



## Overview

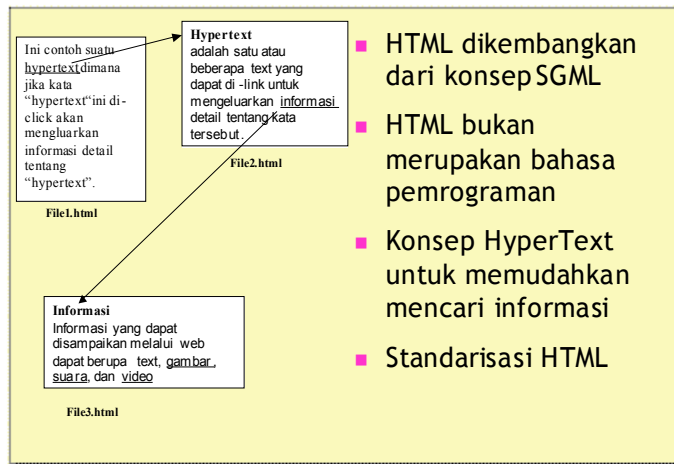
- Pengenalan HTML
- Struktur Dasar sebuah Web Page
- Membuat Web Page Sederhana
- Penambahan Link
- Penambahan Gambar atau Citra

### Tujuan :

Pada akhir modul ini diharapkan peserta training dapat :

1. Mendefinisikan pengertian file HTML (*HyperText Mark up Language*)
2. Memahami konsep *hypertext*
3. Menjelaskan struktur dasar suatu file HTML
4. Mengetahui beberapa tag HTML dasar yang sering digunakan
5. Membuat web *page* sederhana
6. Memahami penggunaan fasilitas link dan penyisipan citra pada file HTML.

# Pengenalan HTML



HTML (*HyperText Mark up Language*) merupakan suatu metoda untuk mengimplementasikan konsep hypertext dalam suatu naskah atau dokumen. HTML sendiri bukan tergolong pada suatu bahasa pemrograman karena sifatnya yang hanya memberikan tanda (*marking up*) pada suatu naskah teks dan bukan sebagai program.

Berdasarkan kata-kata penyusunnya HTML dapat diartikan lebih dalam lagi menjadi :

## Hypertext

Link *hypertext* adalah kata atau frase yang dapat menunjukkan hubungan suatu naskah dokumen dengan naskah-naskah lainnya. Jika kita klik pada kata atau frase untuk mengikuti link ini maka web browser akan memindahkan tampilan pada bagian lain dari naskah atau dokumen yang kita tuju.

## Markup

Pada pengertiannya di sini markup menunjukkan bahwa pada file HTML berisi suatu intruksi tertentu yang dapat memberikan suatu format pada dokumen yang akan ditampilkan pada *World Wide Web*.

## Language

Meski HTML sendiri bukan merupakan bahasa pemrograman, HTML merupakan kumpulan dari beberapa instruksi yang dapat digunakan untuk mengubah-ubah format suatu naskah atau dokumen.

Pada awalnya HTML dikembangkan sebagai subset SGML (Standard Generalized Mark-up Language). Karena HTML didedikasikan untuk ditransmisikan melalui media Internet, maka HTML relatif lebih sederhana daripada SGML yang lebih ditekankan pada format dokumen yang berorientasi pada aplikasi.

## Apa yang dapat dilakukan dengan HTML ?

- Menentukan format suatu teks
- Membuat list tentang sekelompok hal
- Membuat link ke dokumen lain atau bagian lain dari dokumen yang sama.
- Menyisipkan citra atau gambar .
- Menampilkan informasi dalam bentuk tabel

### Memodifikasi format teks

Penggunaan HTML memungkinkan kita untuk memodifikasi tampilan atau format dokumen yang akan kita transmisikan melalui media Internet. Beberapa hal yang dapat dilakukan dalam menentukan format dokumen ini adalah :

- Kita dapat menampilkan suatu kelompok kata dalam beberapa ukuran yang dapat digunakan untuk judul, heading dan sebagainya.
- Kita dapat menampilkan teks dalam bentuk cetakan tebal
- Kita dapat menampilkan sekelompok kata dalam bentuk miring
- Kita dapat menampilkan naskah dalam bentuk huruf yang mirip dengan hasil ketikan mesin ketik
- Kita dapat mengubah-ubah ukuran font untuk suatu karakter tertentu.

