



PEMROGRAMAN DASAR BERBASIS WEB



HERU KARTIKA CANDRA

Pemrograman Dasar Berbasis Web

Undang-Undang No. 28 Tahun 2014 Tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Perlindungan Pasal 26

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap :

- i. penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- ii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- iii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan
- iv. penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp 100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp 500.000.000,00 (lima ratus juta rupiah).
3. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf a, huruf b, huruf e, dan/atau huruf g untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan/atau pidana denda paling banyak Rp 1.000.000.000,00 (satu miliar rupiah).
4. Setiap Orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan/atau pidana denda paling banyak Rp 4.000.000.000,00 (empat miliar rupiah).

Pemrograman Dasar Berbasis Web

Heru Kartika Candra



POLIBAN PRESS

Pemrograman Dasar Berbasis Web

Penulis :
Heru Kartika Candra

ISBN :
978-623-5259-09-3 (PDF)

Editor dan Penyunting :
Reza Fauzan

Desain Sampul dan Tata letak :
Rahma Indera; Eko Sabar Prihatin

Penerbit :
POLIBAN PRESS
Anggota APPTI (Asosiasi Penerbit Perguruan Tinggi Indonesia)
no.004.098.1.06.2019
Cetakan Pertama, 2024

Hak cipta dilindungi undang-undang
Dilarang memperbanyak karya tulis ini dalam bentuk
dan dengan cara apapun tanpa ijin tertulis dari penerbit

Redaksi :
Politeknik Negeri Banjarmasin, Jl. Brigjen H. Hasan Basry,
Pangeran, Komp. Kampus ULM, Banjarmasin Utara
Telp : (0511)3305052
Email : press@poliban.ac.id

Diterbitkan pertama kali oleh :
Poliban Press, Banjarmasin, Januari 2024

KATA PENGANTAR

Bismillahirrahmanirrahim,
Assalamu'alaikum wa rahmatullahi wa barakatuh.

Kami mempersembahkan sebuah buku yang berjudul "Pemrograman Dasar Berbasis Web C#: Kajian Sejarah, Teori, dan Metodologi." Buku ini merupakan suatu langkah strategis dalam mengenalkan Pemrograman Dasar Berbasis Web C# kepada para pembaca. Melalui buku ini, pembaca diajak untuk memahami esensi dan perkembangan pemrograman dasar dalam konteks web.

Buku ini tak hanya membahas konsep-konsep dasar pemrograman web, namun juga memperkenalkan kajian sejarah, teori, dan metodologi yang melandasi kemajuan teknologi informasi. Dengan menyajikan materi secara elementer, diharapkan mahasiswa dapat memahami beberapa aspek pemrograman dasar secara kontekstual. Meskipun sederhana, buku ini diharapkan dapat menjadi panduan yang bermanfaat bagi pembaca, terutama mahasiswa yang sedang menjalani studi pemrograman dasar.

Akhir kata, semoga buku ini dapat menjadi jembatan ilmu bagi pembaca dalam memahami Pemrograman Dasar Berbasis Web C#. Kami berharap buku ini dapat memberikan kontribusi positif bagi perkembangan pemrograman dasar di kalangan mahasiswa dan pembaca pada umumnya.

Wassalamu'alaikum warahmatullahi wabarakatuh.

Banjarmasin, November 2023

Penerbit

PRAKATA

Assalamu'alaikum wa rahmatullahi wa barakatuh.

Pemrograman dasar berbasis web dalam bentuk Kajian Sejarah, Teori dan Metodologi menunjukkan gejala yang terus meningkat melalui berbagai bentuk kegiatan seminar, simposium, workshop, dan forum diskusi lainnya dengan melibatkan unsur-unsur akademisi dan praktisi. Pengembangan Pemrograman Dasar Berbasis Web melalui perkuliahan atau pelatihan Pemrograman Dasar merupakan langkah strategis dalam mengenalkan Pemrograman Dasar Berbasis Web C# sederhana dalam bentuk Kajian Sejarah, Teori dan Metodologi kepada para mahasiswa.

Bahan Pemrograman Dasar Berbasis Web ini merupakan Kajian Sejarah, Teori dan Metodologi teknologi informasi yang lebih banyak mengetengahkan kajian perkembangan pemrograman dasar yang ada saat ini secara elementer.

Mudah-mudahan, meskipun masih serba sederhana, bahan ajar ini bisa membantu mahasiswa untuk memahami beberapa aspek pemrograman dasar secara kontekstual. Penulis yakin bahwa materi dalam bahan kuliah ini masih jauh dari kesempurnaan, sehingga terbuka untuk mendapatkan kritik dan saran untuk perbaikan pada semua sisi penulisannya.

Wassalamu'alaikum warahmatullahi wabarakatuh.

Banjarmasin, November 2023
Penyusun,

Heru Kartika Candra

DAFTAR ISI

KATA PENGANTAR.....	v
PRAKATA	vi
DAFTAR ISI	vii
BAB I WEB SERVER DAN DASAR-DASAR FORMAT TEKS HTML.....	1
1.1. Tujuan Instruksional Umum.....	1
1.2. Tujuan Instruksional Khusus.....	1
1.3. Uraian materi	1
1.3.1. Web Statis dan Dinamis	1
1.3.2. Web Server	2
1.3.2. WAMP.....	2
1.3.3. Struktur Dokumen	3
1.3.4. <i>Style</i> dari Teks	4
1.3.5. Pengaturan format tata letak teks.....	5
1.3.6. Teks Berjalan.....	13
1.4. Rangkuman.....	17
1.5. Latihan.....	18
BAB II DAFTAR URUTAN DAN TABEL	20
2.1. Tujuan Instruksional Umum.....	20
2.2. Tujuan Instruksional Khusus.....	20
2.3. Uraian Materi.....	20
2.3.1. Membuat Daftar.....	20
2.3.2. Tabel.....	26

2.4.	Rangkuman.....	33
2.5.	Latihan.....	34
BAB III PENGATURAN CITRA DAN LINK NAVIGASI		36
3.1.	Tujuan Instruksional Umum.....	36
3.2.	Tujuan Instruksional Khusus	36
3.3.	Uraian Materi.....	36
3.3.1.	Menyisipkan Citra	37
3.3.2.	Tata Letak Citra (Atribut Align).....	38
3.3.3.	Pengaturan Ukuran Citra (Atribut Height dan Width)	39
3.3.4.	Menampilkan Keterangan Citra (Atribut Alt)	40
3.3.5.	Border Citra.....	41
3.3.6.	Hyperlink.....	42
3.4.	Rangkuman.....	47
3.5.	Latihan.....	47
BAB IV TAG INPUT DAN FORM		50
4.1	Tujuan Instruksional Umum.....	50
4.2.	Tujuan Instruksional Khusus	50
4.3.	Atribut-atribut Tag <i>Input</i>	50
4.3.1.	Atribut TEXT	50
4.3.2.	Atribut RADIO	51
4.3.3.	Atribut CHECKBOX.....	52
4.3.4.	Atribut BUTTION dan SUBMIT.....	54
4.3.5.	<i>Tag</i> Select & Option	55
4.3.6.	Teks Area.....	56
4.3.7.	<i>Tag</i> Form	56
4.4.	Rangkuman.....	58
4.5.	Latihan.....	59
BAB V Pembuatan Web Sederhana Menggunakan <i>Tag</i> Dasar HTML		60

5.1.	Tujuan Instruksional Umum	60
5.2.	Tujuan Instruksional Khusus	60
5.3.	Uraian Materi.....	60
5.3.1.	Konsep Dasar Desain Web	60
5.3.2.	Konsep Pewarnaan	61
5.3.3.	Tipografi.....	63
5.3.4.	<i>Layout</i>	64
5.3.5.	Pembuatan desain WEB	64
5.4.	Rangkuman.....	75
5.5.	Latihan.....	76
BAB VI DASAR CSS (CASCADING STYLE SHEET)		77
6.1.	Tujuan Instruksional Umum.....	77
6.2.	Tujuan Instruksional Khusus	77
6.3.	Uraian Materi.....	77
6.3.1.	Tiga teknik menempatkan aturan CSS ke dalam dokumen web.....	79
6.3.2.	Class dan ID Selektor	82
6.3.3.	Padding, Border dan Margin	85
6.3.4.	Nesting.....	89
6.3.5.	Pseudo Class.....	90
6.3.6.	Tipografi.....	93
6.3.7.	BackGround.....	95
6.3.8.	Float dan clear	97
6.4.	Rangkuman.....	99
6.5.	Latihan.....	99
BAB VII PENGATURAN HALAMAN WEB DENGAN DASAR CSS (BAGIAN 1)		101
7.1.	Tujuan Instruksional Umum.....	101
7.2.	Tujuan Instruksional Khusus	101

7.3.	Uraian Materi.....	101
7.3.1.	Pembuatan <i>Layout</i> kasus1	101
7.3.2.	Pengaturan pada Block Content dan Class Content kasus 1	103
7.3.3.	Drop Down Menu.....	111
7.4.	Rangkuman.....	113
7.5.	Latihan.....	114
BAB VIII PENGATURAN HALAMAN WEB DENGAN DASAR CSS (BAGIAN 2)		115
8.1.	Tujuan Instruksional Umum.....	115
8.2.	Tujuan Instruksional Khusus	115
8.3.	Uraian Materi.....	115
8.3.1.	Kasus 1	115
8.3.2.	Kasus 2	121
8.4.	Rangkuman.....	138
8.5.	Latihan.....	139
DAFTAR PUSTAKA.....		141

BAB I

WEB SERVER DAN DASAR-DASAR FORMAT TEKS HTML

1.1. Tujuan Instruksional Umum

1. Memahami fungsi Web Server.
2. Memahami struktur dokumen pada halaman web.

1.2. Tujuan Instruksional Khusus

1. Menggunakan web server Apache dalam WAMP atau XAMPP untuk menjalankan dasar-dasar format teks HTML.
2. Memahami beberapa istilah penting dalam desain web seperti elemen, atribut, *markup* dan *tag*.

1.3. Uraian materi

1.3.1. Web Statis dan Dinamis

Halaman web dikatakan dinamis apabila dapat berinteraksi dengan *user*, dan melayani permintaan *user* yang sedang menggunakan layanan halaman web tersebut, biasanya berkaitan dengan basis data yang ditampilkan pada halaman web. Tampilan-tampilan halaman web dapat berubah-ubah, meskipun *user* membuka halaman web pada halaman yang sama (tidak ganti halaman). Informasi yang ditampilkan pada halaman web dapat berubah secara langsung, seperti penunjukan waktu, tampilan data yang di-*refresh* setiap beberapa detik sekali, dan sebagainya.

Pemrograman web dinamis menggunakan teknik pemrograman *Server-Side*, di mana program dijalankan pada server sedangkan hasilnya dikirimkan ke *user*, ini berbeda dengan pemrograman statis di mana menggunakan teknik pemrograman *Client-Site* yang berarti *script* program dikirimkan ke *user* dan kemudian hasilnya ditampilkan pada browser *user*, setelah *script* terkirim. Pemrograman web statis hanya diperlukan *tag-tag* HTML dan tidak diperlukan pemrograman, sehingga pada pemrograman statis belum dapatlah dikatakan pemrograman sebenarnya, karena hanya menyusun teks dan *<tag>* saja, tanpa ada proses aritmetika, penentuan keputusan, perulangan dan sebagainya. Dikarenakan pemrograman web statis adalah dasar dari pemrograman web dinamis, maka akan diberikan beberapa *<tag>* yang nantinya diperlukan untuk pemrograman web statis dan beberapa *<tag>* tambahan.

1.3.2 Web Server

Apa yang sebenarnya terjadi saat kita mengetikkan sebuah URL, misalnya `www.hasnur.ac.id`? Yang terjadi adalah komputer kita meminta kepada server hasnur untuk dikirimkan data dalam format HTML ke komputer kita. Setelah itu komputer kita akan menampilkan informasi dari HTML tersebut. Jika server kita adalah statis, maka format HTML yang dihasilkan murni format HTML. Jika server kita menggunakan data dinamis, maka yang mengirim data HTML tersebut dihasilkan dari PHP, ASP atau yang sejenis. Jika server itu berada di komputer kita sendiri, maka untuk memanggil halaman web tersebut digunakan perintah *localhost*. Namun jika server tersebut berada jauh di luar area LAN, maka kita sebut internet. Sebab untuk mengaksesnya, data harus melewati lebih dari satu buah *gateway*. Dalam proses pembuatan web, biasanya kita menggunakan *localhost* untuk menyimulasikan server. Program untuk simulasi itu bisa menggunakan IIS dari Microsoft, PHP Triat, XAMPP, SAMPP dan sebagainya.

1.3.2. WAMP

WAMPP adalah salah satu paket server yang terdiri dari APACHE, PHP, MYSQL dan MyAdmin. WAMPP secara otomatis akan menginstall paket PHP, MySQL ke dalam komputer kita dan menjadikannya sebagai server. Sehingga kita dapat melakukan simulasi server pada komputer kita.

File HTML yang dibuat harus diletakkan dalam folder `C:\WAMP\WWW\`, kemudian jalankan pada browser dengan cara memanggil `localhost\namafile.HTML`.



1.3.3. Struktur Dokumen

Format halaman HTML adalah terdiri dari dua bagian yaitu bagian Header dan bagian Body, yang semuanya itu dibuka dan ditutup dengan *<tag>*. Seperti *tag <HTML>* adalah digunakan untuk tanda awal dari suatu halaman HTML, sedangkan *tag </HTML>* (diawali tanda slash) digunakan untuk tanda akhir dari suatu halaman HTML. *Tag-tag* tersebut dapat ditulis dengan huruf kecil maupun huruf besar, karena HTML tidak mengenal *case sensitive* (membedakan huruf kecil dan besar).

```
<HTML>
<head>
<title> Judul ditampilkan pada baris teratas
browser
</title>
</head>
<body>
. . . ISI dari HTML . . .
</body>
</HTML>
```

Penggunaan multipel spasi di antara kata atau kalimat yang dituliskan pada format halaman HTML tanpa menggunakan karakter entitas, hanya akan dianggap sebagai satu spasi jika dijalankan pada browser. Berikut contohnya :

➤ Spasi didokumen WEB

```
<HTML>
<head>
  <title>White Spaces</title>
</head>
<body>
  This paragraph shows how multiple spaces
  between words are      treated as a single
  space. This is known as white space collapsing.
  It also demonstrates how the browser will treat
  multiple carriage returns
  (new lines) as a single space, too.
</body>
</HTML>
```

Hasilnya :



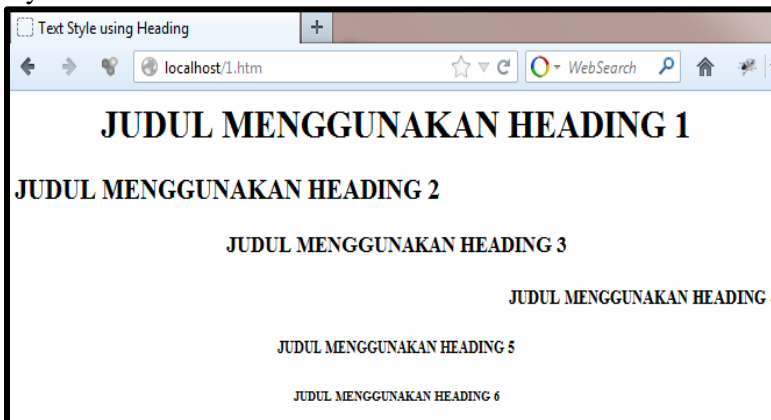
1.3.4. *Style* dari Teks

Ada dua cara untuk pengaturan style teks, yaitu sebagai berikut.

- Menggunakan Heading dengan *tag* <H>
 - Heading menyediakan 6 style teks yang dapat diatur menggunakan angka dari 1 sampai 6, di mana angka 1 adalah ukuran yang paling besar, biasanya digunakan untuk judul atau sub judul, yang dicetak besar dan **tebal (bold)**
 - Heading mempunyai parameter untuk mengatur *justify* atau perataan yaitu LEFT, CENTER dan RIGHT.

```
<HTML>
<HEAD><TITLE>Text                               Style                               using
Heading</TITLE></HEAD>
<BODY>
<H1 ALIGN=CENTER>JUDUL MENGGUNAKAN HEADING 1</H1>
<H2 ALIGN=LEFT>JUDUL MENGGUNAKAN HEADING 2</H2>
<H3 ALIGN=CENTER>JUDUL MENGGUNAKAN HEADING 3</H3>
<H4 ALIGN=RIGHT>JUDUL MENGGUNAKAN HEADING 4</H4>
<H5 ALIGN=CENTER>JUDUL MENGGUNAKAN HEADING 5</H5>
<H6 ALIGN=CENTER>JUDUL MENGGUNAKAN HEADING 6</H6>
</BODY>
</HTML>
```

Hasilnya :



- Menggunakan *font* dengan *tag* .
 - digunakan *tag* di mana tersedia 7 style teks yang dapat dipilih menggunakan tingkatan dari 1 sampai 7, yang merupakan kebalikan dari heading, di mana 1 adalah ukuran yang terkecil sedangkan 7 adalah ukuran terbesar, dan **tidak tebal**.

```
<HTML>
<HEAD>
<TITLE>Font Size & Face</TITLE>
</HEAD>
<BODY>
<FONT SIZE=6>
TEKS INI MENGGUNAKAN UKURAN 6<BR>
DENGAN FONT FACE "DEFAULT" BROWSER<P>
<FONT SIZE=5>
<FONT FACE="ARIAL">
TEKS INI MENGGUNAKAN UKURAN 5<BR>
DENGAN FONT FACE "ARIAL"<P>
</FONT>
</BODY>
</HTML>
```

Hasilnya :



1.3.5. Pengaturan format tata letak teks

Pengaturan format tata letak di sini adalah mengatur tampilan teks agar terlihat rapi dan sesuai dengan keinginan. Adapun *tag* yang digunakan di sini adalah meliputi :

- *tag* paragraph <p>, Meliputi pengaturan paragraph,

```
<HTML>
<head>
<title>Membuat Paragrap</title>
```



```
</head>
<body>
  <p>Ini paragraf pertama</p>
  <p>Ini paragraf    kedua</p>
  <p>Ini paragraf
ketiga</p>
</body>
</HTML>
```

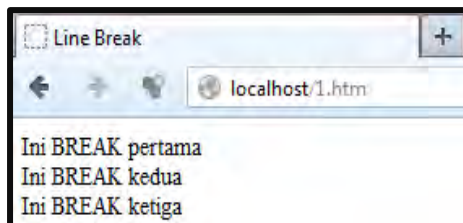
Hasilnya :



- tag ganti baris **
**,
Ganti baris baru,

```
<HTML>
  <head>
    <title>Line Break</title>
  </head>
  <body>
    Ini BREAK pertama<br >
    Ini BREAK kedua<br />
    Ini BREAK ketiga<br />
  </body>
</HTML>
```

Hasilnya :



- tag preformatted **<pre>**,

Terkadang diinginkan untuk menampilkan teks seperti apa adanya, misalkan suatu teks program, di mana susunan dari tiap kalimat harus sama persis dengan aslinya. Untuk keperluan tersebut maka dapat digunakan *tag* `<pre>` yang berfungsi untuk menampilkan teks apa adanya seperti aslinya.

Karena karakter “<” dan “>” merupakan karakter khusus yang digunakan untuk membuat *tag*, maka untuk menampilkan karakter tersebut apa adanya pada halaman browser maka digunakan entitas karakter, yang terdiri dari tiga karakter yaitu & (*ampersand*), # (*crash*) dan ; (*semicolon*). Entitas ini mempunyai sifat *case-sensitive*, di mana memperhatikan penulisan antara huruf kecil dan besar.

Berikut diberikan *table* karakter entitas di mana sering ditemukan pada halaman web dan bahkan sering digunakan untuk keperluan tertentu.

Tabel Daftar Karakter Entitas

Kode	Nama	Tampilan	Arti
&#60;	&lt;	<	Lebih kecil
&#61;	&gt;	>	Lebih besar
&#38;	&amp;	&	Ampersand
&#34;	&quot;	“	Petik ganda
&#174;	&reg;	®	Terdaftar
&#8482;	-	™	Merek Dagang
&#169;	&copy;	©	Hak cipta
&#247	&divide;	÷	Pembagian
&#215	&times;	×	Perkalian

```

<HTML>
  <head>
    <title>Preformatted Teks</title>
  </head>
  <body>
    <p>Teks berikut ditulis dalam elemet &lt;pre>; </p>
    <pre>
      function testFunction(strText){
        alert
(strText)
      }
    </pre>

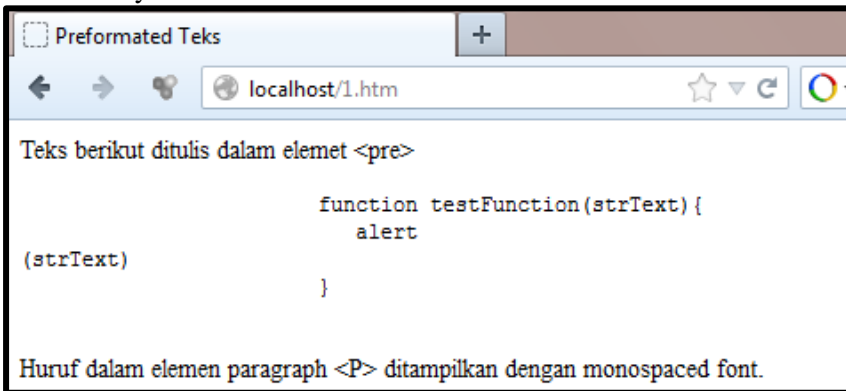
```

```

<p>Huruf dalam elemen paragraph &lt;P&gt; ditampilkan
dengan monospaced font.</p>
</body>
</HTML>

```

Hasilnya:



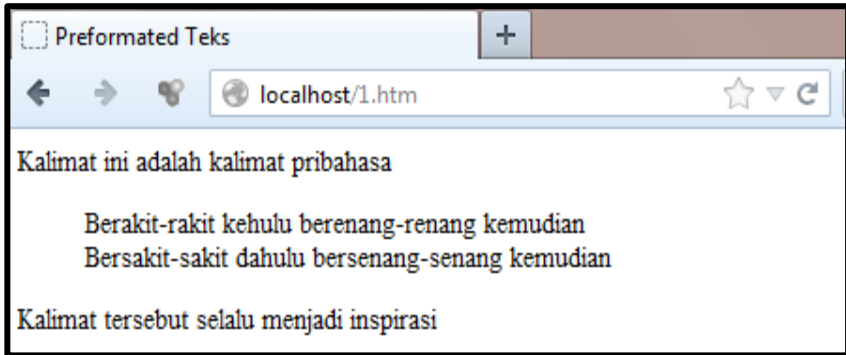
- tag indentasi **<blockquote>**,
Tag `<blockquote>` digunakan untuk pengaturan indentasi teks, sehingga teks ditampilkan masuk beberapa kolom

```

<HTML>
  <head>
    <title>Preformatted Teks</title>
  </head>
  <body>
    Kalimat ini adalah kalimat pribahasa
    <blockquote>
      Berakit-rakit    kehulu    berenang-renang
    kemudian <br>
      Bersakit-sakit   dahulu    bersenang-senang
    kemudian
    </blockquote>
    Kalimat tersebut selalu menjadi inspirasi
  </body>
</HTML>

```

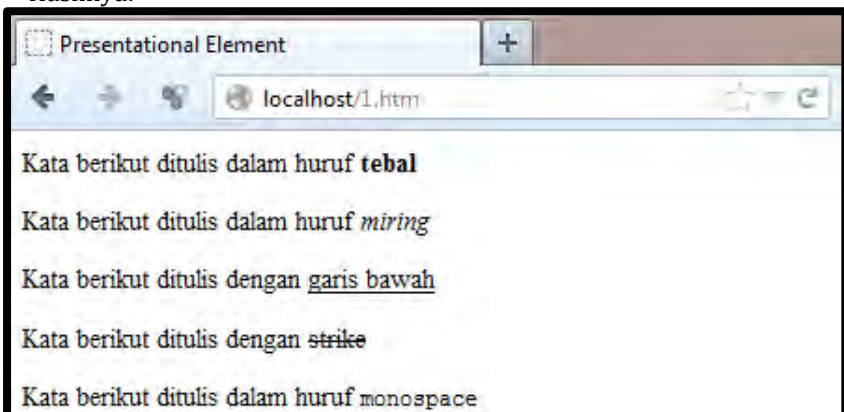
Hasilnya :



- Tag untuk Cetak Miring, Tebal, dan garis bawah, Coret dan teletype

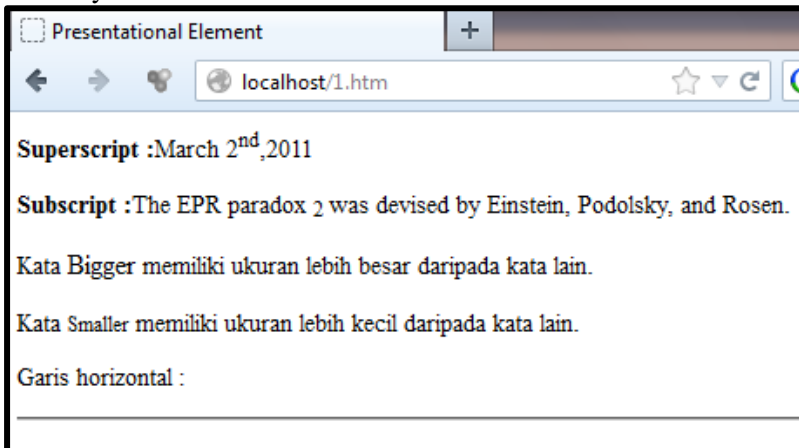
```
<HTML>
  <head>
    <title>Presentational Element</title>
  </head>
  <body>
    <p>Kata berikut ditulis dalam huruf <b>tebal</b></p>
    <p>Kata berikut ditulis dalam huruf <i>miring</i></p>
    <p>Kata berikut ditulis dengan <u>garis bawah</u></p>
    <p>Kata berikut ditulis dengan <s>strike</s></p>
    <tt>monospace</tt></p>
  </body>
</HTML>
```

Hasilnya:



- tag untuk *Superscript*, *subscript*, *small*, *bigger* dan garis horizontal
<HTML>
 <head>
 <title>Presentational Element</title>
 </head>
 <body>
 <p>Superscript :March 2nd,2011</p>
 <p>Subscript :The EPR paradox ₂ was
 devised by Einstein, Podolsky, and Rosen.</p>
 <p>Kata <big>Bigger</big> memiliki ukuran lebih besar
 daripada kata lain.</p>
 <p>Kata <small>Smaller</small> memiliki ukuran lebih
 kecil daripada kata lain.</p>
 <p>Garis horizontal : <hr /></p>
 </body>
</HTML>

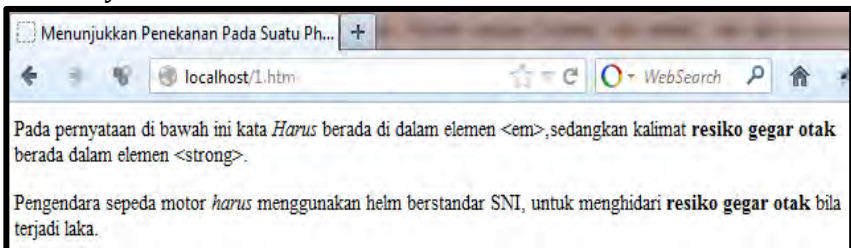
Hasilnya:



- Elemen dan
<HTML>
 <head>
 <title>Menunjukkan Penekanan Pada Suatu Phrase dengan
 Element dan </title>
 </head>
 <body>

```
<p>Pada pernyataan di bawah ini kata <em>Harus</em>
berada di dalam elemen &lt;em&gt;;sedangkan kalimat <strong>resiko
gegar otak</strong> berada dalam elemen &lt;strong&gt;.</p>
<p>Pengendara sepeda motor <em>harus</em> menggunakan helm
berstandar SNI, untuk menghindari <strong>resiko gegar otak</strong>
bila terjadi laka.</p>
</body>
</HTML>
```

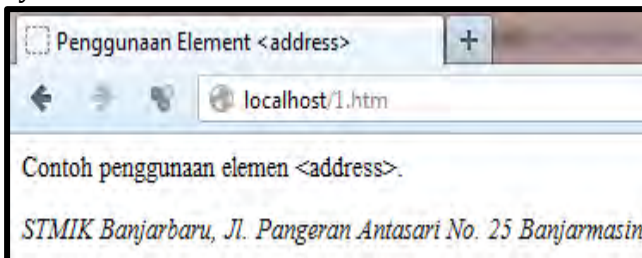
Hasilnya:



- Elemen <address>

```
<HTML>
<head>
  <title>Penggunaan Element &lt;address&gt;</title>
</head>
<body>
  <p>Contoh penggunaan elemen &lt;address&gt;.</p>
  <p><address>STMIK Banjarbaru, Jl. Pangeran Antasari No. 25
Banjarmasin</address></p>
</body>
</HTML>
```

Hasilnya:



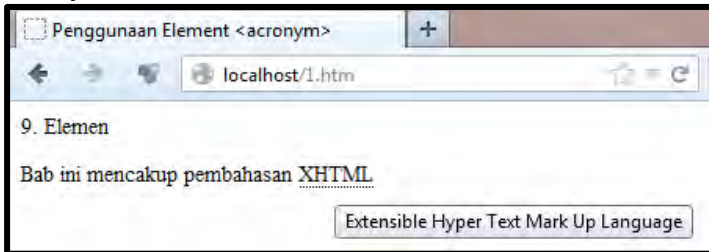
- Elemen `<acronym>`
Saat cursor diarahkan ke kata XHTML, maka akan muncul kepanjangan dari XHTML

```

<HTML>
  <head>
    <title>Penggunaan Element &lt;acronym&gt;</title>
  </head>
  <body>
    <p>Bab ini mencakup pembahasan <acronym
title="Extensible Hyper Text Mark Up
Language">XHTML</acronym></p>
  </body>
</HTML>

```

Hasilnya:



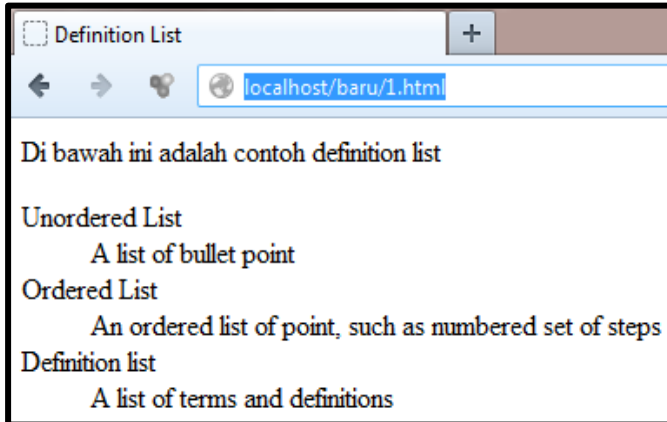
- *Definition List*

```

<HTML>
  <head>
    <title>Definition List</title>
  </head>
  <body>
    <p>Di bawah ini adalah contoh definition list</p>
    <dl>
      <dt>Unordered List</dt>
      <dd>A list of bullet point</dd>
      <dt>Ordered List</dt>
      <dd>An ordered list of point, such as numbered set
of steps</dd>
      <dt>Definition list</dt>
      <dd>A list of terms and definitions</dd>
    </dl>
  </body>
</HTML>

```

Hasilnya:



1.3.6. Teks Berjalan

Pengaturan Teks berjalan menggunakan tag `<marquee>` diakhiri dengan `</marquee>`.

```
<Marquee></Marquee>
```

Jadi secara sederhana, untuk membuat teks bergerak menggunakan Marquee, cukup dengan kode berikut ini:

```
<Marquee>Marque default (standar), teks bergerak dari kanan ke kiri</Marquee>
```

a. Arah Gerakan

Untuk mengatur arah gerakan pada Marquee, tinggal menambahkan kode `direction=""`.

