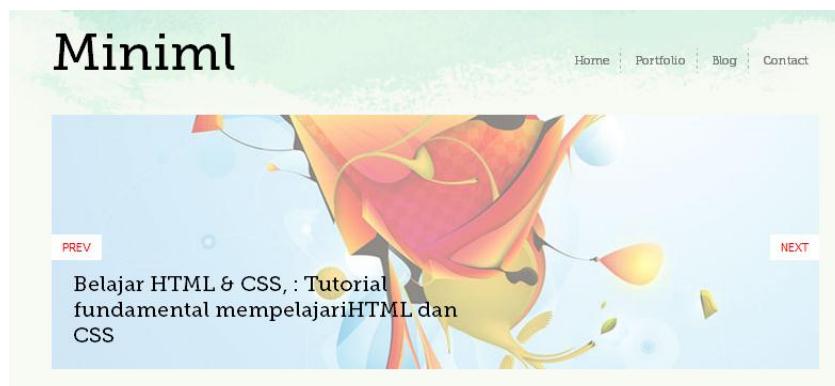
```

1. ...
2. <script type="text/javascript" src="basic-jquery-slider.js"></script>
3. <script type="text/javascript">
4.     $(document).ready(function(){
5.         $('#slideshow').bjqs({
6.             'width' : 720,
7.             'height' : 238,
8.             'showMarkers' : false,
9.             'showControls' : true
10.        });
11.    });
12. </script>

```

Parameter `width` dan `height` mengacu pada lebar dan tinggi gambar slide. Berbeda dengan contoh pemanggilan basic-slider yang telah saya jelaskan sebelumnya, saya hanya menampilkan parameter “`showControls`” untuk navigasi tiap slider dan menyembunyikan Markers dengan memberi nilai **false**.

Kini, jika anda tampilkan dalam web browser, Bagian slider memiliki link “`prev`” dan “`next`”, klik link tersebut untuk memajumundurkan slide.



Gambar 110 Link Prev dan Next untuk navigasi slider

Bab 24

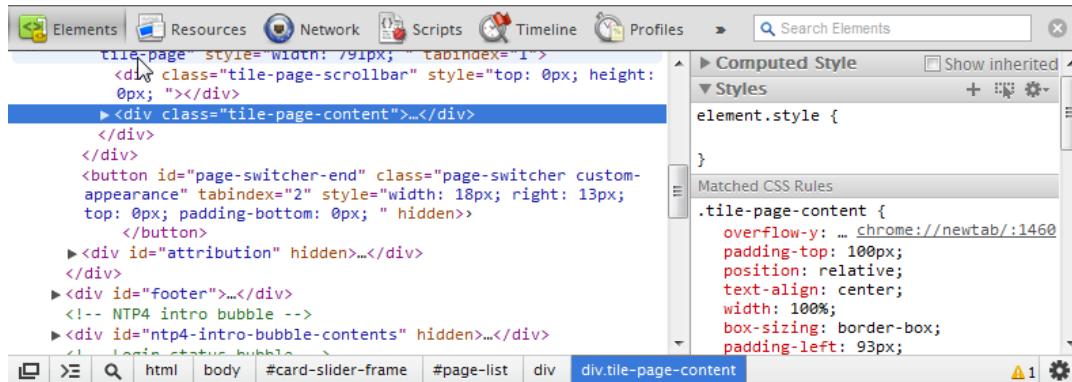
EXTRAS

24.1. Developer Tools dan Firebug

Kedua add-on ini telah saya ceritakan secara singkat di awal buku ini. Fungsi keduanya hampir sama yakni menyediakan berbagai informasi untuk Developer web termasuk informasi tentang dokumen HTML dan CSS yang menjadi konsentrasi dalam buku ini.

Tool ini biasa digunakan untuk mempelajari struktur HTML dan CSS dari web lain, misalnya ketika kita ingin mengetahui bagai mana struktur dari suatu bagian dalam web, anda tinggal klik kanan bagian tersebut, dan pilih menu Inspect Element.