### Menampilkan daftar sesuatu dalam bentuk point-point (item)

Dengan HTML kita dapat menampilkan daftar atau deretan informasi dalam bentuk point-point sehingga lebih mudah dibaca dan dipahami

### Membuat link

Konsep *hypertext* pada HTML memungkinkan kita untuk membuat link pada suatu kelompok kata atau frase untuk menuju ke bagian manapun dalam *World Wide Web*. Ada tiga macam link yang dapat kita gunakan :

- Link menuju bagian lain dari *page*
- Link menuju *page* lain dalam satu *web site*
- Link menuju *resource* atau *web site* yang berbeda

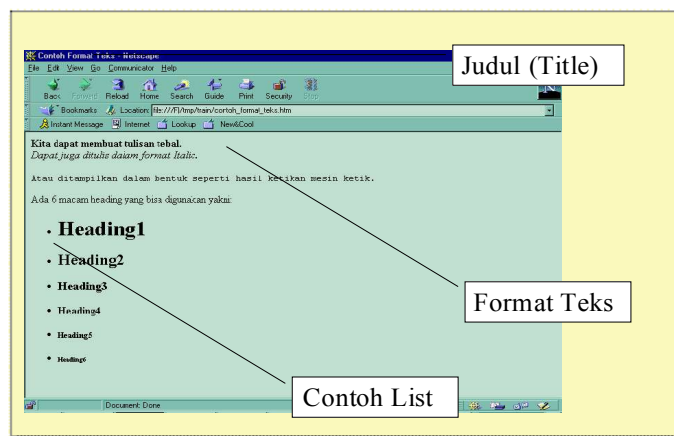
## Menyisipkan citra

Dengan menyisipkan citra maka tampilan *page* kita akan lebih menarik, interaktif dan informatif untuk mendukung data-data lainnya dalam bentuk teks.

## Menampilkan informasi dalam bentuk tabel

Penampilan informasi dalam bentuk tabel ini akan mempermudah pembaca untuk memahami informasi yang kita tawarkan. Penggunaan tabel ini juga dapat dilakukan untuk menambah nilai estetika dari *page* yang akan kita rancang.

## Format Teks

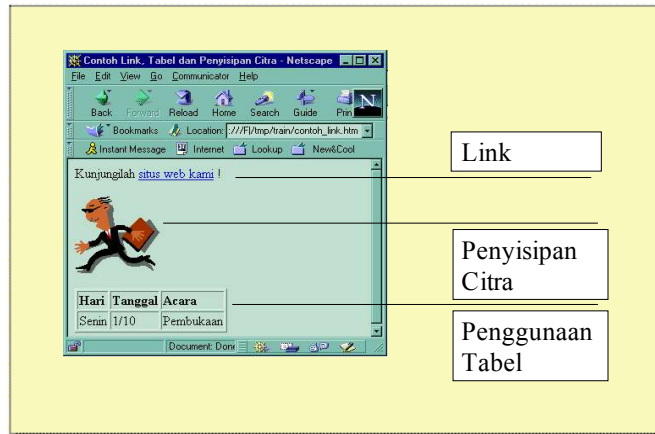


Format Teks yang dapat diimplementasikan dengan menggunakan HTML :

- Kita dapat menampilkan suatu kelompok kata dalam beberapa ukuran yang dapat digunakan untuk judul, heading dan sebagainya.
- Kita dapat menampilkan teks dalam bentuk cetak tebal
- Kita dapat menampilkan sekelompok kata dalam bentuk miring
- Kita dapat menampilkan naskah dalam bentuk huruf yang mirip dengan hasil ketikan mesin ketik

- Kita dapat mengubah-ubah ukuran font untuk suatu karakter tertentu.

## Link dan Penyisipan Citra



### Penambahan link

Penambahan link dimaksudkan agar pengguna dapat mencari informasi yang dibutuhkannya secara cepat tanpa harus terlebih dahulu membaca keseluruhan naskah atau dokumen. Penambahan link ini dapat diimplementasikan pada sekelompok kata, frase atau citra.

### Penyisipan citra

Penyisipan citra atau gambar akan membuat *page* lebih menarik dan interaktif. Informasi yang ditampilkan dalam bentuk gambar relatif lebih informatif daripada yang hanya terdiri dari informasi teks saja. Dalam penyisipan citra, perlu diperhatikan ukuran dan tipe file serta perpaduan warna dengan komponen penyusun *page* lainnya.

## Struktur Dasar Web Page

- Tag HTML
- Tag <HTML> dan </HTML>
- Tag <HEAD> dan </HEAD>
- Tag <BODY> dan </BODY>
- Judul Page (tag <TITLE>)
- Pembuatan paragraf (tag <P>)

File-file HTML dapat ditampilkan sesuai dengan keinginan kita seperti pada contoh-contoh sebelumnya adalah karena terdapat *marker* yang diimplementasikan dalam bentuk tag-tag HTML. Secara umum format tag HTML tersebut dapat dituliskan sebagai berikut:

```
<TAG>Teks yang akan dipengaruhi oleh tag</TAG>
```

Sebagai contoh kalau kita akan membuat tulisan dalam format tebal maka yang harus kita tuliskan dalam file HTML adalah:

```
B>Tulisan yang tercetak tebal</B>
```

Tanda slash (/) menunjukkan akhir dari suatu tag, akan tetapi ada juga beberapa tag HTML yang tidak memerlukan pasangan akhirnya ini.