Ada 4 arah gerakan pada Marquee, yaitu:

- `direction="left"` (untuk gerakan dari kiri ke kanan)
- `direction="right"` (untuk gerakan dari kanan ke kiri)
- `direction="up"` (untuk gerakan dari bawah ke atas)
- `direction="down"` (untuk gerakan dari atas ke bawah)

b. Perilaku Gearakan

Untuk mengatur perilaku gerakan pada Marquee, tinggal menambahkan kode **behavior=""**.

Ada 3 perilaku gerakan Marquee, yaitu:

- `behavior="scroll"`
- `behavior="alternate"`
- `behavior="slide"`

Berikut penjelasan dari masing-masing perilaku gerakan tersebut di atas: Untuk **behavior="scroll"**, teks akan bergerak dari ujung yang satu ke ujung yang lainnya, seperti biasanya.

Kode Marquee untuk perilaku tersebut adalah:

```
<Marquee behavior="scroll">Teks bergerak dari kanan ke kiri</Marquee>
```

Untuk **behavior="alternate"**, teks akan bergerak dari ujung yang satu kemudian kembali lagi (bergerak bolak balik), begitu seterusnya. Kode Marquee untuk perilaku tersebut adalah:

```
<Marquee behavior="alternate">Teks bergerak bolak balik</Marquee>
```

Untuk **behavior="slide"**, teks akan bergerak dari ujung yang satu kemudian berhenti. Kode Marquee untuk perilaku tersebut adalah:

```
<Marquee behavior="slide">Teks bergerak dari ujung yang satu kemudian berhenti</Marquee>
```

c. Kecepatan Gerakan

Kecepatan gerakan teks pada Marquee dinyatakan dalam pixel. Angka-angka yang didefinisikan atau ditetapkan dalam kode ini, jelas akan mempengaruhi kecepatan gerakan teks. Semakin besar angkanya, semakin cepat pula gerakan teks. Begitu pun sebaliknya.

Untuk mengatur kecepatan gerak pada Marquee, tinggal menambahkan kode **scrollamount=""**. Untuk membandingkan kecepatan gerak, maka saya akan membuat kode Marquee untuk angka 2 dan 6.

Kode Marquee untuk angka 2 (scrollamount="2") adalah:

```
<Marquee scrollamount="2">Kecepatan gerak teks dengan  
scrollamount="2"</Marquee>
```

Kode Marquee untuk angka 6 (scrollamount="6") adalah:

```
<Marquee scrollamount="6">Kecepatan gerak teks dengan  
scrollamount="6"</Marquee>
```

d. Jeda Antara Setiap Gerakan

Untuk mengatur jeda antara setiap gerakan pada Marquee, tinggal menambahkan kode **scrolldelay=""**. Semakin tinggi nilainya, maka gerakan teks akan semakin lambat. Nilai terendah adalah 60. Apabila memilih nilai di bawah nilai 60, maka nilai tersebut akan diabaikan dan didefinisikan sebagai nilai 60, kecuali jika menambahkan kode **truestpeed=""** dan di atur pada posisi **"true"** (**truestpeed="true"**).

Berikut kodenya:

```
<Marquee scrolldelay="60">scrolldelay="60"</Marquee>
```

```
<Marquee scrolldelay="100">scrolldelay="100"</Marquee>
```

```
<Marquee scrolldelay="40">scrolldelay="40", diabaikan dan  
didefinisikan dengan nilai 60</Marquee>
```

```
<Marquee scrolldelay="40" truestpeed="true">scrolldelay="40", dengan  
truestpeed="true"</Marquee>
```

e. Tinggi Gerakan

Tinggi gerakan *default* (standar) adalah tinggi teks yang digunakan. Biasanya pengaturan tinggi gerakan pada Marquee ini digunakan, untuk gerakan teks secara vertikal. Untuk mengatur tinggi gerakan pada Marquee, tinggal menambahkan kode **height=""**. Untuk mengatur tinggi gerakan pada Marquee, dapat dilakukan dengan 2 cara, yaitu:

- Dengan Pixel (px)
- Dengan Prosentase (%)

Dengan Pixel (px)

Misalnya ingin menyesuaikan tinggi gerakan dengan tinggi halaman blog, di mana tinggi halaman blog maka kode marquee untuk tersebut adalah:

```
<Marquee direction="up" height="50px">
Teks bergerak setinggi 50px</Marquee>
```

Dengan prosentase (%)

Misalnya hanya ingin tulisan bergerak selebar 50% dari tinggi halaman blog tempat meletakkan tulisan, maka kode Marquee untuk tinggi tersebut adalah:

```
<Marquee direction="up" height="75%">Teks bergerak setinggi
75%</Marquee>
```

f. Warna Latar Belakang

Warna latar belakang *default* (standar) dari teks adalah warna latar belakang di mana teks diletakkan. Untuk mengatur atau mengubah warna latar belakang pada Marquee, tinggal menambahkan kode **bgcolor=""**. Misalnya ingin mengubah warna latar belakang teks dengan warna merah, maka kode Marquee untuk warna tersebut adalah:

```
<Marquee bgcolor="red">Latar belakang teks berwarna
```

```
red</Marquee>
```

g. Tulisan Berjalan

```
<HTML>
<head>
  <title>Tulisan Berjalan</title>
</head>
<body>
  <p><marquee>Tulisan berjalan tanpa atribut sama
sekali</marquee></p>
</body>
</HTML>
```

h. Tulisan berjalan dengan atribut

```
<HTML>
<head>
  <title>Tulisan Berjalan Dengan Atribut</title>
</head>
<body>
  <p><marquee behavior="scroll" width="500" height="30"
bgcolor="#9999FF">
Tulisan berjalan dengan atribut</marquee></p>
<marquee behavior="scroll" width="500" height="30" bgcolor="#9999FF"
scrollamount="20" scrolldelay="50" truespeed>
Tulisan berjalan dengan atribut</marquee></p>
</body>
</HTML>
```

1.4. Rangkuman

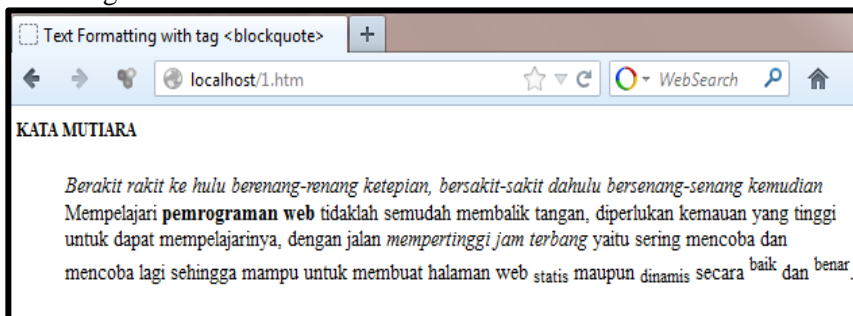
1. Pemrograman web dinamis menggunakan teknik pemrograman *Server-Side*, di mana program dijalankan pada server sedangkan hasilnya dikirimkan ke *user*, ini berbeda dengan pemrograman statis di mana menggunakan teknik pemrograman *Client-Site* yang berarti *script* program dikirimkan ke *user* dan kemudian hasilnya ditampilkan pada browser *user*, setelah *script* terkirim.
2. Beberapa aplikasi WEB SERVER yang sudah terdapat APACHE yang dapat digunakan seperti WAMP, XAMPP. *File* HTML yang

sudah dibuat, diletakkan pada folder c:\wamp\www (untuk WAMP) dan c:\xampp\htdocs\

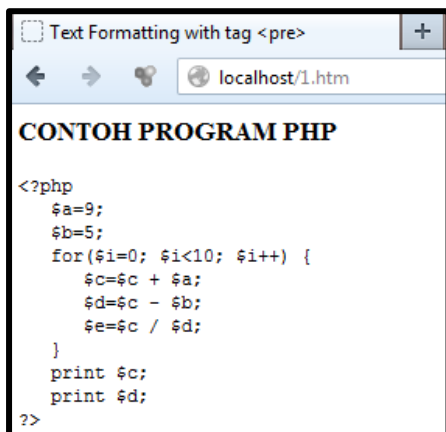
3. Format halaman HTML adalah terdiri dari dua bagian yaitu bagian Header dan bagian Body, yang semuanya itu dibuka dan ditutup dengan `<tag>`. Seperti `tag <HTML>` adalah digunakan untuk tanda awal dari suatu halaman HTML, sedangkan `tag </HTML>` (diawali tanda slash) digunakan untuk tanda akhir dari suatu halaman HTML.
4. Struktur dokumen dapat terdiri dari pengaturan *style* dari *text* dan pengaturan tata letak dari *text*. Seperti `tag paragraph <p>` (Meliputi pengaturan paragraph), `tag ganti baris
` (untuk Ganti baris baru), `tag <pre>` yang berfungsi untuk menampilkan teks apa adanya seperti aslinya, `Tag <blockquote>` digunakan untuk pengaturan indentasi teks, sehingga teks ditampilkan masuk beberapa kolom, `Tag untuk Cetak Miring<i>`, `Tebal`, `garis bawah<U>`, `Coret<s>` dan `teletype<tt>`, Elemen `` dan `` digunakan untuk Penekanan Pada Suatu *Phrase*, `tag <hr>` digunakan untuk membuat garis horizontal, `tag <address>` digunakan untuk penulisan alamat, dan `tag <acronym>` digunakan untuk menunjukkan kepanjangan dari suatu singkata kata.

1.5. Latihan

1. Apa yang dimaksud dengan web statis dan web dinamis?
2. Apa fungsi dari web server?
3. Bagaimana format penulisan halaman HTML ? berikan contoh untuk menampilkan kalimat “selamat datang di web kami”?
4. Buatlah `tag` halaman HTML sehingga pada browser akan tampil sebagai berikut:



5. Buatlah `tag` halaman HTML sehingga pada browser akan tampil sebagai berikut:

A screenshot of a web browser window. The title bar reads "Text Formatting with tag <pre>". The address bar shows "localhost/1.htm". The main content area displays the heading "CONTOH PROGRAM PHP" in bold. Below the heading is a block of PHP code formatted with a monospaced font. The code includes a PHP opening tag, variable assignments for \$a and \$b, a for loop that calculates \$c, \$d, and \$e, and print statements for \$c and \$d. The code ends with a PHP closing tag.

```
<?php
$a=9;
$b=5;
for($i=0; $i<10; $i++) {
    $c=$c + $a;
    $d=$c - $b;
    $e=$c / $d;
}
print $c;
print $d;
?>
```

BAB II

DAFTAR URUTAN DAN TABEL

2.1. Tujuan Instruksional Umum

1. Memahami dan menggunakan fungsi Daftar Urutan.
2. Memahami dan menggunakan Struktur tabel.

2.2. Tujuan Instruksional Khusus

1. Memahami dan menggunakan *tag* daftar urutan tanpa nomor dan dengan nomor, serta *tag* daftar bersarang.
2. Memahami dan menggunakan *tag* tabel mengenai baris, kolom, sistem *merge* dan tabel bersarang.

2.3. Uraian Materi

2.3.1. Membuat Daftar

HTML telah menyediakan *tag* untuk pembuatan daftar, yaitu *tag* untuk membuat daftar tanpa nomor (*Unordered List*) dan *tag* untuk membuat daftar dengan nomor (*Ordered List*).

2.3.1.1 Tag Daftar Tanpa Nomor

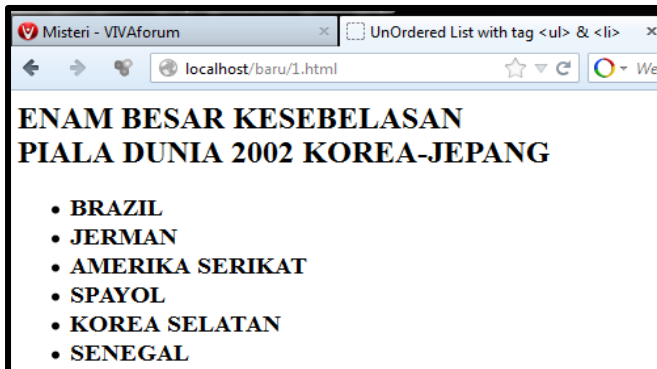
Sesuai dengan namanya maka implementasinya adalah pembuatan daftar tanpa menggunakan nomor baris, contohnya misalkan dibuat suatu daftar negara-negara yang kesebelasannya termasuk 5 besar dalam pertandingan sepak bola piala dunia 2002 di Jepang dan Korea.

Default dari *bullet* adalah titik, sedangkan bila diinginkan untuk mengganti *bullet* dapat ditambahkan atribut TYPE, dengan pilihan “DISC”, “CIRCLE” dan “SQUARE” sehingga *tag* `` menjadi `<UL TYPE=“CIRCLE”>`, di mana akan menampilkan *bullet* berupa lingkaran (○), yang merupakan kebalikan dari *bullet default*-nya.

```
<HTML>
  <HEAD>
    <TITLE>UnOrdered List with tag <ul> & <li></TITLE>
  </HEAD>
  <BODY>
    <H2>ENAM BESAR KESEBELASAN<BR>
    PIALA DUNIA 2002 KOREA-JEPANG
```

```
</H2>
  <UL>
    <H3>
      <LI>BRAZIL</LI>
      <LI>JERMAN</LI>
      <LI>AMERIKA SERIKAT</LI>
      <LI>SPAYOL</LI>
      <LI>KOREA SELATAN</LI>
      <LI>SENEGAL</LI>
    </H3>
  </UL>
</BODY>
</HTML>
```

Hasilnya :



Contoh dengan type disk :

```
<HTML>
  <HEAD>
    <TITLE>UnOrdered List with tag <ul> & <li></TITLE>
  </HEAD>
  <BODY>
    <H2>ENAM BESAR KESEBELASAN<BR>
    PIALA DUNIA 2002 KOREA-JEPANG
    </H2>
    <UL type="disk">
      <H3>
        <LI>BRAZIL</LI>
        <LI>JERMAN</LI>
        <LI>AMERIKA SERIKAT</LI>
        <LI>SPAYOL</LI>
        <LI>KOREA SELATAN</LI>
```

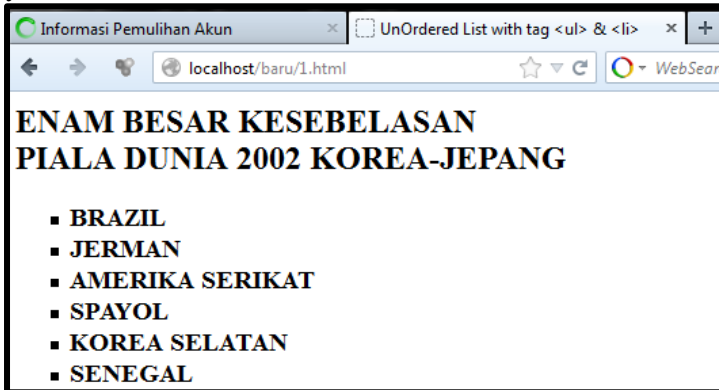


```

        <LI>SENEGAL</LI>
    </H3>
</UL>
</BODY>
</HTML>

```

Hasilnya:



2.3.1.2 Tag Daftar Dengan Nomor

Apabila diinginkan pemberian nomor pada daftar, maka dapat dilakukan dengan mengganti tag `` menjadi tag ``, sedangkan tag lainnya tetap. *Default* dari tag ini adalah angka desimal (`type="1"`), sedangkan bila diinginkan untuk mengganti angka dapat ditambahkan atribut `TYPE`, dengan pilihan "I" (Romawi Besar), "a" (Huruf Kecil) sehingga tag `` menjadi `<OL TYPE="a">`. Untuk Romawi kecil (i) dan untuk Huruf Besar (A). Berikut disajikan bentuk perubahan dari HTML pada bab ini.

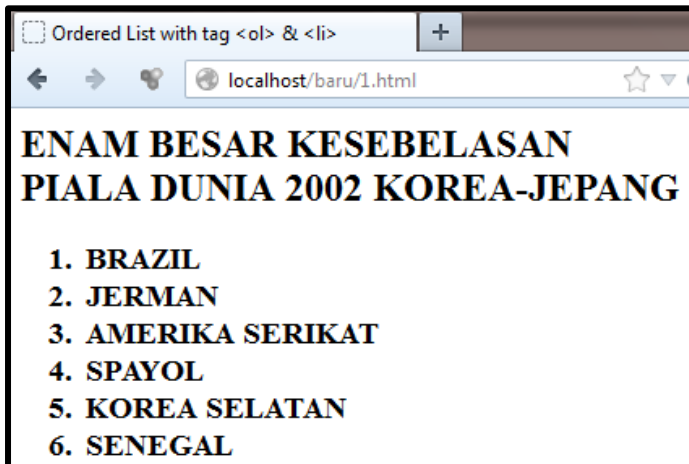
```

<HTML>
  <HEAD>
    <TITLE>Ordered List with tag <ol> & <li></TITLE>
  </HEAD>
  <BODY>
    <H2>ENAM BESAR KESEBELASAN<BR>
    PIALA DUNIA 2002 KOREA-JEPANG
    </H2>
    <OL>
      <H3>
        <LI>BRAZIL</LI>
        <LI>JERMAN</LI>

```

```
        <LI>AMERIKA SERIKAT</LI>
        <LI>SPAYOL</LI>
        <LI>KOREA SELATAN</LI>
        <LI>SENEGAL</LI>
    </H3>
</OL>
</BODY>
</HTML>
```

Hasilnya:



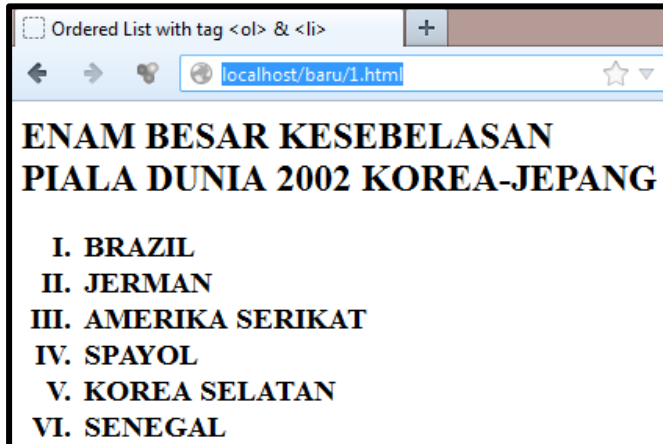
```
<HTML>
  <HEAD>
    <TITLE>Ordered List with tag <ol> &
<li></TITLE>
  </HEAD>
  <BODY>
    <H2>ENAM BESAR KESEBELASAN<BR>
    PIALA DUNIA 2002 KOREA-JEPANG
  </H2>
    <OL TYPE="I">
      <H3>
        <LI>BRAZIL</LI>
        <LI>JERMAN</LI>
        <LI>AMERIKA SERIKAT</LI>
        <LI>SPAYOL</LI>
        <LI>KOREA SELATAN</LI>
```

```

        <LI>SENEGAL</LI>
      </H3>
    </OL>
  </BODY>
</HTML>

```

Hasilnya:



2.3.1.3 Tag Daftar Bersarang (Nested)

Sesuai dengan namanya bersarang yang berarti di dalam daftar ada daftar lagi, ini dapat dilakukan dengan melakukan penulisan *tag* `` di dalam *tag* `` sehingga didapatkan daftar bersarang sebagai berikut:

```

<HTML>
<HEAD>
<TITLE>Ordered List with TYPE="I"
</TITLE>
</HEAD>
<BODY>
<H1>DAFTAR KESEBELASAN<BR>
PIALA DUNIA 2002 KOREA-JEPANG
</H1>
<UL>
  <LI>
    <H2>Grup A</H2>
  </LI>
  <OL>
    <H3>
      <LI>DENMARK</LI>

```

```
<LI>SENEGAL</LI>
<LI>URUGUAY</LI>
<LI>FRANCE</LI>
</H3>
</OL>
</UL>
<UL>
<LI>
<H2>Grup B</H2>
</LI>
<OL>
<H3>
<LI>SPAIN</LI>
<LI>PARAGUAY</LI>
<LI>SOUTH AFRICA</LI>
<LI>SLOVENIA</LI>
</H3>
</OL>
</UL>
</BODY>
</HTML>
```

Hasilnya :



2.3.2. Tabel

Untuk membuat tabel maka diperlukan *tag* <table> dan ditutup dengan *tag* </table>, karena tabel melibatkan banyak pengaturan dan pilihan untuk pembuatan bentuk tabel, maka guna mendukung keperluan pembuatan tabel, telah disediakan atribut-atribut yang khusus disediakan untuk keperluan pembuatan tabel, adapun atribut-atribut tersebut seperti pada tabel di bawah ini:

Tabel Atribut-atribut pada *tag* <table>

Atribut	Nilai	Arti
border	0 s/ 15	Meniadakan atau menampilkan ketebalan garis-garis pada tabel
<tr> </tr>	-	Membuat baris tabel, pengaturan semua teks dapat dilakukan di sini
<th> </th>	-	Membuat kolom judul
<td> </td>	-	Membuat kolom tabel isi
align	LEFT CENTER RIGHT	Mengatur horizontal <i>alignment</i>
valign	TOP MIDDLE BOTTOM	Mengatur <i>vertical alignment</i>
Nowrap	-	Meniadakan pindah baris baru pada saat tabel ditampilkan pada jendela browser yang tidak mencukupi.
rowspan	n	Menggabungkan baris tabel menjadi satu (<i>merge cells</i>)
colspan	n	Menggabungkan kolom tabel menjadi satu (<i>merge cells</i>)

Contoh Tabel:

```
<HTML>
  <HEAD>
    <TITLE>Simple Table 3x4</TITLE>
  </HEAD>
  <BODY>
    <TABLE BORDER=1>
      <TR><TH>NRP</TH><TH>NAMA</TH><TH>NILAI</TH></TR>
      <TR><TD>7401.040.001</TD><TD>ABELDOLOK
      SARIBU</TD><TD>80</TD></TR>
      <TR><TD>7401.040.005</TD><TD>CERAVINA
      SUSANTI K</TD>
```

```

        <TD>76</TD></TR>
        <TR><TD>7401.040.014</TD>
        <TD>M. JARWANTO</TD><TD>90</TD></TR>
    </TABLE>
</BODY>
</HTML>

```

Hasilnya :

NRP	NAMA	NILAI
7401.040.001	ABEL DOLOK SARIBU	80
7401.040.005	CERAVINA SUSANTI K	76
7401.040.014	M. JARWANTO	90

2.2.1.1. Atribut ROWSPAN

Atribut ini digunakan untuk menggabungkan beberapa baris dalam satu kolom menjadi satu baris (merge cell) seperti contoh di bawah ini:

```

<HTML>
<HEAD>
<TITLE>Rowspan on Table 4x4</TITLE>
</HEAD>
<BODY>
    <TABLE BORDER=1>
        <TR><TH>DOSEN</TH><TH>NRP</TH><TH>NAMA</TH><TH>NILAI</TH></TR>
        <TR><TD ROWSPAN=3>Gunawan Rudi C </TD>
            <TD>7401.040.001</TD>
            <TD>ABEL DOLOK SARIBU</TD><TD>80</TD></TR>
        <TR><TD>7401.040.005</TD><TD>CERAVINA SUSANTI K</TD><TD>76</TD></TR>
        <TR><TD>7401.040.014</TD><TD>M. JARWANTO</TD><TD>90</TD></TR>
    </TABLE>
</BODY>
</HTML>

```

Hasilnya:

DOSEN	NRP	NAMA	NILAI
Gunawan Rudi C	7401.040.001	ABEL DOLOK SARIBU	80
	7401.040.005	CERAVINA SUSANTI K	76
	7401.040.014	M. JARWANTO	90

2.2.1.2. Atribut COLSPAN

Atribut ini digunakan untuk menggabungkan beberapa kolom dalam satu baris menjadi satu kolom (*merge cell*), di mana atribut ini juga dapat digabungkan dengan atribut sebelumnya yaitu ROWSPAN seperti contoh di bawah ini:

```
<HTML>
<HEAD>
<TITLE>Colspan on Table 4x4</TITLE>
</HEAD>
<BODY>
<TABLE BORDER=1>
  <TR><TH COLSPAN=4>DAFTAR MAHASISWA SEMESTER GENAP
    2001/ 2002</TH></TR>
  <TR><TH>DOSEN</TH><TH>NRP</TH><TH>NAMA</TH><TH>NILA
    I</TH></TR>
  <TR><TD ROWSPAN=3>Gunawan Rudi
    C</TD><TD>7401.040.001</TD>
    <TD>ABEL DOLOK SARIBU</TD><TD>80</TD></TR>
  <TR><TD>7401.040.005</TD><TD>CERAVINA SUSANTI
    K</TD><TD>76</TD></TR>
  <TR><TD>7401.040.014</TD><TD>M.
    JARWANTO</TD><TD>90</TD></TR>
</TABLE>
</BODY>
</HTML>
```

Hasilnya:

DAFTAR MAHASISWA SEMESTER GENAP 2001/ 2002			
DOSEN	NRP	NAMA	NILAI
Gunawan Rudi C	7401.040.001	ABEL DOLOK SARIBU	80
	7401.040.005	CERAVINA SUSANTI K	76
	7401.040.014	M. JARWANTO	90