Secara otomatis block HTML bagian tersebut akan ditampilkan lengkap dengan style/CSS yang diterapkan.



Gambar 111 Kode HTML beserta Style yang diterapkan

Dengan begitu anda dapat mempelajari style yang digunakan. Anda-pun dapat menambah, menghapus, dan mengeditnya dengan mengklik dua kali pada kode yang bersangkutan dan anda dapat melihat langsung perubahan yang terjadi dalam browser.

Perubahan yang anda lakukan tidak bersifat permanen, ketika browser anda refresh semuanya akan kembali seperti semula.

24.2. Zen Coding

Zen Coding adalah plugin tambahan yang tersedia untuk beberapa text editor, seperti Notepad++, Sublime text dan masih banyak lagi. Untuk daftar text editor yang didukung oleh zen coding bisa dilihat disitus *official*nya

<http://code.google.com/p/zen-coding/>.

Untuk instalasi zen coding di Notepad++, anda cukup masuk ke menu Tools, Plugin Manager. Selanjutnya scroll sampai anda melihat Zen Coding. Beri tanda dan klik tombol install untuk mulai menginstall zen coding.

Penggunaan zen coding sangatlah mudah. Plugin ini dibuat untuk mempercepat penulisan kode HTML dan CSS, sebagai contoh perhatikan kode HTML berikut :

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

Untuk ukuran anda yang sudah terbiasa mengetik dengan sepuluh jari mungkin hanya membutuhkan 5 menit untuk menulis kode di atas. Lalu bagaimana jika anda tidak menguasai teknik tersebut? Mungkin akan memakan waktu 10 menit atau bahkan lebih.

Dengan zen coding, kode tersebut bisa disingkat menjadi satu baris seperti berikut :

```
#wrapper>#content+#sidebar+#footer
```

Selanjutnya anda tinggal menekan kombinasi tombol yang telah ditentukan untuk melebarkan (expand) kode tersebut, dan 5 baris kode diatas pun akan anda dapatkan. Sangat cepat bukan?

Lalu apa arti dari satu baris kode zen coding sebelunya?

#wrapper, #content , #sidebar dan #footer

#wrapper, #content , #sidebar dan #footer berarti kita membuat sebuah div dengan nama id wrapper, content, sidebar dan footer. Penambahan tanda pagar mengacu pada

penulisan selector CSS untuk id (baca kembali bab pengenalan CSS). Dengan begitu untuk penulisan sebuah div dengan nama class artikel anda tinggal menuliskan titik sebelum nama class (.artikel).

Ketika anda tidak menambahkan nama tag yang akan dibuat, maka tag yang akan dibuat adalah tag div. Jika Anda ingin membuat tag tertentu anda tinggal mengetikkan nama tag tersebut. Misal :

```
ul#menu
```

Akan menghasilkan

```
<ul id="menu"></ul>
```

Child

Selanjutnya tanda “lebih besar dari” (>) digunakan untuk membuat child dari tag sebelumnya. Pada contoh kasus di atas adalah : #wrapper>

Ini menandakan kode-kode setelah tanda ">"(yakni #content, #sidebar dan #footer) akan berada di dalam #wrapper (Child dari #wrapper).

Perulangan kode

Untuk menuliskan beberapa baris kode yang sama, anda dapat menggunakan operator *n dimana n adalah jumlah yang diinginkan. Contoh kasusnya adalah pembuatan list item menu.

```
<ul id="menu">
    <li><a href=""></a><li>
    <li><a href=""></a><li>
    <li><a href=""></a><li>
    <li><a href=""></a><li>
    <li><a href=""></a><li>
</ul>
```

Penulisan kode zen coding dari beberapa baris kode diatas adalah seperti berikut :

```
ul#menu>li*5>a
```



*Zen Coding sangat **dianjurkan** untuk anda yang sudah **memahami** HTML dan CSS, jika anda masih belum mengerti sepenuhnya akan HTML dan CSS saya sarankan untuk mengetik kode secara manual agar anda mengingat apa yang sedang anda pelajari.*

Sebenarnya masih banyak lagi tata cara penulisan kode HTML dengan bantuan Zen Coding ini, begitu pula penulisan property-property CSS, anda dapat membaca daftar singkatan zen-coding yang bisa di download dari situs resminya.