Tag yang paling dasar yang digunakan dalam file HTML adalah <HTML> dan </HTML>. Kedua tag ini berfungsi untuk mendefinisikan bahwa teks yang terdapat di antara kedua tag tersebut adalah dalam format HTML.

```
<HTML>  
Teks  
</HTML>
```

Untuk mendefinisikan *head*, kita dapat menambahkan tag <HEAD> dan </HEAD> setelah penggunaan tag <HTML> sedemikian hingga struktur *page* menjadi :

```
<HTML>
<HEAD>
</HEAD>
</HTML>
```

Sedangkan bagian *body* adalah tempat dimana kita dapat menempatkan teks dan berbagai aksesoris pendukung lainnya yang akan ditampilkan pada *web page*. Bagian ini dapat didefinisikan dengan meletakkan tag `<BODY>` dan `</BODY>` di antara teks yang akan kita tampilkan. Kedua tag ini diletakkan sesudah tag `</HEAD>`, sehingga struktur dasar sebuah *page* dapat dituliskan sebagai :

```
<HTML>
<HEAD>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Untuk mendefinisikan judul *page* (*title*) maka kita dapat melakukannya dengan meletakkan naskah judul di antara tag `<TITLE>` dan `</TITLE>`. Judul ini akan ditampilkan pada *title bar* web browser. Misalkan kita akan membuat sebuah *page* dengan judul "Latihan HTML" maka kita harus menambahkan:

```
<TITLE>Latihan HTML</TITLE>
```

Perhatikan bahwa tag `<TITLE>` dan `</TITLE>` ini harus diletakkan pada bagian *head*, sehingga dokumen HTML dasar kita menjadi:

```
<HTML>
<HEAD>
<TITLE>Latihan HTML</TITLE>
</HEAD>
<BODY>
Teks yang akan ditampilkan pada bagian body
</BODY>
</HTML>
```

Pada file HTML, karakter carriage returns (Enter) diabaikan, sehingga untuk membuat atau mendefinisikan naskah dalam bentuk paragraf harus ditambahkan tag khusus yakni `<P>`. Sebagai contoh kita dapat menampilkan beberapa paragraf sekaligus dalam satu dokumen HTML.

```
<HTML>
```

```

<HEAD>

<TITLE>Latihan HTML</HTML>

</HEAD>

<BODY>

Teks yang akan ditampilkan pada bagian body

<P>Paragraf satu </P>

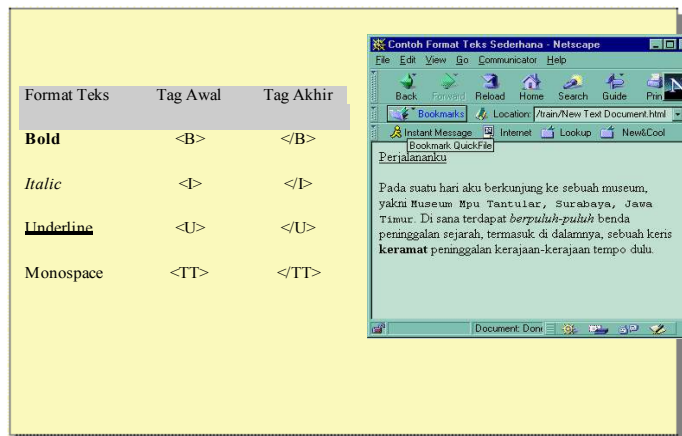
<P>Paragraf dua </P>

</BODY>

</HTML>

```

### Format Teks Dasar



Pada umumnya web browser mendukung empat macam format teks yakni **bold**, *italic*, monospace, and underline. Keempat format teks ini diimplementasikan dalam format HTML dengan menambahkan tag-tag yang bersesuaian untuk masing-masing format tersebut.

Format teks	Tag awal	Tag Akhir
<b>Bold</b>	<B>	</B>
<i>Italic</i>	<I>	</I>
<u>Underline</u>	<U>	</U>
Monospace	<TT>	</TT>



Sebagai contoh kalau kita ingin menggunakan format-format teks ini maka dalam file HTML harus kita tambahkan:

