MODUL KULIAH

WEB PROGRAMMING 1

Penulis

Yoki Firmansyah M.Kom

FAKULTAS TEKNOLOGI INFORMASI

PRODI SISTEM INFORMASI

UNIVERSITAS BINA SARANA INFORMATIKA

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat ALLAH SWT yang telah memberikan kekuatan dan niat baik pada penulis dalam menyusun bahan ajar Web Programming 1 hingga kehadapan pembaca yang budiman.

Pemrograman web merupakan pemrogrman yang mudah untuk dipelajari dan di pahami. Bahan ajar ini memandu dan membantu bagi pembaca agar dapat menguasai dengan mudah dan cepat dalam mempelajarinya.

Bahan ajar ini cocok untuk kalangan siswa/I, mahasiswa/I, dan umum yang baru sedang mempalajari pemrograman dasar web. Materi bahan ajar ini dikemas dalam tahap demi tahap yang saling berhubungan, juga disertai latihan contoh soal untuk mempercepat pemahamannya.

Semoga bahan ajar ini bermanfaat meskipun masih banyak kekurangan yang harus diperbaiki.

TTD

YOKI FIRMANSYAH M.Kom

DAFTAR ISI

KATA PENGANTAR ..... **II**

PERKENALAN ..... **IV**

APA ITU WEBSITE ? IV
APAKAH FUNGSI DARI SERVER SOFTWARE ? IV
BAGAIMANA DAPAT MENGAKSES SITUS WEB? IV

HTML (*HYPERTEXT MAKRUP LANGUAGE*) **1**

APA ITU HTML? 1
STRUKTUR DASAR HTML 1
HTML FORMAT TEXT 4
HTML DAFTAR URUTAN 11
HTML TABEL 14
HTML GAMBAR 18
HTML HYPERLINK 20
HTML FORM 23
HTML FRAME ELEMENT FRAMESET 27

PERKENALAN PHP **29**

APA ITU PHP ? 29
KENAPA BELAJAR PHP? 29
KARAKTERISTIK PHP 30
PERISAPAN LINGKUNGAN PHP 30
SINTAX PHP 34

DAFTAR PUSTAKA **57**

PERKENALAN

Apa itu website ?

Pada dasarnya, web adalah skumpulan computer yang saling terkoneksi dan berbicara satu sama lain. Computer-komputer biasanya terkoneksi melalui jaringan telpon, signal satelit kabel, dan tipe data tranfer lainnya. Data tranfer dapat disederhanakan sebagai cara untuk memindahkan informasi dari poin A ke poin B ke poin C dan seterusnya.

Computer-komputer di dalam satu jaringan dapat saja terkoneksi 24 jam, atau bisa juga hanya terkoneksi pada saat-saat tertentu. Komputer-komputer yang terkoneksi terus selama 24 jam pada umumnya disebut server. Server adalahkomputer-komputer sama saja seperti yang mungkin kamu gunakan sekarang dalam membaca artikel ini. Dengan satu perbedaan besar, mereka memiliki server software.

Apakah fungsi dari Server Software ?

Server software dibuat untuk menampilkan halaman-halaman web dan situs-situs web. Pada dasarnya, computer server memiliki banyak situs web didalamnya dan hanya menuggu untuk orang mengkases seperti: Chrome/Firefox/IE, dll. Untukakses dan membuka situs tersebut. Saat browser meminta akses untuk halaman situs web, server akan mengirimkan halaman tersebut.

Bagaimana dapat Mengakses Situs Web?

Jawaban pendeknya adalah : dengan mengetik di URL, atau dengan kata lain alamt web, contoh: apabila kamu ingin membuka website Facebook, kamu akan mengetik alamt tersebut di kolom alamt web browser address bar atau mungkin juga melalui link favorit/bookmart. Ada cara lain untuk menemukan situs-situs web, seperti search engines (Google, Bing, yahoo, dll) tapi untuk beberapa situs hanya bisa ditemukan apabila kita mengetahui alamatnya.

TTD

Yoki Firmansyah M.Kom

HTML (*Hypertext Makrup Language*)

Apa itu HTML?

HTML merupakan singkatan dari *HyperText Markup Language* adalah script untuk menyusun dokumen-dokumen Web. Dokumen HTML disimpan dalam format teks reguler dan mengandung *tag-tag* yang memerintahkan web browser untuk mengeksekusi perintah-perintah yang dispesifikasikan.

Struktur Dasar HTML

Struktur dasar dokumen HTML adalah sebagai berikut:

```
<html>
<head>
<title>
Disini Judul Dokumen HTML</title>
</head>
<body>
 Disini penulisan informasi Web
</body>
</html>
```

Dari struktur dasar HTML di atas dapat dijelaskan sebagai berikut:

- Tag
  Adalah teks khusus (*markup*) berupa dua karakter "<" dan ">", sebagai contoh <body> adalah tag dengan nama body. Secara umum tag ditulis secara berpasangan, yang terdiri atas tag pembuka dan tag penutup (ditambahkan karakter "/" setelah karakter "<"), sebagai contoh <body> ini adalah tag pembuka isi dokumen HTML, dan </body> ini adalah tag penutup isi dokumen HTML.

- Element
  Element terdiri atas tiga bagian, yaitu tag pembuka, isi, dan tag penutup. Sebagai contoh untuk menampilkan judul dokumen HTML pada web browser digunakan element title, dimana: <title> ini adalah tag pembuka judul dokumen HTML Disini Judul Dokumen HTML ini adalah isi judul dokumen HTML </title> ini adalah tag penutup judul dokumen HTML Tag-tag yang ditulis secara berpasangan pada suatu element HTML, tidak boleh saling tumpang tindih dengan pasangan tag-tag lainnya.

Contoh penulisan tag-tag yang benar:

```
<p>
 <b>
   ...............
 </b>
</p>
```

Contoh penulisan tag-tag yang salah

```
<p>
 <b>
 ................
   </p>
</b>
```

- Attribute

Attribute mendefinisikan property dari suatu element HTML, yang terdiri atas nama dan nilai. Penulisannya adalah sebagai berikut:

```
<TAG
nama-attr="nilai-attr"
nama-attr="nilai-attr"
   .................
>
  .................

</TAG>
```

Secara umum nilai attribute harus berada dalam tanda petik satu atau dua. Sebagai contoh, untuk membuat warna teks menjadi kuning dan latarbelakang halaman web menjadi hitam, penulisannya adalah <body bgcolor="black" text="yellow">

- Element HTML

  Menyatakan pada browser bahwa dokumen Web yang digunakan adalah HTML.

  Sintaks:

  ```
  <html>
  ..........
  </html>
  ```

- Element HEAD

  Merupakan kepala dari dokumen HTML. Tag <head> dan tag </head> terletak di antara tag <html> dan tag </html>. Sintaks:

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

❖ Element TITLE

  Merupakan judul dari dokumen HTML yang ditampilkan pada judul jendela browser. Tag <title> dan tag </title> terletak di antara tag <head> dan tag </head>.

  Sintaks:

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

❖ Element BODY

  Element ini untuk menampilkan isi dokumen HTML. Tag <body> dan tag </body> terletak di bawah tag <head> dan tag </head>. Element BODY mempunyai attribute attribute yang

menspesifikasikan khususnya warna dan latarbelakang dokumen yang akan ditampilkan pada browser.

Sintaks:

```
<body text="v" bgcolor="w" background="uri" link="x" alink="y" vlink="z">
.............
</body>
```

Attribute text memberikan warna pada teks, bgcolor memberikan warna pada latarbelakang dokumen HTML, background memberikan latarbelakang dokumen HTML dalam bentuk gambar, linkmemberikan nilai warna untuk link, alink memberikan warna untuk link yang sedang aktif, vlink memberikan warna untuk link yang telah dikunjungi. Jika attribute bgcolor dan background keduanya dispesifikasikan maka attribute background yang akan digunakan, akan tetapi jika nilai attribute background (gambar) tidak ditemukan pada dokumen HTML maka attribute bgcolor yang akan digunakan.

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Menampilkan teks:

Belajar bahasa pemrograman web ternyata mudah juga :)

Nama file: latihan1_1.html

```
<html>
<head>
<title>Latihan1-1</title>
</head>
<body>
        Belajar bahasa pemrograman web ternyata mudah juga :)
</body>
</html>
```

Tugas tambahan:

Gantilah teks tersebut dengan teks lainnya.

Latihan 2:

Merubah warna teks menjadi merah:

Belajar bahasa pemrograman web ternyata mudah juga :)

Nama file: latihan1_2.html

```
<html>
<head>
<title>Latihan1-2</title>
</head>
<body text="red">
Belajar bahasa pemrograman web ternyata mudah juga :)
</body>
</html>
```

Tugas tambahan:
Cobalah untuk warna: maroon, black, yellow, white, lime, green, blue, red, olive, navy, purple, teal, gray, silver, fuchsia, aqua.

Latihan 3:
Merubah warna background menjadi hitam.
Nama file: latihan1_3.html

```
<html>
<head>
<title>Latihan1-3</title>
</head>
<body text="red" bgcolor="black">
Belajar bahasa pemrograman web ternyata mudah juga :)
</body>
</html>
```

Tugas tambahan:
Cobalah untuk warna: maroon, black, yellow, white, lime, green, blue, red, olive, navy, purple, teal, gray, silver, fuchsia, aqua.

Latihan 4:
Merubah background dengan suatu gambar.
Nama file: latihan1_4.html

```
<html>
<head>
<title>Latihan1-4</title>
</head>
<body text="red" bgcolor="aqua" background="./images/image027.jpg">
Belajar bahasa pemrograman web ternyata mudah juga :)
</body>
</html>
```

catatan:
./images/ = nama direktori file gambar disimpan image027.jpg = nama file gambar

Tugas tambahan:
Cobalah untuk file gambar: image050.jpg, image052.jpg, image058.jpg dan image060.jpg.

HTML Format Text
ELEMENT BR (Line Break)
Element BR berfungsi untuk ganti baris. Dan element BR ini tidak mempunyai tag penutup.
Sintaks:
**
**

ELEMENT P (Paragraph)
Element P berfungsi untuk ganti paragraf yang diikuti dengan baris kosong di awal dan akhir paragraf. Tag penutup </p> sifatnya optional jika suatu paragraf diikuti oleh paragraf berikutnya. Jika tag </p> diabaikan, maka paragraf itu tidak akan diikuti dengan baris kosong di akhir paragraf.
Element P mempunyai attribute yaitu *align* yang bernilai *"left"*, *"center"*, *"right"* yang menspesifikasikan posisi tepi horizontal dari paragraf (default: "left").
Sintaks:
<p align="left"|"center"|"right">
.....
</p>

ELEMENT H1,H2,H3,H4,H5,H6 (Header)
Element H1,H2,H3,H4,H5,H6 berfungsi untuk membuat header dengan format 6 jenis ukuran huruf dan tercetak tebal. Nilai ukuran huruf terbesar adalah H1 dan terkecil adalah H6. Element ini mempunyai attribute yaitu *align* yang bernilai *"left"*, *"center"*, *"right"* yang menspesifikasikan posisi horizontal dari header (default: "left").
Sintaks:
<hx align="left"|"center"|"right">
.....
</hx>

x : 1 ... 6

ELEMENT B (Bold)
Element B berfungsi untuk membuat tampilan teks tercetak tebal (bold).

Sintaks:
****
.....
****

ELEMENT I (Italic)
Element I berfungsi untuk membuat tampilan teks tercetak miring (italic).

Sintaks:
****
.....
****

ELEMENT U (Underline)
Element U berfungsi untuk membuat tampilan teks tercetak garis bawah (underline).

Sintaks:
<u>
.....
</u>

ELEMENT PRE (Preformated text)
Element PRE berfungsi untuk menampilkan teks seperti apa adanya.

Sintaks:
<pre>
.....
</pre>

ELEMENT CENTER
Element CENTER berfungsi untuk menampilkan teks dengan posisi horizontal ditengah.