24.3. Menambahkan Dummy Text

Ketika anda membutuhkan sejumlah teks/kalimat/paragraf untuk kepentingan visual, Anda akan membutuhkan dummy text. Dummy text adalah sekumpulan kalimat/paragraf tanpa makna seperti :

```
 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
```

Paragraf tersebut tidak memiliki makna namun berguna ketika kita membutuhkan konten cepat.

Untuk mendapatkan teks tersebut, jika anda menggunakan text editor sublime text, anda tinggal mengetikkan `lorem` lalu tekan tombol Tab.

Anda juga dapat membuatnya dengan menggunakan Aplikasi pengolah kata Microsoft Word, caranya dengan mengetikkan :

```
=lorem(jumlahParagraf, jumlahKalimat)
```

Lalu tekan Enter untuk mendapatkan text dummy sesuai dengan jumlah kalimat dan paragraf yang ditentukan.

Jika anda menggunakan text editor Sublime text, anda tinggal mengetikkan kata lorem dan tekan tab untuk mendapatkan dummy text ini.

24.4. Buat situs anda “online”

Untuk membuat halaman web yang anda buat “online” sehingga dapat diakses oleh semua orang melalui internet, anda harus menyimpan seluruh file yang anda buat di tempat penyimpanan online yang biasa disebut dengan **hosting** dan mengarahkannya dengan **domain**.

Mengenal Hosting dan Domain

Sebelumnya telah dijelaskan secara singkat bahwa Hosting adalah tempat penyimpanan online untuk file-file website anda sehingga semua orang dapat mengaksesnya di internet. sedangkan untuk mengarahkan orang-orang agar membuka halaman web anda, diperlukan sebuah alamat yang biasa disebut dengan **domain** misalnya www.facebook.com.

Untuk menyederhanakan istilah-istilah tersebut, bayangkanlah **Hosting** itu sebuah Rumah, dan domain sebagai **Alamat** dari rumah tersebut.

Lalu dimana saya bisa mendapatkan domain dan hosting ini? Jawabannya, Anda dapat menyewa keduanya. Sudah banyak jasa-jasa Hosting indonesia yang menawarkan jasa sewa Hosting dan domain dengan harga yang murah, anda dapat menyewanya perbulan atau bahkan pertahun dengan berbagai paket sesuai kebutuhan.

Tips memilih Hosting

Murah saja tidaklah cukup, jangan terpengaruh oleh harganya yang miring, tapi lakukan survey terlebih dahulu akan kinerja dari hosting yang akan kita sewa. Karena biasanya kendala dari sewa hosting sangat banyak, mulai dari server yang selalu down sehingga halaman web anda tidak bisa dibuka oleh orang lain.

Sesuaikan pemilihan paket data sesuai dengan ukuran situs anda, semakin kecil ukuran hosting yang akan disewa semakin murah pula harga yang harus anda bayar.

Tips memilih Domain

Gunakanlah nama domain yang pendek atau yang lebih mudah diingat oleh semua orang, misalnya toko-bagus.com, ilmu-komputer.com atau nama lain yang akan mudah diingat.