```
<HTML>
<HEAD>
<TITLE>Yer Basic Text Formatting Styles</TITLE>
</HEAD>
<BODY>
<U>My Excellent Bookstore Adventure</U>
<P>
The other day, I went to a unique bookstore called
<TT>Mary, Mary, Quite Contrary</TT>. There were
<I>tons</I> of unexpected delights, including, believe
it or not, a <B>Self-Helpless</B> section! For real.
I saw titles like <I>Got a 50-Cent Head? Here's How To
Get a Ten Dollar Haircut!</I> and <I>A Few Geese Shy of
a Gaggle-And Proud Of It!</I>
</BODY>
</HTML>
```

Kita juga dapat mengkombinasikan dua macam format teks secara bersamaan. Misalkan untuk menampilkan kata-kata dalam cetak tebal dan miring kita dapat menuliskan kode HTML seperti ini:

```
<B><I>Kata-kata yang tercetak tebal dan miring</I></B>
```

Selain itu kita juga dapat menampilkan suatu karakter khusus dengan menggunakan kode yang tertentu pula. Ada dua macam kode yang dapat kita gunakan untuk menampilkan karakter-karakter alternatif tersebut yakni dengan menggunakan Character Reference atau Entity Name. Berikut ini terdapat daftar yang memuat beberapa karakter khusus yang dapat ditampilkan dengan menggunakan suatu kode tertentu.

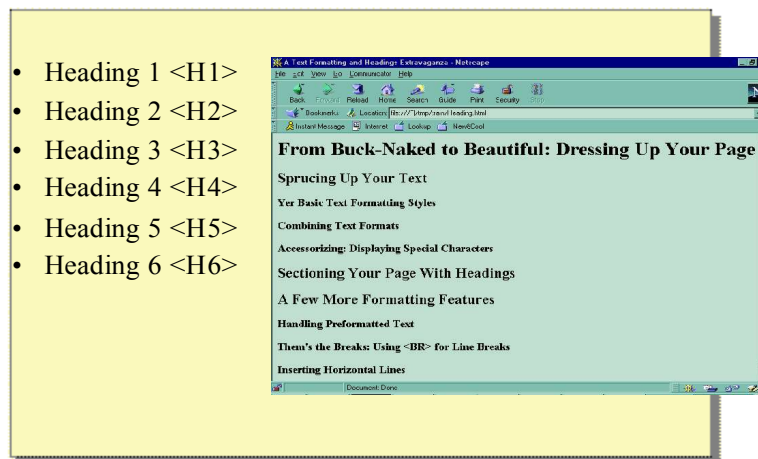
Misalkan untuk menuliskan "Tag <HTML>" dapat kita implementasikan:

Tag &lt;HTML&gt;

Symbol	Character Reference	Entity name
<	&#060;	&lt;
>	&#062;	&gt;
¢	&#162;	&cent;
£	&#163;	&pound;
¥	&#165;	&yen;

©	&#169;	&copy;
®	&#174;	&reg;
°	&#176;	&deg;
$\frac{1}{4}$	&#188;	$\frac{1}{4}$
$\frac{1}{2}$	&#189;	$\frac{1}{2}$
$\frac{3}{4}$	&#190;	$\frac{3}{4}$
&times;	&#215;	&times;

## Penggunaan Heading



- Heading 1 <H1>
- Heading 2 <H2>
- Heading 3 <H3>
- Heading 4 <H4>
- Heading 5 <H5>
- Heading 6 <H6>

Seperti halnya pada pembuatan buku, kita dapat membagi naskah atau dokumen HTML ke dalam beberapa seksi untuk memudahkan proses pembacaan. Masing-masing seksi atau bagian itu terdapat satu subjudul. Untuk menangani format tampilan yang seperti ini kita dapat menggunakan tag HTML yang khusus untuk kepentingan tersebut. Terdapat enam tag yang dapat digunakan untuk membentuk format heading yakni mulai dari <H1> yang menggunakan ukuran font paling besar hingga ke <H6> yang ukuran font-nya paling kecil.

Contoh penggunaannya dalam file HTML adalah sebagai berikut:

```
<HTML>

<HEAD>

<TITLE>A Text Formatting and Headings Extravaganza</TITLE>

</HEAD>

<BODY>

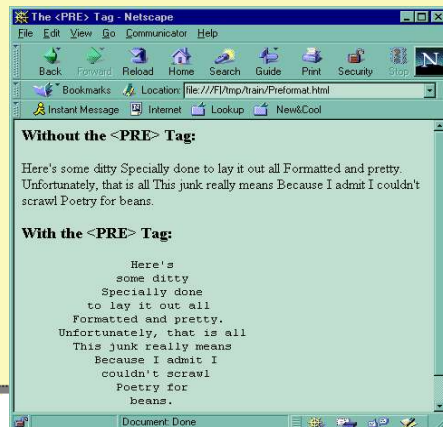
<H1>From Buck-Naked to Beautiful: Dressing Up Your Page</H1>