Sintaks:
** **
.....
** **

ELEMENT BASEFONT
Element BASEFONT berfungsi untuk merubah dasar ukuran huruf tampilan dari web browser. Attribute dari element ini adalah size yang merupakan ukuran huruf dengan nilai "1" sampai dengan "7" pixel, atau berupa ukuran relatif (+ atau -), default nya adalah "3" pixel. Element ini tidak mempunyai tag penutup.
Sintaks:
<basefont size="pixel">

ELEMENT FONT
Element FONT berfungsi untuk merubah jenis, ukuran dan warna dari tampilan huruf. Element ini mempunyai attribute color untuk merubah warna huruf, face untuk merubah jenis huruf (jika lebih dari satu jenis huruf, harus dibatasi dengan koma), dan size untuk merubah ukuran huruf (pixel).
Sintaks:
****
.....
****

ELEMENT HR (Horizontal Rule)

Element HR berfungsi untuk membuat suatu garis horizontal. Element ini tidak mempunyai tag penutup dan mempunyai attribute align untuk menempatkan posisi secara horizontal ("left", "center", "right"), size untuk ukuran ketebalan garis ("pixel"), width untuk ukuran panjang garis ("persen"), dan noshade (garis solid).

Sintaks:

<hr align="left"|"center"|"right" size="pixel" width="persen" noshade>

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Menampilkan teks dengan ganti baris (element BR):

Belajar bahasa pemrograman web. Dengan materi dasar HTML. Dan JSP Belajar bahasa pemrograman web. Dengan materi dasar HTML. Dan JSP.

Nama file: latihan2_1.html

```
<html>
<head>
<title>Latihan2-1</title>
</head>
<body>
Belajar bahasa pemgrograman web. Dengan mater dasar HTML. Dan JSP. <br>
<br>Belajar bahasa pemrograman web.<br>Dengan materi dasar HTML.<br>Dan JSP.
</body>
</html>
```

Latihan 2:

Menampilkan teks dalam bentuk paragraf (element P):

Tampilan teks sebelum paragraf.

Tampilan teks paragraf pertama.

Tampilan teks paragraf kedua.

Tampilan teks setelah paragraf.

Nama file: latihan2_2.html

```
<html>
<head>
<title>Latihan2-2</title>
</head>
<body>
Tampilan teks sebelum paragraf.
<p>Tampilan teks paragraf pertama.</p>
<p>Tampilan teks paragraf kedua.</p>
Tampilan teks setelah paragraf.
</body>
</html>
```

Tugas tambahan:
Gantilah setiap tag </p> dengan element BR, dan lihat hasilnya.

Latihan 3:

Menampilkan posisi horizontal kelurusan paragraf:
Paragraf ini pada posisi rata kiri (default)
Paragraf ini pada posisi rata tengah
Paragraf ini pada posisi rata kanan
Paragraf ini pada posisi rata kiri

Nama file: latihan2_3.html

```
<html>
<head>
<title>Latihan2-3</title>
</head>
<body>
<p>Paragraf ini pada posisi rata kiri (default)</p>
<p align="center">Paragraf ini pada posisi rata tengah</p>
<p align="right">Paragraf ini pada posisi rata kanan</p>
<p align="left">Paragraf ini pada posisi rata kiri</p>
</body>
</html>
```

Latihan 4:

Menampilkan teks sebagai header (element H):

Heading-1
Ini Heading-2
Ini Heading-3
Ini Heading-4
Ini Heading-5
Ini Heading-6

Nama file: latihan2_4.html

```
<html>
<head>
<title>Latihan2-4</title>
</head>
<body>
<h1>Ini Heading-1</h1>
<h2>Ini Heading-2</h2>
<h3>Ini Heading-3</h3>
<h4>Ini Heading-4</h4>
<h5>Ini Heading-5</h5>
<h6>Ini Heading-6</h6>
</body>
</html>
```

Tugas tambahan:
Tempatkanlah tampilan isi element H3 di kanan dan H5 di tengah.

Latihan 5:

Menampilkan teks dalam format tebal, miring dan garis bawah:
Ini normal teks
Ini teks tercetak tebal (bold)
Ini teks tercetak miring (italic)
Ini teks tercetak garis bawah (underline)
Nama file: latihan2_5.html

```
<html>
<head>
<title>Latihan2-5</title>
</head>
<body>
Ini normal teks<br>
<b>Ini teks tercetak tebal (bold)</b><br>
<i>Ini teks tercetak miring (italic)</i><br>
<u>Ini teks tercetak garis bawah (underline)</u>
</body>
</html>
```

Tugas tambahan:
Buat teks tercetak miring dan tebal**.**

Latihan 6:

Menampilkan teks dalam bentuk apa adanya (element PRE):
Ini bentuk penulisan preformated text apapun bentuknya tulisan ini akan ditampilkan pada web browser seperti apa adanya, alias yang ditulis.
Nama file: latihan2_6.html

```
<html>
<head>
<title>Latihan2-6</title>
</head>
<body>
<pre>
Ini bentuk penulisan preformated text apapun bentuknya tulisan ini akan ditampilkan
pada web browser seperti apa adanya, alias yang ditulis
</pre>
</body>
</html>
```

Latihan 7:

Menampilkan teks pada posisi tengah jendela web browser (element CENTER):
Teks ini terletak di tengah layar web browser dengan menggunakan element CENTER

Nama file: latihan2_7.html

```
<html>
<head>
<title>Latihan2-7</title>
</head>
<body>
<center>
Teks ini terletak di tengah layar web browser<br>
 dengan menggunakan element CENTER
</center>
</body>
</html>
```

Latihan 8:

Menampilkan teks berdasarkan element BASEFONT:

Teks ini mempunyai size = 1
ini mempunyai size = 2
ini mempunyai size = 3
Teks ini mempunyai size = 4
ini mempunyai size = 5
ini mempunyai size = 6
Teks ini mempunyai size = 7

Nama file: latihan2_8.html

```
<html>
<head>
<title>Latihan2-8</title>
</head>
<body>
<basefont size="1">Teks ini mempunyai size=1<br>
<basefont size="2">Teks ini mempunyai size=2<br>
<basefont size="3">Teks ini mempunyai size=3<br>
<basefont size="4">Teks ini mempunyai size=4<br>
<basefont size="5">Teks ini mempunyai size=5<br>
<basefont size="6">Teks ini mempunyai size=6<br>
<basefont size="7">Teks ini mempunyai size=7
</body>
</html>
```

Latihan 9:

Menampilkan teks dengan menggunakan element FONT:

dengan format color=fuchia; size=+2; face=verdana Teks dengan format color=gray; size=6; face=Algerian dengan format color=red; size=-1; face=impact

Nama file: latihan2_9.html

```
<html>
<head>
<title>Latihan2-9</title>
</head>
<body>
<font color="fuchsia" size="+2" face="verdana">
Teks dengan format color=fuchsia; size=+2; face=verdana<br>
</font>
<font color="gray" size="6" face="algerian">
Teks dengan format color=gray; size=6; face=algerian<br>
</font>
<font color="red" size="-1" face="impact">
Teks dengan format color=red; size=-1; face=impact
</font>
</body>
</html>
```

Latihan 10:

Menampilkan garis horizontal menggunakan elemen HR:

Nama file: latihan2_10.html

```
<html>
<head>
<title>Latihan2-10</title>
</head>
<body>
<hr align="left" size="6" width="30%">
<hr align="left" size="6" width="30%" noshade>
</body>
</html>
```

Tugas tambahan:

Buatlah garis horizontal di bawah ini dengan size=5, align=center dengan urutan width 40%, 20%, 10%:

HTML Daftar urutan

ELEMENT OL (Orderd List)

Element OL berfungsi untuk membuat nomor daftar urut. Element ini mempunyai attribute yaitu start, dan type. Attribute start berfungsi untuk memberikan nilai awal dari daftar urutan, attributetype berfungsi untuk memilih bentuk daftar urutan yaitu A untuk abjad huruf besar, a untuk abjad huruf kecil, I untuk abjad romawi besar, i untuk abjad romawi kecil, 1 untuk angka decimal (default: start="1" type="1"). Dalam element OL harus terdapat beberapa element LI.

Sintaks:
<ol start="number" type="A"|"a"|"I"|"i"|"1">
.....

ELEMENT UL (Unordered List)

Element UL berfungsi untuk membuat daftar tanpa nomor urut (dalam format bullet). Element ini mempunyai attribute yaitu type. Attribute type berfungsi untuk memilih bentuk bullet yang digunakan. Nilai dari pada attribute type ini adalah "circle", "square" dan "disc" . Dalam element UL harus terdapat beberapa element LI.

Sintaks:
<ul type="circle"|"square"|"disc">
.....

ELEMENT LI (List Item)

Element LI merupakan isi dari pada daftar. Element ini harus berada di dalam element OL atau UL. Element ini mempunyai attribute type dan value. Nilai attribute type adalahA,a,I,i,1,circle,square,disc dan attribute value berisikan nilai nomor urutan dari attribute type.

Sintaks:
<li type="A"|"a"|"I"|"i"|"1"|"circle"|"square"|"disc">
.....

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Menampilkan teks dengan daftar urut (element OL):
Matakuliah TKJ Semester 1:
1. Bahasa Inggris 1
2. Matematika 1
3. Jaringan Komputer 1
4. Konsep Teknologi Informasi
5. Keterampilan Komputer dan Pengelolaan Informasi
6. Perangkat Keras Komputer
7. Sistem Operasi

Nama file: latihan3_1.html

```
<html>
<head>
<title>Latihan3-1</title>
</head>
<body>
```

```
Matakuliah TKJ Semester 1:
<ol>
<li>Bahasa Inggris 1</li>
<li>Matematika 1</li>
<li>Jaringan Komputer 1</li>
<li>Konsep Teknologi Informasi</li>
<li>Keterampilan Komputer dan Pengelolaan Informasi</li>
<li>Perangkat Keras Komputer</li>
<li>Sistem Operasi</li>
</ol>
</body>
</html>
```

Latihan 2:

Menampilkan teks dengan daftar urut (element OL):

Matakuliah TKJ Semester 1:

a. Bahasa Inggris 1
b. Matematika 1
c. Jaringan Komputer 1
d. Konsep Teknologi Informasi
e. Keterampilan Komputer dan Pengelolaan Informasi
f. Perangkat Keras Komputer
g. Sistem Operasi

Nama file: latihan3_2.html

```
<html>
<head>
<title>Latihan3-2</title>
</head>
<body>
Matakuliah TKJ Semester 1:
<ol start="1" type="a">
<li>Bahasa Inggris 1</li>
<li>Matematika 1</li>
<li>Jaringan Komputer 1</li>
 <li>Konsep Teknologi Informasi</li>
<li>Keterampilan Komputer dan Pengelolaan Informasi</li>
<li>Perangkat Keras Komputer</li>
<li>Sistem Operasi</li>
</ol>
</body>
</html>
```

Latihan 3:

Menampilkan teks dengan daftar urut (element OL):

I. Matakuliah TKJ Semester 1:
a. Bahasa Inggris 1
b. Matematika 1
c. Jaringan Komputer 1
d. Konsep Teknologi Informasi
e. Keterampilan Komputer dan Pengelolaan Informasi
f. Perangkat Keras Komputer
g. Sistem Operasi

Nama file: latihan3_3.html

```
<html>
<head>
<title>Latihan3-3</title>
</head>
<body>
<ol start="1" type="I">
<li>Matakuliah TKJ Semester 1:</li>
<ol start="1" type="a">
<li>Bahasa Inggris 1</li>
<li>Matematika 1</li>
<li>Jaringan Komputer 1</li>
<li>Konsep Teknologi Informasi</li>
<li>Keterampilan Komputer dan Pengelolaan Informasi</li>
<li>Perangkat Keras Komputer</li>
<li>Sistem Operasi</li>
</ol>
</ol>
</body>
</html>
```

Tugas tambahan:
Ubahlah tampilannya dalam format kombinasi ordered list dan unordered list.

HTML Tabel
ELEMENT TABLE

Element TABLE befungsi untuk membuat suatu data multidimensi yang terdiri atas kolom dan baris.

Element ini mempunyai attribute seperti align, bgcolor, border, cellpadding(jarak antara tepi sel dengan isi sel), cellspacing(jarak antara sel), width(lebar tabel), height(tinggi tabel). Element TABLE berisikan element CAPTION, TH, TR dan TD.

Sintaks:

```
<table align="left"|"center"|"right" bgcolor="color" border="pixel" cellpadding="pixel" cellspacing="pixel" width="pixel"|"%" height="pixel"|"%" >
.....
</table>
```

ELEMENT CAPTION

Element CAPTION berfungsi untuk membuat judul dari tabel. Element ini harus berada di dalam element TABLE dan mempunyai attribute align dengan nilai top(judul terletak di atas tabel), danbottom(judul terletak di bawah tabel).