2.2.1.3. Atribut Latar Belakang Tabel

Untuk memberikan warna latar belakang di dalam tabel maka dapat digunakan atribut bgcolor disertai dengan latar yang diinginkan, misalkan latar belakang tabel akan diisi dengan warna biru, maka dapat dilakukan **BGCOLOR="BLUE"**, atau latar belakang diisi dengan gambar maka dapat dilakukan dengan **BACKGROUND="namafile.jpg"**, *file* yang

dapat digunakan untuk latar belakang adalah *.JPG, *.GIF, *.BMP dan *.PNG.

```

<HTML>
<HEAD>
<TITLE>BgColor & Background on Table</TITLE>
</HEAD>
<BODY>
<TABLE BORDER=1>
<TR><TH COLSPAN=4 BGCOLOR=LIGHTGREEN>
DAFTAR MAHASISWA SEMESTER
GENAP 2001/ 2002</TH>
<TR><TH>DOSEN</TH><TH>NRP</TH><TH>NAMA</TH><TH>NILAI</
TH></TR>
<TR><TD
ROWSPAN=3
BACKGROUND="keluarag1.jpg">Gunawan Rudi C
</TD>
<TD>7401.040.001</TD><TD>ABEL DOLOK
SARIBU</TD><TD>80</TD></TR>
<TR><TD>7401.040.005</TD><TD>CERAVINA SUSANTI
K</TD><TD>76</TD></TR>
<TR><TD>7401.040.014</TD><TD>M.
JARWANTO</TD><TD>90</TD></TR>
</TABLE>
</BODY>
</HTML>

```

Hasilnya:

DAFTAR MAHASISWA SEMESTER GENAP 2001/ 2002			
DOSEN	NRP	NAMA	NILAI
	7401.040.001	ABEL DOLOK SARIBU	80
	7401.040.005	CERAVINA SUSANTI K	76
	7401.040.014	M. JARWANTO	90

2.2.1.4. Atribut Align

Atribut ini digunakan untuk pengaturan pelurusan dari suatu *cell*, adapun macam pelurusan ini ada tiga yaitu LEFT, CENTER, dan RIGHT. Pelurusan *cell* ini diperlukan terutama untuk informasi data mengenai mata uang, atau yang berhubungan dengan nilai uang misalkan daftar harga. Bila ada daftar harga maka sebaiknya digunakan align RIGHT, sehingga akan memudahkan pembacaan, dari pada menggunakan align

default dari tabel yaitu menggunakan align LEFT. Lebih jelasnya perhatikan *script* dan tampilan hasilnya di bawah ini:

```

<HTML>
<HEAD>
<TITLE>Align on Table</TITLE>
</HEAD>
<BODY>
<TABLE BORDER=1>
  <TR><TH COLSPAN=4>DAFTAR HARGA HANDPHONE NOKIA</TH>
  <TR><TH>TIPE</TH><TH>GARANSI</TH><TH>NON
GARANSI</TH>
  <TH>BEKAS</TH></TR>
  <TR ALIGN=RIGHT><TD>3210</TD><TD>&nbsp;</TD>
  <TD>&nbsp;</TD>
  <TD>700.000</TD></TR>
  <TR
  <TD align=RIGHT><TD>3310</TD><TD>1.350.
000</TD><TD>1.100.000</TD>
  <TD>800.000</TD></TR>
  <TR ALIGN=RIGHT><TD>3330</TD><TD>1.400.000</TD>
  <TD>1.200.000</TD>
  <TD>950.000</TD></TR>
  <TR
  ALIGN=RIGHT><TD>3350</TD><TD>1.750.000</TD><TD>1.500.0
00</TD> <TD>1.100.000</TD></TR>
</TABLE>
</BODY>
</HTML>

```

Hasilnya:

TIPE	GARANSI	NON GARANSI	BEKAS
3210			700.000
3310	1.350.000	1.100.000	800.000
3330	1.400.000	1.200.000	950.000
3350	1.750.000	1.500.000	1.100.000

2.2.1.5. Atribut Border

Untuk menampilkan garis pada tabel maka digunakan atribut border sebagai berikut, **BORDER=1** atau **BORDER=n**, di mana n merupakan ketebalan dari border, yang dapat bernilai dari 0 sampai dengan 1000. Bila tab <table> tanpa disertai atribut border, maka tabel akan ditampilkan tanpa border atau sama dengan bila atribut border diberikan angka nol yaitu <TABLE BORDER=0>. Pada contoh tabel di atas dan di bawah terdapat karakter entitas yaitu ** ** (*non backspace*) yang artinya mengisi *cell* dengan karakter kosong.

```
<HTML>
<HEAD>
<TITLE>Border=14 on Table 4x4</TITLE>
</HEAD>
<BODY>
  <TABLE BORDER=14>
    <TR><TH COLSPAN=4>DAFTAR HARGA HANDPHONE
      NOKIA</TH>
    <TR><TH>TIPE</TH><TH>GARANSI</TH><TH>NON
      GARANSI</TH> <TH>BEKAS</TH></TR>
    <TR ALIGN=RIGHT><TD>3210</TD><TD>&nbsp;</TD>
      <TD>&nbsp;</TD> <TD>700.000</TD></TR>
    <TR ALIGN=RIGHT><TD>3310</TD><TD>1.350.000</TD>
      <TD>1.100.000</TD><TD>800.000</TD></TR>
    <TR ALIGN=RIGHT><TD>3330</TD><TD>1.400.000</TD>
      <TD>1.200.000</TD><TD>950.000</TD></TR>
    <TR ALIGN=RIGHT><TD>3350</TD><TD>1.750.000</TD>
      <TD>1.500.000</TD><TD>1.100.000</TD></TR>
  </TABLE>
</BODY>
</HTML>
```

Hasilnya:

DAFTAR HARGA HANDPHONE NOKIA			
TIPE	GARANSI	NON GARANSI	BEKAS
3210			700.000
3310	1.350.000	1.100.000	800.000
3330	1.400.000	1.200.000	950.000
3350	1.750.000	1.500.000	1.100.000

← BORDER=14

2.2.1.6. Tabel Bersarang (Nested Table)

Kata bersarang (*nested*) sering digunakan pada istilah-istilah pemrograman dan sering dijumpai dalam setiap pemrograman. Demikian pula dalam pembuatan *script* HTML yaitu tabel bersarang, yang artinya adalah di dalam suatu tabel terdapat table yang lain, kemungkinan ini bisa saja terjadi dengan meletakkan *tag* <table> di dalam *tag* <table> lainnya, sehingga didapatkan tabel yang bersarang di dalam tabel lainnya.

```
<HTML>
<HEAD><TITLE>Nested Table</TITLE></HEAD>
<BODY>
<TABLE BORDER=14>
<TR><TH COLSPAN=2>
DAFTAR HARGA HANDPHONE DI JAKARTA MEI 2002</TH></TR>
<TR><TD><TABLE BORDER=1>
  <TR><TH COLSPAN=4>HANDPHONE NOKIA</TH></TR>
  <TR><TH>TIPE</TH><TH>GARANSI</TH><TH>NON
    GARANSI</TH>      <TH>BEKAS</TH></TR>
  <TR ALIGN=RIGHT><TD>3210</TD><TD>&nbsp;</TD>
    <TD>&nbsp;</TD><TD>700.000</TD></TR>
  <TR
    ALIGN=RIGHT><TD>3310</TD><TD>1.350.000</TD>
    <TD>1.100.000</TD><TD>800.000</TD></TR>
  <TR
    ALIGN=RIGHT><TD>3330</TD><TD>1.400.000</TD>
    <TD>1.200.000</TD><TD>950.000</TD></TR>
  <TR
    ALIGN=RIGHT><TD>3350</TD><TD>1.750.000</TD>
    <TD>1.500.000</TD><TD>1.100.000</TD></TR>
  </TABLE>
<TD><TABLE BORDER=1>
  <TR><TH COLSPAN=4>HANDPHONE ERICSON</TH></TR>
  <TR><TH>TIPE</TH><TH>GARANSI</TH>
    <TH>NON GARANSI</TH><TH>BEKAS</TH></TR>
  <TR ALIGN=RIGHT><TD>T20</TD><TD>875.000</TD>
    <TD>650.000</TD><TD>600.000</TD></TR>
  <TR ALIGN=RIGHT><TD>T28s</TD><TD>&nbsp;</TD>
    <TD>&nbsp;</TD><TD>700.000</TD></TR>
  <TR ALIGN=RIGHT><TD>T66</TD><TD>2.800.000</TD>
    <TD>2.550.000</TD><TD>&nbsp;</TD></TR>
</TABLE>
</TD></TR>
</TABLE>
```

```

<TR
  ALIGN=RIGHT><TD>R380</TD><TD>4.000.000</TD>
  >
  <TD>3.000.000</TD><TD>2.500.000</TD></TR>
</TABLE>
</TD></TR>
</TABLE>
</BODY>
</HTML>

```

Hasilnya:

DAFTAR HARGA HANDPHONE DI JAKARTA MEI 2002							
HANDPHONE NOKIA				HANDPHONE ERICSON			
TIPE	GARANSI	NON GARANSI	BEKAS	TIPE	GARANSI	NON GARANSI	BEKAS
3210			700.000	T20	875.000	650.000	600.000
3310	1.350.000	1.100.000	800.000	T28s			700.000
3330	1.400.000	1.200.000	950.000	T66	2.800.000	2.550.000	
3350	1.750.000	1.500.000	1.100.000	R380	4.000.000	3.000.000	2.500.000

2.4. Rangkuman

- HTML telah menyediakan *tag* untuk pembuatan daftar, yaitu *tag* untuk membuat daftar tanpa nomor (*Unordered List*) dan *tag* untuk membuat daftar dengan nomor (*Ordered List*).
- *Default* dari (*Unordered List*) adalah titik, sedangkan bila diinginkan untuk mengganti bullet dapat ditambahkan atribut TYPE, dengan pilihan “DISC”, “CIRCLE” dan “SQUARE” sehingga *tag* menjadi <UL TYPE=“CIRCLE”>, di mana akan menampilkan *bullet* berupa lingkaran (o).
- *Default* dari *tag* (*Ordered List*) adalah angka desimal (type=“1”), sedangkan bila diinginkan untuk mengganti angka dapat ditambahkan atribut TYPE, dengan pilihan “I” (Romawi Besar), “a” (Huruf Kecil) sehingga *tag* menjadi <OL TYPE=“a”>. Untuk Romawi kecil (i) dan untuk Huruf Besar (A).
- Untuk membuat tabel maka diperlukan *tag* <table> dan ditutup dengan *tag* </table>. Di antara <table> dan </table> untuk membuat baris digunakan *tag* <tr> dan </tr>. Dan di antara <tr> dan </tr> untuk membuat kolom digunakan <td> dan </td>.

- Atribut Rowspan `<TR><TD ROWSPAN=3>` digunakan untuk menggabungkan 3 baris dalam satu kolom menjadi satu baris (*merge cell*). Atribut Colspan `<TR><TH COLSPAN=4>` digunakan untuk menggabungkan 4 kolom dalam satu baris menjadi satu kolom (*merge cell*)

2.5. Latihan

1. Perhatikan *screenshot* di bawah ini kemudian tuliskan kode sumber HTML-nya!
- 2.

Group dan Zona Babak Kualifikasi Piala Dunia 2014

I. Group A

- a. Zona Asia dan Australia
 - Indonesia
 - Jepang
 - Korea Selatan
 - Australia
 - Selandia Baru
- b. Zona Eropa
 - Inggris
 - Jerman
 - Perancis
 - Belanda
 - Italia

II. Group B

- a. Zona Amerika
 - Amerika Serikat
 - Meksiko
 - Uruguay
 - Brasil
 - Argentina
 - Kanada
- b. Zona Afrika
 - Afrika Selatan
 - Kamerun
 - Tunisia
 - Mesir
 - Nigeria
 - Sudan

3. Perhatikan *screenshot* di bawah ini kemudian tuliskan kode sumber HTML-nya!

Classic Movie Day				
	5 pm	7 pm	9 pm	11 pm
Screen one	Star Wars	Empire Strikes Back	Return of the Jedi	The Exorcist
Screen two	Dances with Wolves		Gone With the Wind	
Screen three	2001: A Space Odyssey		The Conversation	5 Easy Pieces
Screen Four	Adiboo		One Peace	Good Warriors
Screen Five			Star Trek	Hulk

BAB III

PENGATURAN CITRA DAN LINK NAVIGASI

3.1. Tujuan Instruksional Umum

1. Memahami menampilkan citra pada WEB.
2. Memahami dan menggunakan *link* navigasi.

3.2. Tujuan Instruksional Khusus

1. Memahami cara menyisipkan citra, pengaturan tata letak citra, pengaturan ukuran dan memberikan keterangan pada citra dalam web.
2. Menghubungkan antar halaman web di satu situs web, menghubungkan suatu situs web dengan situs web yang lain dan menghubungkan bagian tertentu dalam halaman situs web yang sama

3.3. Uraian Materi

File citra dengan ekstensi *.BMP sangat tidak baik untuk digunakan untuk menghiasi laman, karena tidak menggunakan teknik kompresi sehingga ukurannya besar.

Berikut ini adalah *file* citra yang mempunyai teknik kompresi tinggi dan sering dijumpai dan digunakan untuk menghiasi laman di Internet, yaitu:

- Citra dengan format kompresi PNG (*Portable Network Graphics*), menggunakan teknik kompresi pengelompokan jumlah pixel, yaitu pixel yang sama disimpan sekali dan diikuti dengan jumlah pixel pada citra aslinya, sehingga format citra ini sesuai untuk citra yang mempunyai banyak pixel sama, seperti pada citra statistik seperti diagram batang, pie chart, scatter diagram dan sebagainya.
- Citra dengan format kompresi JPEG (*Joint Photographic Experts Groups*), menggunakan format warna 24 bit (16 juta/ 16.776.960), menggunakan teknik *progressive*, yaitu menampilkan sebagian citra secara bertahap, sehingga gambar tidak perlu diunduh secara keseluruhan (serupa dengan teknik *interlacing* pada GIF), biasanya digunakan untuk menampilkan citra yang memerlukan kualitas tinggi. Untuk memperkecil citra maka sebaliknya dilakukan teknik **blur** (teknik perataan pixel, sehingga citra tampak buram) dahulu kemudian baru dikecilkan.

- Citra dengan format kompresi GIF (*Graphics Interchange Format*), menggunakan format warna 8 bit (256), teknik penyimpanan *interlacing* sehingga citra dapat dilihat pada jendela browser secara bertahap (tidak perlu menunggu sampai semua *file* terkirim). Di samping itu juga menggunakan teknik transparan sehingga citra dapat digunakan sebagai latar belakang yang terlihat secara samar-samar. Dengan menggabungkan beberapa citra maka didapatkan suatu animasi citra yang bergerak, contohnya menggunakan *software* GIF Animator.

3.3.1. Menyisipkan Citra

Untuk dapat menyisipkan citra maka digunakan *tag* `` diikuti dengan atribut nama *file* citra yang akan ditampilkan pada laman. Untuk itu perhatikan *script* di bawah ini:

```
<HTML>
<HEAD><TITLE>IMAGE Inserting</TITLE></HEAD>
<BODY>
<TABLE>
<TR>
<TD><IMG SRC="POLIBAN.jpg"></TD>
<TD>
<FONT SIZE=4>
POLITEKNIK NEGERI BANJARMASIN<BR>
JURUSAN TEKNIK ELEKTRO<BR>
Program Studi Teknik Informatika
</FONT>
</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Hasilnya :

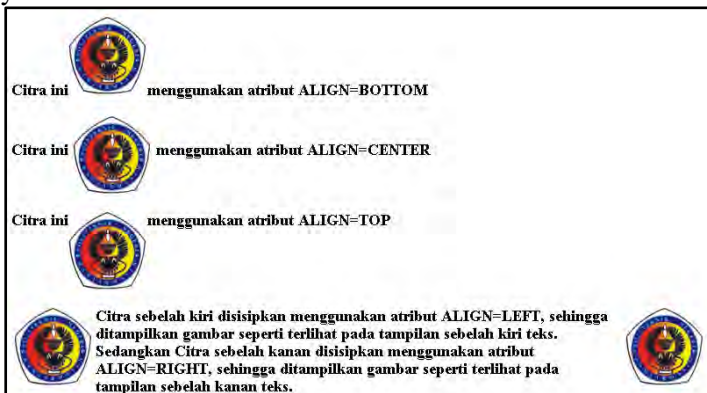


3.3.2. Tata Letak Citra (Atribut Align)

Penempatan citra dapat diatur sedemikian sehingga tampilan citra sesuai dengan teks yang menyertainya, yaitu citra dapat diletakkan di atas (*default*), di tengah, di bawah, kiri atau kanan teks seperti yang ditampilkan pada Gambar berikut.

```
<HTML>
<HEAD><TITLE>IMAGE Alignment</TITLE></HEAD>
<BODY>
<H3>
Citra ini<IMG SRC="POLIBAN.jpg"
ALIGN=BOTTOM>menggunakan atribut ALIGN=BOTTOM<P>
Citra ini <IMG SRC="POLIBAN.jpg " ALIGN=CENTER>
menggunakan atribut ALIGN=CENTER<P>
Citra ini<IMG SRC="POLIBAN.jpg "
ALIGN=TOP>menggunakan atribut ALIGN=TOP<P>
<IMG SRC="POLIBAN.jpg " ALIGN=LEFT >
<IMG SRC="POLIBAN.jpg " ALIGN=RIGHT>
Citra sebelah kiri disisipkan menggunakan atribut
ALIGN=LEFT, sehingga ditampilkan gambar seperti
terlihat pada tampilan sebelah kiri teks. Sedangkan
Citra sebelah kanan disisipkan menggunakan
atribut ALIGN=RIGHT, sehingga ditampilkan gambar
seperti terlihat pada tampilan sebelah kanan teks.
</H3>
</BODY>
</HTML>
```

Hasilnya:



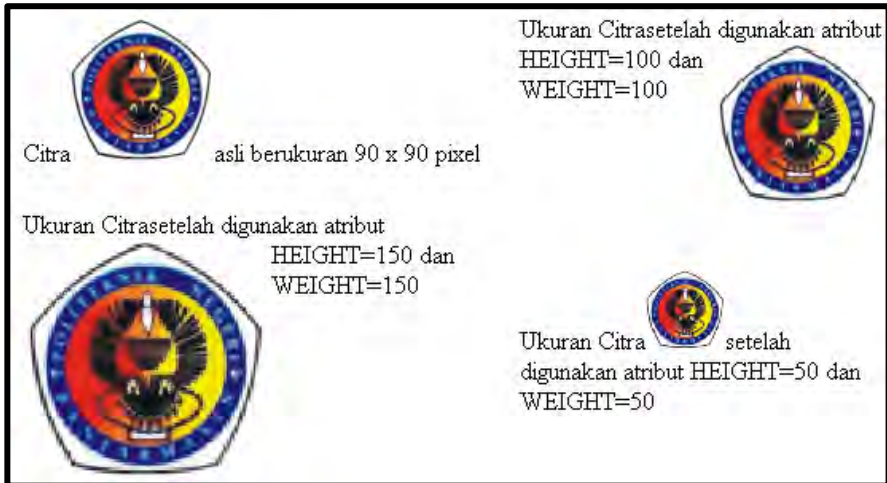
3.3.3. Pengaturan Ukuran Citra (Atribut Height dan Width)

Untuk mengatur lebar dan tinggi ukuran pixel dari citra yang akan ditampilkan pada jendela browser, maka digunakan atribut **HEIGHT** dan **Width**, di mana **HEIGHT** adalah untuk mengatur ukuran lebar citra sedangkan **Width** untuk mengatur ukuran panjang citra. Penggunaan atribut ini tidak disarankan untuk memperkecil ukuran gambar, karena meskipun ukuran tampilan citra pada jendela browser kecil, tampilan tetap saja lambat karena citra asli yang dikirimkan ukurannya sebenarnya lebih besar dari ukuran yang ditampilkan pada jendela browser. Untuk itu disarankan menggunakan ukuran citra yang sebenarnya atau menggunakan atribut ini untuk memperbesar citra. Sedangkan untuk memperkecil citra sebelum disisipkan pada jendela browser sebaliknya diedit menggunakan editor citra (seperti *Photoshop*) untuk memperkecil ukuran citra.

Berikut ini diberikan contoh penggunaan atribut **HEIGHT** dan **Width** pada citra yang berukuran 75x75 pixel, di mana ukurannya diperbesar menjadi 100 x 100 dan 150 x 150 pixel serta diperkecil menjadi 50 x 50.

```
<HTML>
<HEAD>
<TITLE>
IMAGE Height & Width
</TITLE>
</HEAD>
<BODY>
<H3><TABLE>
<TR><TD>Citra<IMG SRC="POLIBAN.jpg">asli berukuran 90
x 90 pixel<P></TD><TD>Ukuran Citra<IMG
SRC="POLIBAN.jpg" HEIGHT=100 Width ="100"
ALIGN="RIGHT">setelah digunakan atribut HEIGHT=100 dan
Width =100</TD>
<TR><TD>Ukuran Citra<IMG SRC="POLIBAN.jpg" HEIGHT=150
Width ="150" ALIGN="LEFT">setelah digunakan atribut
HEIGHT=150 dan Width =150</TD>
<TD>Ukuran Citra<IMG SRC="POLIBAN.jpg" HEIGHT=50 Width
="50">setelah digunakan atribut HEIGHT=50 dan Width
=50</TD>
</TABLE></H3>
</BODY>
</HTML>
```

Hasilnya:

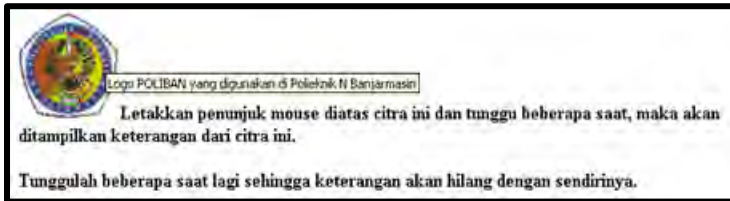


3.3.4. Menampilkan Keterangan Citra (Atribut Alt)

Untuk memperjelas maksud atau untuk memberikan keterangan pada suatu citra, maka dapatlah suatu citra tersebut diberikan suatu penjelasan mengenai maksud atau arti dari citra yang berada pada halaman web tersebut. Sehingga apabila penunjuk mouse berada di atas citra tersebut, maka akan muncul keterangan mengenai citra yang bersangkutan.

```
<HTML>
<HEAD>
<TITLE>Tag Alt on Image</TITLE>
</HEAD>
<BODY>
<H3>
<IMG SRC="POLIBAN.jpg" ALT="Logo POLIBAN yang
digunakan di Polieknik N Banjarmasin">Letakkan
penunjuk mouse di atas citra ini dan tunggu beberapa
saat, maka akan ditampilkan keterangan dari citra
ini.<P>
Tunggulah beberapa saat lagi sehingga keterangan akan
hilang dengan sendirinya.
</H3>
</BODY>
</HTML>
```

Hasilnya:



3.3.5. Border Citra

Untuk memberikan bingkai pada citra yang ditampilkan di web, maka digunakan atribut border. Tebal tipisnya border tergantung nilai angka yang diberikan pada border tersebut.

```
<HTML>
<head>
  <title>Attribute border </title>
</head>
<body>
  <h1>Border = "1"</h1>
  
  <h1>Border = "2"</h1>
  
  <h1>Border = "3"</h1>
  
</body>
</HTML>
```

Hasilnya :

Border = "1"



Border = "2"



Border = "3"



3.3.6. Hyperlink

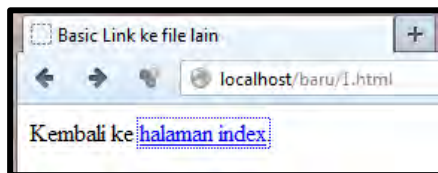
Untuk melakukan *link* ke lain dokumen, Menggunakan *tag* alamat yaitu *tag* `<a href> `

3.3.6.1. Tag Alamat Relatif

Untuk keperluan pembuatan link ini maka paling tidak sudah tersedia minimal dua file HTML, di mana satu sebagai *script* utama dan lainnya sebagai *script* link. Berikut ini Link ke halaman lain dalam satu situs web.

```
<HTML>
  <head>
    <title>Basic Link ke file lain</title>
  </head>
  <body>
    <p>Kembali ke <a href="index.HTML">halaman
index</a>.</p>
  </body>
</HTML>
```

Pada contoh berikut, harus disediakan juga *file* HTML dengan nama `index.HTML`. agar tujuan *link* dapat membuka alamatnya.



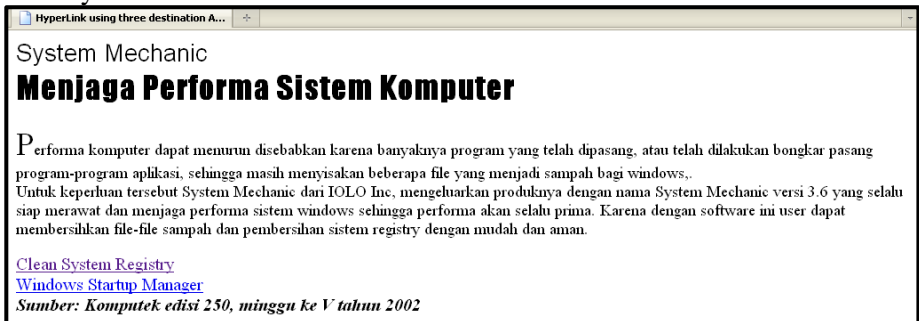
Contoh lainnya sebagai berikut :

```
<HTML>
<HEAD><TITLE>
HyperLink using three destination Address
</TITLE></HEAD>
<BODY>
<FONT SIZE=5 FACE="ARIAL">System Mechanic</FONT><BR>
<FONT SIZE=6 FACE="IMPACT">Menjaga Performa Sistem
Komputer</FONT><BR><B>
><BR><FONT SIZE=6>P</FONT>erforma komputer dapat
menurun disebabkan karena banyaknya program yang telah
dipasang, atau telah dilakukan bongkar pasang program-
```

program aplikasi, sehingga masih menyisakan beberapa file yang menjadi sampah bagi windows,.
Untuk keperluan tersebut System Mechanic dari IOLO Inc, mengeluarkan produknya dengan nama System Mechanic versi 3.6 yang selalu siap merawat dan menjaga performa sistem windows sehingga performa akan selalu prima. Karena dengan software ini user dapat membersihkan file-file sampah dan pembersihan sistem registry dengan mudah dan aman.

```
<P>
<FONT SIZE=4><a href="SYSTEM.HTML">Clean System Registry</a><BR>
<a href="STARTUP.HTML">Windows Startup Manager</a><BR>
<B><I>Sumber: Komputek edisi 250, minggu ke V tahun 2002</I></B>
</BODY>
</HTML>
```

Hasilnya:



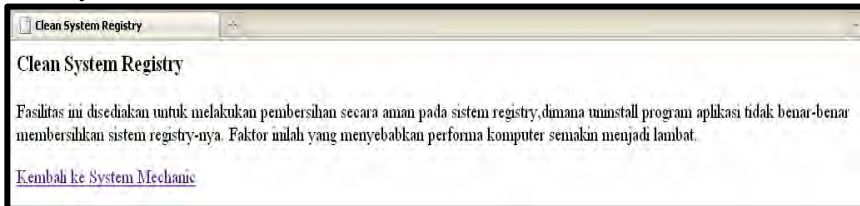
Berikut file HTML guna mendukung HyperLink dari Gambar di atas, file ini disimpan terpisah dengan file HTML utama di atas, dan disimpan dengan nama "SYSTEM.HTML".

```
<HTML>
<HEAD><TITLE>Clean System Registry</TITLE></HEAD>
<BODY>
<H3>Clean System Registry</H3>
```

Fasilitas ini disediakan untuk melakukan pembersihan secara aman pada sistem registry, di mana uninstall program aplikasi tidak benar-benar membersihkan sistem registry-nya. Faktor inilah yang menyebabkan performa komputer semakin menjadi lambat.

```
<P>
<A HREF="LINK1.HTML">Kembali ke System Mechanic</A>
</BODY>
</HTML>
```

Hasilnya:

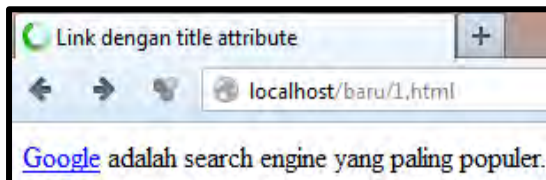


3.3.6.2. Tag Alamat Absolut dan Title

Hampir sama dengan *link* alamat relatif di atas, perbedaannya hanyalah pada alamat yang digunakan, yaitu menggunakan alamat URL (*Uniform Resources Locator*), contohnya www.poliban.ac.id, yang berarti menampilkan *homepage* Politeknik Negeri Banjarmasin yaitu ditulis `Poliban `.