<H2>Sprucing Up Your Text</H2>
```

```
<H3>Yer Basic Text Formatting Styles</H3>
<H3>Combining Text Formats</H3>
<H3>Accessorizing: Displaying Special Characters</H3>
<H2>Sectioning Your Page With Headings</H2>
<H2>A Few More Formatting Features</H2>
<H3>Handling Preformatted Text</H3>
<H3>Them's the Breaks: Using &lt;br> for Line Breaks</H3>
<H3>Inserting Horizontal Lines</H3>
</BODY>
</HTML>
```

## Penanganan Preformat

- Menggunakan tag `<pre>` dan `</pre>`



Pada kondisi biasa web browser akan mengabaikan pemakaian karakter-karakter multi spasi, tab, dan carriage return, namun kita dapat mencegahnya dengan menggunakan tag `<PRE>`. Penggunaan tag `<PRE>` ini membuat web browser akan menampilkan dokumen dalam bentuk apa adanya (karakter multi spasi, tab dan carriage return tidak diabaikan).

Tag `<PRE>` merupakan kependekan dari "*preformatted*", yang artinya naskah ditampilkan dalam bentuk layout yang asli. Pada contoh berikut ditampilkan perbedaan antara naskah dokumen yang menggunakan tag `<PRE>` dan naskah yang lain tidak menggunakannya. Layout asli kedua naskah tersebut dibuat serupa sehingga kita dapat melihat perbedaan dari hasil tampilannya.

```
<HTML>
<HEAD>
<TITLE>The &lt;PRE&gt; Tag</TITLE>
</HEAD>
<BODY>
<H3>Without the &lt;PRE&gt; Tag:</H3>
```

```
    Here's
    some ditty
    Specially done
    to lay it out all
    Formatted and pretty.
Unfortunately, that is all
    This junk really means
    Because I admit I
    couldn't scrawl
    Poetry for
    beans.
```

```
<P>
<H3>With the &lt;PRE&gt; Tag:</H3>
<PRE>
```

```
    Here's
    some ditty
    Specially done
    to lay it out all
    Formatted and pretty.
Unfortunately, that is all
    This junk really means
    Because I admit I
    couldn't scrawl
```

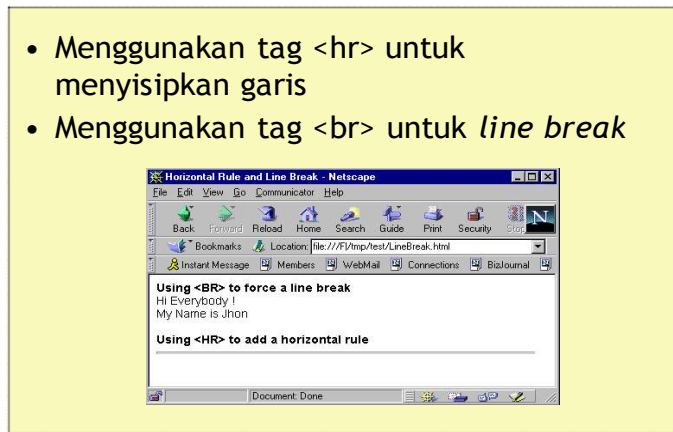
```
Poetry for  
beans.  
  
</PRE>  
  
</BODY>  
  
</HTML>
```

Catatan :

Pada penggunaan tag <PRE>, teks akan ditampilkan dengan menggunakan font dengan format monospace.

## Penyisipan Garis dan *Line Break*

- Menggunakan tag <hr> untuk menyisipkan garis
- Menggunakan tag <br> untuk *line break*



### Penyisipan Garis

Untuk menambah nilai estetika *page* maupun untuk memisahkan suatu bagian informasi dari bagian yang lainnya, kita dapat menyisipkan sebuah garis horisontal pada *page*. Penyisipan garis ini dapat dilakukan dengan menambahkan tag <HR>. Tag <HR> ini juga memiliki beberapa parameter tambahan yang memungkinkan kita untuk memodifikasi tampilan dari garis horisontal yang digunakan ini.

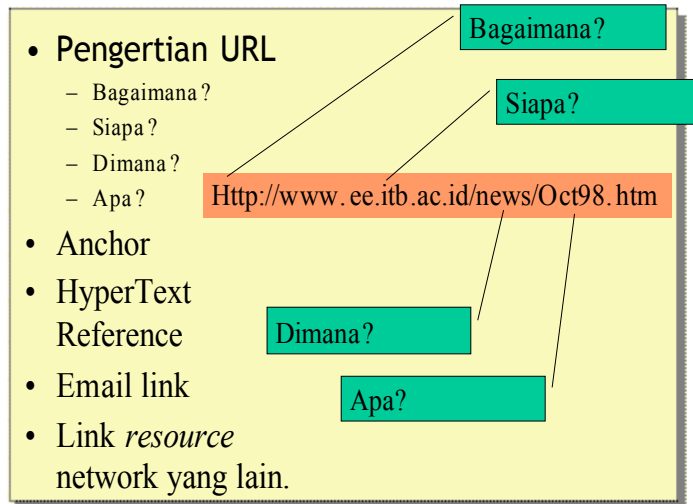
## Line Break

Pada pemaparan bagian sebelum ini telah dijelaskan bahwa tag <P> dapat digunakan untuk memisahkan suatu paragraf dengan paragraf yang berikutnya. Persoalan berikutnya bagaimana jika ingin menyisipkan *line break* pada akhir baris agar jarak dengan baris berikutnya tidak terlalu jauh (kalau kita menggunakan tag <P> jarak antara kedua baris relatif lebih jauh). Tentu saja kita dapat melakukan dengan menggunakan tag <PRE> untuk menampilkan dokumen dalam bentuk *preformat*, akan tetapi hal ini kurang diminati karena font ditampilkan dalam format *monospace*. Untuk mengatasi hal itu kita dapat menggunakan tag <BR>. Tag <BR> ini akan menyisipkan *line break* pada akhir baris sehingga kita dapat pindah ke baris berikutnya tanpa harus berganti paragraf.

Pada contoh berikut akan ditampilkan pada *page* berupa penggunaan *line break* dan penyisipan sebuah garis horizontal.