Sintaks:

```
<caption align="top"|"bottom">
.........................
</caption>
```

ELEMENT TR (*Table Row*)

Element TR mendefinisikan baris pada tabel dan element ini harus berada di dalam element TABLE. Pada element TR terdapat element TH dan TD. Attribute yang terdapat pada element ini adalahalign, valign(posisi vertikal), dan bgcolor.

Sintaks:

```
<tr  align="left"|"center"|"right" valign="top"|"middle"|"bottom" bgcolor="color">
.........................
</tr>
```

ELEMENT TH (*Table Header*) dan TD (*Table Data*)

Element TH dan TD merupakan informasi pada tabel. TH mendefinisikan isi sel sebagai header sel pada kolom tabel dan datanya tercetak tebal dan dengan posisi ditengah sel. Element TH dan TD ini harus terletak di dalam element TR. Attribute kedua element ini adalah align, valign, bgcolor, colspan, rowspan

Sintaks:

```
<th align="left"|"center"|"right" valign="top"|"middle"|"bottom" bgcolor="color"
colspan="number"  rowspan="number">
.........................
</th>

<td align="left"|"center"|"right" valign="top"|"middle"|"bottom" bgcolor="color"
colspan="number" rowspan="number">
.........................
</td>
```

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Buat tabel untuk teks di bawah ini:

Tabel 1.1

No.	Nama
1.	Rina Hastuti
2.	Ali Amran
3.	Rahmat

Nama file: latihan4_1.html

```
<html>
<head>
<title>Latihan4-1</title>
</head>
<body>
<table align="center" border="2" bgcolor="cyan" cellpadding="5" cellspacing="0" >
<caption align="top"><b>Tabel 1.1</b></caption>
<tr>
<th>No.</th>
<th>Nama</th>
</tr>
<tr>
   <td>1.</td>
<td>Rina Hastuti</td>
</tr>
<tr>
 <td>2.</td>
<td>Ali Amran</td>
</tr>
<tr>
<td>3.</td>
<td>Rahmat</td>
</tr>
</table>
</body>
</html>
```

Tugas tambahan:
1. Ganti ukuran cellpadding, cellspacing, border dengan lainnya.
2. Berikan nilai width dan height pada tabel hingga 100%.

Latihan 2:

Dari Latihan 1 gantilah warna background baris tabel menjadi baris-1=green baris-2=yellow baris-3=red baris-4=blue

Nama file: latihan4_2.html

```
<html>
<head>
<title>Latihan4-2</title>
</head>
```

```
<body>
<table align="center" border="2" bgcolor="cyan" cellpadding="5" cellspacing="0" >
<caption align="top"><b>Tabel 1.1</b></caption>
<tr bgcolor="green">
<th>No.</th>
<th>Nama</th>
</tr>
<tr bgcolor="yellow">
<td>1.</td>
   <td>Rina Hastuti</td>
</tr>
<tr bgcolor="red">
<td>2.</td>
<td>Ali Amran</td>
</tr>
<tr bgcolor="blue">
<td>3.</td>
<td>Rahmat</td>
</tr>
</table>
</body>
</html>
```

Tugas tambahan:
Ubahlah align pada baris-2 kolom-1 dan baris-4 kolom-2 ke kanan

Latihan 3:

Buat tabel untuk teks di bawah ini:
Tabel 1.1
Data Siswa

No.	Nama
1.	Rina Hastuti
2.	Ali Amran
3.	Rahmat

Nama file: latihan4_3.html

```
<html>
<head>
<title>Latihan4-3</title>
</head>
<body>
<table align="center" border="2" bgcolor="cyan" cellpadding="5" cellspacing="0" >
<caption align="top"><b>Tabel 1.1</b></caption>
<tr bgcolor="fuchsia">
<th colspan="2">Data Siswa</th>
</tr>
<tr bgcolor="green">
```

```
<th>No.</th>
<th>Nama</th>
</tr>
<tr bgcolor="yellow">
<td>1.</td>
<td>Rina Hastuti</td>
</tr>
<tr bgcolor="red">
<td>2.</td>
<td>Ali Amran</td>
</tr>
<tr bgcolor="blue">
<td>3.</td>
<td>Rahmat</td>
</tr>
</table>
</body>
</html>
```

HTML Gambar
ELEMENT IMG

Element IMG befungsi untuk menampilkan gambar. Element ini tidak mempunyai tag penutup dan mempunyai attribute yaitu align, alt (teks alternatif jika gambar tidak tampil), border, height,width, hspace (jarak kosong posisi horizontal, vspace (jarak kosong posisi vertikal), src (lokasi dari gambar itu disimpan beserta nama file-nya).

Sintaks:

```
<img align="left"|"right"|"top"|"middle"|"bottom" alt="text" border="pixel" width="pixel"|"%" height="pixel"|"%" hspace="pixel" vspace="pixel" src="uri">
```

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Tampilkan gambar image027.jpg dengan ukuran width dan height 160pixel dan border 8pixel :
Nama file: latihan5_1.html

```
<html>
<head>
<title>Latihan5-1</title>
</head>
<body>
<img width="160" height="160" src="image027.jpg">
</body>
</html>
```

Tugas tambahan:
Letakkan gambar tersebut ke kanan, setelah itu ke kiri.

Latihan 2:
Dari Latihan 1 ubah gambar menjadi yang terdapat dalam direktori /images dengan nama image050.jpg

Nama file: latihan5_2.html

```
<html>
<head>
<title>Latihan5-1</title>
</head>
<body>
<img width="160" height="160" src="./images/image050.jpg">
</body>
</html>
```

Latihan 3:
Tampilkan gambar image027.jpg dengan ukuran width dan height 80pixel dan border 8pixel serta teks seperti di bawah ini:

Kehadiran Web Teknisi Jardiknas dapat memudahkan penyebaran informasi antara Biro Perencanaan dan Kerja Sama Luar Negeri - dan anggota Teknisi Jardiknas. Keuntungan bila sistem jaringan informasi ini terbentuk, komunikasi antar Biro Perencanaan dan Kerja Sama Luar Negeri dan Provider dan anggota Teknisi Jardiknas akan lebih cepat terjalin, informasi dapat dengan cepat sampai dan tepat sasaran pula. Salam Pendidikan...! Dr. Ir. Gatot Hari Priowirjanto

Nama file: latihan5_3.html

```
<html>
<head>
<title>Latihan5-3</title>
</head>
<body>
<img width="80" height="80" src="image027.jpg" align="left" border="8">
Kehadiran Web Teknisi Jardiknas dapat memudahkan penyebaran informasi antara Biro Perencanaan dan Kerja Sama Luar Negeri - dan anggota Teknisi Jardiknas. Keuntungan bila sistem jaringan informasi ini terbentuk, komunikasi antar Biro Perencanaan dan Kerja Sama Luar Negeri dan Provider dan anggota Teknisi Jardiknas akan lebih cepat terjalin, informasi dapat dengan cepat sampai dan tepat sasaran pula.Salam Pendidikan...! Dr. Ir. Gatot Hari Priowirjanto
</body>
</html>
```

HTML Hyperlink

ELEMENT A

Element A befungsi untuk menghubungkan ke internal ataupun eksternal halaman web. Element ini mempunyai attribute href, name, dan target. Attribute href berfungsi sebagai hypertext reference yang mempunyai nilai URI (uniform resource identifier) yang akan menghubungkan ke eksternal halaman web. Attribute name berfungsi untuk menghubungkan ke tempat tujuan dalam halaman web (anchor). Attribute target berfungsi untuk menampilkan halaman web pada suatu jendela frame atau jendela baru.

Sintaks:

```
<a href="uri" name="name" target="_blank"|"parent"|"_top"|"_self">
.........................
</a>
```

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Membuat link eksternal:

Link:

1. yahoo
2. google
3. d3tkj

Nama file: latihan6_1.html

```
<html>
<head>
<title>Latihan6-1</title>
</head>
<body>
```

```
Link:
<ol>
<li>
<a href="http://www.yahoo.com">yahoo</a>
</li>
<li>
<a href="http://www.google.com">google</a>
</li>
<li>
<a href="http://teknisi.jardiknas.org">d3tkj</a>
</li>
</ol>
</body>
</html>
```

Tugas tambahan:
Tambahkan setiap link dengan target "_blank", "_parent", "_self", "_top". Amatilah hasilnya.

Latihan 2:

Membuat link internal dan anchor:
Link internal dan anchor adalah link yang menunjuk ketempat yang ada dalam dokument ini.

kebagian akhir

Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.

Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
kebagian pertama

Nama file: latihan6_2.html

```
<html>
<head>
<title>Latihan6-2</title>
</head>
<body>
Link internal dan anchor adalah link yang menunjuk ketempat yang
ada dalam dokument ini.<br>
<a href="#akhir">kebagian akhir</a>
<a name="pertama"></a>
<p>Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini.
Ini adalah teks bagian pertama dalam dokumen ini. </p>
<p>Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.
Ini adalah teks bagian kedua dalam dokumen ini.</p>
<p>Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
```

```
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.
Ini adalah teks bagian ketiga dalam dokumen ini.</p>
<p>Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.
Ini adalah teks bagian keempat dalam dokumen ini.</p>
<p>Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.
Ini adalah teks bagian kelima dalam dokumen ini.</p>
<a name="akhir"></a>
<p>Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.
Ini adalah teks bagian keenam dalam dokumen ini.</p>
<a href="#pertama">kebagian pertama</a>
</body>
</html>
```

Tugas tambahan:
Buatlah link ke bagian ketiga dari posisi bagian keenam.

HTML Form

ELEMENT FORM
Element FORM berfungsi untuk mendefinisikan form interaktif. Element ini mempunyai attribute yaitu action, dan method. Attribute action berisikan aksi terhadap form yang dikirim (URI) dan attribute method berisikan metode form melakukan proses pengiriman(GET/POST).

Sintaks:

```
<form action="action" method="GET"|"POST">
.........................
</form>
```

ELEMENT INPUT

Element INPUT berfungsi untuk mendefinisikan input yang akan dimasukkan pengguna. Element ini mempunyai attribute yaitu name, size, type, value, checked. Attribute name mendefinisikan nama dari input kontrol form, attribute size mendefinisikan ukuran teks pada input kontrol, type mendefinisikan bentuk-bentuk input kontrol, value mendefinisikan nilai awal/reset/submit, checked mendefinisikan pilihan terpilih pada type radio/checkbox. Element ini tidak mempunyai tag penutup dan harus berada di dalam element FORM.

Sintaks:

```
<input name="name"  size="number"  type="text"|"checkbox"|"radio"|"submit"|"reset"
value="value" checked >
```

ELEMENT SELECT

Element SELECT berfungsi untuk mendefinisikan opsi pilihan pada form kontrol. Element ini mempunyai attribute yaitu name, size, multiple(diizinkan banyak pilihan). Element ini harus berada di dalam element FORM.

Sintaks:

```
<select name="name"  size="number" multiple >
.....
</select>
```

ELEMENT OPTION

Element OPTION mendefinisikan opsi pilihan pada menu SELECT. Element ini mempunyai attribute selected, dan value. Attribute selected opsi terpilih dan attribute value berisikan nilai element OPTION.