```
<HTML >
<head>
  <title>Link dengan title attribute</title>
</head>
<body>
  <p><a href="http://www.Google.com/" title="Search
the web with Google" >
  Google</a> adalah search engine yang paling
populer.</p>
</body>
</HTML>
```

Hasilnya:



3.3.6.3. Tag Alamat Halaman

Berbeda dengan *link* yang dilakukan di atas, yaitu melompat pada halaman lain selain halamannya sendiri. Berikut ini *link* dilakukan pada halaman yang sama, dengan syarat bahwa bagian yang akan dilompati melebihi dari satu jendela browser.

`Link dan Navigasi</h1>` → posisi alamat halaman

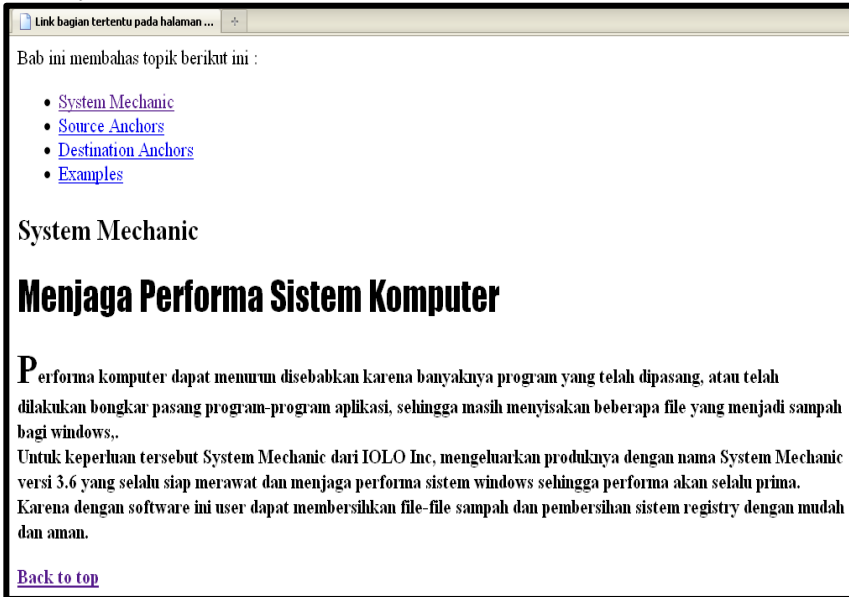
`Back to top</p>` → menuju ke alamat link dari halaman

```
<HTML>
<head>
  <title>Link bagian tertentu pada halaman web</title>
</head>
<body>
<h1><a name="top">Link dan Navigasi</a></h1>
<p>Bab ini membahas topik berikut ini :</p>
  <ul>
    <li><a href="#System Mechanic">System
Mechanic</a></li>
    <li><a href="#SourceAnchors">Source
Anchors</a></li>
    <li><a href="#DestinationAnchors">Destination
Anchors</a></li>
    <li><a href="#Examples">Examples</a></li>
  </ul>
<h2><a name="System Mechanic">System Mechanic</a></h2>
<FONT SIZE=6 FACE="IMPACT">Menjaga Performa Sistem
Komputer</FONT><BR><B>
<BR><FONT SIZE=6>P</FONT>erforma komputer dapat
menurun disebabkan karena banyaknya program yang telah
dipasang, atau telah dilakukan bongkar pasang program-
program aplikasi, sehingga masih menyisakan beberapa
file yang menjadi sampah bagi windows, .<BR>
Untuk keperluan tersebut System Mechanic dari IOLO
Inc, mengeluarkan produknya dengan nama System
Mechanic versi 3.6 yang selalu siap merawat dan
menjaga performa sistem windows sehingga performa akan
selalu prima. Karena dengan software ini user dapat
membersihkan file-file sampah dan pembersihan sistem
registry dengan mudah dan aman. <P>
<p><a href="#top">Back to top</a></p>
```



```
</body>
</HTML>
```

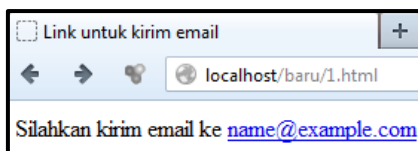
Hasilnya:



3.3.6.4. Link untuk kirim email

```
<HTML>
<head>
  <title>Link untuk kirim email</title>
</head>
<body>
<p>Silahkan kirim email ke <a
href="mailto:name@example.com">name@example.com</a></p>
>
</body>
</HTML>
```

Hasilnya:



3.4. Rangkuman

- Untuk dapat menyisipkan citra maka digunakan *tag* `` diikuti dengan atribut nama *file* citra yang akan ditampilkan pada laman ``, untuk *perataan top (atas), bawah (bottom), kiri (left) dan kanan (right)* dengan menambah atribut `align` ``, sementara Untuk mengatur lebar dan tinggi ukuran pixel dari citra yang akan ditampilkan pada jendela browser, maka digunakan atribut `Width` dan `HEIGHT` ``.
- Untuk melakukan link ke lain dokumen, menggunakan *tag* alamat yaitu *tag* `<a href> `. Hyperlink dapat dibagi menjadi *tag* alamat relative (untuk keperluan pembuatan *link* ini maka paling tidak sudah tersedia minimal dua file HTML, di mana satu sebagai *script* utama dan lainnya sebagai *script link* `halaman index`), *tag* alamat absolute dan *title* (Hampir sama dengan *link* alamat relatif di atas, perbedaannya hanyalah pada alamat yang digunakan, yaitu menggunakan alamat URL (*Uniform Resources Locator*), contohnya www.poliban.ac.id, yang berarti menampilkan homepage Politeknik Negeri Banjarmasin yaitu ditulis `Poliban `), *tag* alamat halaman (yaitu melompat pada halaman lain selain halamannya sendiri. Link ini juga dapat dilakukan pada halaman yang sama, dengan syarat bahwa bagian yang akan dilompati melebihi dari satu jendela browser `Link dan Navigasi</h1>` → **posisi alamat halaman**, `Back to top</p>` → **menuju ke alamat link dari halaman**, **Tag link untuk kirim email** (`name@example.com `, dengan syarat harus mempunyai mail server)

3.5. Latihan

Perhatikan beberapa *screenshot* di bawah ini, kemudian tulis *source code* HTML -nya !

JEJAMUR CAFE

[HOME](#) [MENU](#) [RECIPES](#) [CONTACT](#)

A community cafe serving home cooked, locally sourced, organic food

With stunning views of the ocean, Example Cafe offers the perfect environment to unwind and recharge the batteries.

Our menu offers a wide range of breakfasts, brunches and lunches, including a range of vegetarian options.

Whether you sip on a fresh, hot coffee or a cooling smoothie, you never need to feel rushed - relax with friends or just watch the world go by.

This weekend's special brunch

This weekend, our season of special brunches continues with scrambled egg on an English muffin. Not for the faint hearted, the secret to these eggs is that they are made with half cream and cooked in butter, with no more than four eggs in the pan at a time.

JEJAMUR CAFE

[HOME](#) [MENU](#) [RECIPES](#) [CONTACT](#)

[Starters](#) | [Main Courses](#) | [Desserts](#)

Menu

Starters

- Chestnut and Mushroom Goujons [\(v\)](#)
- Goat Cheese Salad [\(v\)](#)
- Honey Soy Chicken Kebabs
- Seafood Salad

[Back to top](#)

Main courses

- Spinach and Ricotta Roulade [\(v\)](#)
- Beef Tournados with Mustard and Dill Sauce
- Roast Chicken Salad
- Icelandic Cod with Parsley Sauce
- Mushroom Wellington [\(v\)](#)

[Back to top](#)

Desserts

- Lemon Sorbet [\(v\)](#)

JEJAMUR CAFE

[HOME](#) [MENU](#) [RECIPES](#) [CONTACT](#)

Recipes - World's Best Scrambled Eggs

I adapted this recipe from a book called *Jobney Food* by Bill Granger. Ever since making these eggs on my 1st visit to Bill's restaurant in Hazy Crest, Sydney, I have been able to replicate them at home. I really believe these are the best scrambled eggs I have ever tasted.

This recipe is what I call a "very special breakfast", just look at the ingredients to see why. It has to be tasted to be believed.

Ingredients

The following ingredients make one serving.

- 2 eggs
- 1 tablespoon of butter (10g)
- 1/2 cup of cream (125ml of sour cream)
- A pinch of salt
- Freshly milled black pepper
- 4 fresh chives (chopped)

Instructions

1. Whisk eggs, cream, and salt in a bowl.
2. Melt the butter in a non-stick pan over a high heat (taking care not to burn the butter).
3. Pour egg mixture into pan and whisk until it starts setting around the edge of the pan (around 20 seconds).
4. Using a wooden spatula, bring the mixture into the center as if it were an omelet, and let it cook for another 20 seconds.
5. Fold contents in again, leave for 20 seconds, and repeat until the eggs are only just done.

JEJAMUR CAFE

[HOME](#) [MENU](#) [RECIPES](#) [CONTACT](#)

Contact

A. Yani Street Number 4 Banjarmasin

[Find us on Google Maps](#)

[Email Jejamur Cafe](#)

BAB IV

TAG INPUT DAN FORM

4.1 Tujuan Instruksional Umum

1. Memahami dan menggunakan jenis *tag input* dalam memasukkan data melalui WEB .
2. Memahami dan menggabungkan penggunaan *form* untuk *tag input*.

4.2. Tujuan Instruksional Khusus

1. Memahami dan menggunakan *tag input* type text, password, radio, checkbox, textarea, button dan submit.
2. Menggabungkan penggunaan form pada masing-masing *tag input*, serta memahami tentang penggunaan atribut method,action pada form.

4.3. Atribut-atribut Tag Input

Guna menunjang pemrograman dinamis tersebut maka diperlukan *tag input* HTML untuk memasukkan data yang diperlukan sebagai masukkan permintaan dari pengguna (pengunjung) halaman web. Di sini digunakan beberapa atribut dari *tag input* seperti TEXT, RADIO, CHECKBOX, BUTTON dan SUBMIT.

4.3.1. Atribut TEXT

Atribut ini digunakan untuk membuat baris masukan yang digunakan untuk memasukkan data, yang dapat berupa data numeric maupun data karakter dan string maupun data keduanya. Seperti halnya pada pemrograman pascal atau C, untuk menampung data maka diperlukan variabel, demikian juga dengan *input* di sini, diperlukan atribut NAME untuk mendefinisikan variabel yang digunakan untuk menampung data masukan tersebut.

```
<HTML>
<HEAD>
<TITLE>Tag INPUT with Text Attribute</TITLE>
</HEAD>
<BODY>
<H2>ISIAN BUKU TAMU</H2>
```

```
<TABLE BORDER=0>
<TR><TD>Nama</TD>
<TD><INPUT TYPE=text NAME=nama size=25 MAXLENGTH=25>
</TD></TR>
<TR><TD>Email</TD>
<TD><INPUT TYPE=TEXT NAME=EMAIL SIZE=60
MAXLENGTH=60></TD></TR>
</TABLE>
</BODY>
</HTML>
```

Hasilnya:

ISIAN BUKU TAMU	
Nama	<input type="text"/>
Email	<input type="text"/>

4.3.2. Atribut RADIO

Input dengan atribut radio adalah menampilkan beberapa pilihan untuk dipilih satu dari beberapa pilihan tersebut. Seperti pilihan jurusan di HASNUR (Informatika, Mesin, Perkebunan), jenis kelamin (Pria/Wanita), usia (Anak-anak, Remaja, Dewasa), jenis mobil (sedan, pickup, station, combi, kodok, jeep, truck, trailer), jawaban pilihan soal (A, B, C, D, E) dan lain-lain. Berikut ini diberikan contoh *tag input* menggunakan atribut radio, dengan memberikan 4 (empat) pilihan hobi, yaitu Komputer, Membaca, Elektronika, dan Olah-raga. Keempat pilihan tersebut hanya dapat dipilih satu saja dari keempat pilihan, sehingga apabila sudah memilih hobi Komputer maka pilihan lain diabaikan, kemudian bila mengganti pilihan, misalkan dari Komputer menjadi Elektronika maka pilihan yang ada di Komputer akan dihapus dan pilihan berganti pada Elektronika, sehingga pilihan tetap satu saja

```
<HTML>
<HEAD>
<TITLE>Tag INPUT with RADIO Attribute</TITLE>
</HEAD>
<BODY>
<H2>ISIAN BUKU TAMU</H2>
<TABLE BORDER=0>
<TR>
```

```

<TD><B>Nama</B></TD>
<TD><INPUT TYPE=text NAME=NAMA size=25 Maxlength=25></TD>
</TR>
<TR>
<TD><B>Email</B></TD>
<TD><INPUT TYPE=TEXT NAME=EMAIL SIZE=60 MAXLENGTH=60>
</TD></TR>
<TR><TD><B>Hobby</B></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY
value="K">Komputer</TD></TR>
<TR><TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY
value="M">Membaca</TD></TR>
<TR><TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY value="E">Elektronika</TD>
</TR>
<TR><TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY value="O">Olah-
raga</TD></TR>
</TABLE>
</BODY>
</HTML>

```

Hasilnya:

ISIAN BUKU TAMU

Nama

Email

Hobby Komputer
 Membaca
 Elektronika
 Olah-raga

4.3.3. Atribut CHECKBOX

Seperti telah dikatakan di atas bahwa untuk membuat pilihan ganda maka digunakan *tag input* dengan atribut CHECKBOX, sehingga dapat dipilih beberapa pilihan dari pilihan yang telah disediakan. Jadi memungkinkan untuk memilih semua pilihan yang tersedia. Dengan memodifikasi dari *script* HTML di atas yaitu mengganti atribut RADIO

dengan atribut CHECKBOX, maka didapatkan tampilan yang berbeda, di mana pada tampilan gambar di atas pilihan disajikan dalam bentuk lingkaran dengan titik sebagai tanda pilihannya sedangkan bila menggunakan atribut CHECKBOX, maka pilihan disajikan dalam bentuk pilihan kotak dengan tanda cawang sebagai pilihannya. Pada tampilan contoh di sini diberikan empat pilihan hobi yang dapat dipilih semuanya mulai dari Komputer, Membaca, Elektronika dan Olah-raga.

```
<HTML>
<HEAD><TITLE>Tag INPUT with CHECKBOX Attribute</TITLE>
</HEAD>
<BODY>
<H2>ISIAN BUKU TAMU</H2>
<TABLE BORDER=0><TR><TD><B>Nama</B></TD>
<TD><INPUT TYPE=text NAME=NAMA size=25 Maxlength=25>
</TD></TR><TR><TD><B>Email</B></TD>
<TD><INPUT TYPE=TEXT NAME=EMAIL SIZE=60 MAXLENGTH=60>
</TD></TR><TR><TD><B>Hobby</B></TD>
<TD><INPUT TYPE=CHECKBOX NAME=HOBBY1
value="k">Komputer</TD> </TR><TR><TD></TD>
<TD><INPUT TYPE=CHECKBOX NAME=HOBBY2
value="m">Membaca</TD></TR>
<TR><TD></TD>
<TD><INPUT TYPE=CHECKBOX NAME=HOBBY3
value="e">Elektronika</TD> </TR><TR><TD></TD>
<TD><INPUT TYPE=CHECKBOX NAME=HOBBY4 value="o">Olah-
raga</TD> </TR>
</TABLE>
</BODY>
</HTML>
```

Hasilnya:

ISIAN BUKU TAMU	
Nama	<input type="text" value="Sigit Wasista"/>
Email	<input type="text" value="wasista@yahoo.com.sg"/>
Hobby	<input checked="" type="checkbox"/> Komputer <input checked="" type="checkbox"/> Membaca <input checked="" type="checkbox"/> Elektronika <input type="checkbox"/> Olah-raga

4.3.4. Atribut BUTTON dan SUBMIT

Penggunaan atribut ini berkaitan erat dengan tag `<form>` (dijelaskan lebih lanjut pada Sub Bab berikutnya), di mana semua data *input* yang berada di dalam tag `<form>` akan dikirimkan ke server apabila *input* yang berupa tampilan tombol diklik. Berikut ini contoh penggunaan tag *input* tombol tanpa tag `<form>`.

```

<HTML>
<HEAD>
<TITLE>Tag INPUT with SUBMIT, RESET & BUTTON Attribute
</TITLE>
</HEAD>
<BODY>
<H2>ISIAN BUKU TAMU</H2>
<TABLE BORDER=0>
<TR>
<TD><B>Nama</B></TD>
<TD><INPUT TYPE=text NAME=NAMA size=25 Maxlength=25>
</TD>
</TR>
<TR>
<TD><B>Email</B></TD>
<TD><INPUT TYPE=TEXT NAME=EMAIL SIZE=60 MAXLENGTH=60>
</TD></TR><TR><TD><B>Hobby</B></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY>Komputer</TD>
</TR>
<TR>
<TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY
value="M">Membaca</TD>
</TR>
<TR>
<TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY
value="E">Elektronika</TD> </TR>
<TR>
<TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY value="O">Olah-
raga</TD>
</TR>
<TR><TD></TD><TD ALIGN=RIGHT>
<INPUT TYPE=SUBMIT NAME=TOMBOL1 VALUE=Kirim>
<INPUT TYPE=RESET NAME=TOMBOL2 VALUE=Batal>

```

```
<INPUT TYPE=BUTTON NAME=TOMBOL3 VALUE=Home></TD></TR>
</TABLE>
</BODY>
</HTML>
```

Hasilnya:

4.3.5. Tag Select & Option

Di samping hal tersebut di atas *tag* `<form>` juga dapat digunakan untuk pembuatan menu ‘drop down’, sehingga pilihan akan ditampilkan bila diklik pada pilihan menu tersebut, dan dapat dipilih satu dari beberapa pilihan kemudian ditampilkan pada pilihan paling atas, di mana pilihan yang ditampilkan tersebut merupakan pilihan yang dipilih.

```
<HTML>
<HEAD><TITLE>SELECT & OPTION TAG</TITLE></HEAD>
<BODY>
PILIHAN JURUSAN
<SELECT NAME="JURUSAN">
<OPTION VALUE="E">Elektronika</option>
<OPTION VALUE="L">Listrik</option>
<OPTION VALUE="I">Informatika</option>
<OPTION VALUE="T">Telekomunikasi</option>
</Select>
</BODY>
</HTML>
```

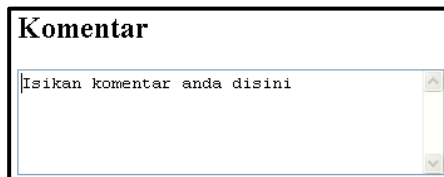
Hasilnya:

4.3.6. Teks Area

Tag ini digunakan untuk memasukkan data teks dengan ukuran melebar ke bawah (seperti editor), sehingga harus ditentukan panjang dan lebar dari area teks yang akan digunakan sebagai data *input* data.

```
<HTML>
<HEAD>
<TITLE>
Text Input using Area Tag
</TITLE>
</HEAD>
<BODY>
<H2>Komentar</H2>
<TEXTAREA ROWS=5 COLS=40>
Isikan komentar anda di sini
</TEXTAREA>
</BODY>
</HTML>
```

Hasilnya:

The image shows a browser window with a title bar. The page content includes a heading 'Komentar' and a text area below it. The text area contains the placeholder text 'Isikan komentar anda disini'. The text area has a vertical scrollbar on the right side.

4.3.7. Tag Form

Dengan adanya *tag* `<form>` maka data *input* dapat dikirimkan ke server, dalam hal ini terdapat dua metode pengiriman yaitu menggunakan atribut GET atau POST.

- Metode GET akan mengirimkan data melalui alamat URL, sehingga data yang dikirimkan akan terlihat pada isian alamat pada browser, Metode GET baik bila digunakan untuk pengiriman data yang pendek-pendek dan tidak rahasia serta memudahkan pemeriksaan data yang dikirimkan,
- Metode POST akan mengirimkan data melalui header dari *script* HTML, sehingga data yang dikirimkan tidak terlihat. Metode POST baik digunakan untuk pengiriman data yang panjang serta rahasia.

Dapat disimpulkan bahwa metode GET digunakan pada awal pembuatan form setelah dirasa data yang dikirimkan benar, maka dapat diubah menjadi metode POST.

Dengan metode GET maka data *input* akan sertakan semua pada alamat URL dengan diawali tanda tanya (?) dan diikuti dengan Nama variable yang dikirimkan beserta isinya, untuk batas antara variable satu dengan yang lainnya adalah digunakan tanda ampersand (&). Setelah data *input* tersebut benar maka dapat diubah menggunakan metode POST.

```
<HTML>
<HEAD>
<TITLE>
Tag FORM-GET
</TITLE>
</HEAD>
<BODY>
<H2>ISIAN BUKU TAMU</H2>
<FORM ACTION="FORM-GET.PHP" METHOD=GET>
<TABLE BORDER=0>
<TR><TD><B>Nama</B></TD>
<TD><INPUT TYPE=text NAME=NAMA size=25 Maxlength=25>
</TD></TR><TR><TD><B>Email</B></TD>
<TD><INPUT TYPE=TEXT NAME=EMAIL SIZE=60 MAXLENGTH=60>
</TD></TR>
<TR><TD><B>Hobby</B></TD>
<TD>
<INPUT TYPE=RADIO NAME=HOBBY VALUE="K">Komputer</TD>
</TR>
<TR><TD></TD>
<TD><INPUT TYPE=RADIO NAME=HOBBY VALUE="M">Membaca
</TD>
</TR>
<TR><TD></TD>
<TD>
<INPUT TYPE=RADIO NAME=HOBBY VALUE="E">Elektronika
</TD>
</TR>
<TR>
<TD></TD>
<TD>
<INPUT TYPE=RADIO NAME=HOBBY VALUE="O">Olah-raga</TD>
</TR>
```

```
<TR>
<TD></TD>
<TD ALIGN=RIGHT>
<INPUT TYPE=SUBMIT NAME=TOMBOL VALUE=Kirim>
<INPUT TYPE=RESET NAME=TOMBOL VALUE=Batal>
<INPUT TYPE=BUTTON NAME=TOMBOL VALUE=Home>
</TD>
</TR>
</TABLE>
</FORM>
</BODY>
</HTML>
```

Hasilnya:



ISIAN BUKU TAMU

Nama

Email

Hobby Komputer
 Membaca
 Elektronika
 Olah-raga

ACTION="FORM-GET.PHP" pada form di atas, digunakan browser untuk menjalankan file FORM-GET.PHP setelah tombol submit ditekan.

4.4. Rangkuman

- Guna menunjang pemrograman dinamis maka diperlukan *tag input* HTML untuk memasukkan data yang diperlukan sebagai masukan permintaan dari pengguna (pengunjung) halaman web. Digunakan beberapa atribut dari *tag input* seperti TEXT, RADIO, CHECKBOX, BUTTON dan SUBMIT, TEKS AREA dan Select & option untuk combo box.
- Dengan adanya *tag <form>* maka data *input* dapat dikirimkan ke server, dalam hal ini terdapat dua metode pengiriman yaitu menggunakan atribut GET atau POST.
- Metode GET akan mengirimkan data melalui alamat URL, sehingga data yang dikirimkan akan terlihat pada isian alamat pada browser, Metode GET baik bila digunakan untuk pengiriman data yang pendek-

pendek dan tidak rahasia serta memudahkan pemeriksaan data yang dikirimkan.

- Metode POST akan mengirimkan data melalui header dari *script* HTML, sehingga data yang dikirimkan tidak terlihat. Metode POST baik digunakan untuk pengiriman data yang panjang serta rahasia.

4.5. Latihan

Tuliskan kode HTML-nya dan gunakan fasilitas *TAG* FORM dengan atribut method POST dan GET supaya menghasilkan halaman web seperti di bawah ini !

User Registration

Please complete the following form to register with our site:

About You (ALT + Y)

User name:

Password:

Confirm Password:

First name:

Last name:

Email address:

Gender: Male
 Female

About Us (ALT + U)

How did you hear about us?:

Please select this box if you wish to be added to our mailing list
We will not pass on your details to any third party.

BAB V

Pembuatan Web Sederhana Menggunakan *Tag* Dasar HTML

5.1. Tujuan Instruksional Umum

1. Membangun WEB sederhana dengan tampilan yang memiliki penataan warna dan gambar secara proporsional.
2. Membangun halaman WEB sederhana yang memiliki beberapa menu dan halaman.

5.2. Tujuan Instruksional Khusus

1. Memahami dan menggunakan seluruh *tag* dasar HTML mulai dari struktur dokumen, link dan navigasi, tabel, menyisipkan citra, *tag input* dan form untuk membangun WEB sederhana.
2. Membangun WEB sederhana dengan konsep dasar desain, pewarnaan, tipografi dan *layout*.

5.3. Uraian Materi

5.3.1. Konsep Dasar Desain Web

a. Konsep Keseimbangan

Merupakan hasil susunan satu atau lebih elemen dari desain yang sama antara yang satu dengan lainnya ada dua jenis keseimbangan, yaitu :

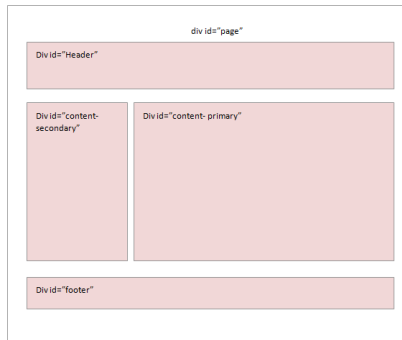
b. Keseimbangan Simetris (Formal)

Keseimbangan yang memiliki elemen - elemen dengan bobot yang sama pada dua sisi dari garis vertikal imajiner pada halaman, hal ini memberikan kesan kestabilan dan ketetapan Keseimbangan simetris menghasilkan desain yang formal. Cocok diterapkan pada aplikasi perkantoran, iklan yang menekankan pada mutu dan kualitas. Gambaran konsep dasar keseimbangan simetris adalah sebagai berikut.



c. Keseimbangan Asimetris (Nonformal)

Kebanyakan dijumpai pada ketidaksamaan dalam posisi dan intensitasnya. Untuk membuat sesuatu seimbang dalam konsep asimetris yaitu dengan menambahkan intensitas untuk mengimbangi perubahan posisi. intensitas dapat dilakukan dengan mengubah ukuran, warna dan objek. Terdapat kesulitan dalam penggunaannya, karena seorang desainer harus merancang tata letak dengan sangat hati-hati untuk memastikan bahwa desainnya benar – benar terlihat seimbang.



5.3.2. Konsep Pewarnaan

Warna pada tampilan sebuah halaman aplikasi, harus tetap menarik karena berhubungan dengan tema aplikasi, dan efek psikologis juga. Kategori aspek psikologis warna :

WARNA	MAKNA POSITIF	MAKNA NEGATIF
Merah	Kekuatan, energi, tenaga, hasrat, cinta, dengan sedikit sentuhan warna merah memberikan efek semangat	Bahaya, kekejaman, api darah, terlalu banyak warna merah berarti terlalu agresif
Merah Muda	Kewanitaan (feminim) dan masih remaja	Naif, kelemahan, kekurangan
Orange	Kehangatan, ceria, semangat, keseimbangan, menimbulkan getaran, saat ini merupakan warna yang paling sering digunakan	Meminta dan mencari perhatian

Kuning	Sinar matahari, emas, kekayaan, keberuntungan, kehidupan	Tidak jujur, pengecut, cemburu, iri hati, penghianatan, penipuan, kebohongan, resiko, sakit.
Hijau	Alam, lingkungan hidup, pertumbuhan, stabil, santai, kesuburan, harapan	Kecemburuan, nasib buruk, iri hari dan dengki
Biru	Kepercayaan, kesetiaan, kedamaian, harmonis, kebersihan, percaya diri.	Kesedihan, kedinginan, depresi, penurunan vitalitas.
Ungu	Kebangsawanan, spiritual	Kesombongan, duka cita
Coklat	Tanah, bumi, netral	Tumpul, kotor, membosankan
Abu – abu	Modern, cerdas, bersih, kokoh, masa depan, intelektual	Umur tua, kesedihan, bosan
Putih	Kesucian, kebersihan, kemurnian, kesederhanaan, damai, kebaikan, disiplin, perawan, perkawinan, musim dingin, musim salju.	Kematian (budaya timur), dingin, mandul, steril, klinik, hampa
Hitam	Kekuatan, keanggunan, kemewahan, misteri, kecanggihan, kemakmuran, kepuasan, pengalaman, keras, kokoh, sangat kuat.	Kematian (budaya barat), takut, setan, kesedihan, duka cita, marah, anonim, penyesalan

Jenis warna adalah sebagai berikut:

a. Warna primer

Warna ini berdiri sendiri, tidak bisa dicampurkan dengan warna lain. Terdiri atas merah, kuning dan biru.

b. Warna sekunder

Warna ini dibuat dengan mengombinasikan dua warna primer. Warna sekunder terdiri atas oranye, hijau dan ungu. Warna ini terletak di antara warna primer.

c. Warna tersier

Merupakan jenis warna menengah. Warna ini dibuat dengan mengombinasikan warna primer dengan perbatasan warna sekunder. Terdiri atas kuning-hijau, kuning-oranye, merah-oranye, merah-ungu, biru-ungu dan biru-hijau. Warna ini terletak di antara warna primer dan sekunder yang digunakan

5.3.3. Tipografi

Merupakan seni huruf, meliputi pemilihan huruf, penentuan ukuran yang tepat, di mana teks dapat diputus, spasi jarak dan bagaimana teks dapat dengan mudah dibaca. Huruf di web dapat dibuat sebagai bagian dari grafik (*image*) atau HTML atau dengan *style sheet*. Desainer dapat menentukan pilihan huruf, tapi yang menggunakan HTML atau *style sheet* dianjurkan untuk menggunakan huruf yang ada (*default*).

Jenis huruf secara garis besar :

a. Serif

- Mempunyai stroke (ekor).
- Berkesan formal, elegan/intelektual, anggun dan konserpatif pada desain.
- Cocok dipakai untuk teks yang panjang dengan jarak spasi yang sedikit.
- Cocok dipakai untuk organisasi, pemerintahan, pendidikan dan perusahaan.

Huruf serif terbagi menjadi 4 jenis, yaitu :

- Style (Caslon, Garamond, Goudy, Palatino),
- Transitional (Baskerville, Century, Time New Roman),
- Modern (Bodoni), dan
- Egyptian (Clarendon, Lubalin, Memphis).

b. Dekoratif

Jenis huruf yang mempunyai desain rumit, sesuatu yang baru, dll. Biasa digunakan untuk judul, banner, dll. Contoh : Stencil, crackling, rosewood, dll

c. Skrip

Menyerupai tulisan tangan, sering disebut kursif (*cursive*). Contoh : *Brush Script*, Larissa, dll.

d. Monospace

Jenis huruf yang mempunyai jarak dan lebar yang sama untuk setiap hurufnya. Contoh : Monospace, Courier New, dll.

5.3.4. *Layout*

Merupakan proses penataan dan pengaturan teks atau grafik pada halaman. Meliputi penyusunan, pembagian tempat dalam suatu halaman, pengaturan jarak spasi, pengelompokan teks dan grafik, serta penekanan pada suatu bagian tertentu. Sebelum merancang *layout*, harus mengetahui jawaban pertanyaan berikut.

- Di manakah akan diletakkan *layout* tersebut?
- Siapa yang akan melihatnya?
- Hasil apakah yang diperoleh?
- Bagaimana menempatkan teks dan gambar sehingga memberikan pengaruh yang baik?
- Akan seperti apakah *layout* tersebut?

Layout yang baik setidaknya memiliki 3 kriteria dasar, yaitu sebagai berikut.

a. Mencapai tujuan

- Apakah tujuan yang akan dicapai sebuah halaman web? Informasi apakah yang akan disampaikan?
- Siapakah yang akan membaca, menggunakan, atau mengunjungi web tersebut.
- Di manakah letak halaman web tersebut terhadap halaman yang lain.

b. Pemetaan visual

- Penataan dan penekanan pada beberapa titik informasi.
- Penentuan item apakah yang akan dibaca atau dilihat oleh pengunjung.

c. Menarik perhatian (berbeda dengan yang lain dan memiliki daya tarik tersendiri).

5.3.5. **Pembuatan desain WEB**

Pada percobaan membuat web sederhana berikut ini, belum menggunakan *cascade style sheet* atau CSS. Karena materinya akan dibahas di bab berikutnya. Penggunaan konsep dasar desain web, konsep pewarnaan, tipografi dan *layout* lebih memaksimalkan pada penggunaan *tag-tag* dasar HTML beserta atributnya.

Konsep pembuatan *layout* halaman menggunakan table dan terdiri dari dua halaman yang dihubungkan dengan link dan navigasi. Contoh berikut ini menggunakan navigasi menu Home dan Reservasi. Untuk halaman home dengan nama file home.HTML, dan untuk halaman reservasi dengan nama file reservasi.HTML.

Hotel Victoria



[Home](#)
[Profil](#)
[Fasilitas](#)
[Tipe Kamar](#)
[Reservasi](#)
[Contact](#)

» WELCOME

Selamat Datang di Hotel Victoria Banjarmasin,


Sebuah hotel berbintang yang terletak di pusat kota Banjarmasin yang akan memesona anda. Prioritas kami adalah anda. Kehadiran anda adalah tujuan kami. Karena kenyamanan anda adalah kebanggaan kami.



Address: Jl. Lambung Mangkurat no 48 Banjarmasin, Kalimantan Selatan

April 2013

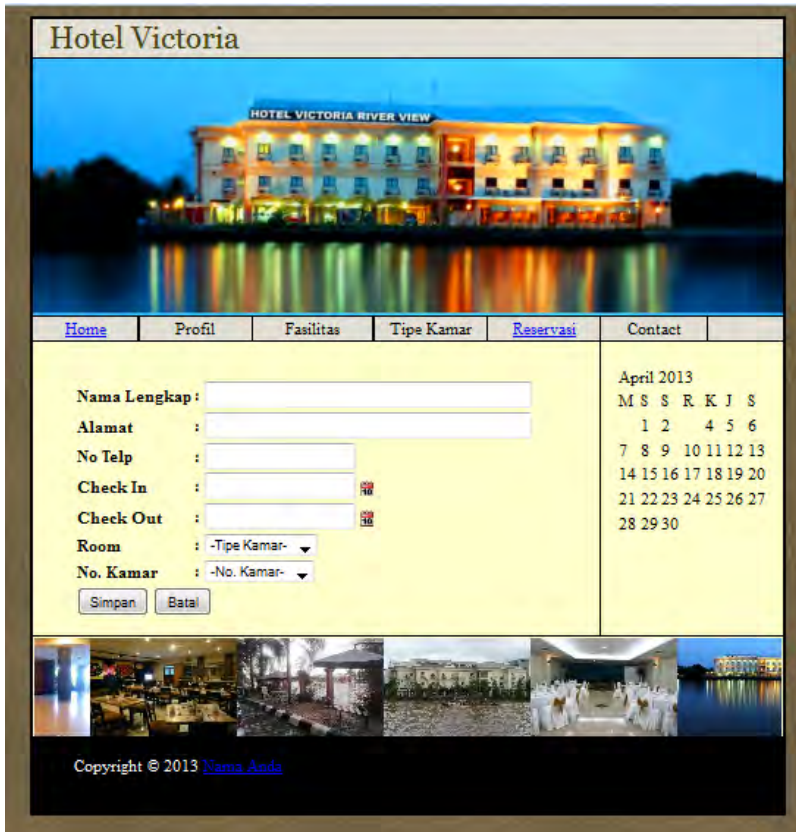
M	S	S	R	K	J	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				




Copyright © 2013 [Nama Anda](#)

Halaman home



Halaman Reservasi

Berikut ini proses pembuatannya :

1. Pembuatan Halaman Home

- a. Pembuatan warna background pada body, dengan cara memanggil gambar bg.jpg pada folder img.

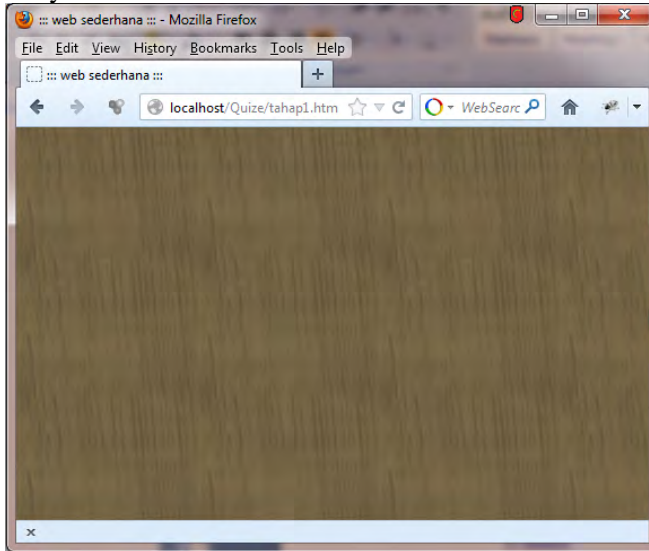
```