```
<HTML>
<HEAD>
<TITLE>Horizontal Rule and Line Break</TITLE>
</HEAD>
<BODY>
<P><B>Using &lt;BR&gt; to force a line break</B><BR>
Hi Everybody ! <BR>
My Name is Jhon <BR></P>
<P><B>Using &lt;HR&gt; to add a horizontal rule</B>
<HR>
</BODY>
</HTML>
```

### Penambahan Link



Pada modul-modul sebelumnya kita telah dikenalkan pada konsep URL (*Uniform Resource Locator*), yakni pada dasarnya semua *resource* yang ada di Internet memiliki alamat sendiri. Selanjutnya kita juga telah memahami konsep *hypertext* dari sebuah file HTML yang memungkinkan dibuat suatu link yang menghubungkan pada bagian lain dari dokumen HTML atau langsung mengarah pada suatu *resource* Internet tertentu.

URL yang digunakan untuk mengimplementasikan penambahan link pada suatu page dapat kita sederhanakan dalam model berikut :

Bagaimana://Siapa/Dimana/Apa

### Bagaimana

Pada bagian pertama dari URL ini adalah menunjukkan bagaimana cara data dikirimkan atau ditransmisikan melalui media jaringan. Secara teknis lebih sering kita sebut dengan istilah protokol. Misalnya untuk aplikasi *World Wide Web* ini kita menggunakan protokol yang disebut dengan protokol HTTP. Oleh karena itu pada bagian awal URL ini kita deklarasikan:

http://

### Siapa

Pada bagian selanjutnya menjelaskan "siapa" yang memiliki URL. Bagian ini berisi nama komputer tempat *resource* Internet tersebut berada. Istilah yang lebih teknis kita sebut dengan *hostname*. Misalkan page kita diletakkan di mesin *www.yahoo.com* maka penambahan URL sampai dengan bagian ini menjadi:

http://www.yahoo.com

### Dimana

Bagian selanjutnya menunjukkan lokasi yang lebih spesifik tempat menyimpan *resource* Internet tersebut. Secara fisik bagian ini berisi nama direktori tempat penyimpanan *resource* yang bersangkutan.

```
http://www.yahoo.com/Home
```

## Apa

Bagian terakhir ini menunjukkan nama file yang akan kita gunakan. Untuk aplikasi web file ini berisi kode-kode HTML dan teks. Secara keseluruhan URL dapat dituliskan sebagai berikut:

```
http://www.yahoo.com/Home/homepage.html
```

Pembuatan link dalam suatu file HTML dapat dilakukan dengan menggunakan tag `<A>` yang diikuti dengan parameter referensinya HREF.

```
<A HREF="URL">
```

Contoh penggunaannya:

```
<A HREF=http://www.yahoo.com/Home/homepage.html> Klik di sini  
</A>
```

Selain link yang mengarah ke luar, contohnya mengarah ke file lain, *web site* lain atau ke *resource* lain, kita dapat juga menggunakan internal link untuk menunjukkan suatu bagian tertentu dari sebuah naskah HTML. Internal link ini sering dikenal dengan istilah Anchor. Tag ini digunakan untuk menandai bagian naskah yang nantinya akan ditunjuk oleh HREF (*Hypertext REFerence*).