Sintaks:

```
<option selected   ="number" >
.....
</option>
```

ELEMENT TEXTAREA

Element TEXTAREA berfungsi sebagai input kontrol form untuk memasukkan teks lebih dari satu baris. Element ini mempunyai attribute yaitu name, cols, rows. Attribute name mendefinisikan nama input kontrol form element textarea, attribute cols mendefinisikan jumlah kolom textarea yang terlihat, dan attribute rows mendefinisikan jumlah baris textarea yang terlihat. Element ini harus berada di dalam element FORM

Sintaks:

```
<textarea name="name" ="number" ="number" >
.....
</textarea>
```

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Buat form dengan input kontrol form seperti text, checkbox, radio, reset, submit, seperti tampilan di bawah ini:

Nama file: latihan7_1.html

```
<html>
<head>
<title>Latihan7-1</title>
</head>
<body>
<FORM METHOD="POST" ACTION=mailto:teknisi@jardiknas.org>
<H4>FORM</H4>
<INPUT TYPE="text" NAME="var1" SIZE="30" VALUE="Enter your name here.">
<BR><BR>
<B>Are you a student?</B>
<INPUT TYPE="checkbox" NAME="var2">
<BR><BR>
<B>How old are you?</B>
<BR>
<INPUT TYPE="radio" NAME="var3" VALUE="r1">10 – 15
<BR>
<INPUT TYPE="radio" NAME="var3" VALUE="r2">16 – 20
<BR>
<INPUT TYPE="radio" NAME="var3" VALUE="r3">21 – 25
<BR><BR>
<INPUT TYPE="submit" NAME="var4" VALUE="Send">
<INPUT TYPE="reset" NAME="var5" VALUE="Clear">
```

```
</FORM>
</body>
</html>
```

Jika diisi namanya adalah Rahmat ; student di check ; umur 21-25 ; maka jika form ini di submit data form yang terkirim adalah var1="Rahmat"&var2="1"&var3="r3"

Latihan 2:

Dari latihan 1 tambahkan menu select seperti gambar di bawah ini:

Nama file: latihan7_2.html

```
<html>
<head>
<title>Latihan7-2</title>
</head>
<body>
<FORM METHOD="POST" ACTION="mailto:teknisi@jardiknas.org">
<H4>FORM</H4>
<INPUT TYPE="text" NAME="var1" SIZE="30" VALUE="Enter your name here.">
<BR><BR>
<B>Are you a student?</B>
<INPUT TYPE="checkbox" NAME="var2">
<BR><BR>
<B>How old are you?</B>
<BR>
<INPUT TYPE="radio" NAME="var3" VALUE="r1">10 – 15
<BR>
<INPUT TYPE="radio" NAME="var3" VALUE="r2">16 – 20
<BR>
<INPUT TYPE="radio" NAME="var3" VALUE="r3">21 – 25
<BR><BR>
<B>Where are you from?</B>
<BR>
<SELECT NAME="var6" SIZE="1">
<OPTION VALUE="BG">Bulgaria
```

```
<OPTION VALUE="UK">United Kingdom
<OPTION VALUE="USA" SELECTED>USA
</SELECT>
<BR><BR>
<INPUT TYPE="submit" NAME="var4" VALUE="Send">
<INPUT TYPE="reset" NAME="var5" VALUE="Clear">
</FORM>
</body>
</html>
```

Catatan:
Pemberian nilai attribute action pada element FORM dengan menuju ke alamat email adalah untuk memperlihatkan nilai-nilai data form yang dikirimkan.

HTML FRAME

ELEMENT FRAMESET

Element FRAMESET berfungsi untuk membagi isi jendela web browser menjadi beberapa bagian. Element ini mempunyai attribute yaitu cols (membagi jendela berdasarkan kolom), rows (membagi jendela berdasarkan baris), border , bordercolor, frameborder (untuk menampilkan border frame), dan framespacing (spasi antara frame). Element FRAMESET terletak di antara element HEAD dan BODY.

Sintaks:

```
<frameset cols="list" rows="list" ="pixel" bordercolor="color" frameborder="yes"|"no" framespacing="pixel">
.........................

</frameset>
```

ELEMENT FRAME

Element FRAME mendefinisikan frame yang terdapat dalam frameset. Element ini mempunyai attribute yaitu name (nama frame), marginheight, marginwidth, bordercolor, frameborder, noresize, scrolling, src (dokumen yang ditampilkan pada frame). Element ini tidak mempunyai tag penutup.

Sintaks:

```
<frame name="name" marginheight="pixel" maginwidth="pixel" bordercolor="color" frameborder="yes"|"no" noresize scrolling="yes"|"no" src="uri">
```

ELEMENT NOFRAMES

Element NOFRAME mendefinisikan isi yang akan ditampilkan jika suatu frame tidak dapat diakses

Sintaks:

```
<noframes>
....................
</noframes>
```

LATIHAN

Gunakan teks editor misalkan "Notepad" untuk menyunting dan menyimpan script latihan di bawah ini. Untuk melihat hasilnya bukalah file tersebut dengan menggunakan web browser atau gunakan editor yang telah tersedia pada modul ini dengan mengklik menu Editor.

Latihan 1:

Bagilah jendela menjadi dua frame secara vertikal dengan komposisi 40% untuk dokumen left.html dan 60% untuk dokumen right.html.

Nama file: latihan8_1.html

```
<html>
<head>
<title>Latihan8-1</title>
</head>
<frameset cols="40%,60%">
<frame src="left.html">
<frame src="right.html">
</frameset>
<body>
</body>
</html>
```

Catatan: penulisan nilai attribute cols dapat juga ditulis cols="40%,*".

Latihan 2:

Dari latihan 1 ubahlah dengan memberikan warna merah pada border dengan ukuran 10pixel.

Nama file: latihan8_2.html

```
<html>
<head>
<title>Latihan8-2</title>
</head>
<frameset cols="40%,*" border="10" bordercolor="red">
<frame src="left.html">
<frame src="right.html">
</frameset>
<body>
</body>
</html>
```

PERKENALAN PHP

Apa itu PHP ?

Php merupakan salah satu bahasa pemrograman yang bayak digunakan oleh developer untuk mengembagkan suatu system informasi. Dia antaranya adalah di kalangan Pendidikan, perkantoran, kesehatan, dan lain sebagainya. Secara sederahana PHP adalah Bahasa pemrograman script server-side yang di desain untuk perkembangan web dan juga PHP bias diguankan sebagai Bahasa pemrogrman umum.

Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni kata dimana kepanjangannya terdiri dari singkatatan itu sendiri. PHP bersifat open source sehingga banyak digunakan untuk proyek open source.

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakan PHP/FI dengan perilisan kode sumber ini menjadi terbuka, maka banyak pemrograman yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0 dimana intreperter sudah diimplementasi dengan pemrogrmana C. dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall software server.

Versi terbaru dan stabil dari bahasa pemograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

Kenapa Belajar PHP?

Mungkin diluar sana banyak sekali bahasa pemrograman yang di tawarkan untuk di kembangkan. Namun pembelajaran dibuku ini memberikan sebuah pilihan untuk para pemula yang belum kenal dengan pemrograman bisa memilih dan belajar bagaimana belajar dengan PHP.

Berikut mengapa kita harus belajar PHP:

1. Gratis

Artinya siapa saja bisa menggunakan bahasa pemrograman ini, dan banyak domain menggunakan PHP mudah untuk diguanakan.

2. Cross Platform
   Artinya di guankan di berbagasi system operasi, mulai dari windows, linux dan mac.

3. Mendukung banyak database
   PHP telah banyak mendukung database untuk pengembangannya, seperti Frontbase mSQL, Direct MS-SQL, MySQL, ODBC, Oracle, Ovrimos, PostgresSQL, SQL Lite, dan lain-lain.

4. On The Fly
   Artinya dapat membuat dokumen text, word, exel, pdf, menciptakan images dan flas, dan lain sebagainya.

Karakteristik PHP

Berikut karakteristik pada pemrograman PHP,

a) Extension harus .php
   Setiap sintak / skrip PHP harus disimpan dengan extention .php, misal test.php. Jika didalam file .php tidak ada skrip PHP tidak menjadi masalah akan tetap diproses misal didalam file .php isinya hanya skrip html saja maka akan tetap di proses.
b) Sintax harus ditulis dalam dilementer atau sering disebut dengan tag PHP
   Sintak PHP tidak akan perjalan atau tidak akan diproses jika sintak tersebut tidak diletakan didalam delimenter php yaitu
   <?php dan ?>.
   <?php //pembuka
   //Disini sintak PHP
   ?> //penutup
c) Sintak PHP mengikuti bahasa induknya bahasa C
   Seperti pada artikel **sejarah php** awalnya PHP merupakan sebuah program yang dibuat menggunakan bahasa C untuk menanganai sebuh form untuk koneksi ke database. sehingga sintak PHP hampir sama dengan baha C, seperti case sensitive dan diakhiri tanda ; disetiap akhir dari sintak.
d) PHP dapat diguankan bersamaan dengan HTML
   PHP karana merupakan bahasa pemograman khusus untuk web maka sangat dengan mudah digunakan bersmaan dengan html, baik tag html didalam PHP atau sebaliknya.
e) PHP merupakan bahasa server side scripting
   Artinya perlu penerjemah atau kompilasi dari sisi server. Salah satu software yang mendukung PHP adalah apache2.
f) PHP Open Source
   PHP dapat digunakan secara gratis dan bebas dikembangkan.
g) PHP Multi Platform
   Dapat dijalankan diberbagai platfrom OS seperti linux, windows, dan mac yang membuat bahasa pemograman ini banyak diminati.

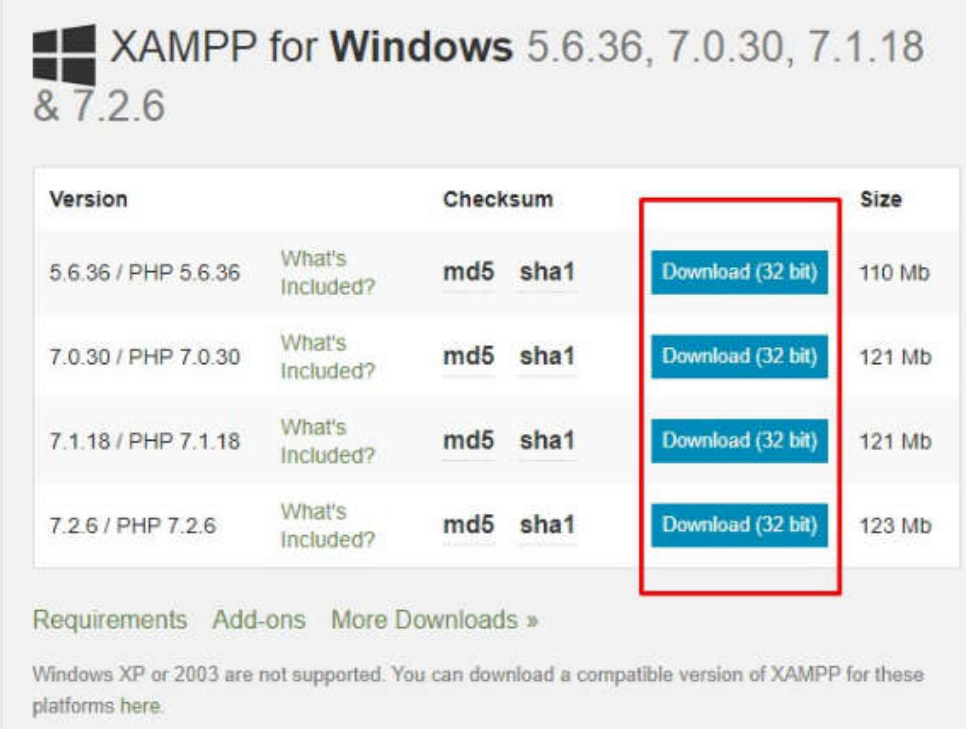
Perisapan Lingkungan PHP

Untuk bekerja dengan PHP, perlu beberapa aplikasi yang dibutuhkan, yaitu

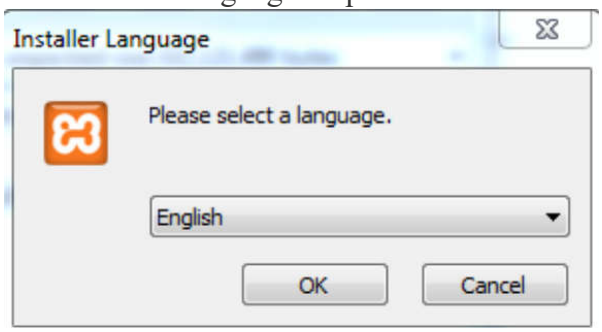
1. Web server (Apache, IIS, Personal Web Server/PWS)
2. PHP server (dapat di download di php.net)
3. Database Server (MySQL dan lain-lain)
4. Web editor (Sublime Text, Visual Studi Code)

Untuk menggunakan tools aplikasi yang didalam nya sudah terdapat web server (Apache), PHP Server, dan MYSQL yang terintegrasi menjadi satu. Tools tersebut dapat diinstal di PC sebagai saran belajar PHP. Salah satu contohnya adalah XAMPP yang bisa dijalankan di windows, LAMP yang bisa diajalankan di Linuk, dan lain -lain.