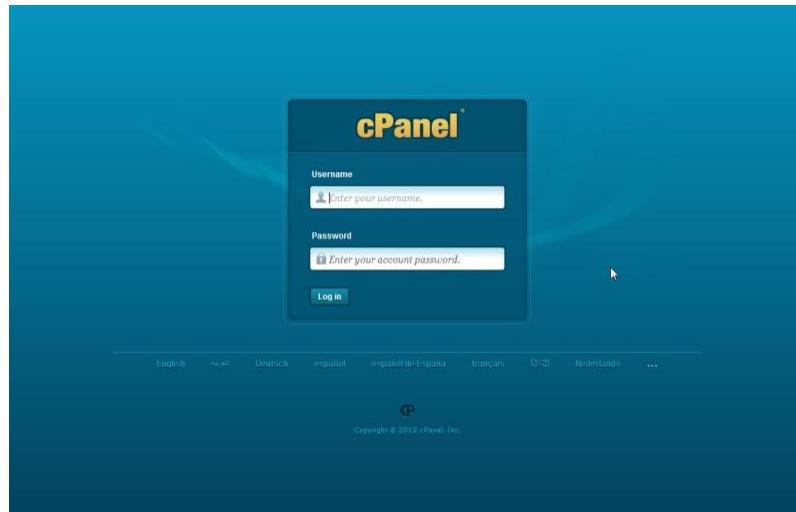
Harga domain tergantung dari ekstensi yang anda pilih misalnya .com , .net , .org , .co.id dan lain sebagainya.

Pada Contoh yang akan saya berikan, kita akan menyewa hosting dari salah satu jasa hosting lokal. Sebagai contoh saya akan memilih paket domain dan hosting dengan size 150mb, cukup untuk menampung seluruh situs saya. Setelah proses pembayaran selesai, kita tinggal menggunakan hosting tersebut tanpa harus bingung mengatur hosting dan domain karena semuanya sudah dilakukan oleh penyedia jasa.

Small	Medium	Large
Rp. 120.000 Per Tahun	Rp. 140.000 Per Tahun	Rp. 160.000 Per Tahun
50 MB Disk Space	100 MB Disk Space	150 MB Disk Space
4 GB Data Transfer	5 GB Data Transfer	6 GB Data Transfer
1 Addon/Park Domain	1 Addon/Park Domain	1 Addon/Park Domain
10 Email Account	10 Email Account	10 Email Account
10 Sub Domain	10 Sub Domain	10 Sub Domain
10 MySQL Database	10 MySQL Database	10 MySQL Database
Rendah Prioritas Bantuan	Rendah Prioritas Bantuan	Rendah Prioritas Bantuan
Gratis Domain	Gratis Domain	Gratis Domain
Pesan (Server ID)	Pesan (Server ID)	Pesan (Server ID)
Pesan (Server USA)	Pesan (Server USA)	Pesan (Server USA)

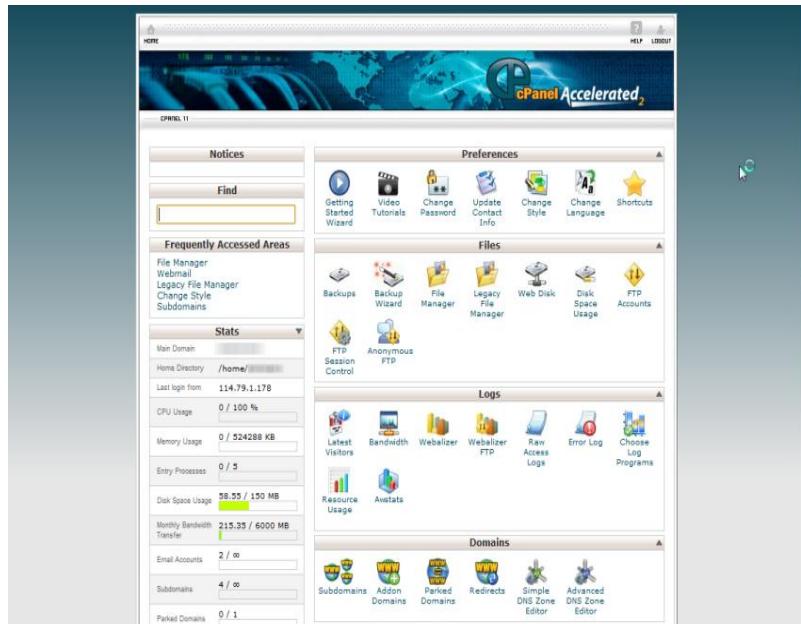
Gambar 112 Tabel harga hosting+domain di salah satu jasa hosting lokal

Pertama-tama, kita masuk ke pusat pengaturan Hosting yang biasa disebut dengan cpanel. Biasanya dapat diakses melalui alamat :www.domain-anda.com/cpanel. Andapun akan dibawa ke halaman berikut.