```
<A NAME="Nama">Bagian teks yang akan diberi tanda</A>
```

Contoh penggunaannya:

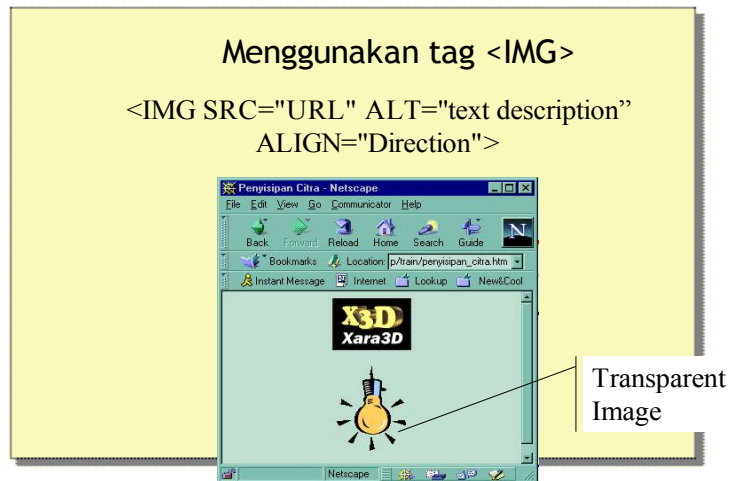
```
<A NAME="Bagian1">Bagian 1</A>
```

maka untuk merujuknya digunakan:

```
<A HREF="#Bagian1">Klik di sini untuk menuju Bagian 1</A>
```

## Penyisipan Citra atau Gambar





Penyisipan citra pada page dimaksudkan agar tampilan dari page tersebut dapat lebih menarik dan informatif. Ada beberapa hal yang perlu diperhatikan dalam penyisipan citra atau gambar ke page, yakni:

- Ukuran file citra

Ukuran file yang terlalu besar akan membuat pengunjung web harus menunggu lebih lama dalam mengakses web karena musti terlebih dahulu mendownload file citra atau gambar yang ukurannya besar

- Tipe file citra

Tipe file yang mendukung untuk transmisi melalui aplikasi web adalah JPEG atau GIF

File citra dapat disisipkan ke dalam suatu file HTML dengan menggunakan tag <IMG>, yakni dalam format penuh:

```
<IMG SRC="URL" ALT="deskripsi teks" ALIGN="Arah">
```

Atribut ALT berfungsi sebagai alternatif jika browser yang digunakan oleh pengguna tidak dapat menampilkan citra. Sedangkan atribut ALIGN dapat diisi dengan nilai-nilai TOP, MIDDLE, BOTTOM, LEFT, dan RIGHT.

Contoh penggunaan tag <IMG>:

```
<HTML>
<HEAD>
<TITLE>Aligning Text and Images</TITLE>
</HEAD>
<BODY>
```

```
<IMG SRC="constru1.gif" ALIGN=TOP>This text appears at the top of  
the image.
```

```
<P>
```

```
<IMG SRC="constru1.gif" ALIGN=MIDDLE>This text appears in the  
middle of the image.
```

```
<P>
```

```
<IMG SRC="constru1.gif" ALIGN=BOTTOM>This text appears at the  
bottom of the image.
```

```
</BODY>
```

```
</HTML>
```

## TABEL

Tabel merupakan cara untuk menampilkan informasi dalam bentuk sel yang terdiri dari kolom dan baris . untuk menampilkannya digunakan tag <TABLE>...</TABLE> , untuk mengisi baris(table row) digunakan tag <TR>...</TR> yang didalamnya terdapat tag <TD>...</TD> untuk menampilkan data pada setiap sel.

### Latihan 7 : TABEL

```
<HTML><HEAD><TITLE>LATIHAN 7 </TITLE></HEAD>
```

```
<BODY>
```

```
<TABLE BORDER=3>
```

```
<TR>
```

```
<TD> baris 1, kolom 1</TD>
```

```
<TD> baris 1, kolom 2</TD>
```

```
<TD> baris 1, kolom 3</TD>
```

```
</TR>
```

```
<TR>
```

```
<TD> baris 2, kolom 1</TD>
```

```
<TD> baris 2, kolom 2</TD>
```

```
<TD> baris 2 ,kolom 3</TD>
```

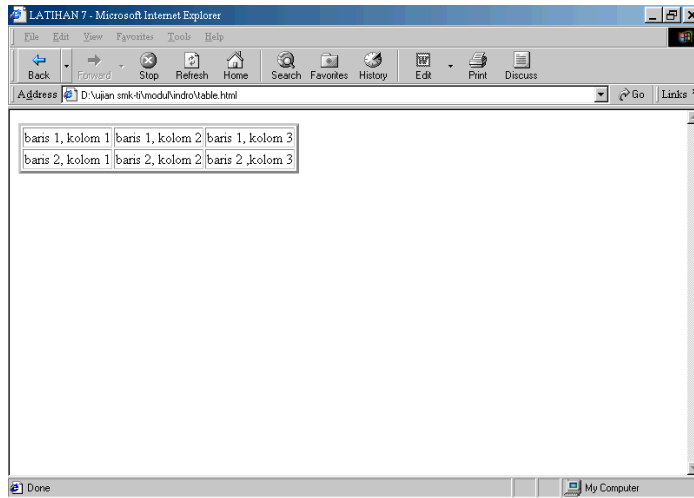
```
</TR>
```