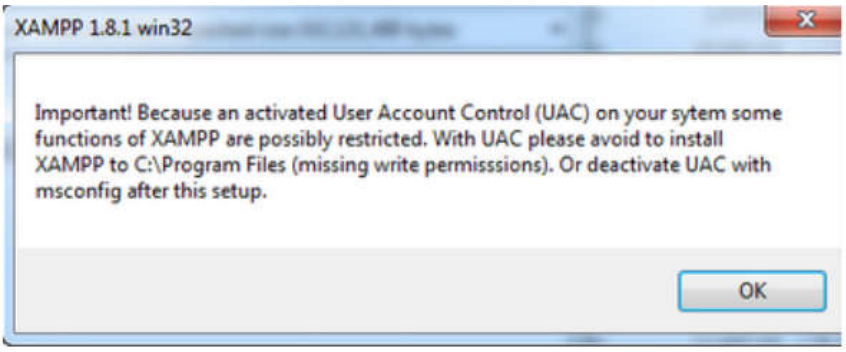
- Instal XAMPP (untuk windows)
  - Download aplikasi xampp terlebih dahulu di **https://www.apachefriends.org/download.html** jika bingung memilih versi yang diguankan maka gunakan versi 7.2.6/ PHP 7.2.6. seperti gambar di bawah ini

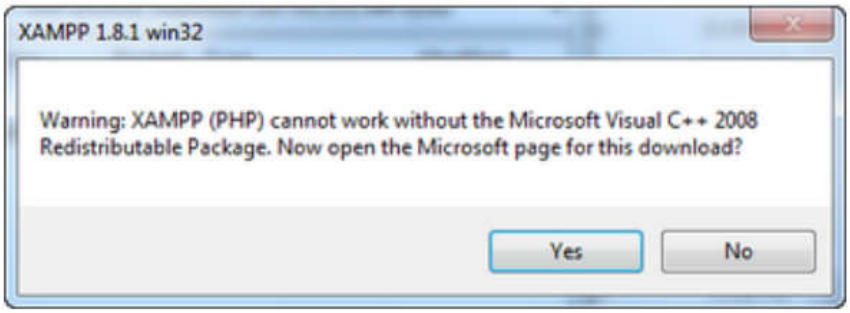


  - Dobel klik file XAMPP yang baru saja Anda download, nanti selanjutnya akan muncul jendela "installer language" seperti di bawah ini:



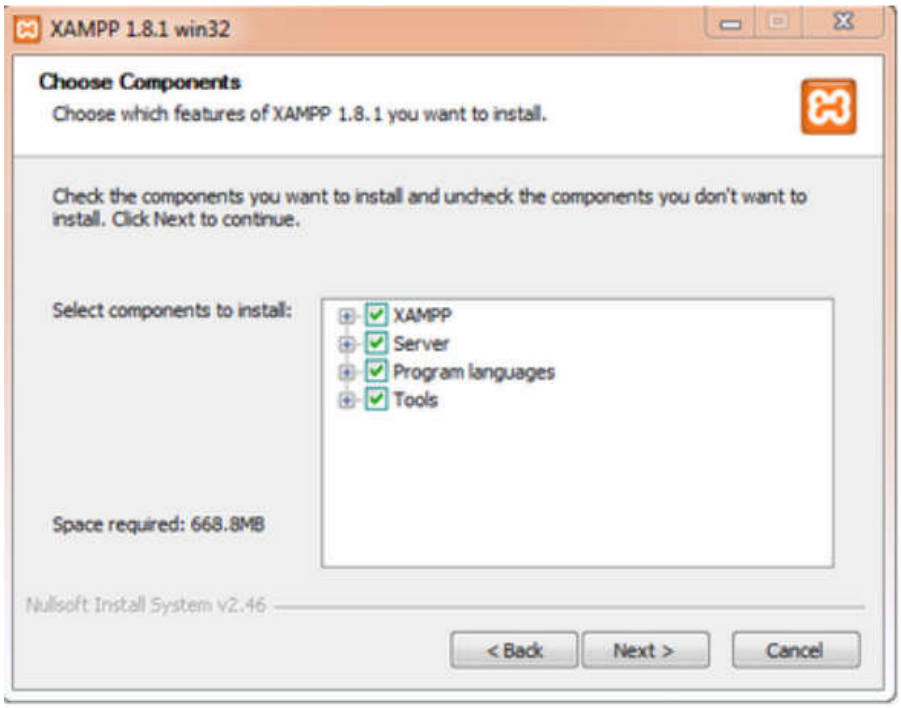
  - Selanjutnya pilih bahasa. Pilih yang Bahasa Inggris (English). Klik OK.
  - Kadang pada proses ini muncul pesan error. Jika ada, abaikan saja dan lanjutkan dengan klik OK dan YES.



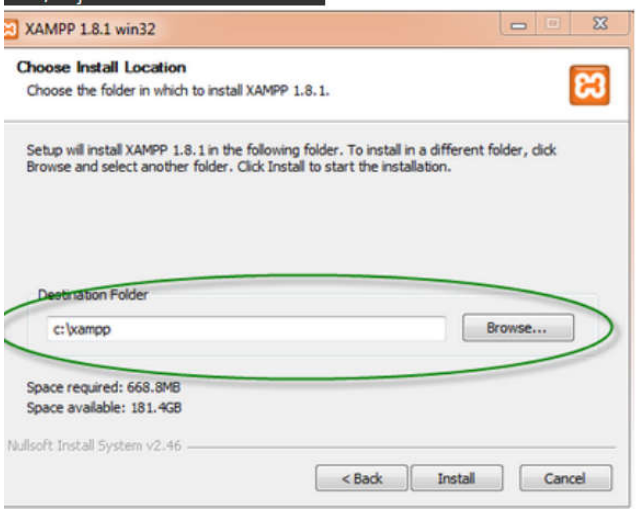


- Berikutnya akan muncul jendela yang isinya meminta Anda menutup semua aplikasi yang sedang berjalan. Jika semua aplikasi sudah ditutup, maka klik tombol Next.

- Selanjutnya akan diminta untuk memilih aplikasi yang mau diinstal. Centang saja semua pilihan dan klik tombol **Next**.



- Kemudian Anda akan diminta untuk menentukan lokasi folder penyimpanan file-file dan folder XAMPP. Secara default akan diarahkan ke lokasi c:\xampp. Namun jika Anda ingin menyimpannya di folder lain bisa klik browse dan tentukan secara manual folder yang ingin digunakan. Jika sudah selesai, lanjutkan dan klik tombol Install.



- Tunggu beberapa menit hingga proses intalasi selesai. Jika sudah muncul jendela seperti di bawah ini, klik tombol Finish untuk menyelesaikannya.

Berikutnya, akan muncul jendela dialog seperti gambar di bawah ini yang menanyakan Anda apakah mau langsung menjalankan aplikasi XAMPP atau tidak. Jika ya, maka klik YES.

❖ Instal Text Editor untuk PHP

Banyak tools yang menyediakan untuk text editor php, misalnya seperti notepade, notepade++, sublime text, visual studio code, dreamweaver, dan lain sebagainya. Silahkan pilih salah satunya saja . untuk instalannya silahkan cek di google.com karena sudah banyak sekali tutorialnya.

Sintax PHP

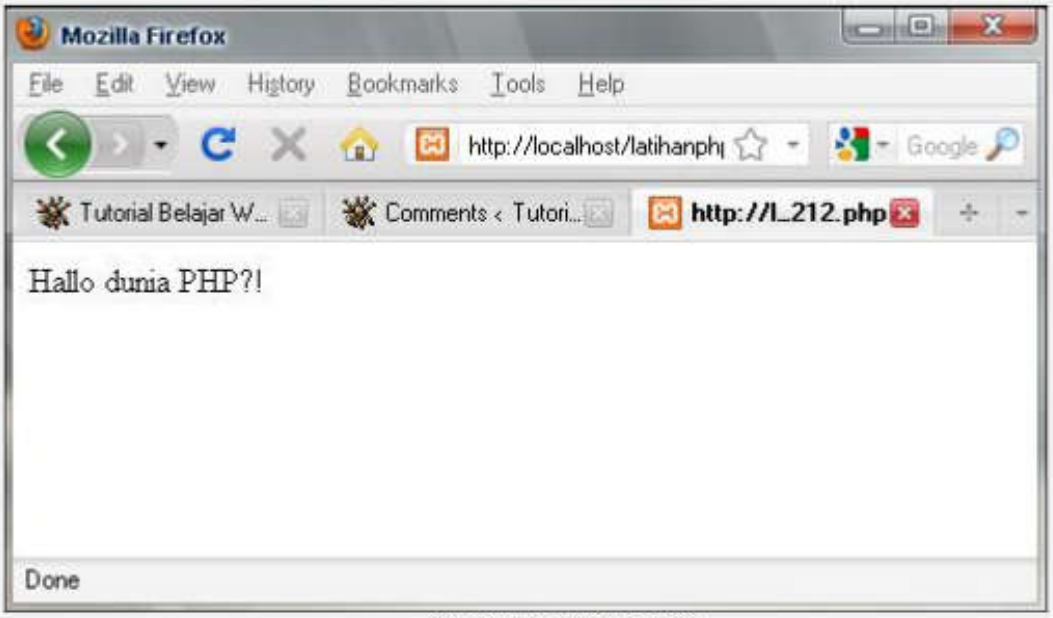
Secara sederhana sintax penulisan PHP sangat simple yaitu :

```
>?php
//syntax php di letakkan disini …
//kode php di akhiri Degnan titik koma  ‘;’
?>
```

Untuk mengetikkan baris code php, maka anda perlu mengapitnya dengan tag <?php dan di akhir dengan ?> karena di luar dari itu php engine tidak menganggapnya sebagai script php, melainkan hanya html biasa. Sebagai contoh kita akan menggunakan php, untuk menampilkan kalimat "Hallo dunia PHP?!" kita cukup membuka notepad, lalu ketikan seperti ini di notepad :

```
<?php
Echo”Hallo dunia PHP?! “;
?>
```

Maka akan muncul tampilan seperti berikut ini



Gambar 2.2 (Hallo PHP?)

❖ **Variable**

Misalkan dalam PHP kita akan menyimpan suatu nilai berupa angka atau string dalam suatu variable, caranya adalah membuat nama variable terlebih dahulu kemudian diberikan suatu assigmnet pada nilai yang di inginkan. Perhatikan sintaxs berikut ini;

$nama_variabel = nilai;

Note : jangan lupa tanda dollar ($)

Contoh:

```
<?php
$hello = "Hello World";
$sebuah_bilangan = 4;
$bilanganYangLain = 8;
```

Dari contoh di atas tampak bahwa dalam PHP, nama variable tidak perlu dideklarasikan terlebih dahulu seperti halnya bahasa pemrogrman C/C++.

Ada beberapa aturan dalam penamaan variable yaitu :

- Nama variable harus diawali dengan huruf atau underscore(_)
- Nama variable hanya boleh dituliskan dengan alpha numeric a-z, A-Z,)-9 dan underscore
- Nama variable yang terlebih dahulu dari satu kata, dapat dipisahkan dengan underscore

❖ **Echo**

Seperti yang kita lihat pada contoh-contoh kode PHP sebelumnya, bahwa printah echo digunakan untuk menampilkan teks ke dalam browser. Suatu teks atau string dapat dituliskan di browser dengan langsung dituliskan dalam echo yang diapit oleh tanda petik ganda (quotes) atau menyimpan string atau teks terlebih dahulu dalam suatu variable kemudian dituliskan dalam echo. Berikut contohnya:

```
<?php
$StringKu = "Hello";
```

```
Echo” $StringKu;
Echo”<h1>I Love using PHP!</h1>”;
```

❖ **Operator**

- Operator Aritmatika

Berikut ini adalah table operator aritmatik, makna dan contohnya

Operator	Makna	Contoh
+	Pejumlahan	2 + 4
-	Pengurangan	6 – 3
*	Perkalian	2 * 2
/	Pembagian	10 / 2
%	Modulus	43 % 10

Contoh :

```
<?php
$penjumlahan = 2 + 4;
$pengurangan = 6 - 2;
$perkalian = 5 * 3;
$pembagian = 15 / 3;
$modulus = 5 % 2;
echo "Menampilkan penjumlahan: 2 + 4 = ".$penjumlahan."<br>";
echo "Menampilkan pengurangan: 6 - 2 = ".$pengurangan."<br>";
echo "Menampilkan perkalian: 5 * 3 = ".$perkalian."<br>";
echo "Menampilkan pembagian: 15 / 3 = ".$pembagian."<br>";
echo "Menampilkan modulus: 5 % 2 = " . $modulus.";
?>
```

- Operator Perbandingan

Perbandingan digunakan untuk menguji hubungan antara nilai dan atau variabel. Operator ini digunakan dalam suatu statement bersyarat yang selalu menghasilkan nilai TRUE atau FALSE.