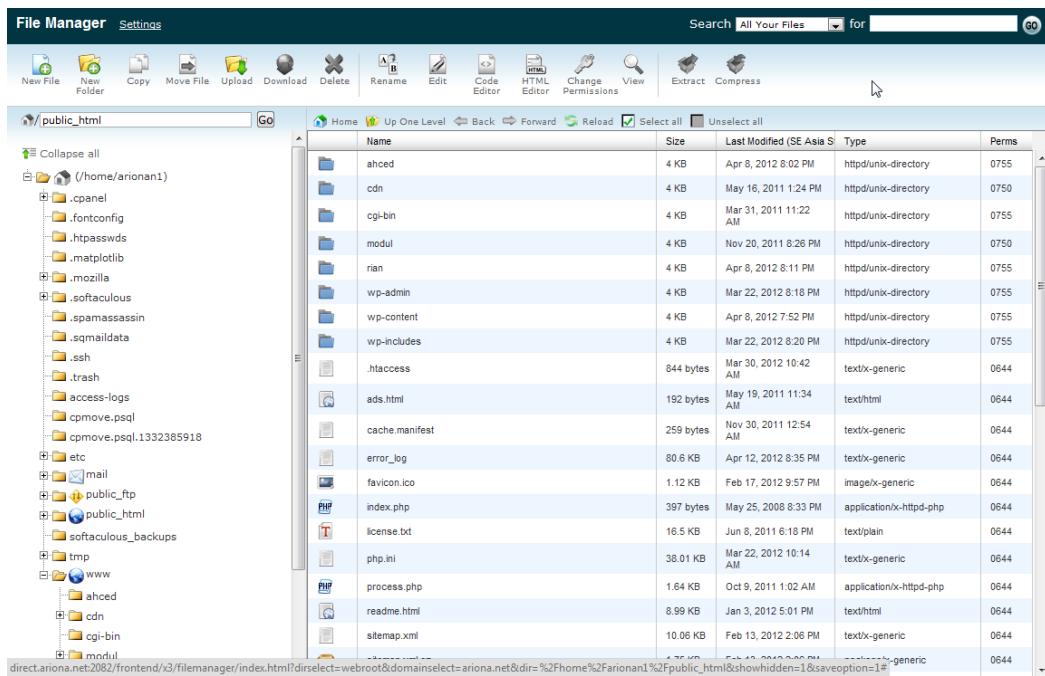
Gambar 113 Halaman login cPanel

Masukkan username dan password yang telah diberikan oleh jasa hosting yang anda gunakan untuk memasuki cpanel. Setelah anda login, berikut ini adalah tampilan halaman web yang akan anda dapatkan.