<html>
<head>
<title>::: web sederhana :::</title>

</head>

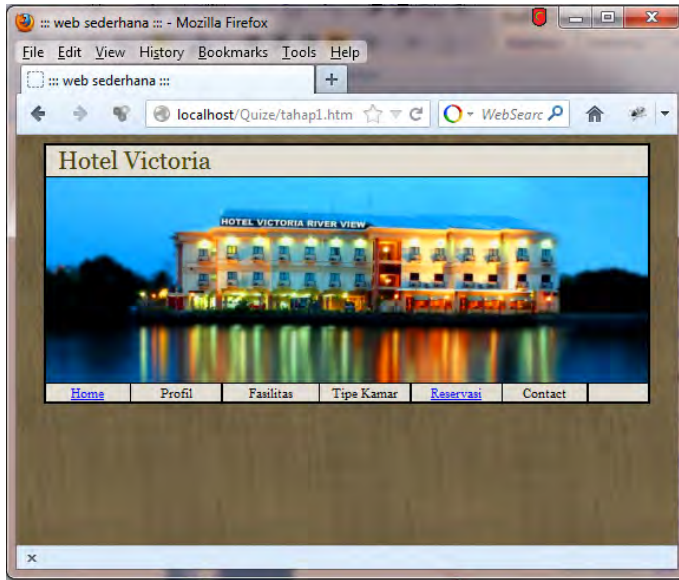
<body background="img/bg.jpg">
</body>
</html>
    
```

Simpan filenya dengan nama home.HTML ke folder c:\xampp\htdocs\ atau c:\wamp\www\
Hasilnya :



- b. Selanjutnya pembuatan tabel untuk background warna hitam, seperti yang ditunjukkan tanda panah.





f. Pembuatan menu *content* dan isi berita, menggunakan 2 kolom

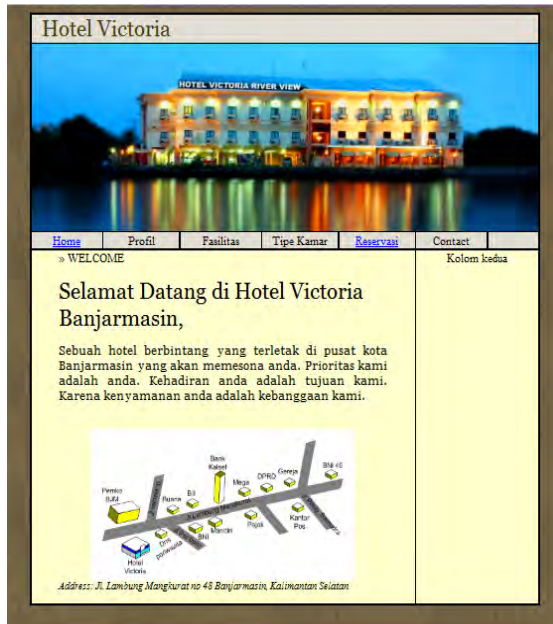
```

<td width="154" align="center">Profil</td>
<td width="168" align="center">Fasilitas</td>
<td width="154" align="center">Tipe Kamar</td>
<td width="154" align="center"><a href="reservasi.htm">Reservasi</a></td>
<td width="148" align="center">Contact</td><td width="104"></td>
</tr>

<tr ><td valign="top" bgcolor="#FFFFFF" colspan="5">
<blockquote> WELCOME <br />
<p><font face="Georgia, Times New Roman, Times, serif" size="+3">
Selamat Datang di Hotel Victoria Banjarmasin,</font></p>
<p align="justify"><font face="Georgia, Times New Roman, Times, serif" size="3">
Sebuah hotel berbintang yang terletak di pusat kota Banjarmasin yang akan memesona anda.
Prioritas kami adalah anda. Kehadiran anda adalah tujuan kami.
Karena kenyamanan anda adalah kebanggaan kami.</font>
<br /><br />
<center></center>
<address><font size="2">Address:
Jl. Lambung Mangkurat no 48 Banjarmasin, Kalimantan Selatan</font></address>
</blockquote>
</td>
<td valign="top" bgcolor="#FFFFFF" colspan="2" align="center">
Kolom kedua
</td>
</tr>
</table>
</td>
</tr>
</table>

```

Hasilnya :



g. Pembuatan kalender pada kolom kedua

```

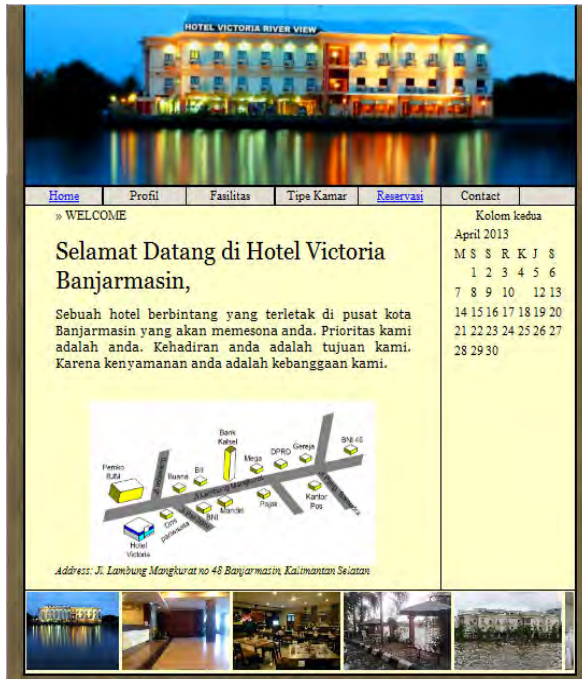
Jl. Lambung Mangkurat no 48 Banjarmasin, Kalimantan Selatan</address>
</blockquote>
</td>
<td valign="top" bgcolor="#FFFFFF" colspan="2" align="center">
Kolom kedua

<br />
<table>
<tr><td colspan="7">April 2013 </td></tr>
<tr><td>M</td><td>S</td><td>S</td><td>R</td><td>R</td><td>K</td><td>S</td></tr>
<tr><td> </td><td>1</td><td>2</td><td>3</td>
<td>4</td><td>5</td><td>6</td>
<tr><td>7</td><td>8</td><td>9</td><td>10</td>
<td colspan="2">
<blink color="#000099">11</font></blink></td>
<td>12</td><td>13</td>
</tr>
<tr><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td></tr>
<tr>
<td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td>
</tr>
<tr>
<td>28</td><td>29</td><td>30</td><td> </td><td> </td><td> </td><td> </td>
</tr>
</table>
</td>

</td>
</tr>
</table>

```


Hasilnya :



i. Pembuatan Footer

```
<tr><td height="30" colspan="7" valign="middle">
  <blockquote><font color="#FFFFFF">Copyright &copy; 2013
    <a href="indexh.php?module=home">Nama Anda</a></font></blockquote></td></tr>
```

Hasilnya :



2. Pembuatan Halaman Reservasi

Langkah-langkah untuk pembuatan bagian judul, header, navigasi, content dan footer sama seperti dengan pembuatan halaman Home. Yang membedakan hanya pada isi content. Dan nama filenya adalah Reservasi.HTML

Hotel Victoria

[Home](#) | [Profil](#) | [Fasilitas](#) | [Tipe Kamar](#) | [Reservasi](#) | [Contact](#)

Nama Lengkap :

Alamat :

No Telp :

Check In :

Check Out :

Room : -Tipe Kamar- ▾

No. Kamar : -No. Kamar- ▾

April 2013

M	S	S	R	K	J	S
1	2	3	4	5	6	
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

Copyright © 2013 [Nama Anda](#)

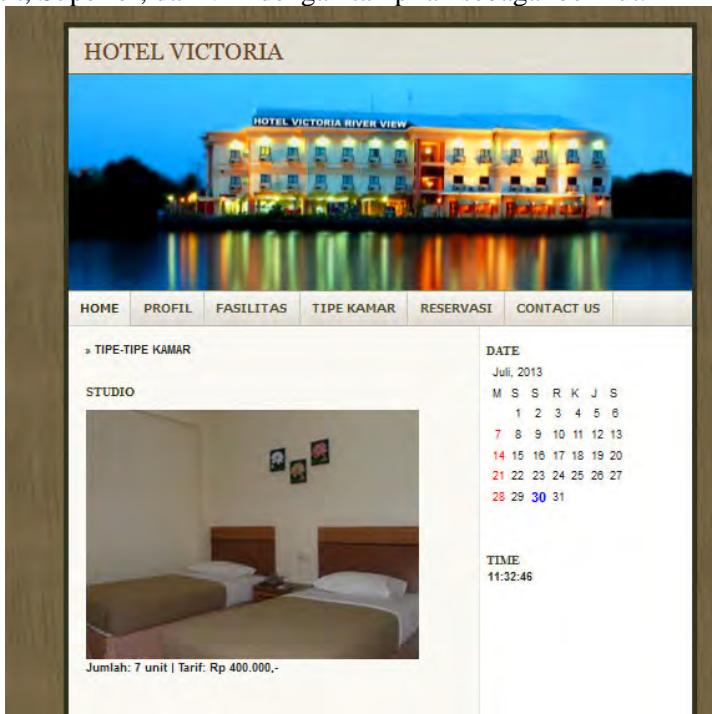
5.4. Rangkuman

- Dalam pembuatan web harus diperhatikan beberapa kriteria diantaranya : keseimbangan simetris dan asimetris, Tipografi, Pewarnaan dan *Layout*.

- Warna pada tampilan sebuah halaman aplikasi, harus tetap menarik karena berhubungan dengan tema aplikasi, dan efek psikologis juga.
- Pada desain web pemilihan huruf, penentuan ukuran yang tepat, di mana teks dapat diputus, spasi jarak dan bagaimana teks dapat dengan mudah dibaca, menjadikan hala paling dominan agar pembaca dapat bertahan lama mengunjungi web kita.
- *Layout* yang baik setidaknya memiliki 3 kriteria dasar : Mencapai tujuan, Pemetaan visual, Menarik perhatian (berbeda dengan yang lain dan memiliki daya tarik tersendiri)

5.5. Latihan

Tambahkan desain untuk halaman tipe kamar yang dibagi menjadi Standart, Superior, dan VIP dengan tampilan sebagai berikut.



BAB VI

DASAR CSS (CASCADING STYLE SHEET)

6.1. Tujuan Instruksional Umum

Memahami cara menggunakan perintah CSS dalam membangun halaman WEB.

6.2. Tujuan Instruksional Khusus

1. Memahami dasar cara penulisan CSS menggunakan inline text, embedded dan link.
2. Memahami dasar perintah CSS dan penggunaannya dari masing-masing perintah CSS tersebut.

6.3. Uraian Materi

CSS berperan bagaimana menampilkan dokumen web yang menarik. CSS adalah singkatan dari Cascading Style Sheets. Kalau baca di kamus, cascading itu artinya air terjun. Tapi dalam hal ini, yang di maksud adalah, aliran dari suatu kode ke kode lain yang saling berhubungan. Jadi kalau di tulis lengkap dalam bahasa Indonesia kira-kira arti CSS adalah: kumpulan kode-kode yang berurutan dan saling berhubungan untuk mengatur format / tampilan suatu halaman HTML. Ada keuntungan menggunakan CSS untuk mengatur tampilan web yaitu adanya pemisahan antara presentation sebuah dokumen web dengan content dokumen akan memudahkan dalam melakukan pengembangan dan pemeliharaan situs web. Tanpa adanya CSS, maka ketika melakukan perubahan terhadap tampilan web seorang web designer harus mengubah setiap file web. Tapi di sisi lain juga ada kekurangan, yaitu setiap *User Agent* atau browser memiliki *default* style sheet. Sehingga diperlukan teknik – teknik khusus agar tampilan web dapat sesuai di semua web browser.

Satuan Dalam CSS

a. Statik

- in -- satuan inchi

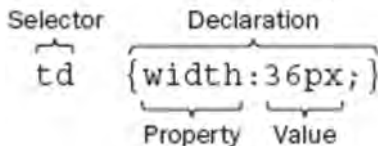
- cm -- satuan centimeter
- mm -- satuan milimeter
- pt -- satuan point (1point = 1/72 inchi)
- pc -- satuan pica (1pica = 12 point)
- px -- satuan pixel (satu titik gambar terkecil dalam layar monitor)

b. Relatif

- % -- satuan persen
- em -- atau ems (1em)
- ex -- 1ex = x-height suatu font (x-height biasanya setengah ukuran font)
- **1em = 12pt = 16px = 100%.**

Syntax CSS

Syntax / kalimat CSS terdiri dari beberapa set peraturan yang memiliki: selector, property, dan value. Format penulisan kalimat CSS:



Selector itu untuk menunjukkan bagian mana yang hendak diatur / diformat. Property untuk menunjukkan, bagian (properti) dari selector yang hendak diatur. Value adalah nilai dari pengaturannya.

Contoh Syntax:

```
h1 { color: red }
```

Contoh di atas menunjukkan

- Selector: h1
- Property: color
- Value: red

Kalau diterjemahkan ke kalimat bahasa Indonesia kira-kira begini:
Mengatur color dari *h1* ke warna *merah* (*red*).

Pengelompokan Selectors

Anda dapat menulis satu kode CSS untuk berbagai macam selector dengan cara menggunakan koma.

Misalkan anda mau mengatur agar tag *h1*, *h2* dan *h3* semua menggunakan warna *merah*, maka kode

CSS nya menjadi:

```
h1,h2,h3 { color: red }
```

Perhatikan penulisan h1,h2,h3 yang dipisahkan oleh koma.

Penggunaan Banyak *Properties*

Untuk mengatur lebih dari satu properties gunakan pemisah titik koma (;). Contoh:

```
h1,h2,h3 {  
color:red;  
font-family:arial;  
font-size:150%;  
}
```

CSS Comment

Kadang kala, ada baiknya anda menuliskan komentar ke dalam kode CSS anda untuk memberi catatan pengingat. Anda bisa menggunakan syntax pembuka `/*` dan penutup `*/` untuk menuliskan komentar. Segala teks yang berada di antara `tag /*` dan `*/` tidak akan dibaca sebagai kode, tapi hanya sebagai catatan untuk diri anda.

```
/* Tulis komentar anda di sini */  
p{  
text-align: justify;  
/* Tulis komentar anda di sini */  
color: blue;  
font-family: arial;  
}
```

6.3.1. Tiga teknik menempatkan aturan CSS ke dalam dokumen web

- Inline style yaitu aturan CSS langsung disertai dalam elemen web
- Embedded style yaitu aturan CSS disertai dalam dokumen web
- Linked style yaitu aturan CSS terdapat di file tersendiri yang kemudian dipanggil oleh dokumen web.

a. Inline CSS

Kode CSS dituliskan langsung ke dalam `tag` HTML yang ingin di format. Penulisan cara ini tidak memerlukan penulisan selector dalam kode CSS. Cara ini sebaiknya hanya digunakan jika anda mau memformat suatu elemen satu kali saja.

Contoh:

```
<P style="color:blue">
Isi paragraf.
</p>
```

Pada contoh di atas, elemen paragraf <P> di format agar tulisannya menggunakan warna *biru*. Elemen paragraf lain, tidak akan menggunakan warna biru, karena format ini hanya berlaku pada elemen paragraf yang ditentukan kode CSS nya. Penulisan CSS dengan cara ini di mulai dengan kata *style:* lalu di ikuti dengan syntax *property: value.*

b. Inline style

```
<HTML>
  <head>
    <title>Inline Style</title>
  </head>
  <body>
    <h1 style="color:maroon">Inline Style</h1>
    <p style="color:blue">Paragraf ini menggunakan css
sehingga
    bisa berwarna biru</p>
  </body>
</HTML>
```

Hasilnya:



c. Embedded CSS

Anda bisa juga menempelkan kode CSS di antara tag <head> dan </head> atau dapat juga diadalam body <body> dan </body>. Penulisan CSS dengan cara ini diawali dengan tag <style> dan diakhiri dengan tag </style>.

Contoh:

```
<head>
<style type="text/css" media=screen>
p {color:blue;}
</style>
</head>
```

Dalam contoh di atas semua elemen *<P>* dalam halaman web tersebut akan diformat menggunakan font berwarna *biru*.

```
<HTML>
  <head><title>Embedded Style</title>
  <style type="text/css">
    body {color: navy}
    h1 {font-family: Arial}
    b {text-decoration: underline}
  </style>
</head>
<body>
  <h1>Ini <b>Homepage</b> Saya</h1>
  Ini adalah homepage yang menggunakan CSS.
</body></HTML>
```

Hasilnya :



d. Link ke *sheet* lainnya

Apabila Anda menginginkan *style* yang sama untuk halaman HTML Anda yang lain, disarankan mempergunakan *sheet* terpisah namun satu dan lainnya terhubung dengan cara *link*. Anda dapat mengikuti cara berikut ini :

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

Kode *tag* untuk *link* ini ditempatkan di bagian "head" dokumen Anda. Perintah *rel* perlu diatur dengan pernyataan "stylesheet" agar supaya browser dapat menemukan perintah *href* sebagai penunjukan ke alamat Web (URL) *sheet* Anda.

Buat *file* baru lagi, berekstensi CSS, Ketik code di bawah ini dan simpan dengan nama *link.CSS*

```
body {color: navy}
h1   {font-family: Arial}
b    {text-decoration: underline}
```

```
<HTML>
  <head>
    <title>Embedded Style</title>
    <link rel="stylesheet" href="link.css"
type="text/css">
  </head>
  <body>
    <h1>Ini <b>Homepage</b> Saya</h1>
    Ini adalah homepage yang menggunakan CSS.
  </body></HTML>
```

Hasilnya :



6.3.2. Class dan ID Selektor

Pada contoh-contoh di pelajaran sebelumnya, Anda melihat penggunaan *tag* HTML sebagai selector. Misalkan Anda membuat kode CSS untuk *tag* `<h1>`. Sekarang bagaimana jika Anda ingin memformat *tag* `<h1>` dengan warna / property berbeda? Misalkan, Anda ingin *tag* `<h1>` di kolom kiri berwarna biru sementara *tag* `<h1>` di kolom tengah berwarna hitam. Untuk kasus seperti ini, Anda bisa menggunakan Class selector dan ID selector.

a. Class Selector

Class selector adalah penggabungan beberapa properties yang digunakan lebih dari satu kali. Cara penulisan Class Selector:

```
.nama-class {property:value;}
```

Atau

```
tagHTML.nama-class {Property:value;}
```

Perhatikan tanda titik di setiap awal nama Class.

Penulisan kode CSS, simpan dengan nama link1.css

```
.tengah {text-align:center;}
  p {color:red;}
h1.kiri {color:blue;}
  h1 {color:yellow;}
```

Pemakaian kode CSS pada HTML

```
<HTML>
  <head>
    <title>
      Embedded Style
    </title>
    <link rel="stylesheet" href="link1.css"
type="text/css">
  </head>
  <body>
    <p class="tengah">
      Teks tengah akan berwarna merah.
    </p>
    <h1 class="tengah">
      Tag H1 tengah akan berwarna kuning
    </h1>
    <h1 class="kiri">
      Tag H1 kiri akan berwarna biru
    </h1>
  </body>
</HTML>
```

Hasilnya :



b. ID Selector

ID Selector mirip dengan Class selector. Untuk membedakannya, gunakanlah ID selector untuk memformat bagian yang hanya muncul satu kali dalam satu halaman web, misalnya untuk memformat bagian menu / sidebar. Cara penulisan ID Selector:

```
#nama-ID {property:value;}
```

Untuk menempelkan ID selector ke dalam tag HTML:

```
tagHTML#nama-ID {Property:value;}
```

Perhatikan tanda # di setiap awal nama ID. Jika anda ingin menggunakan class selector di luar kode HTML anda menggunakan tag `<div id=nama-ID>` dan di akhiri dengan tag `</div>`.

Coding CSS simpan dengan format1.css:

```
#top {background-color:#ccc;
padding:1em;}
.intro{color: red; font-weight:bold;}
```

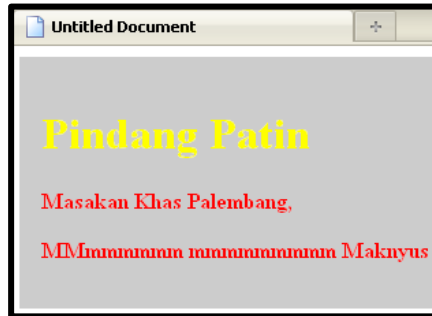
Coding HTML :

```
<HTML>
<head>
<title>Untitled Document</title>
<link rel="stylesheet" href="format1.css"
type="text/css">
</head>

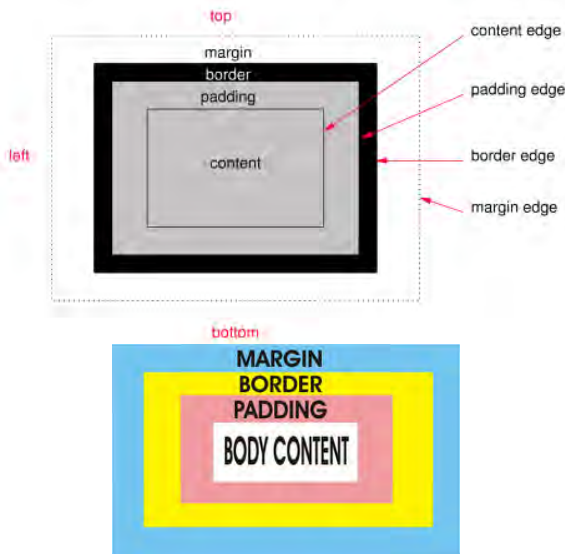
<body>
<div id="top">
<h1>Pindang Patin</h1>
<p class="intro">Masakan Khas Palembang, </p>
```