Misalkan:

$x = 4; $y = 5;

berikut ini adalah beberapa contoh penggunaan operator perbandingan dan hasilnya.

Operator	Makna	Contoh	Hasil
==	Sama dengan	$x == $y	False
!=	Tidak sama dengan	$x != $y	True
<	Lebih kecil dari	$x < $y	True
>	Lebih besar dari	$x > $y	False
<=	Lebih kecil atau sama dengan dari	$x <= $y	True
>=	Lebih besar atau sama dengan dari	$x >= $y	False

❖ Modularitas

Suatu program yang baik seharusnya program yang besar dipecahkan menjadi program-program yang kecil yang selanjutnya disebut modul. Modul-modul kecil tersebut dapat dipanggil sewaktu-waktu diperlukan. Dalam PHP juga sudah mendukung konsep tersebut

yang selanjutnya diberinama modularitas. Kita dapat menyisipkan isi suatu file/modul lain ke dalam file/modul tertentu.

- Include ()

  Untuk memudahkan pemahaman, diberikan contoh berikut. Misalkan kita akan membuat menu link sejumlah 4 buah yaitu index, about, links, dan contact pada setiap halaman web yang kita buat. Teknik yang digunakan adalah membuat menu link dalam suatu modul tersendiri kemudian modul tersebut dipanggil pada setiap halaman web yang diinginkan terdapat menu link di dalamnya.

  menu.php

```
<html>
<body>
<a href="index.php">Home</a> -
<a href="about.php">About Us</a> -
<a href="links.php">Links</a> -
<a href="contact.php">Contact Us</a> <br>
```

  index.php

```
<?php
include("menu.php");
?>
<p>Ini adalah halaman index</p>
</body>
</html>
```

  about.php

```
<?php
include("menu.php");
?>
<p>Ini adalah halaman about</p>
</body>
</html>
```

  Dari teknik di atas tampak adanya kemudahan dalam membuat halaman web. Dalam hal ini, kita tidak perlu membuat menu link di setiap halaman web yang ada. Bayangkan seandainya kita punya halaman web sejumlah 100 buah yang kesemuanya ingin diberi menu link tanpa menggunakan teknik di atas, tentu hal tersebut sangat merepotkan. Meskipun secara teknis, kode pembangun web dipecah dalam modul-modul, namun ketika di browser akan terlihat menyatu. Berikut ini adalah kode HTML yang dihasilkan oleh browser ketika membuka halaman web index.php

```
<html>
<body>
        <a href="index.php">Home</a> -
        <a href="about.php">About Us</a> -
        <a href="links.php">Links</a> -
        <a href="contact.php">Contact Us</a> <br>
        <p>Ini adalah halaman index</p>
</body>
```

```
</html>
```

- Require()

  Cara penggunaan maupun fungsi dari require() sama dengan include(). Jadi apa perbedaannya? Sebaiknya mana yang kita gunakan? Perhatikan contoh berikut ini.

```
<?php
include("noFileExistsHere.php");
echo "Hello World!";
?>
```

  dengan asumsi bahwa file noFileExistxHere.php tidak ada.

  Maka dengan menggunakan include() akan dihasilkan tampilan:

```
Warning: main(noFileExistsHere.php): failed to open stream: No such file or directory in include.php on line 2
Warning: main(): Failed opening 'noFileExistsHere.php' for inclusion (include_path='.:/usr/lib/php:/usr/local/lib/php') in include.php on line 2
Hello World
```

  Selanjutnya kita akan gunakan require().

```
<?php
        require("noFileExistsHere.php");
        echo "Hello World!";
?>
```

  dan hasilnya

```
Warning: main(noFileExistsHere.php): failed to open stream: No such file or directory in require.php on line 2
Fatal error: main(): Failed opening required 'noFileExistsHere.php' (include_path='.:/usr/lib/php:/usr/local/lib/php') in require.php on line 2
```

  Bandingkan kedua hasil di atas, khususnya yang tercetak merah. Pada include(), error yang dihasilkan hanya berupa Warning saja dan statement berikutnya masih dapat dijalankan. Hal ini terlihat bahwa teks Hello World! Masih ditampilkan di browser. Sedangkan pada require(), error yang dihasilkan berupa Fatal Error. Dengan demikian statement selanjutnya tidak akan dijalankan.

❖ Array

Dalam PHP, index untuk array dapat berupa numerik dan dapat pula berupa satu nilai atau sering disebut Degnan array assosiatif.

- Array Berindeks Numerik

  Sintaks umum untuk menyatakan suatu array berindeks numerik beserta nilainya adalah

```
$nama_array[$key] = value;
```

  Dalam hal ini $key berupa bilangan bulat mulai dari 0, 1, 2, …

  Contoh:

```
$karyawan[0] = "Bob";
$karyawan[1] = "Sally";
$karyawan[2] = "Charlie";
$karyawan[3] = "Clare";
```

Sedangkan berikut ini adalah contoh untuk menampilkan nilai dari suatu array berindeks numerik.

```
<?
        $karyawan[0] = "Bob";
        $karyawan[1] = "Sally";
        $karyawan[2] = "Charlie";
        $karyawan[3] = "Clare";
echo "Berikut ini adalah 2 orang karyawan saya, yaitu ". $karyawan[0] . " & " . $karyawan[1];
echo "<br>Dua orang karyawan saya yang lain adalah ". $karyawan[2] . " & " . $karyawan[3];
?>
```

- Array Assosiatif

  Untuk array assosiatif, sintaksnya sama dengan array berindeks numerik namun perbedaannya adalah pada $key. Pada array assosiatif, $key dapat berupa suatu string. Berikut ini adalah

  Contohnya

```
$gaji["Bob"] = 2000;
$gaji["Sally"] = 4000;
$gaji["Charlie"] = 600;
$gaji["Clare"] = 0;
```

dan berikut ini adalah contoh kode untuk menampilkan nilai dari array assosiatif

```
<?
        $gaji["Bob"] = 2000;
        $gaji["Sally"] = 4000;
        $gaji["Charlie"] = 600;
        $gaji["Clare"] = 0;
        echo "Bob digaji - $" . $gaji["Bob"] . "<br>";
        echo "Sally digaji - $" . $gaji["Sally"] . "<br>";
        echo "Charlie digaji - $" . $gaji["Charlie"] . "<br>";
        echo "dan Clare digaji - $" . $gaji["Clare"];
?>
```

❖ Statement Kontrol

- Statement If …

Sintaks dari statement tersebut dalam PHP adalah

```
if (syarat)
{
    statement1;
    statement2;
    .
    .
```

```
}
```

Untuk menyatakan syarat, biasanya digunakan operator perbandingan. Apabila syarat bernilai TRUE maka statement-statement yang diapit dengan tanda kurung kurawal akan dijalankan. Bentuk lain dari sintaks IF adalah

```
    if (syarat)
    {
    statement1;
    statement2;
    .
    .
    }
    else
    {
    statement3;
    statement4;
    .
    .
    }
```

Untuk sintaks kedua di atas, statement3, statement4, dst akan dijalankan apabila syarat bernilai FALSE.

Contoh:

```
<?
    $my_name = "nada";
    if ($my_name == "nada")
    {
    echo "Your name is ".$my_name."!"<br>";
    }
    echo "Welcome to my homepage!";
    ?>
    <?
    $number = 3;
    if ($number == 4)
    {
    echo "Benar";
    }
    else
    {
    echo "Salah";
    }
?>
```

Terdapat pula bentuk sintaks berikutnya dari IF … yaitu dengan ditambahkan elseif

```
if (syarat1)
{
```

```
    statement11;
    statement12;
    .
    .
    }
    elseif (syarat2)
    {
    statement21;
    statement22;
    .
    .
    }
    .
    .
    else
    {
    statement1;
    statement2;
    .
    .
}
```

Jika syarat1 bernilai TRUE, maka statement11, statement12 dst akan dijalankan. Sedangkan jika syarat1 FALSE maka selanjutnya akan dicek untuk syarat2. Jika syarat2 TRUE maka statemen21, statement22, dst akan dijalankan, sedangkan jika syarat2 FALSE akan dicek syarat berikutnya (jika masih ada). Statement1, statement2, dst baru akan dijalankan apabila semua syarat sebelumnya bernilai FALSE.

Contoh:

```
<?php
$karyawan = "Bob";
if($karyawan == "Tanner")
{
    echo "Hello Tanner!";
    }
    elseif($karyawan == "Bob")
    {
    echo "Hello Bob!";
    }
    else
    {
    echo "Hello!";
}
?>
```

- Statement SWITCH

Sintaks dari statement ini adalah

```
switch (variabel)
{
    case option1:
    statement11;
    statement12;
    .
    .
    break;
    case option2:
    statement21;
    statement22;
    .
    .
    break;
    .
    .
    default:
    statementdefault1;
    statementdefault2;
    .
    .
    break;
}
```

Pada sintaks di atas, nilai dari variabel akan dicek pada setiap option yang ada (terletak di bagian case). Jika ada option yang sama dengan nilai variabel, maka statement-statement di bawah option tersebutlah yang akan dijalankan. Bagian default adalah optional (boleh ada, boleh tidak).

Contoh:

```
<?php
$tujuan = "Tokyo";
echo "Biaya Perjalanan Menuju $tujuan adalah ";
switch ($tujuan){
    case "Las Vegas":
    echo " $500";
    break;
    case "Amsterdam":
    echo " $1500";
    break;
    case "Egypt":
    echo " $1750";
    break;
    case "Tokyo":
    echo " $900";
    break;
    case "Caribbean Islands":
```

```
    echo " $700";
    break;
    }
?>
```

Contoh:

```
<?php
$tujuan = "New York";
echo "Biaya Perjalanan Menuju $tujuan adalah ";
switch ($tujuan){
    case "Las Vegas":
    echo " $500";
    break;
    case "Amsterdam":
    echo " $1500";
    break;
    case "Egypt":
    echo " $1750";
    break;
    case "Tokyo":
    echo " $900";
    break;
    case "Caribbean Islands":
    echo " $700";
    break;
    default:
    echo " $100";
    break;
    }
?>
```

- Statement WHILE

Statement ini digunakan untuk mengerjakan suatu statement secara berulang-ulang sampai suatu syarat dipenuhi. Sintaksnya adalah

```
while (syarat)
{
    statement;
    statement;
    .
    .
}
```

Pada sintaks di atas, selama syarat bernilai TRUE maka statement-statement di dalam while akan terus dijalankan secara berulang-ulang. Perulangan baru akan berhenti apabila syarat bernilai FALSE. Sebelum statement yang diulang-ulang dilakukan, terlebih dahulu akan dicek syarat nya apakah bernilai TRUE atau FALSE. Apabila TRUE maka statement akan

dijalankan. Sedangkan apabila FALSE, perulangan akan langsung berhenti. Dengan kata lain, statement dalam WHILE bisa jadi tidak akan pernah dilakukan, yaitu apabila syaratnya langsung bernilai FALSE.

Contoh:

```
<?
    $harga_sikat = 1500;
    $jumlah_sikat = 10;
    echo "<table border=\"1\" align=\"center\">";
    echo "<tr><td><b>Jumlah Sikat</b></td>";
    echo "<td><b>Harga</b></tr></td>";
    while ( $jumlah_sikat <= 100 )
    {
    echo "<tr><td>";
    echo $jumlah_sikat;
    echo "</td><td>";
    echo "Rp. ".$harga_sikat * $jumlah_sikat;
    echo "</td></tr>";
    $jumlah_sikat = $jumlah_sikat + 10;
    }
    echo "</table>";
?>
```

Kode di atas akan menampilkan hasil di browser berupa tabel yang berisi jumlah sikat dan harganya, dengan asumsi harga sebuah sikat adalah Rp. 1.500. Jumlah sikat yang ditampilkan adalah kelipatan 10 dengan batas sampai 100 buah.