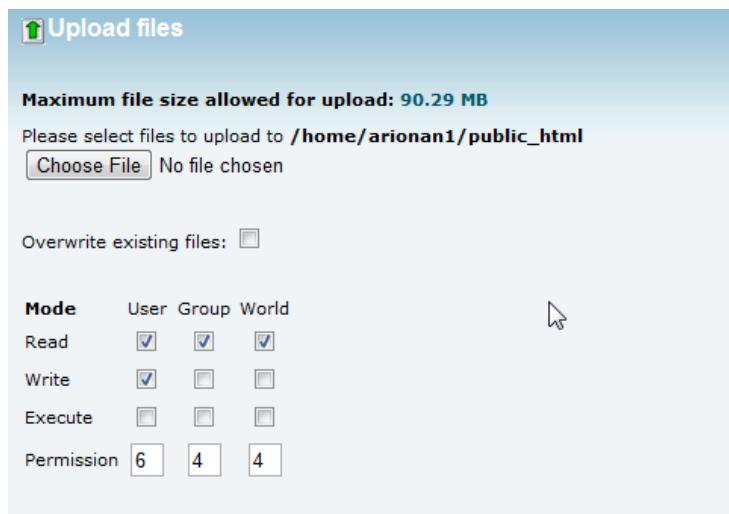
Gambar 114 Halaman cPanel

Kita mulai dengan mengupload seluruh file-file HTML dan file-file yang bersangkutan (CSS, gambar dan lain sebagainya), carilah Icon File Manager, ketika anda klik, berikut tampilan yang akan anda dapatkan, hampir sama dengan explorer atau aplikasi pencari berkas yang terdapat dalam komputer anda.



Gambar 115 File Manager untuk mengatur file di hosting

Lalu dimana kita harus meletakkan file yang kita punya agar dapat diakses oleh orang lain di internet? Anda harus mengupload file-file web anda ke folder **www** atau **public_HTML**. Klik tombol upload, dan uploadlah file anda satu persatu.



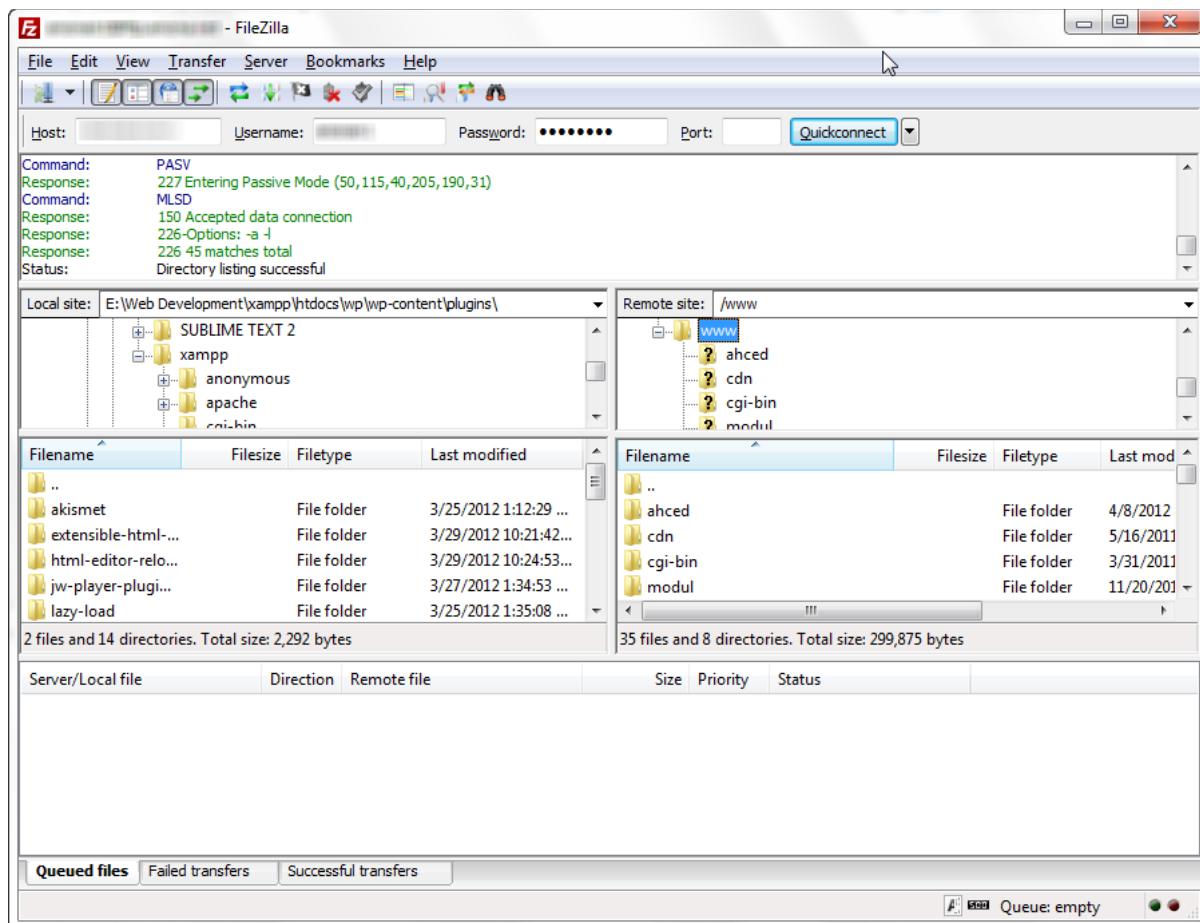
Gambar 116 Halaman untuk mengupload file