```
<p class="intro">MMmmmmmmmm mmmmmmmmm Maknyus</p>  
</div>  
</body>  
</HTML>
```

Hasilnya :



6.3.3. Padding, Border dan Margin



a. Padding

Adalah Untuk menentukan jarak komponen body ke border atau Ukuran jarak bagian dalam ditulis dengan CSS `padding:5px 5px 5px 5px;` urutan nilai angkanya adalah atas, kanan, bawah dan kiri, atau Anda bisa

menggunakan `padding-left:5px;` ini adalah untuk pengaturan padding bagian kiri, `padding-right:5px;` ini adalah untuk pengaturan padding kanan, `padding-top:5px;` untuk bagian atas, dan `padding-bottom:5px;` untuk bagian bawah, ingat satuan px(pixels) bisa kamu ganti sesuai satuan yang lain yang sesuai.

b. Margin

Adalah Ukuran jarak bagian luar atau ukuran jarak sesudah Border Ditulis dengan CSS `margin:5px 5px 5px 5px;` urutannya atas, kanan, bawah dan kiri, atau bisa menggunakan seperti padding di atas `margin-left:5px; margin-right:5px; margin-top:5px; margin-bottom:5px;`

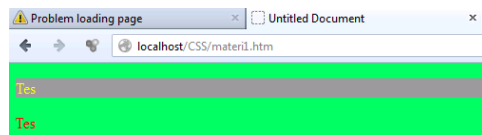
Buatlah *file* CSS, dan ketikan *code* css berikut, dan simpan dengan nama *file* padding.css:

```
body{background-color:#00FF66}
.tengah{color:#FF0000;}
.kiri{color:#FFFF00}
#top{background-color:#999999;}
```

Ketik coding HTML berikut :

```
<HTML>
<head>
<title>Untitled Document</title>
<link rel="stylesheet" type="text/css"
href="padding.css">
</head>
<body>
<div id="top">
  <p class="kiri">Tes</p>
</div>
  <p class="tengah">Tes</p>
</body>
</HTML>
```

Hasilnya:



Kemudian tambahkan file CSS pada baris #top dengan yang dicetak tebal

```
#top{background-color:#999999; padding-left:50px;}
```

Lihat hasilnya!!!!

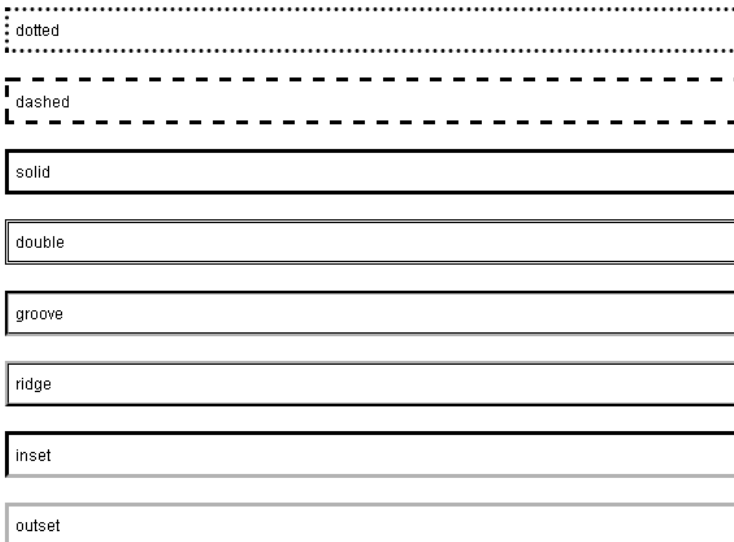
Kemudian tambahkan file CSS pada baris #top dengan yang dicetak tebal

```
#top{background-color:#999999; padding-left:50px; margin-left:50px}
```

Lihat hasilnya!!!!

c. Border

Adalah garis tepi dari komponen. Ditulis dengan CSS `border:1px dotted #000000`; urutan penggunaannya adalah ukuran border, *style* border dan warna border, atau bisa menggunakan `border-width:1px`; ini adalah ketebalan border `border-style:dotted`; ini adalah jenis bordernya bisa kamu ganti dengan dashed, solid, double, groove, ridge, inset, outset dan lainnya `border-color:#FFFFFF`; ini adalah warna dari border.. kamu bisa mengganti *code* warnanya.



d. Border -width

Buatlah file CSS dengan nama format5.css

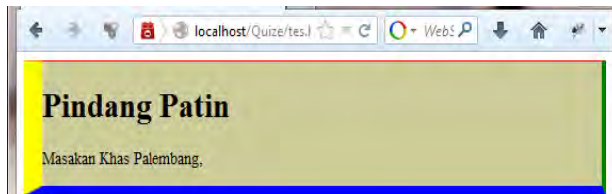
```
/* CSS Document */
```

```
div { border-width:1px 5px 10px 20px; /*top right
bottom left*/
      border-color: red green blue yellow;
      border-style:solid;
      background-color:#CC9;}
```

Ketiklah coding HTML berikut ini :

```
<head>
<title>Untitled Document</title>
<link rel="stylesheet" href="format5.css"
type="text/css">
</head>
<body>
<div>
  <h1>Pindang Patin</h1>
  <p>Masakan Khas Palembang, </p>
</div>
</body>
</HTML>
```

Hasilnya :



Jika keseluruhan nilai border dengan ukuran sama untuk top right bottom dan left, serta warna dan bentuknya maka dapat ditulis sebagai berikut :

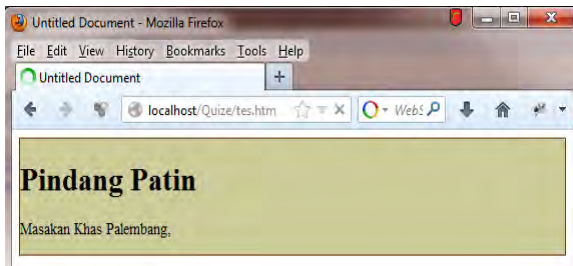
```
/* CSS Document */
div { border : 1px #630 solid; /*top right bottom
left warna dan bentuk border*/
      background-color:#CC9;}
```

```
<head>
<title>Untitled Document</title>
<link rel="stylesheet" href="format4.css"
type="text/css">
</head>
```

```
<body>
```

```
<div>
  <h1>Pindang Patin</h1>
  <p>Masakan Khas Palembang, </p>
</div>
</body>
</HTML>
```

Hasilnya :



6.3.4. Nesting

Bila dokumen HTML yang dibuat memiliki struktur yang baik, Anda tidak perlu repot-repot membuat banyak class dan ID. Karena Anda dapat mendefinisikan CSS dengan cara penulisan selector dalam selector.

Buatlah file CSS dengan nama file format2.css

```
#top {background-color:#ccc;
padding:1em;}

#top h1{color:#ff0;}
#top p{color:red; font-weight:bold;}
```

Ketiklah coding HTML berikut ini

```
<HTML>
<head>
<title>Untitled Document</title>
<link rel="stylesheet" href="format2.css"
type="text/css" >
</head>

<body>
<div id="top">
  <h1>Pindang Patin</h1>
```


6.3.6. Tipografi

a. Font style

Buatlah coding css berikut ini dengan nama font1.css

```
body {
  color:#899900;
  background-color:#ffffff;
  font-family:arial, verdana, sans-serif;
  font-size:12px;}
td {width:200px;}
p.one {font-style:normal;}
p.two {font-style:italic;}
p.three {font-style:oblique;}
```

Kemudian buatlah coding HTML berikut :

```
<HTML >
<head>
  <title>Font-Style</title>
  <link rel="stylesheet" type="text/css"
href="font1.css" />
</head>
<body>
<h1>font-style</h1>
<p class="one">This is a normal font</p>
<p class="two">This is an italic font</p>
<p class="three">This is an oblique
font</p>
</body>
</HTML>
```

Hasilnya :



b. Text Decoration

Buatlah coding HTML berikut :

```
<HTML >
<head>
  <title>Text Decoration</title>
  <link      rel="stylesheet"      type="text/css"
href="ch07_eg10.css" />
</head>
<body>
<h1>text-decoration</h1>
<p  class="underline">This      text      should      be
underlined</p>
<p class="overline">This text should have a line
over it</p>
<p class="line-through">This text should have a line
through it</p>
<p class="blink">This text should be blinking</p>
<a href="#">Home</a>
<ul>
  <li>home</li>
</ul>
</body>
</HTML>
```

Kemudian buatlah file CSS dengan nama file ch07_eg10.css

```
li {list-style:none}
body {
  color:#000000;
  background-color:#ffffff;
  font-family:arial, verdana, sans-serif;
  font-size:12px;}
table {width:500px;}

p.underline {text-decoration:underline;}
p.overline {text-decoration:overline;}
p.line-through {text-decoration:line-through;}
p.blink {text-decoration:blink;}
a {text-decoration:none;}
```

c. Drop shadow

Buatlah file HTML berikut ini:

```
<HTML>
<head>
  <title>Drop Shadow</title>
  <link      rel="stylesheet"      type="text/css"
```

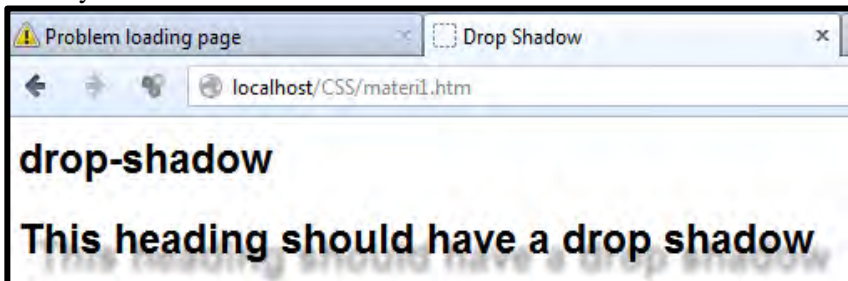
```
href="drop.css" />
</head>
<body>
<h1>drop-shadow</h1>
<h1 class="shadow">This heading should have a drop
shadow</h1>
</body>
</HTML>
```

Buatlah file CSS dengan nama drop.css

```
body {
  color:#000000;
  background-color:#ffffff;
  font-family:arial, verdana, sans-serif;
  font-size:12px;}

.shadow {text-shadow: #999999 10px 10px 3px;}
```

Hasilnya:



6.3.7. BackGround

Properti CSS yang bisa digunakan untuk efek background adalah:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

a. Background Color (Warna Latar belakang)

Properti background-color biasanya digunakan untuk menentukan warna latar belakang pada suatu elemen.

Kode:

```
<p style="background-color:yellow;">Ini adalah text yang mempunyai warna latar belakang.</p>
```

Hasilnya:

Ini adalah text yang mempunyai warna latar belakang.

b. Background Image (Gambar Latar Belakang)

Properti `background-image` digunakan untuk menampilkan gambar sebagai latar belakang pada suatu elemen. Secara *default*, gambar akan tampil berulang dan menutupi seluruh bagian pada elemen.

Kode:

```
<p style="height: 35px; background-image: URL(true_small.png);">Ini adalah text yang mempunyai latar belakang gambar.</p>
```

Hasilnya:



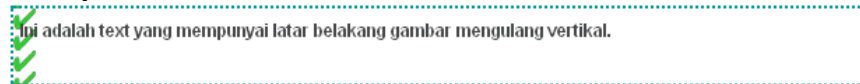
c. Background Image Repeat

Gambar background bisa diatur hanya tampil berulang untuk vertikal atau horisontal saja.

Vertikal:

```
<p style="height: 40px; background-repeat: repeat-y; background-image: URL(icon_true_small.png);">Ini adalah text yang mempunyai latar belakang gambar mengulang vertikal.</p>
```

Hasilnya:



Horizontal:

```
<p style="height: 40px; background-repeat: repeat-x; background-image: URL(icon_true_small.png);"><strong>Ini adalah text
```

yang mempunyai latar belakang gambar mengulang horizontal.</p>

Hasilnya:



Ini adalah text yang mempunyai latar belakang gambar mengulang horizontal.

d. Background Image Position (Posisi gambar latar belakang)

Biasanya digunakan untuk mengatur posisi tampil dari gambar latar belakang.

```
<p style="height: 40px; background-image:
URL('code/icon_true_small.png'); background-repeat:
no-repeat; background-position: 100px;">Mengatur
posisi gambar latar belakang 100px dari kiri.</p>
```

Hasilnya:



Mengatur posisi gambar latar belakang 100px dari kiri.

e. Mempersingkat properti background

Seperti yang dapat Anda lihat dari contoh di atas, ada banyak properti yang digunakan untuk menentukan efek background. Dengan mempersingkat kode, kita dapat menentukan semua properti dalam satu properti tunggal.

Contoh:

```
body {background:#ffffff URL(icon_true_small.png) no-
repeat right top;}
```

6.3.8. Float dan clear

Float sering digunakan untuk mengatur posisi image atau gambar tetapi hal ini juga dapat digunakan untuk mengatur *layout* elemen-elemen [website](#). Sesuai dengan artinya, float, memungkinkan kita untuk “memaksa” elemen tersebut ke kanan dan ke kiri. Tetapi hanya horizontal saja, jadi kita tidak bisa memaksanya untuk posisi ke atas atau ke bawah. Contoh penggunaan float adalah sebagai berikut :

```
.image{float:left;}
```

Pada saat kita menempatkan ke kanan atau ke kiri, maka elemen tersebut akan menempati posisi di kanan atau kiri hingga “mentok” dengan elemen yang menaunginya dan elemen yang berada sebelum elemen float tidak akan terpengaruh tetapi elemen-elemen yang terletak setelah float akan terpengaruh.

Float memiliki beberapa value sebagai berikut.

a. float:right

Seperti di atas fungsi float adalah menekan bidang ke atas, maksud dari float right ialah menekan ke atas kanan.

b. float:left

Seperti fungsi float adalah menekan bidang ke atas, sedangkan maksud dari float left ialah menekan ke atas kiri.

c. float:none

Ini adalah nilai defaultnya, jadi jika ditambahkan float none sama saja dengan tidak perlu ditulis

Karena semua elemen setelah elemen floating akan terpengaruh, maka perlu properti khusus agar sebuah elemen tidak terpengaruh walaupun tempatnya setelah elemen floating. Properti yang kita gunakan adalah clear. Contoh penggunaan clear adalah sebagai berikut:

```
.judul{clear:both;}
```

Nilai yang dimasukkan pada properti tersebut adalah :

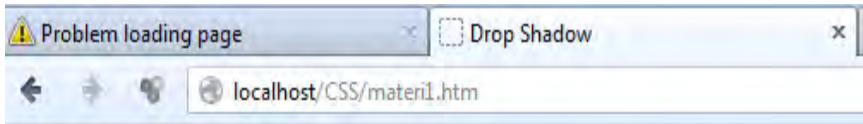
Properti	Nilai	Arti
Clear	left right both none inherit	sisi elemen di mana tidak terpengaruh oleh floating elemen
Float	left right none inherit	posisi dari floating atau tidak float sama sekali

6.4. Rangkuman

- Tiga teknik menempatkan aturan CSS ke dalam dokumen web adalah Inline style yaitu aturan CSS langsung disertai dalam elemen web, Embedded style yaitu aturan CSS disertai dalam dokumen web dan Linked style yaitu aturan CSS terdapat di file tersendiri yang kemudian dipanggil oleh dokumen web.
- Class selector adalah penggabungan beberapa properties yang digunakan lebih dari satu kali. ID selector untuk memformat bagian yang hanya muncul satu kali dalam satu halaman web, misalnya untuk memformat bagian menu / sidebar.
- Padding digunakan untuk menentukan jarak komponen body ke border atau Ukuran jarak bagian dalam, border adalah garis tepi dari komponen, sementara margin adalah Ukuran jarak bagian luar atau ukuran jarak sesudah Border
- Pseudo Class adalah atribut tambahan yang diberikan pada selector , biasanya dipisahkan dengan tanda titik dua (:), contoh penulisannya
 - **:link { color:#00ff; } /* untuk warna link yang belum diakses */**
 - **:visited { color:#00ff; } /* untuk warna link yang telah diakses */**
 - **:active { color:#00ff; } /* untuk warna link ketika link diklik */**
 - **:hover { color:#00ff; } /* untuk warna link ketika mouse di atasnya*/**
- Float sering digunakan untuk mengatur posisi image atau gambar tetapi hal ini juga dapat digunakan untuk mengatur *layout* elemen-elemen website. Sesuai dengan artinya, float, memungkinkan kita untuk “memaksa” elemen tersebut ke kanan dan ke kiri. Tetapi hanya horizontal saja, jadi kita tidak bisa memaksanya untuk posisi ke atas atau ke bawah. Sementara Clear merupakan properti khusus agar sebuah elemen tidak terpengaruh walaupun tempatnya setelah elemen floating

6.5. Latihan

1. Bagaimana cara penulisan tiga teknik menempatkan aturan CSS ke dalam dokumen web, berikan contohnya.
2. Bagaimana perbedaan cara penulisan Class dan ID selector dalam dokumen web. Berikan contohnya dengan menggabungkan beberapa *tag* HTML dalam Class dan ID selector
3. Jika diberikan tampilan sebagai berikut :



drop-shadow

This heading should have a drop shadow

Berikan contoh coding HTML dan CSS nya, sehingga hasil tampilan pada browser seperti pada gambar di atas.

BAB VII

PENGATURAN HALAMAN WEB DENGAN DASAR CSS (BAGIAN 1)

7.1. Tujuan Instruksional Umum

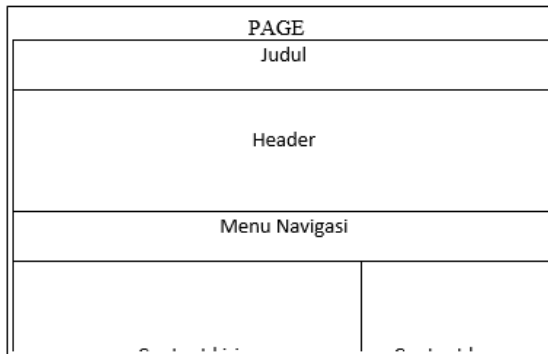
Memahami cara menggunakan perintah CSS dalam membangun halaman WEB.

7.2. Tujuan Instruksional Khusus

1. Memahami dan menggunakan dasar cara penulisan CSS untuk membangun sebuah *layout* web.
2. Memahami dasar perintah CSS untuk mengatur pewarnaan, dan tipografi halaman WEB.

7.3. Uraian Materi

7.3.1. Pembuatan *Layout* kasus1



Buatlah coding HTML seperti di bawah ini :

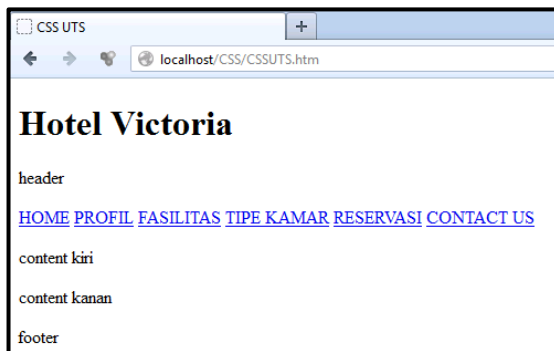
```
<HTML>  
<head>  
<title>Kasus 1</title>  
</head>  
  
<body>
```



```
<div id="page">
  <div class="judul"><h1>Hotel Victoria</h1></div>
  <div id="header"> <p>header</p></div>
  <div id="navigation">
    <a href="#" id="active">HOME</a>
    <a href="#">PROFIL</a>
    <a href="#">FASILITAS</a>
    <a href="#">TIPE KAMAR</a>
    <a href="#">RESERVASI</a>
    <a href="#">CONTACT US</a>

  </div>
  <div id="kiri"> <p>content
kiri</p></div>
  <div id="kanan"> <p>content kanan</p></div>
  <div id="footer"><p>footer</p></div>
</div>
</body>
</HTML>
```

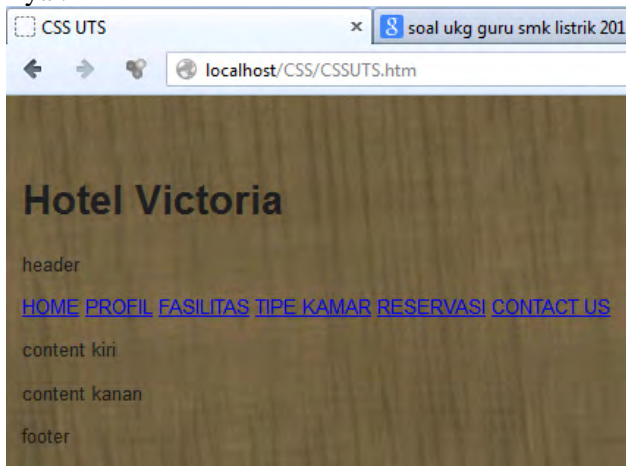
Hasilnya :



1. Buatlah file dengan nama kasus1.css
2. Tambahkan link file kasus1.css ke file HTML yang sudah dibuat sebelumnya
3. Kemudian ketik coding berikut ini pada file kasus1.css:

```
Body { background-image:URL(img/bg.jpg);
          color:#222;
          font: normal 62.5% sans-serif;
          padding: 2% 0;}
```

pada body HTML akan menggunakan background dari file gambar bg.jpg, kemudian seluruh tulisan dalam body berwarna dengan code #222, sanserif, ukurannya 62.5% dan normal. Padding 2% untuk pengaturan bagian atas dan bawah
Hasilnya :

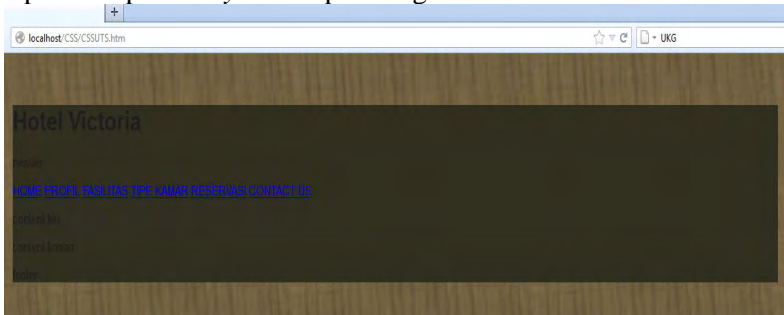


7.3.2. Pengaturan pada Block Content dan Class Content kasus 1

1. Buatlah ID untuk page di kasus1.css

```
#page{background: #332 URL(img/wrapper.gif) no-repeat;  
margin: 0 auto;  
width: 762px;}
```

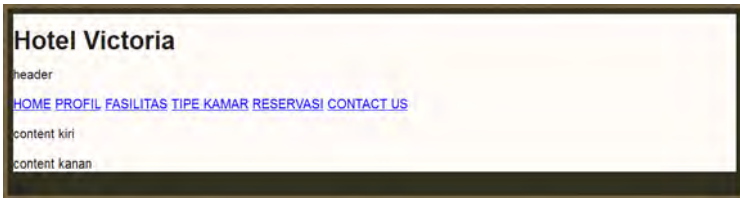
pengaturan,margin agar pada saat layar browser diperbesar dan diperkecil posisi *layout* tetap di tengah.



2. Buatlah ID untuk container

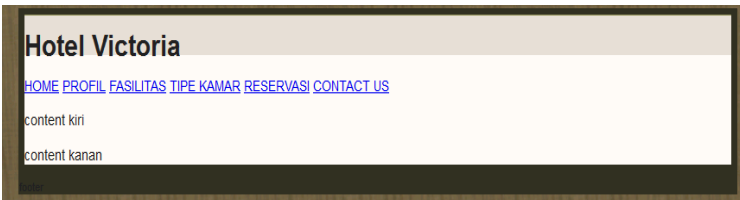
```
#container {  
    background: #FBF9F4;  
    border-top: 6px solid #332;  
    font-size: 1.3em;  
    line-height: 1.3em;  
    margin: 0 auto;  
    width: 750px;  
}
```

Jarak antar kalimat tiap barisnya diatur tingginya dengan line-height.



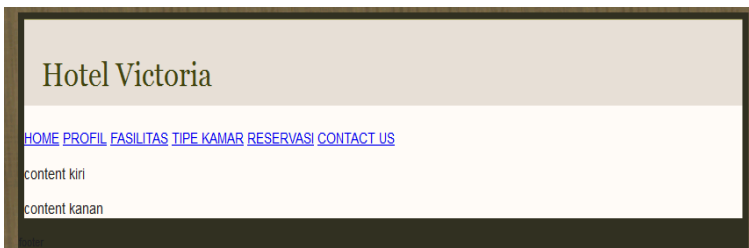
3. Buatlah CLASS untuk judul

```
.judul {  
    background: #E2DED4;  
    border-top: 1px solid #996;}  
}
```



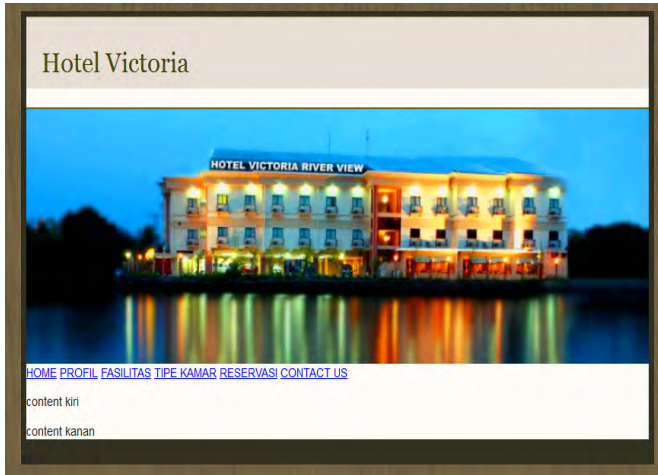
4. Buatlah CLASS-SELECTOR untuk H1 pada judul

```
.judul h1 {  
    color: #441;  
    font: normal 2.2em Georgia, "Trebuchet MS", sans-serif;  
    padding: 10px 0 10px 18px;  
}
```



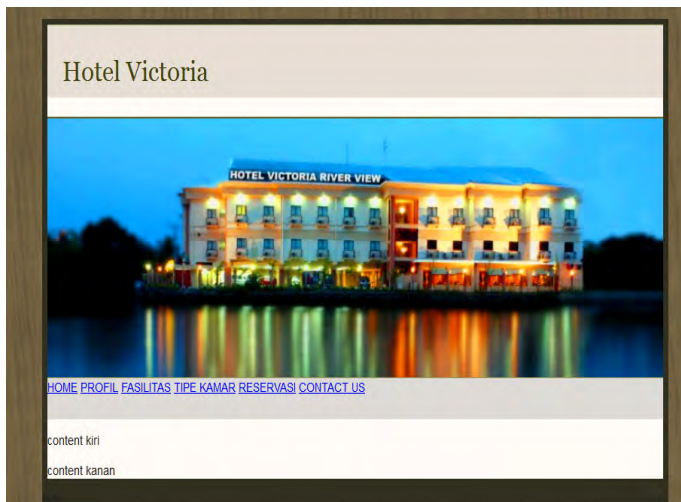
5. Buatlah ID untuk Header

```
#header {  
  border-top: 2px solid #663;  
  background: URL(img/header.png) no-repeat;  
  height: 260px;  
}
```



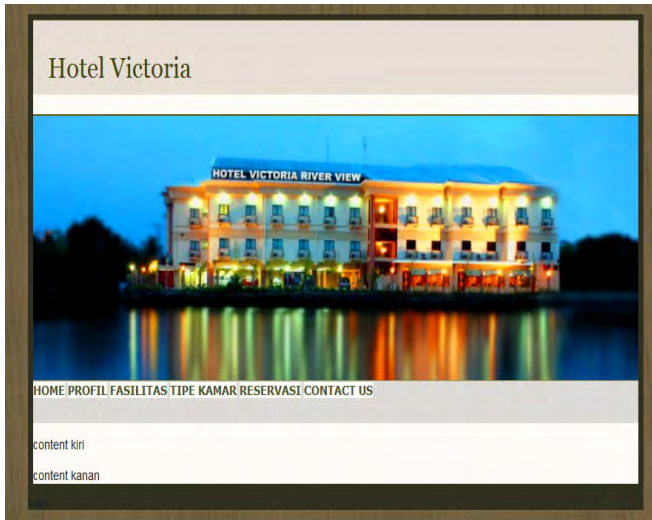
6. Buatlah CLASS untuk navigation

```
.navigation {  
  background: #DED9D0 URL(img/nav.gif) repeat-x;  
  height: 41px;  
  border-top: 1px solid #996;  
}
```



7. Buatlah CLASS-SELECTOR untuk a pada navigation