```
</TABLE>
```

</BODY>

</HTML>

Hasil Tampilan :



## FRAME

Frame HTML dapat digunakan untuk membuat tampilan halaman HTML yang terbagi-bagi menjadi beberapa bagian dimana setiap bagiannya merupakan satu halaman HTML yang terpisah. Ini dilakukan untuk membuat tampilan halaman HTML yang salah satu atau beberapa bagian berganti-ganti isinya sedang bagian lain tetap sehingga dapat menghemat bandwidth internet.

### **Latihan FRAME**

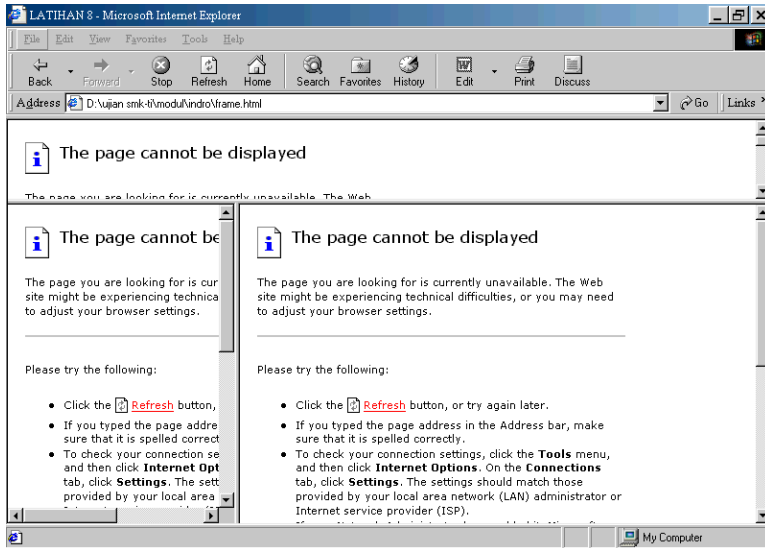
```
<HTML>
<HEAD>
<TITLE>LATIHAN 8 </TITLE>
</HEAD>
<FRAMESET ROWS=20%,*>
<FRAME NAME=header SRC="header.html">

<FRAMESET COLS=30%,*>
    <FRAME NAME=kiri SRC="kiri.html">
    <FRAME NAME=kanan SRC="kanan.html">
</FRAMESET>

</FRAMESET>

<BODY>
</BODY>
</HTML>
```

Hasil Tampilan :



## FORM HTML

Web page memungkinkan kita interaktif dengan pengunjung web , salah satunya dalah dengan penggunaan FORM . Metoda pengiriman data pada form terdapat dua yaitu GET dan POST. Pada HTML metoda ini dinyatakan dengan atribut METHOD di dalam elemen FORM. Metoda GET merupakan default pengiriman suatu data form .metode GET mengirimkan data pada server dengan cara meletakkannya pada bagian akhir URL yang menunjuk pada script pemroses Jadi jika URL anda menunjuk ke CGI Script pada URL “cgi-bin/scriptform” dan ata formnya adalah “Salman” dan “2503645”, maka URL akhir yang dikirim ke server adalah :

/cgi-bin/scriptform?kota=Jakarta&telepon=2503645

Metode POST tidak seperti metode GET yang mengirimkan datanya secara terpisah pada suatu standar input. Script mengambil data form dari standar input ini, dengan adanya penyimpanan data secara tersendiri membuat metoda POST bisa menyimpan data input dalam jumlah banyak.

### Latihan FORM

```
<HTML>
<HEAD>
<TITLE>LATIHAN 9</TITLE>
</HEAD>
<BODY>
<FORM METHOD="POST" ACTION="kirim.php3">
<PRE>
Nama      :<INPUT TYPE=TEXT NAME="Nama">
Alamat    :<INPUT TYPE=TEXT NAME="Alamat" SIZE=40 MAXLENGTH=60>
E-mail    :<INPUT TYPE=TEXT NAME="Email" SIZE=20>
Telepon   :<INPUT TYPE=TEXT NAME="Telepon" SIZE=7>
<INPUT TYPE=SUBMIT VALUE="OK">    <INPUT TYPE=RESET VALUE="BATAL">
</PRE>
</FORM>
</BODY>
</HTML>
```

Contoh form diatas :