Jika semua file-sudah anda upload, andapun dapat mengaksesnya melalui alamat domain anda www.domain-anda.com.

Jika anda membuat folder lagi dalam folder www, anda dapat mengaksesnya lewat alamat : www.domain-anda.com/nama-folder.

24.5. Menggunakan FTP Client

Jika sebelumnya kita mengupload file satu persatu, sekarang kita akan belajar menggunakan FTP⁹ client untuk mengupload file-file web ke hosting kita. FTP Client adalah aplikasi kecil, untuk mengupload file-file dengan mudah ke hosting tanpa harus membuka Cpanel. Salah satu aplikasi yang paling sering digunakan adalah filezilla yang bisa anda download dari filezilla.com.



Gambar 117 FileZilla, Aplikasi FTP client free dan ringan

Masukkan, Host (domain-anda.com), username dan password cpanel anda, pada kolom login. Setelah login, maka dibagian kanan akan ditampilkan seluruh file dan folder dalam hosting anda. Dan di bagian kiri adalah file-dan folder di komputer anda.

⁹ FTP merupakan kependekan dari File Transfer Protocol, FTP digunakan untuk berinteraksi dengan file-file yang terdapat dalam server hosting dengan mudah

Cara mengupload file yang diinginkan sangatlah mudah, anda tinggal menyeleksi file-file yang dinginkan dan klik upload atau drag ke bagian kanan dan secara otomatis file-file tersebut akan diupload oleh FileZilla.

Selain mengupload, andapun dapat mendownload file yang terdapat dalam hosting anda dengan cara yang sama, yaitu memindahkannya dari kolom sebelah kanan ke kiri.

EPILOG

That's it, Sekarang anda telah mempelajari dasar-dasar dari HTML, CSS dan sedikit pengetahuan tentang Javascript. Buku ini hanyalah sebagian kecil dari pembahasan seputar Web Design, HTML, CSS dan Javascript.

Setelah selesai memahami buku ini bukan berarti perjalanan anda dalam mempelajari HTML dan CSS telah usai, masih banyak yang harus anda pelajari seperti Tag-tag baru HTML5, Efek-efek yang bisa diterapkan dengan CSS3, Penggunaan Javascript untuk interaktifitas website, Teknik Mendesain Website dengan Photoshop dan lain sebagainya.

Materi-materi yang saya sebutkan di atas bisa didapatkan dengan mudah di internet ataupun dalam buku-buku tentang Web Design/Web Development yang terdapat di toko-toko buku.

Daftar Pustaka

Buku

Way, Jeffrey Jordan. *From Photoshop to HTML & CSS*. 2009.Rockable Press.

Ta'eed, Collis. How to be Rockstar Wordpress Designer. 2009.Rockable Press.

Website

<http://www.w3schools.com>

<http://learncss.tutsplus.com>

<http://www.ariona.net>

Tentang Penulis



Rian Ariona adalah Seorang Web Designer dan Developer dari Bandung yang memulai karirnya sebagai Desainer Grafis dan pada akhirnya Memfokuskan diri dalam dunia Web Design dan Development.

Penulis sering berbagi pengetahuannya mengenai Web di situs pribadinya www.ariona.net, melalui tweetnya di [@ariona_rian](https://twitter.com/ariona_rian) atau di Channel youtubenya youtube.com/user/arionarian.