```
.navigation a {  
    background: #FFF URL(img/nav.gif) repeat-x;  
    color: #553;  
    font: bold 1em Tahoma,sans-serif;  
    text-align: center;  
    text-decoration: none;  
}
```



Masih pada Class-selector untuk **a** pada navigation, tambahkan perintah ini :

```
border-right: 1px solid #C9C6B3;  
line-height: 41px;
```

Amati hasilnya!!!, dan buat kesimpulan

Masih pada Class-selector untuk **a** pada navigation, tambahkan perintah ini :

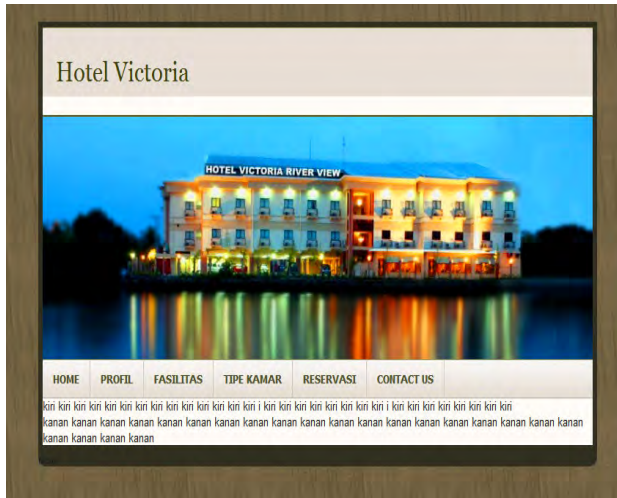
```
padding: 0 14px;
```

Amati hasilnya!!!, dan buat kesimpulan

Masih pada Class-selector untuk **a** pada navigation, tambahkan perintah ini :

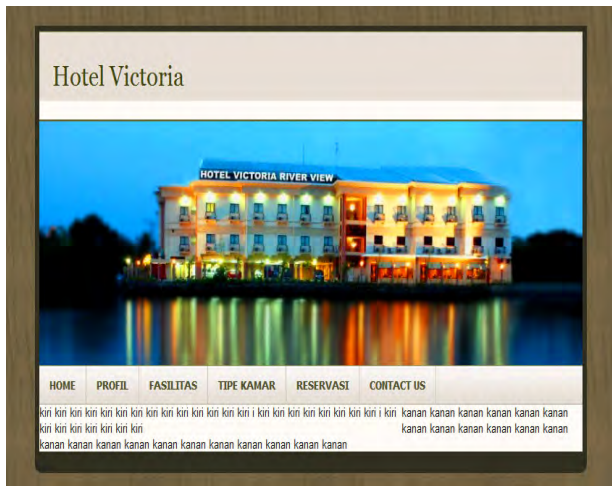
```
float: left;
```

Amati hasilnya!!! Dan buat kesimpulan



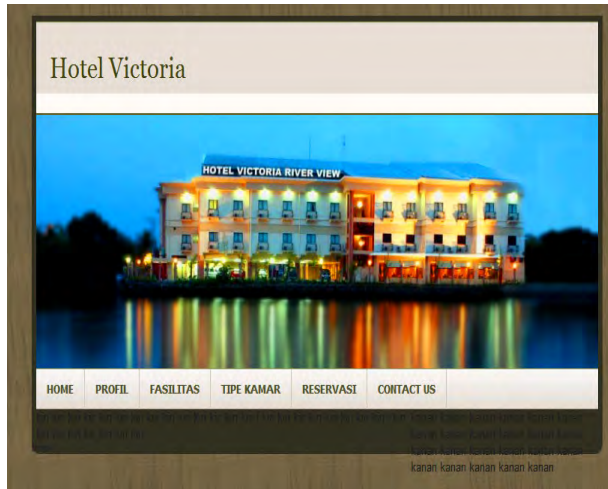
8. Tambahkan ID kiri pada kasus1.css

```
#kiri{float:left; width:500px}
```



9. Tambahkan ID kanan pada kasus1.css

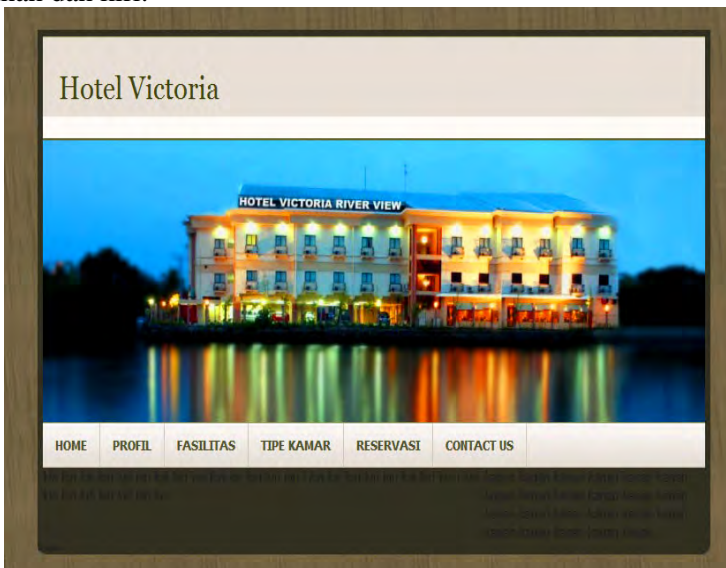
```
#kanan{float:right; width:250px}
```



10. Tambahkan ID footer pada kasus 1.css

```
#footer{clear:both;}
```

Perintah clear akan membebaskan footer dari pengaruh float pada ID kanan dan kiri.



Tambahkan Class main, class isi sesuai dengan petunjuk di bawah ini pada file HTMLnya.

```

16     <a href="#">TIPE KAMAR</a>
17     <a href="#">RESERVASI</a>
18     <a href="#">CONTACT US</a>
19 </div>
20 <div class="main">
21   <div id="kiri">
22     <div class="isi">kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri kiri </div>
23   </div>
24   <div id="kanan">
25     <div class="isi">kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan kanan </div>
26   </div>
27 </div>
28 </div>
29 </div>
30 <div id="footer"><p>footer</p></div>
31 </div>
32 </div>
33 </body>

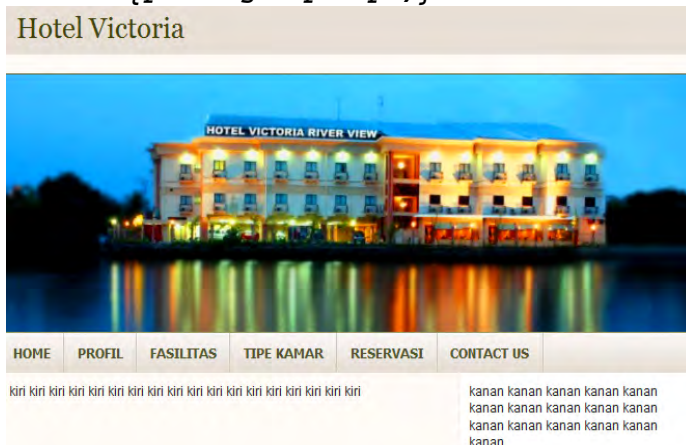
```

11. Tambahkan CLASS main dan isi pada kasus1.css

```

.main{background: #FFF URL(img/two-columns.gif)
repeat-y; min-height:200px}
#kiri .isi{padding:10px 10px;}
#kanan .isi{padding:10px 5px;}

```

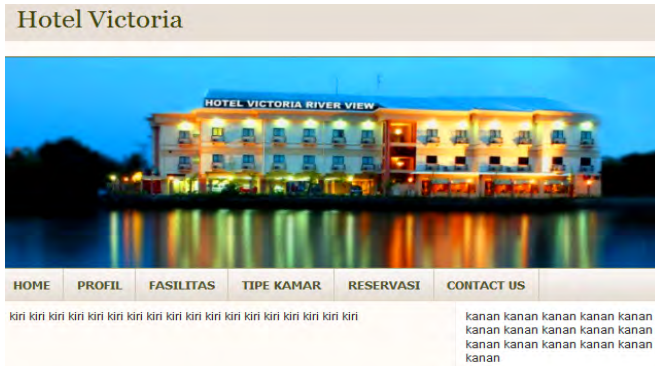


12. Tambahkan code pada ID footer sebagai berikut :

```

#footer{clear:both;
background: #332;
color: #EED;
padding: 10px;
}

```

13. Tambahkan pada file HTML perintah berikut ini

```

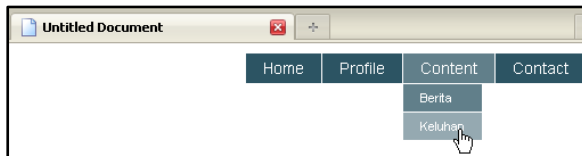
<div id="kanan">
  <div class="isi">kanan kanan kanan kanan kanan kanan kanan
  kanan kanan kanan kanan kanan kanan kanan kanan
  </div>
</div>
<div class="bottom">
  <h2>Galery</h2>
  <table>
  <tr>
    <td></td>
    <td style="width:5px"></td>
    <td></td>
    <td style="width:5px"></td>
    <td></td>
    <td style="width:5px"></td>
    <td></td>
    <td style="width:5px"></td>
    <td></td>
    <td style="width:5px"></td>
    <td></td>
    <td style="width:5px"></td>
  </tr>
  </table>
</div>
<div id="footer">Copyright &copy; 2011 <a href="#">hotelvictoria_bjm.com</a></div>
</div>
</div>
</body>
</html>

```

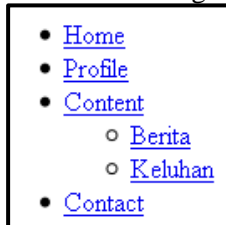
14. Tambahkan Class bottom pada file kasus1.css

```
.bottom {  
    background: #DCDAC9 URL(img/bottom.gif) no-repeat left bottom;  
    clear: both;  
    border-top: 1px solid #AA8;  
    color: #444;  
    padding: 14px 20px;  
}
```

7.3.3. Drop Down Menu



Langkah: Buatlah list item file HTML sebagai berikut:



Buatlah file menu.CSS dan lakukan link ke file HTML yang sudah dibuat di atas.

a. Ketik coding berikut :

```
ul {  
    font-family: arial, Verdana;  
    font-size: 14px;  
    margin: 0;  
    padding: 0;  
    list-style: none;  
}
```

Jalankan pada browser



b. Ketik coding berikut :

```
ul li {display:block;
      position:relative;
      float:left;}
```

Jalankan pada browser



c. Ketik coding berikut :

```
li ul {display:none;}
```

Jalankan pada browser



d. Ketik coding berikut :

```
ul li a {display:block;
        border-top:1px solid #ffffff;
        padding:5px 15px 5px 15px;
        background:#2c5463;
        text-decoration:none;
        color:#FFFFFF;
        margin-left:1px;
      }
```



Jalankan pada browser, apa fungsi margin-left, padding, text-decoration tersebut?

e. Ketik coding berikut :

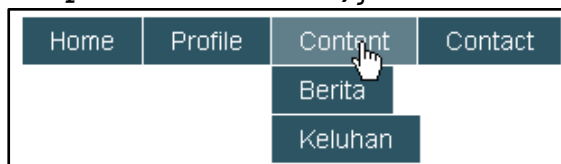
```
ul li a:hover { background:#617F8A;}
```



Jalankan pada browser

f. Ketik coding berikut :

```
li:hover ul{display:block;
            position:absolute;}
```



Jalankan pada browser

g. Ketik coding berikut :

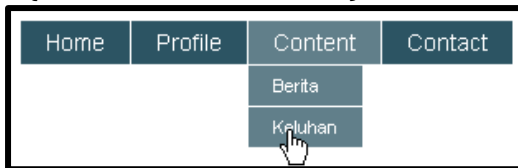
```
li:hover li { float:none;
font-size:11px;}
```



Jalankan pada browser

h. Ketik coding berikut :

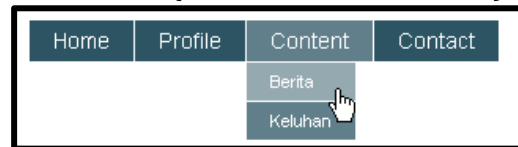
```
li:hover a{background:#617F8A;}
```



Jalankan pada browser

i. Ketik coding berikut :

```
li:hover li a:hover{background:#95A9B1;}
```



Jalankan pada browser

7.4. Rangkuman

- Pada coding di file CSS berikut :

```
#page{ background: #332 URL(img/wrapper.gif) no-repeat;
margin: 0 auto;
width: 762px;}
```

Pengaturan margin dimaksudkan agar pada saat layar browser diperbesar dan diperkecil posisi *layout* tetap ditengah.

- Pada coding di file CSS berikut :

```
#kiri{float:left; width:500px}
#kanan{float:right; width:250px}
```

Pengaturan ID Kiri dimaksudkan agar seluruh isi pada block content kiri dipaksa diimbangkan ke posisi kiri dan dengan lebar 500px, sementara untuk ID kanan dipaksa diimbangkan ke posisi kanan

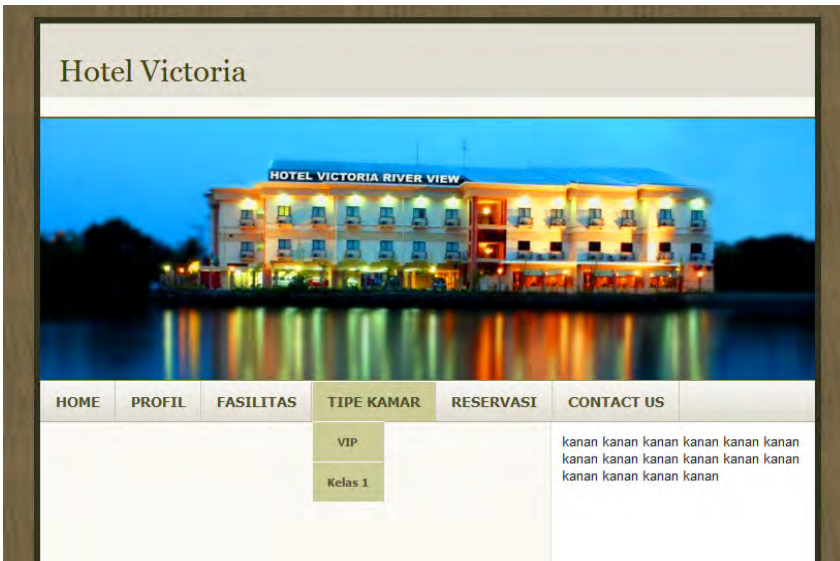
- Pada coding di file CSS berikut :

```
#footer{clear:both;}
```

Pengaturan ID footer dimaksudkan agar seluruh isi footer tidak terpengaruh oleh floating dari ID sebelumnya, yaitu ID kiri dan ID kanan.

- Position relative yaitu posisi suatu elemen akan relative terhadap eemen lainnya. Sebagai contoh, jika elemen dengan property position : relative berada disebelah kiri elemen gambar, kemudian diatur koordinat top dan left : 10px, maka elemen tersebut akan digeser dari arah atas dan kiri sejauh 10px dari posisi gambar. Sedangkan posisi absolute akan mengabaikan posisi elemen lainnya dan menggunakan sumbu X dan Y jendela browser sebagai koordinatnya.

7.5. Latihan



Buatlah codingnya dari gambar di atas, dengan cara menggabungkan kasus1 dengan drop dwon menu, sehingga pada kasus 1 memiliki navigasi dengan dropdown menu

BAB VIII

PENGATURAN HALAMAN WEB DENGAN DASAR CSS (BAGIAN 2)

8.1. Tujuan Instruksional Umum

Memahami cara menggunakan perintah CSS dalam membangun halaman WEB.

8.2. Tujuan Instruksional Khusus

1. Memahami dan menggunakan dasar cara penulisan CSS untuk membangun sebuah *layout* web.
2. Memahami dasar perintah CSS untuk mengatur pewarnaan, dan tipografi halaman WEB.

8.3. Uraian Materi

8.3.1. Kasus 1

Pada pembahasan kasus CSS berikut ini, ditekankan pada pembahasan pengaturan tipografi suatu halaman web sederhana, dan penggunaan position absolute untuk pengaturan tata letak navigasi menu.

- [Home page](#)
- [Renungan](#)
- [Kotaku](#)
- [Link-link](#)

Halaman dengan style saya yang pertama

Selamat datang ke halaman dengan style saya!

Halaman ini tidak ada gambar, tetapi paling tidak memiliki style. Dan juga memiliki link, meskipun mereka tidak pergi kemana-mana...

Disini seharusnya ada lebih banyak lagi, tetapi saya belum tahu apa.

*Dibuat 5 April 2004
oleh diriku.*

Untuk coding HTML nya adalah sebagai berikut :

```
<html>
<head>
  <title>Halaman dengan style saya yang pertama</title>
</head>
<body>
<ul class="navbar">
  <li><a href="#">Home page</a></li>
  <li><a href="#">Renungan</a></li>
  <li><a href="#">Kotaku</a></li>
  <li><a href="#">Link-link</a></li>
</ul>
<h1>Halaman dengan style saya yang pertama</h1>
<p>Selamat datang ke halaman dengan style saya</p>
<p>Halaman ini tidak ada gambar, tetapi paling tidak
memiliki style. Dan juga <br>
memiliki link, meskipun mereka tidak pergi
kemana-mana...</p>
<p>Disini seharusnya ada lebih banyak lagi,
tetapi saya belum tahu apa</p>
<p>
<address>Dibuat 5 April 2004<br>
oleh diriku</address>
</p>
</body>
</html>
```

Kemudian ketiklah code CSS dari kalimat yang berwarna merah

```
<HTML>
<head>
  <title>Halaman dengan style saya yang
pertama</title>
  <style type="text/css">
    body {
      color: purple;
      background-color: #d8da3d }
  </style>
</head>
```

```
<body>
```

Dan seterusnya...



Kemudian ketiklah code CSS dari kalimat yang berwarna merah

```
<HTML>
<head>
  <title>Halaman dengan style saya yang
pertama</title>
  <style type="text/css">
    body {
      font-family: Georgia, "Times New Roman",
        Times, serif;
      color: purple;
      background-color: #d8da3d }
    h1 {
      font-family: Helvetica, Geneva, Arial,
        SunSans-Regular, sans-serif }
  </style>
</head>
```

<body>

Dan seterusnya...



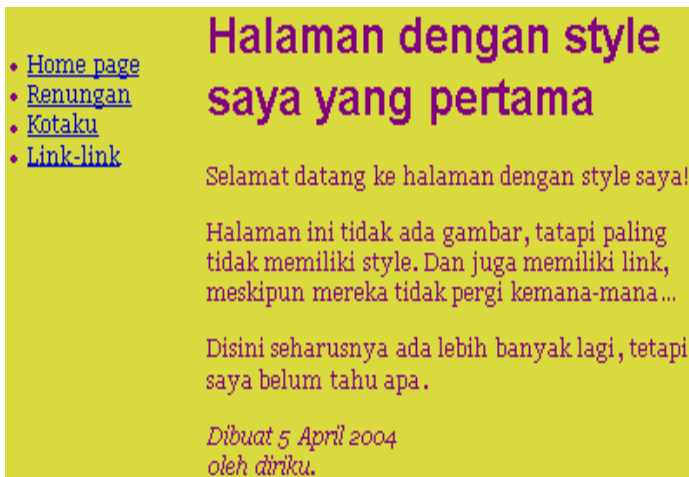
Kemudian ketiklah code CSS dari kalimat yang berwarna merah

```
<HTML>
<head>
  <title>Halaman dengan style saya yang
pertama</title>
  <style type="text/css">
  body {
    padding-left:11em;
    font-family: Georgia, "Times New Roman",
      Times, serif;
    color: purple;
    background-color: #d8da3d }

    ul.navbar {
      position: absolute;
      top: 2em;
      left: 1em;
      width: 9em }

  h1 {
    font-family: Helvetica, Geneva, Arial,
      SunSans-Regular, sans-serif }
  </style>
</head>

<body>
[dsb.]
```



Kemudian ketiklah code CSS dari kalimat yang berwarna merah

```
<HTML>
<head>
  <title>Halaman dengan style saya yang
pertama</title>
  <style type="text/css">
  body {
    padding-left: 11em;
    font-family: Georgia, "Times New Roman",
      Times, serif;
    color: purple;
    background-color: #d8da3d }
  ul.navbar {
    list-style-type: none;
    padding: 0;
    margin: 0;
    position: absolute;
    top: 2em;
    left: 1em;
    width: 9em }
  h1 {
    font-family: Helvetica, Geneva, Arial,
      SunSans-Regular, sans-serif }

  ul.navbar li {
    background: white;
    margin: 0.5em 0;
    padding-left: 12px;
    border-right: 1em solid black }

  ul.navbar a {
    text-decoration: none }
  a:link {
    color: blue }
  a:visited {
    color: purple }
  a:hover {color:yellow}
  ul.navbar a:hover{color:red}
  </style>
</head>
<body>
[dsb.]
```

Kemudian ketiklah code CSS dari kalimat yang berwarna merah

```
<HTML>
<head>
  <title>Halaman dengan style saya yang
pertama</title>
  <style type="text/css">
    body {
      padding-left: 11em;
      font-family: Georgia, "Times New Roman",
        Times, serif;
      color: purple;
      background-color: #d8da3d }
    ul.navbar {
      list-style-type: none;
      padding: 0;
      margin: 0;
      position: absolute;
      top: 2em;
      left: 1em;
      width: 9em }
    h1 {
      font-family: Helvetica, Geneva, Arial,
        SunSans-Regular, sans-serif }
    ul.navbar li {
      background: white;
      margin: 0.5em 0;
      padding: 0.3em;
      border-right: 1em solid black }
    ul.navbar a {
      text-decoration: none }
    a:link {
      color: blue }
    a:visited {
      color: purple }
    address {
      margin-top: 1em;
      padding-top: 1em;
      border-top: thin dotted }
  </style>
</head>

<body>
[dsb.]
```



8.3.2. Kasus 2

Konsep Desain:



Buatlah design awal tampilan sebagai berikut



```
<HTML>
<head>
<title>Untitled Document</title>
</head>
<body>
  <div id="page">
    <div id="Header">
      .....isi dari Header.....
    </div>
    <div id="content-primary">
      .....isi dari content-primary.....
    </div>
    <div id="content-secondary">
      .....isi dari content-secondary.....
    </div>
    <div id="footer">
      .....isi dari footer.....
    </div>
  </div>
</body>
</HTML>
```

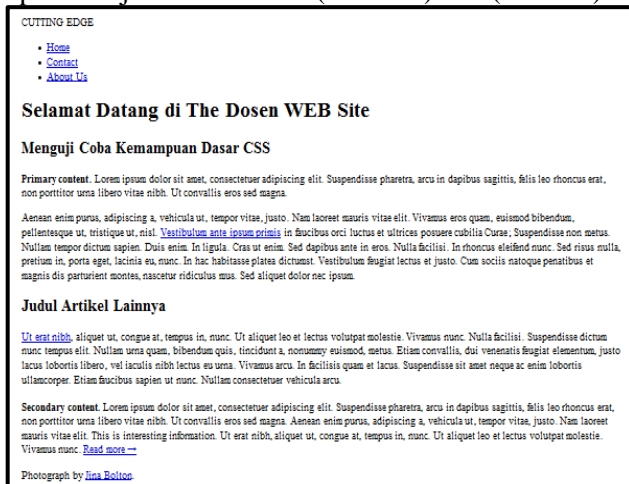
Buatlah file baru dengan format ekstensi .css

a. Ketik coding berikut :

```
#page { width:960px;
        Margin:0 auto;
        text-align:left;}
```

Ujikan :

- file CSS yang baru dibuat, kemudian panggil dari file HTML yang sudah anda buat sebelumnya.
- Jalankan pada browser, kemudian Uji dengan memperlebar dan memperkecil jendela browser (CTRL +) dan (CTRL -)



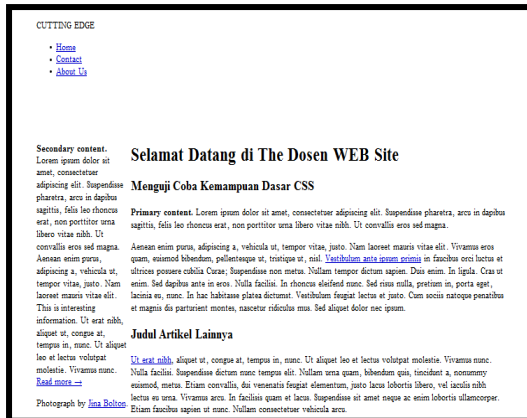
- b. Ketik coding berikut :
- ```
#header {height:192px;}
```
- Jalankan pada browser,



c. Ketik coding berikut :

```
#content-primary {width:768px;
float:right;}
```

Jalankan pada browser



d. Ketik coding berikut :

```
#content-secondary {width: 192px;
float:right;}
```

Jalankan pada browser

e. Ketik coding berikut :

```
#footer {clear:right;}
```

Jalankan pada browser, apa fungsi Clear?

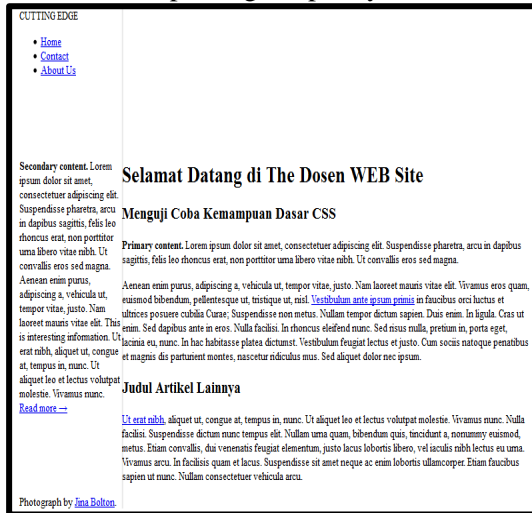


f. Tambahkan coding yang bercetak tebal berikut :

```
#page {text-align:left;
width:968px;
margin: 0 auto;}
```

```
background:URL(image/page.gif) repeat-y;}
```

Jalankan pada browser, apa fungsi repeat-y?



g. Ketik coding berikut :

```
body {color:#333;}
```

Jalankan pada browser

h. Tambahkan coding yang bercetak tebal berikut :

```
#header {height:192px;
background:#829F60}
```

Jalankan pada browser

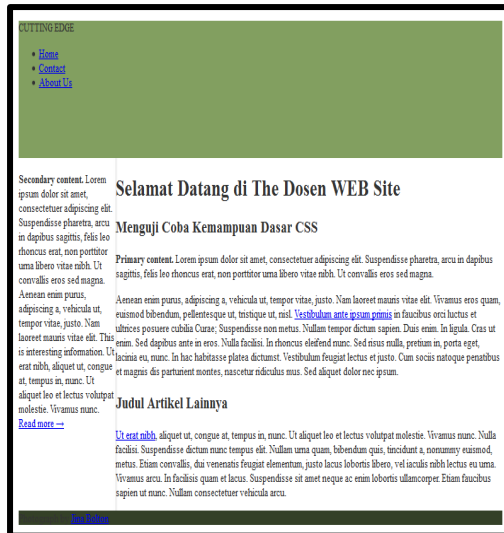




- i. Tambahkan coding yang bercetak tebal berikut :

```
#footer {clear:right;
background:#344027
```

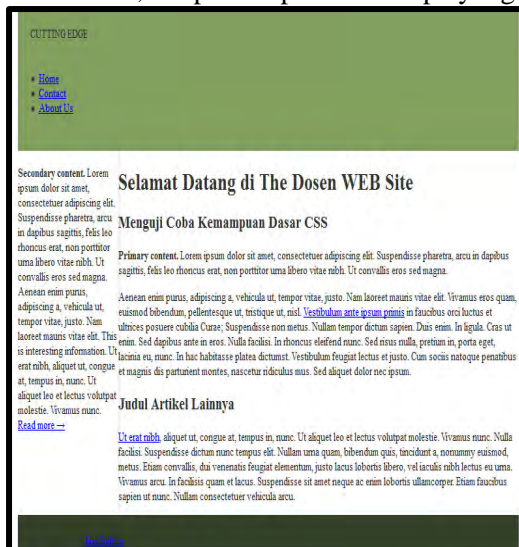
Jalankan pada browser



- j. Ketik coding berikut :

```
#header p {padding:24px;}
#footer p {padding:24px;}
```

Jalankan pada browser, simpulkan perubahan apa yang terjadi?



k. Tambahkan coding yang bercetak tebal berikut :