- Statement FOR

Statement FOR mirip dengan WHILE yang memiliki sintaks berikut ini

```
for (inisialisasi counter; syarat; increment/decrement counter)
{
    statement;
    .
    .
}
```

Untuk memperjelas pemahaman tentang FOR, berikut ini adalah contoh kode dengan for untuk menghasilkan tampilan yang sama dengan contoh while sebelumnya (tentang jumlah sikat dan harganya). Coba bandingkan dengan kode contoh while sebelumnya.

Contoh:

```
<?
    $harga_sikat = 1500;
    echo "<table border=\"1\" align=\"center\">";
    echo "<tr><td><b>Jumlah Sikat</b></td>";
    echo "<td><b>Harga</b></td></tr>";
    for ($jumlah_sikat = 10; $jumlah_sikat <= 100; $jumlah_sikat+=10)
    {
    echo "<tr><td>";
```

```
    echo $jumlah_sikat;
    echo "</td><td>";
    echo "Rp. ".$harga_sikat * $jumlah_sikat;
    echo "</td></tr>";
    }
    echo "</table>";
?>
```

- Statement Foreach

Misalkan Anda punya data berupa array assosiatif yang akan diproses secara berulang-ulang, maka PHP menyediakan statement foreach yang mudah digunakan.
Sintaksnya adalah:

```
foreach(variabelarray as kunci => value)
{
    statement;
    .
    .
}
```

Sebagai contoh, misalkan Anda memiliki 5 orang karyawan dengan usianya masing-masing yang ditulis dalam kode PHP sebagai berikut

```
$UsiaKaryawan["Lisa"] = "28";
$UsiaKaryawan["Jack"] = "16";
$UsiaKaryawan["Ryan"] = "35";
$UsiaKaryawan["Rachel"] = "46";
$UsiaKaryawan["Grace"] = "34";
```

Berikut ini adalah contoh kode PHP yang akan menampilkan semua karyawan beserta usianya dengan menggunakan foreach.

```
<?php
    $UsiaKaryawan["Lisa"] = "28";
    $UsiaKaryawan["Jack"] = "16";
    $UsiaKaryawan["Ryan"] = "35";
    $UsiaKaryawan["Rachel"] = "46";
    $UsiaKaryawan["Grace"] = "34";
    foreach($UsiaKaryawan as $Nama => $umur)
    {
    echo "Nama Karyawan: $Nama, Usia: $umur"." th <br>";
    }
?>
```

- Statement Do While

Statement ini merupakan bentuk modifikasi dari WHILE. Sintaksnya adalah sebagai berikut
Do

```
    {
    statement;
    .
    .
    }
while (syarat);
```

Coba bandingkan dengan sintaks WHILE sebelumnya. Dilihat dari posisi statement yang diulang, posisi statement yang diulang pada DO WHILE terletak di atas syarat. Dengan demikian, sebelum syarat dicek TRUE atau FALSE nya, statement akan dikerjakan terlebih dahulu. Sedangkan pada WHILE, sebelum statement yang diulang dikerjakan, terlebih dahulu syarat akan dicek. Prinsip kerja DO WHILE sama dengan WHILE yaitu statement akan terus dikerjakan selama syarat bernilai TRUE dan perulangan akan berhenti apabila FALSE. Perhatikan contoh berikut ini yang membandingkan DO WHILE dengan WHILE
Contoh:

```
<?php
    $kue = 0;
    while($kue > 1)
    {
    echo "Mmmmm...Aku suka kue! Nyam..nyam..nyam..";
    }
    ?>
    <?
    $kue = 0;
    do
    {
    echo "Mmmmm... Aku suka kue! Nyam..nyam..nyam..";
    } while ($kue > 1);
?>
```

Pada kode WHILE, teks "Mmmm…. " dst tidak akan ditampilkan karena syaratnya langsung bernilai FALSE (perulangan berhenti). Sedangkan pada DO WHILE, teks akan ditampilkan dahulu kemudian perulangan berhenti (syarat bernilai FALSE).

- Bekarja dengan Form

  Setelah kita belajar dasar-dasar perintah PHP, sekarang saatnya mengaplikasikannya pada aplikasi web. Biasanya PHP digunakan sebagai pengolah data yang diinputkan melalui form yang dibuat dengan HTML. Sebagai contoh, andaikan Anda memiliki toko virtual dalam web yang menjual alat-alat tulis seperti pensil, buku tulis, dan ballpoint. Berikut ini adalah salah satu bentuk kode HTML yang digunakan untuk membuat form pemesanan pembelian barang-barang tersebut.

```
<html><body>
<h2>Toko Alat Tulis Amalia</h2>
<form action="proses.php" method="post">
        <select name="barangpesanan">
        <option>Pensil</option>
        <option>Buku Tulis</option>
```

```
        <option>Ballpoint</option>
        </select>
        Jumlah pesanan: <input name="jumlah" type="text">
        <input type="submit" value="Submit">
</form>
</body></html>
```

Seperti yang Anda lihat pada kode HMTL di atas, perintah action="proses.php" digunakan untuk mengarahkan ke file PHP yang digunakan untuk memproses barang pembelian ketika tombol submit ditekan. Dalam form yang dihasilkan dari kode di atas terdapat 2 buah komponen input yaitu berbentuk combobox dan textbox. Untuk combobox, diberi nama “barangpesanan” (perhatikan perintah <select name="barangpesanan">) dan textbox diberi nama “jumlah” (perhatikan perintah <input name="jumlah" type="text">).
Penjelasan mengenai method="post" akan dijelaskan pada bab berikutnya. Sedangkan berikut ini adalah salah satu contoh kode PHP untuk memproses input dari form di atas. Kode PHP ini disimpan dengan nama proses.php.

```
<html><body>
        <?php
        $jumlah = $_POST['jumlah'];
        $barangpesanan = $_POST['barangpesanan'];
        echo "Anda memesan ". $jumlah . " " . $barangpesanan . ".<br>";
        echo "Terima kasih atas kesediaan Anda memesan barang dari kami!";
        ?>
</body></html>
```

Kalau Anda perhatikan, terdapat keterkaitan perintah $_POST['xxx']; dengan "xxx" pada name = "xxx" (nama komponen input).

- Post and Get Method

Pada contoh sebelumnya, kita mengirim data input dari form menuju ke file PHP untuk diproses menggunakan metode post. Selain metode tersebut, terdapat pula metode get. Lantas perbedaannya apa? Kapan kita gunakan metode post atau get? Itulah yang akan dibahas pada bab ini. Untuk melihat perbedaan post dan get, kita akan sedikit mengubah file HTML form dan file PHP proses.php sebelumnya.

```
<html><body>
<h2>Toko Alat Tulis Amalia</h2>
<form action="proses.php" method="get">
        <select name="barangpesanan">
        <option>Pensil</option>
        <option>Buku Tulis</option>
        <option>Ballpoint</option>
        </select>
        Jumlah pesanan: <input name="jumlah" type="text">
        <input type="submit" value="Submit">
</form>
</body></html>
```

dan isi proses.php nya adalah

```
<html><body>
<?php
	$jumlah = $_GET['jumlah'];
	$barangpesanan = $_GET['barangpesanan'];
	echo "Anda memesan ". $jumlah . " " . $barangpesanan . ".<br>";
	echo "Terima kasih atas kesediaan Anda memesan barang dari kami!";
?>
</body></html>
```

Perbedaan kode HTML dan PHP di atas dengan sebelumnya adalah yang dicetak merah. Metode pengiriman data input dari form menggunakan get, dan dalam proses.php $_POST diganti dengan $_GET.

Apabila aplikasi di atas dijalankan, maka secara sekilas hasil yang tampak sama dengan ketika digunakan metode post. Namun, coba perhatikan URL yang tampak ketika proses.php muncul. Pada URL tersebut terdapat tambahan ?barangpesanan=...&jumlah=... setelah nama file (proses.php). Titik-titik tersebut akan diisi dengan data sesuai dengan yang diinputkan pada form. Coba bandingkan dengan URL ketika digunakan metode post. Data isian pada form tidak ditampilkan pada URL. Sehingga inilah perbedaan antara keduanya. Dengan demikian, hendaknya kita jangan menggunakan metode get ketika akan memproses data input melalui form. Bayangkan seandainya form tersebut digunakan untuk login atau untuk keperluan yang menyangkut privasi. Apabila Anda gunakan metode get, maka semua input data akan ditampilkan pada URL. Bisa-bisa password Anda akan kelihatan di URL (jika terdapat input password ketika login). Untuk metode get, biasanya digunakan untuk input data melalui link (bukan melalui form). Untuk contoh aplikasinya dapat dilihat pada contoh-contoh aplikasi pada bab-bab berikutnya.

- Function

Sebuah function merupakan sebuah nama yang kita berikan pada suatu blok program yang sewaktu-waktu dapat kita panggil dan gunakan. Sebuah function dapat diletakkan di bagian manapun, bisa di awal, tengah, dan akhir dari keseluruhan bagian kode PHP.

Berikut ini adalah contoh membuat sebuah function sederhana yang di dalamnya tidak ada statementnya.

Contoh:

```
<?php
	function myCompanyMotto()
	{
	}
	myCompanyMotto();
?>
```

Pada contoh di atas, myCompanyMotto merupakan nama function. Nama function inilah yang dapat dipanggil sewaktu-waktu diperlukan. Aturan membuat nama function sama dengan ketika membuat nama variabel (lihat di Bab II). Statement/perintah dari function dituliskan di dalam kurung kurawal {}. Sedangkan perintah myCompanyMotto(); bagian paling bawah dari kode di atas merupakan cara memanggil function. Perhatikan contoh berikutnya:

Contoh:

```
<?php
        function myCompanyMotto()
        {
        echo "Sabar adalah bagian dari keimanan";
        }
        myCompanyMotto();
?>
```

Pada contoh tersebut, terdapat perintah echo di dalam function. Sehingga begitu nama function dipanggil, PHP akan menampilkan teks yang di-echo-kan tersebut.
Sebuah function dapat dipanggil berulang-ulang, seperti pada contoh berikut.

```
<?php
        function myCompanyMotto()
        {
        echo "Sabar adalah bagian dari keimanan";
        }
        echo "Selamat datang di PT. Nada Corp. <br>";
        myCompanyMotto();
        echo "Terima kasih atas kunjungan Anda<br>";
        echo "dan ingatlah selalu... <br>";
        myCompanyMotto();
?>
```

❖ Operasi File

- Membuka File

Secara umum terdapat 3 cara membuka file, yaitu membuka file hanya untuk dibaca (read: ‘r’), hanya untuk ditulisi baru (write: ‘w’), dan hanya untuk ditambahi isinya (append: ‘a’). Selain 3 cara membuka file tersebut, terdapat pula cara lain membuka file yaitu dapat dibaca dan ditulisi (read/write: ‘r+’), serta dapat dibaca dan ditambahi isinya (append: ‘a+’).
Berikut ini adalah contoh kode PHP untuk membuka file

```
<?
    $nama_file = "test.txt";
    $fh = fopen($nama_file, 'X') or die("File tidak bisa dibuka");
    fclose($fh);
?>
```

dengan 'X' dapat diganti dengan 'w', 'r', 'a', 'r+', 'a+'.
Apabila nama file yang akan dibuka ternyata salah, atau letak filenya yang tidak tepat, maka function die() yang akan dijalankan. Function die() akan menampilkan teks sebagai peringatan apabila proses membuka file gagal.

- Menutup File

Setelah file dibuka, hendaknya file tersebut juga ditutup ketika pemrosesan selesai. File yang yang tidak ditutup kemungkinan dapat terjadi kerusakan pada strukturnya. Berikut ini adalah contoh kode PHP untuk menutup file yang telah dibuka

```
$NamaFile = "testFile.txt";
```

```
$FileHandle = fopen($NamaFile, 'w') or die("File tidak bisa dibuka");
fclose($FileHandle);
```

File yang telah ditutup tidak bisa untuk dibaca, ditulisi, dan ditambah. Untuk bisa melakukan hal itu kembali, file terlebih dahulu harus dibuka lagi seperti sebelumnya.

- Menulis ke File

File yang telah dibuka dapat ditulisi dengan data di dalamnya. Berikut ini adalah contoh kode PHP untuk menulis suatu string ke dalam file.

```
<?php
    $FileKu = "testFile.txt";
    $FileHandle = fopen($Fileku, 'w') or die("File gagal dibuka");
    $DataString = "Hallo Amalia... \n";
    fwrite($FileHandle, $DataString);
    $DataString = "Hallo Faza dan Nada... \n";
    fwrite($FileHandle, $DataString);
    fclose($FileHandle);
?>
```

Apabila Anda menggunakan mode 'w' pada fopen(), dan selanjutnya Anda menuliskan data pada file, maka isi file yang lama (jika sebelumnya terdapat isi pada file tersebut) akan terhapus dan diganti dengan isi yang baru. Sedangkan apabila Anda menginginkan data yang lama pada suatu file tidak dihapus, maka Anda gunakan mode append 'a+' atau 'a'.

- Membaca isi File

Misalkan kita memiliki file dengan nama test.txt yang isinya adalah sbb:
Selamat berjumpa lagi kawan!!
Senang bertemu Anda.
Kita dapat menggunakan kode PHP untuk membaca file tersebut dan selanjutnya hasil pembacaan dapat ditampilkan di browser atau diproses lebih lanjut. Berikut ini adalah contoh kode pembacaannya.

```
<?php
    $FileKu = "test.txt";
    $FileHandle = fopen($FileKu, 'r');
    $Data = fread($FileHandle, 5);
    fclose($FileHandle);
    echo $Data;
?>
```

Apabila kode di atas dijalankan, maka pada browser hanya akan menampilkan teks
Selam
Hal ini dikarenakan adanya nilai 5 pada fread(). Nilai tersebut menyatakan jumlah karakter awal yang dibaca dari file. Supaya seluruh isi file dapat dibaca, gunakan function filesize().

```
<?php
    $FileKu = "test.txt";
    $FileHandle = fopen($FileKu, 'r');
    $Data = fread($FileHandle, filesize($FileKu));
    fclose($FileHandle);
```

```
    echo $Data;
?>
```

Setelah Anda lihat di browser, ternyata ganti baris pada isi file test.txt diabaikan. Meskipun teks dalam file tersebut ditulis dalam 2 baris, oleh perintah fread() akan dibaca dalam satu baris. Hal ini dikarenakan dalam file test.txt tidak terdapat tag html <br> untuk pindah baris. Anda juga dapat menggunakan function fgetc() untuk membaca file. Function ini membaca isi file karakter demi karakter. Berikut ini adalah contoh penggunaannya.

```
<?php
    $FileKu = "test.txt";
    $FileHandle = fopen($FileKu, 'r');
    while(!feof($FileHandle))
    {
    $Data = fgetc($FileHandle);
    echo $Data;
    }
    fclose($FileHandle);
?>
```

Keterangan:
Function fgetc() pada kode di atas diletakkan dalam perulangan. Function feof() digunakan untuk menyelidiki apakah pointer sudah berada di akhir dari file (end of file) atau belum. Selama belum end of file, proses pembacaan karakter akan berjalan terus. Setiap kali pembacaan, karakter yang dibaca akan ditampilkan di browser.

- Menghapus File

Dalam PHP, function untuk menghapus file adalah unlink(). Berikut ini contohnya.

```
<?
    $myFile = "testFile.txt";
    unlink($myFile);
?>
```

- Menambah isi File

Maksud dari menambah isi file di sini adalah, menambah data baru pada file (diasumsikan data sudah ada sebelumnya). Untuk menambah isi file dalam PHP, mode pembukaan file nya menggunakan 'a' atau 'a+' dan fwrite() untuk menulis data ke dalam file. Berikut ini adalah contohnya.
Misalkan kita mempunyai file buah.txt yang isinya adalah sbb:
Jeruk
Apel
Mangga
Selanjutnya kita punya kode PHP sbb:

```
<?php
    $myFile = "buah.txt";
    $fh = fopen($myFile, 'a') or die("File tidak bisa dibuka");
    $buah1 = "Anggur\n";
```

```
    fwrite($fh, $buah1);
    $buah2 = "Nanas\n";
    fwrite($fh, $buah2);
    fclose($fh);
?>
```

Kode di atas akan menambahkan 2 buah data baru ke file buah.txt. Perintah \n (new line) digunakan untuk ganti baris pada file.

- Upload File

Di beberapa aplikasi web, sering kita menjumpai proses upload file ke server. Berikut ini akan dibahas cara melakukan hal itu. Langkah pertama untuk membuat aplikasi web guna upload file adalah membuat formnya terlebih dahulu. Berikut ini adalah salah satu contoh form dalam bentuk HTML

```
<form enctype="multipart/form-data" action="upload.php" method="post">
    <input type="hidden" name="MAX_FILE_SIZE" value="30000" />
    Nama File : <input name="userfile" type="file" />
    <input type="submit" value="Upload" />
</form>
```

Pada form di atas, kita membatasi ukuran file yang dapat diupload adalah 30 Kb. Selanjutnya kita membuat file upload.php untuk proses uploadnya.

```
<?php
    $uploaddir = 'uploads/';
    $uploadfile = $uploaddir . $_FILES['userfile']['name'];
    if (move_uploaded_file($_FILES['userfile']['tmp_name'], $uploadfile)) {
    echo "File telah berhasil diupload";
    } else {
    echo "File gagal diupload";
    }
?>
```

❖ Bekerja dengan Session

Session digunakan untuk mengirim data ke beberapa halaman web. Sebuah halaman web, secara normal tidak akan mengirim suatu data dari halaman yang satu ke halaman yang lain. Dengan kata lain, semua informasi tentang data tersebut akan hilang begitu halaman web direload. Berikut ini akan diberikan contoh tentang penjelasan di atas, mengenai perlunya menggunakan session.

Contoh:

Diberikan source code sbb.

Form.php

```
<form action=submit.php method=post>
Username <input type=text name=username><input type=submit name=submit
value=Submit>
</form>
```

Submit.php

```
<?php
$username = $_POST['username'];
echo "Nama user Anda adalah: $username<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal1.php

```
<?php
echo "Ini adalah halaman 1<br>";
echo "Nama user Anda adalah: $username<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal2.php

```
<?php
echo "Ini adalah halaman 2<br>";
echo "Nama user Anda adalah: $username<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal3.php

```
<?php
echo "Ini adalah halaman 3<br>";
echo "Nama user Anda adalah: $username<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Dengan file-file di atas, user diminta melakukan login terlebih dahulu (memasukkan nama user) melalui form.php. Selanjutnya akan muncul submit.php yang menampilkan nama user yang dimasukkan tadi dan beberapa link ke halaman lain. Begitu user mengakses 3 buah halaman web yang ada tersebut, diharapkan nama user yang telah dimasukkan dalam form sebelumnya tetap ditampilkan dalam setiap halaman web yang diakses. Akan tetapi apa yang terjadi? Ternyata untuk ketiga halaman web yang diakses tersebut tidak menampilkan nama user. Hal ini dikarenakan hilangnya data/informasi dari nama user yang dimasukkan sebelumnya. Nah… di sinilah perlunya session. Dengan session, data dapat disimpan dan selanjutnya dapat diakses di beberapa halaman web. Penggunaan session sering diterapkan pada aplikasi web yang bersifat multiuser, seperti online shopping, web based mail, e-banking, dll. Data yang yang tersimpan dalam session bersifat temporary/ sementara. Biasanya akan terhapus secara otomatis begitu user menutup browser, atau melakukan logout.

- Memulai PHP-Session

Sebelum Anda menyimpan data dalam session, terlebih dahulu harus memulai session. Untuk memulai session, perintahnya adalah:

```
session_start()
```

- Menyimpan Data ke dalam Session

  Untuk menyimpan data ke dalam session, digunakan perintah

```
$_SESSION['nama_session'] = data;
```

Contoh:

Berikut ini adalah code dalam file submit.php (contoh sebelumnya) yang telah dimodifikasi. Nama user akan disimpan dalam session.

```
<?php
        session_start();
        $username = $_POST['username'];
        $_SESSION['namauser'] = $username;
        echo "Nama user Anda adalah: $username<br><br>";
        echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
        href=hal3.php>Hal 3</a>";
?>
```

- Memanggil Data yang Tersimpan dalam Session

  Setelah data disimpan dalam session, selanjutnya dapat dipanggil kembali apabila diperlukan.Untuk memanggil data dalam session, caranya cukup menuliskan

```
$_SESSION['nama_session'].
```

Contoh :

Berikut ini adalah code dari file hal1.php, hal2.php, dan hal3.php sehingga dapat menampilkan nama user yang telah disimpan dalam session.

Hal1.php

```
<?php
session_start();
echo "Ini adalah halaman 1<br>";
echo "Nama user Anda adalah: ".$_SESSION['namauser']. "<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal2.php

```
<?php
session_start();
echo "Ini adalah halaman 1<br>";
echo "Nama user Anda adalah: ".$_SESSION['namauser']. "<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal3.php

```
<?php
session_start();
echo "Ini adalah halaman 1<br>";
echo "Nama user Anda adalah: ".$_SESSION['namauser']. "<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

- Menghapus Data Session

  Untuk menghapus data yang tersimpan dalam suatu session, digunakan perintah

```
unset($_SESSION['nama_session']);
```

  Misalkan terdapat lebih dari satu session dan kita ingin menghapus semuanya, maka caranya dengan memberikan perintah

```
session_destroy();
```

  Biasanya menghapus data session yang terkait dengan username diterapkan pada proses logout. Dengan proses logout, maka username yang telah tersimpan dalam session akan terhapus.

  Contoh:

  Akan dibuat script untuk proses logout dari kasus di atas.

```
<?php
session_start();
unset($_SESSION['namauser']);
?>
```

- Penerapan Session untuk Security

  Pada contoh kasus di atas, misalkan kita buat aturan bahwa untuk dapat mengakses halaman 1, 2, dan 3, user harus terlebih dahulu melakukan login. Dari script yang kita buat di atas, seorang user bisa saja langsung melakuan by pass ke tiga halaman tersebut tanpa login terlebih dahulu. Tentu saja hal ini bisa berbahaya untuk aplikasi multiuser yang harus menjamin keamanan data dari para usernya. Untuk mencegah proses by pass tersebut, dapat kita akali dengan menggunakan session. Logikanya adalah, apabila user melakukan login, maka nama user yang dia masukkan akan disimpan dalam session. Sedangkan apabila ada seorang user yang mencoba mem by pass, maka dengan kata lain username tidak akan disimpan dalam session (session untuk user masih kosong), dengan catatan bahwa user lain yang sebelumnya login harus sudah me-logout. Dari hal ini, kita dapat melakukan cek apakah seorang user sudah melakukan login atau belum dengan melihat session, masih kosong atau tidak. Untuk melihat sebuah session masih kosong atau tidak dengan menggunakan perintah

```
isset($_SESSION['nama_session'])
```

  Perintah di atas akan menghasilkan nilai TRUE apabila session sudah tidak kosong, dan akan menghasilkan nilai FALSE apabila session masih kosong.

  Contoh:

Kita akan membuat script untuk mencegah by pass dalam kasus sebelumnya. Script ini selanjutnya akan disisipkan pada setiap halaman yang diinginkan, dalam hal ini adalah halaman 1, 2, dan 3

Cek.php

```
<?php
session_start();
if (!isset($_SESSION['namauser']))
{
echo "Anda belum login";
exit;
}
?>
```

Dan berikut adalah script hal1.php, hal2.php, dan hal3.php yang sudah dimodifikasi

Hal1.php

```
<?php
session_start();
include "cek.php";
echo "Ini adalah halaman 3<br>";
echo "Nama user Anda adalah: ".$_SESSION['namauser']. "<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal2.php

```
<?
session_start();
include "cek.php";
echo "Ini adalah halaman 3<br>";
echo "Nama user Anda adalah: ".$_SESSION['namauser']. "<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

Hal3.php

```
<?
session_start();
include "cek.php";
echo "Ini adalah halaman 3<br>";
echo "Nama user Anda adalah: ".$_SESSION['namauser']. "<br><br>";
echo "<a href=hal1.php>Hal 1</a> <a href=hal2.php>Hal 2</a> <a
href=hal3.php>Hal 3</a>";
?>
```

DAFTAR PUSTAKA

1. PHP Documentation Group, 1997-2007
2. MySQL 5.1 Reference Manual. 1997-2007
3. Dasar Pemrograman Web Dinamis Menggunakan PHP, Abdul Kadir, Andi Offset, 2002, 2003.
4. http://www.sitepoint.com/ , 2007.
5. http://www.gscripts.net/, 2007.
6. http://www.hotscripts.com/php, 2007.