```
#content-primary {width: 768px; float:right; margin:24px;}
```

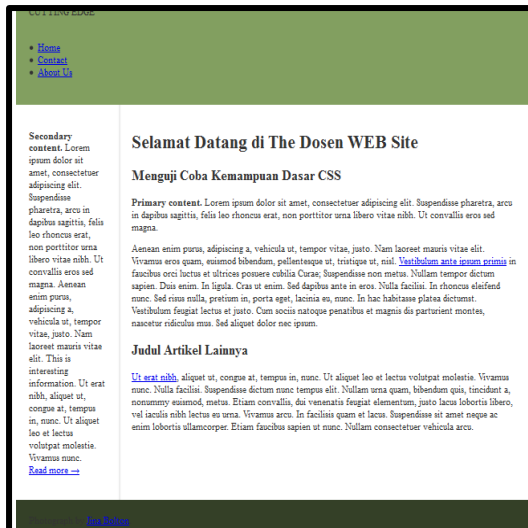
Jalankan pada browser, apa yang terjadi?, **Ganti width menjadi 720px**, simpulkan apa yang terjadi mengapa bisa sampai begitu?



l. Tambahkan coding yang bercetak tebal berikut dan ganti ukuran width:

```
#content-secondary { width: 144px; float:right; margin:24px;}
```

Jalankan pada browser.



Tambahkan gambar berikut , pada file HTML:

**Selamat Datang di The Dosen website**

## Menguji coba kemampuan dasar CSS

**Primary content.** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse convallis eros sed magna.



Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet ma [ipsum primis](#) in faucibus orci luctus et ultrices posuere cubilia Curae; Suspendisse Nulla facilisi. In rhoncus eleifend nunc. Sed risus nulla, pretium in, porta eget, lac penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed aliquet da

Menjadi :

CUTTING EDGE

- » [Home](#)
- » [Contact](#)
- » [About Us](#)

**Secondary content.** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse pharetra, arcu in dapibus sagittis, felis leo rhoncus erat, non portitor urna libero vitae nibh. Ut convallis eros sed magna. Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet maucis vitae est. This is interesting information. Ut erat nibh, aliquet ut, congue at, tempus in, suso. Ut aliquet leo et lectus volutpat molestie. Vivamus suso. [Read more ->](#)

## Selamat Datang di The Dosen WEB Site

### Menguji Coba Kemampuan Dasar CSS

**Primary content.** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse pharetra, arcu in dapibus sagittis, felis leo rhoncus erat, non portitor urna libero vitae nibh. Ut convallis eros sed magna.



Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet maucis vitae est. Vivamus eros quam, eximium bibendum, pellentesque ut, tristique ut, nisi. [Vestibulum ante ipsum primis](#) in faucibus orci luctus et ultrices posuere cubilia Curae; Suspendisse suso metus. Nullam tempus dictum sapien. Duis enim. In ligula. Cras ut enim. Sed dapibus ante in eros. Nulla facilisi. In rhoncus eleifend suso. Sed risus nulla, pretium in, porta eget, lacinia eu, suso. In hac habitasse platea dictumst. Vestibulum feugiat lectus et justo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed aliquet dolor suso ipsum.

### Judul Artikel Lainnya

[Ut erat nibh](#), aliquet ut, congue at, tempus in, suso. Ut aliquet leo et lectus volutpat molestie. Vivamus suso. Nulla facilisi. Suspendisse dictum suso tempus est. Nullam urna quam, bibendum quis, necididunt a, nonummy susumod, suso. Etiam convallis, dai venenatis feugiat elementum, justo lacus lobortis libero, vel lacula nibh lectus eu suso. Vivamus arcu. In facilisis quam et lectus. Suspendisse sit amet neque ac enim lobortis ullamcorper. Etiam faucibus sapien ut suso. Nullam consectetur vehicula arcu.

[Read more](#)

m. Ketik coding berikut :

```
#header ul{padding-top:62px;
padding-right:12px;
text-align:right;
list-style:none;
}
```

Jalankan pada browser, apa fungsi list-style:none dan text-align:right?



n. Ketik coding berikut :

```
#header li{display:inline;
margin:0 12px;}
```

Jalankan pada browser, apa fungsi display:inline pada coding tersebut?

CUTTING EDGE

[Home](#) [Contact](#) [About Us](#)

Secondary context. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse pharetra, arcu in dapibus sagittis, felis leo rhoncus erat, non porttitor urna libero vitae nibh. Ut convallis eros sed magna. Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet mauris vitae elit. This is interesting information. Ut erat nibh, aliquet ut, congue at, tempus in, musc. Ut aliquet leo et lectus volutpat molestie. Vivamus musc. [Read more →](#)

## Selamat Datang di The Dosen WEB Site

### Menguji Coba Kemampuan Dasar CSS

Primary context. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse pharetra, arcu in dapibus sagittis, felis leo rhoncus erat, non porttitor urna libero vitae nibh. Ut convallis eros sed magna.



Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet mauris vitae elit. Vivamus eros quam, euismod bibendum, pellentesque ut, tristique ut, nisl. [Vestibulum ante ipsum primis](#) in faucibus orci luctus et ultrices posuere cubilia Curae; Suspendisse non metus. Nullam tempor dictum sapien. Duis enim. In ligula. Cras ut enim. Sed dapibus ante in eros. Nulla facilisi. In rhoncus eleifend musc. Sed rarius nulla, pretium in, porta eget, lacinia eu, musc. In hac habitasse platea dictumst. Vestibulum feugiat lectus et justo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed aliquet dolor nec ipsum.

### Judul Artikel Lainnya

[Ut erat nibh](#), aliquet ut, congue at, tempus in, musc. Ut aliquet leo et lectus volutpat molestie. Vivamus musc. Nulla facilisi. Suspendisse dictum musc tempus elit. Nullam urna quam, bibendum quis, tincidunt a, nonummy euismod, metus. Etiam convallis, dui venenatis feugiat elementum, justo lacus lobortis libero, vel iaculis nibh lectus eu urna. Vivamus arcu. In facilisis quam et lacus. Suspendisse sit amet neque ac enim lobortis ullamcorper. Etiam faucibus sapien ut musc. Nullam consectetur vehicula arcu.

[Jwa Soltra](#)

o. Ketik coding berikut :

```
#header li a{text-decoration:none;
 color:#fff;
 font-size:1.25em;
 line-height:1em;}
```

Jalankan pada browser, apa fungsi text-decoration:none dan line-height?

CUTTING EDGE


Home Contact About Us

Secondary content. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse pharetra, arcu in dapibus sagittis, felis leo rhoncus erat, non porttitor urna libero vitae nibh. Ut conwallis eros sed magna. Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet mauris vitae elit. This is interesting information. Ut erat nibh, aliquet ut, congue at, tempus in, musc. Ut aliquet leo et lectus volutpat molestie. Vivamus musc. [Read more →](#)

### Selamat Datang di The Dosen WEB Site

#### Menguji Coba Kemampuan Dasar CSS

Primary content. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse pharetra, arcu in dapibus sagittis, felis leo rhoncus erat, non porttitor urna libero vitae nibh. Ut conwallis eros sed magna.



Aenean enim purus, adipiscing a, vehicula ut, tempor vitae, justo. Nam laoreet mauris vitae elit. Vivamus eros quam, euismod bibendum, pellentesque ut, tristique ut, nisi. [Vestibulum ante ipsum primis](#) in faucibus orci luctus et ultrices posuere cubilia Curae; Suspendisse non metus. Nullam tempor dictum sapien. Duis enim. In ligula. Cras ut enim. Sed dapibus ante in eros. Nulla facilisi. In rhoncus eleifend musc. Sed risus nulla, pretium in, porta eget, lacinia eu, musc. In hac habitasse platea dictumst. Vestibulum feugiat lectus et justo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed aliquet dolor nec ipsum.

#### Judul Artikel Lainnya

[Ut erat nibh](#), aliquet ut, congue at, tempus in, musc. Ut aliquet leo et lectus volutpat molestie. Vivamus musc. Nulla facilisi. Suspendisse dictum musc tempus elit. Nullam urna quam, bibendum quis, tincidunt a, nonummy euismod, metus. Etiam conwallis, dui venenatis feugiat elementum, justo lacus lobortis libero, vel iaculis nibh lectus eu urna. Vivamus arcu. In facilisis quam et lacus. Suspendisse sit amet neque ac enim lobortis ullamcoeper. Etiam faucibus sapien ut musc. Nullam consectetur vehicula arcu.

[Iina Rofiq](#)

p. Ketik coding berikut :

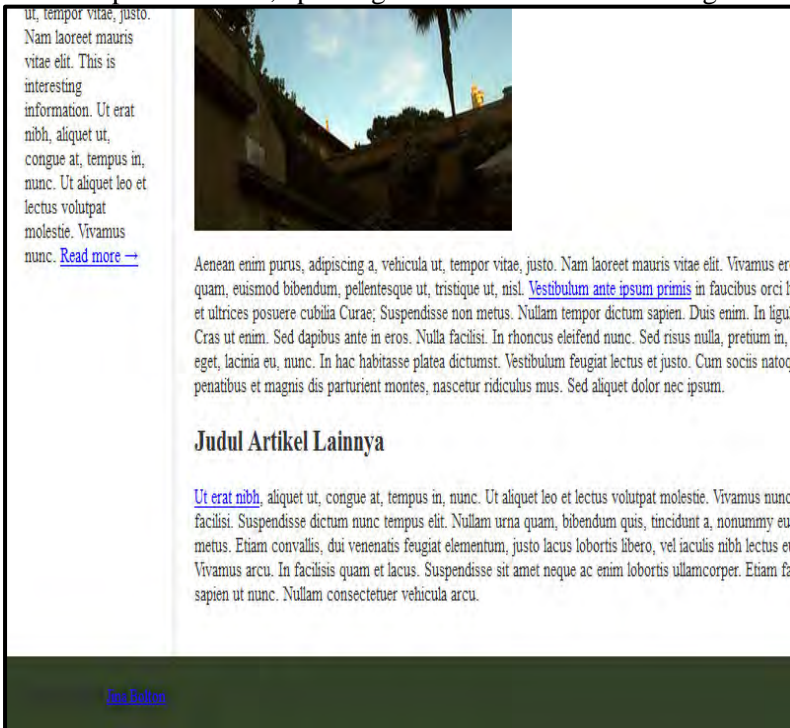
```
#header li a:hover{color:#300;}
```

Jalankan pada browser, arahkan kursor ke atas menu Home, apa fungsi hover ?

- q. Tambahkan coding yang bercetak tebal berikut :

```
#footer {
 clear:right;
 background:#344027;
 height:192px;
 font-size:14px;
 text-shadow:1px 1px 3px #333;
 line-height:1em;
 color:#fff;
}
```

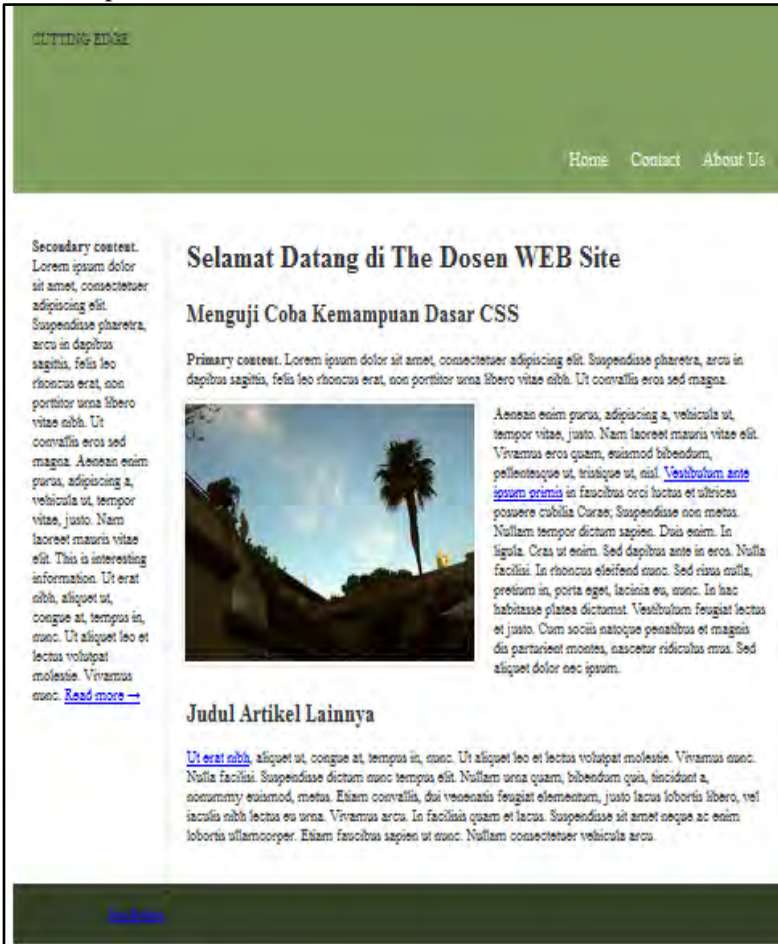
Jalankan pada browser, apa fungsi text-shadow dan line-heigh?



- r. Ketik coding berikut :

```
#content-primary img {float:left;
margin:-6px 18px 0 -6px;
border:3px solid #eee;
background:#fff;
padding:3px;}
```

Jalankan pada browser



s. Ketik coding berikut :

```
#content-secondary p {padding-bottom: 18px;}
```

Jalankan pada browser

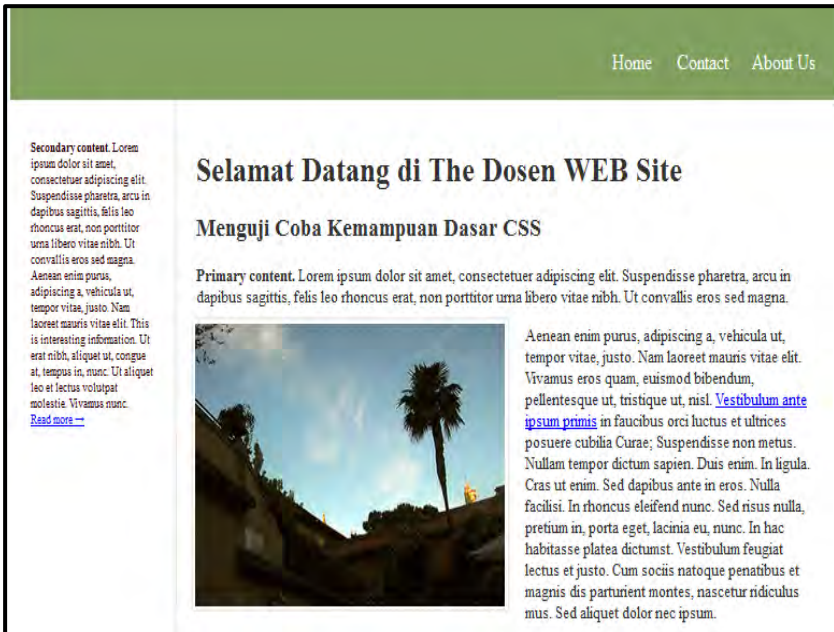
t. Tambahkan coding yang bercetak tebal berikut :

```
#content-secondary {
 float: right;
 margin: 24px;
 width: 144px;
 font-size: 11px;
 color: #300;}

```

Jalankan pada browser





u. Ketik coding berikut :

```

h1 {
 margin-bottom: 11px;
 padding-top: 15px;
 padding-bottom: 15px;
 border-bottom: 1px solid #ccc;
 color: #663;
 font-family: Georgia, serif;
 font-size: 1.5em;
 font-style: italic;
 line-height: 18px;
 font-weight: normal;
 font-variant: normal;
}
h2, h3 {
 font-weight: normal;
 font-size: 1em;
 line-height: 18px;
}
h2 {
 padding-top: 10px;
 padding-bottom: 20px;
 font-size: 1.25em;
}

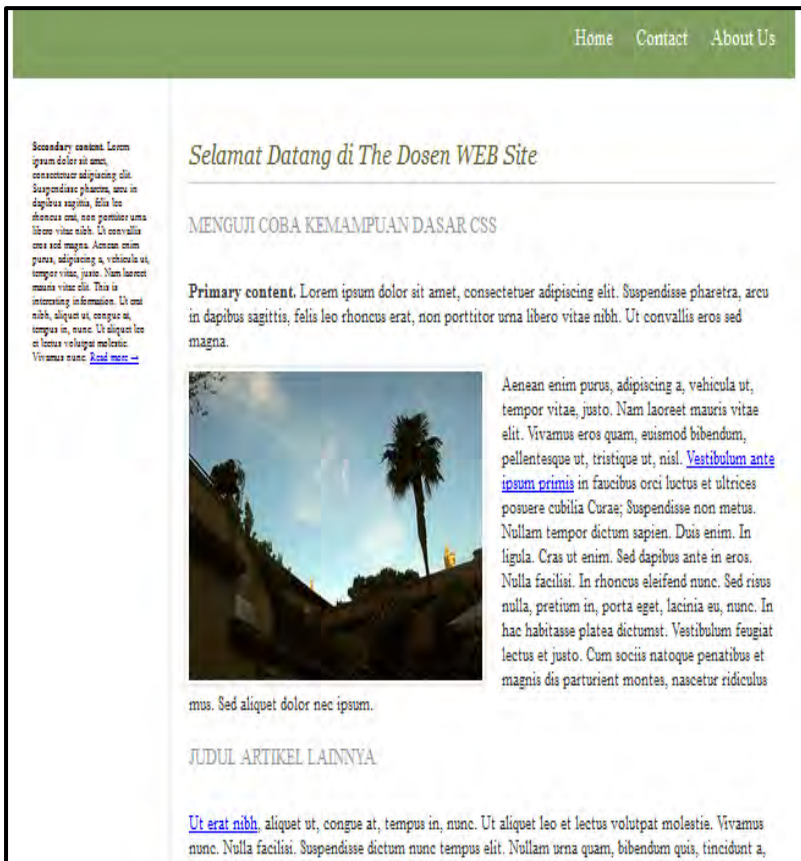
```

```

text-transform: uppercase;
color: #999;
}
h3 {
padding-bottom: 6px;
font-weight: bold;
text-transform: uppercase;
color: #633;
}

```

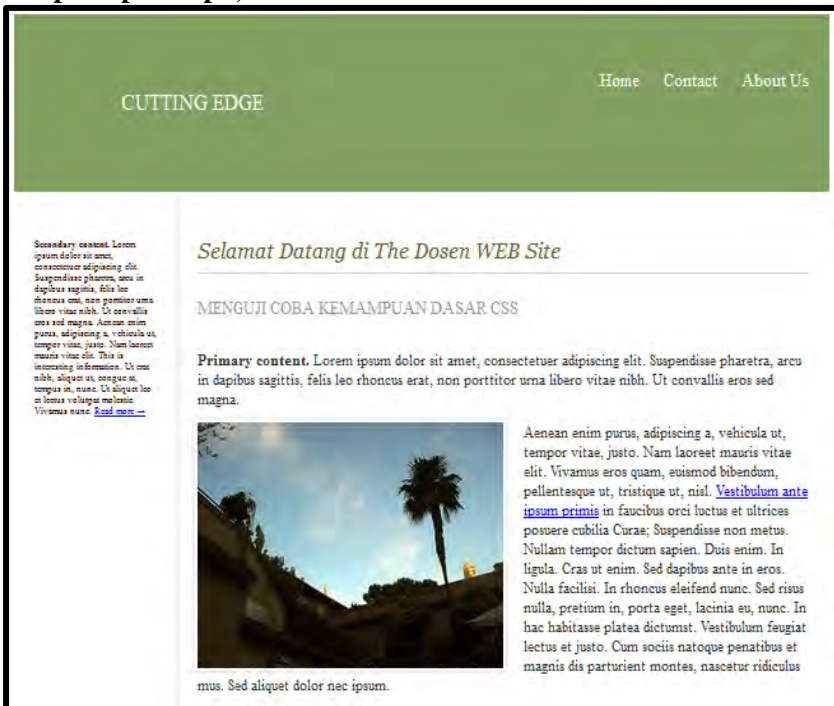
Jalankan pada browser, apa fungsi **border-bottom: 1px solid #ccc;** pada **h1**, **padding-top:10px** dan **text-transform:uppercase** pada **h2**, dan **padding-bottom: 6px;** pada **h3**?



v. Ganti #header p dengan coding berikut :

```
#header p { float: left;
 font-size: 1.5em;
 line-height: 1em;
 color: #fff;
 padding: 60px 24px 24px 126px;
 }
```

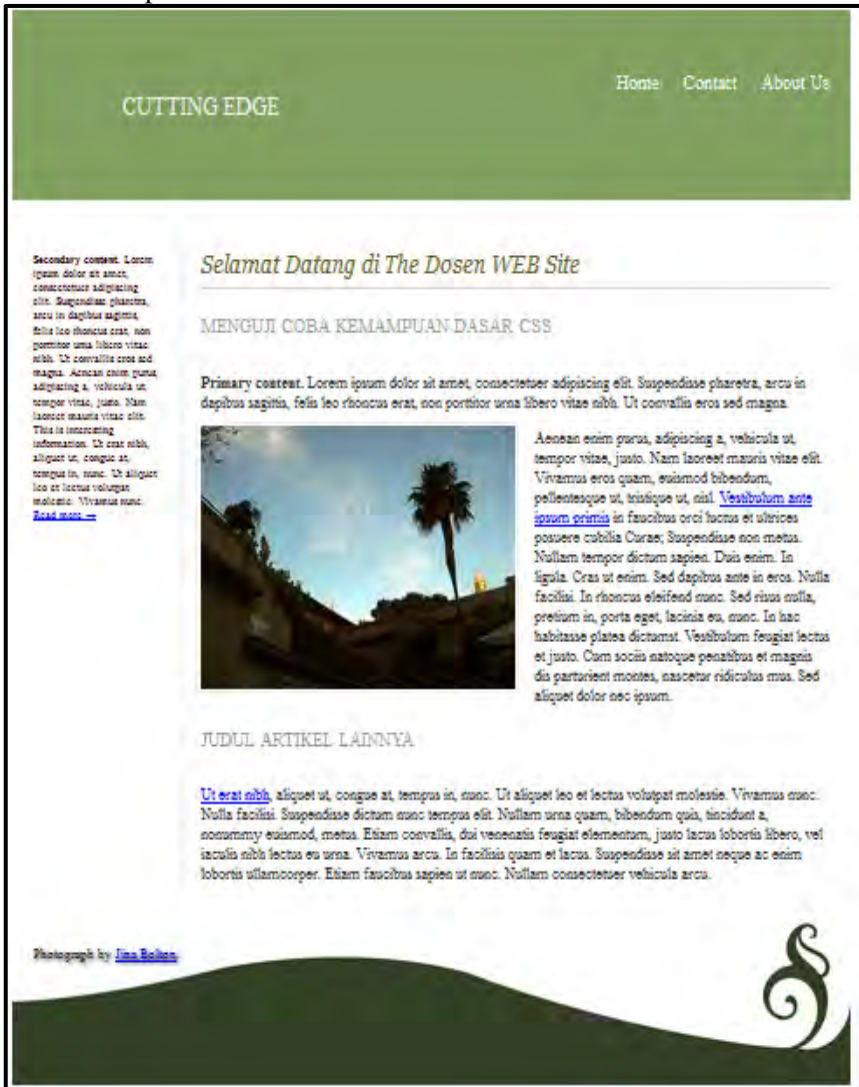
Jalankan pada browser, apa fungsi **float: left;** dan **padding: 60px 24px 24px 126px;**?



w. Tambahkan coding yang bercetak tebal berikut:

```
#footer {
 clear: right;
 height: 192px;
 font-size: 10px;
 line-height: 1em;
 text-shadow: 1px 1px 1px #333;
 color: #FFF;
 background-color: #344027;
 background-image: URL(footer.gif);
 background-repeat: no-repeat;
 background-position: top;
}
```

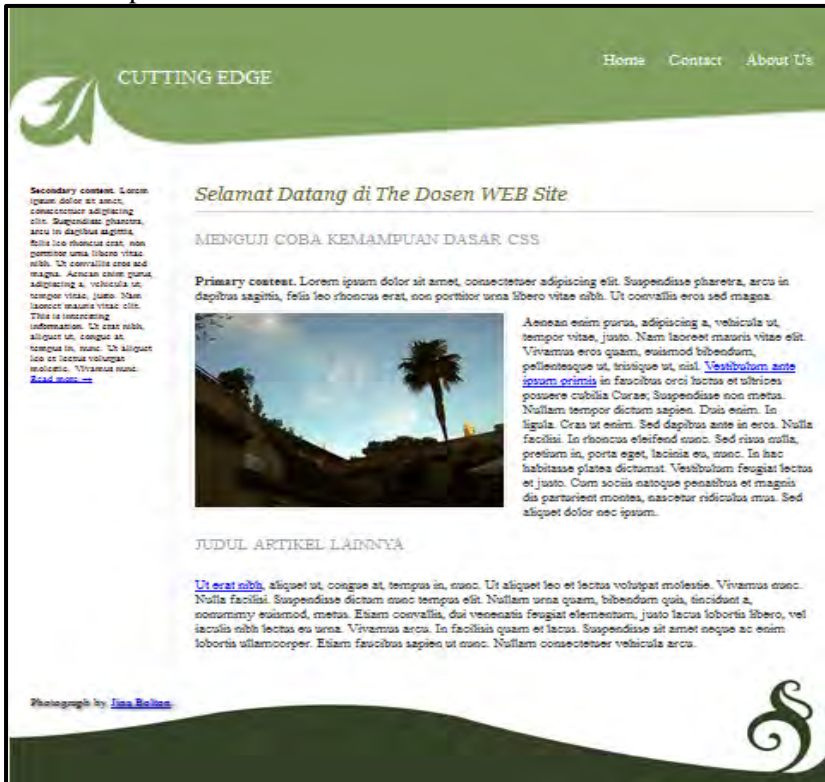
Jalankan pada browser



x. Tambahkan coding berikut :

```
#header {
 height: 192px;
 background-color: #829F60;
 background-image: URL(header.gif);
 background-repeat: no-repeat;
 background-position: bottom;}
```

Jalankan pada browser



## 8.4. Rangkuman

- Pada kasus 1 coding di file CSS berikut :

```
ul.navbar {
 position: absolute;
 top: 2em;
 left: 1em;
 width: 9em }
```

pengaturan, posisi absolute di atas akan mengabaikan posisi elemen lainnya dan menggunakan sumbu X dan Y jendela browser sebagai koordinatnya. Untuk pengaturan sumbu Y perintah **top:2em** akan mengatur jarak 2 em dari koordinat atas browser, dan sumbu X digunakan **left: 1 em** akan mengatur jarak 1 em dari koordinat sebelah kiri browser.

- Pada kasus 2 coding di file CSS berikut :

```
#header li{display:inline;
margin:0 12px;}
```

pengaturan, display: inline akan membuat posisi kalimat dalam 1<sup>st</sup> index menjadi horizontal (satu baris). Pengaturan margin : 0 12px akan berpengaruh pada jarak antar list index pada posisi kanan dan kiri berjarak 12px.

## 8.5. Latihan



Buatlah codingnya dari gambar di atas, dengan cara membuat halaman tersebut menjadi 3 kolom. Namun untuk Footer tetap seperti tampilan pada kasus 2.

## DAFTAR PUSTAKA

---

- Jayan, CSS Untuk Orang Awam, Maxicom, 2002.
- Hakim lukmanul, Membongkar Trik Rahasia Para master PHP, Lokomedia, 2008.
- PENS ITS, Modul Pemrograman WEB, Surabaya 2007.
- Solichin Achmad, S.Kom, Pemrograman Web dengan PHP dan MySQL, Jakarta, 2008.
- Rezal, M. Faisal. 2016. Seri Belajar ASP NET : Membangun Aplikasi Web Mudah & Cepat. INDC: Banjarmasin.
- Rezal, M. Faisal. 2017. Seri Belajar ASP NET : Pemrograman Web ASP.NET Core dengan MySQL dan Visual Studio Code. INDC: Banjarmasin.



# PEMROGRAMAN DASAR BERBASIS WEB

## HERU KARTIKA CANDRA

Pemrograman web dinamis menggunakan teknik pemrograman Server-Side, dimana program dijalankan pada server sedangkan hasilnya dikirimkan ke user, ini berbeda dengan pemrograman statis dimana menggunakan teknik pemrograman Client-Site yang berarti script program dikirimkan ke user dan kemudian hasilnya ditampilkan pada browser user, setelah script terkirim. Pemrograman web statis hanya diperlukan tag-tag HTML dan tidak diperlukan pemrograman, sehingga pada pemrograman statis belum dapatlah dikatakan pemrograman sebenarnya, karena hanya menyusun teks dan <tag> saja, tanpa ada proses aritmatika, penentuan keputusan, perulangan dan sebagainya. Dikarenakan pemrograman web statis adalah dasar dari pemrograman web dinamis, maka akan diberikan beberapa <tag> yang nantinya diperlukan untuk pemrograman web statis dan beberapa <tag> tambahan.



Penerbit Poliban Press

Redaksi :

Politeknik Negeri Banjarmasin, Jl. Brigjen H. Hasan Basry,  
Pangeran, Komp. Kampus ULM, Banjarmasin Utara

Telp : (0511)3305052

Email : [press@poliban.ac.id](mailto:press@poliban.ac.id)

ISBN 978-623-5259-09-3 (PDF)

