Dasar Pemograman Web

Penyusun:

Miftah Farid Adiwisasta, S.T., M.Kom
Agung Baitul Hikmah, S.Kom, M.Kom
Ai ilah Warnilah, S.T., M.Kom

Penerbit CV. SARNU UNTUNG
KATA PENGANTAR

Dengan mengucapkan Puji syukur kehadirat Allah SWT, yang senantiasa melimpahkan rahmat dan karunia-Nya, sehingga penyusun dapat menyelesaikan buku ajar mata kuliah Web Programming I. Tujuan penyusunan buku ajar ini sebagai bahan pengajaran mata kuliah web programming I. Penyusun menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penyusunan buku ajar tidak akan lancar. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada semua pihak yang telah membantu dalam proses buku ajar ini sehingga dapat diselesaikan dengan baik.

Penulis menyadari bahwa dalam penyusunan buku ajar ini masih banyak kekurangannya. Oleh karena itu penyusun mohon kritik dan saran yang membangun demi kesempurnaan penulisan dimasa yang akan datang.

Akhir kata semoga buku ajar ini dapat berguna bagi penyusun khususnya dan bagi para pembaca yang berminat pada umumnya.

Tasikmalaya, 25 September 2019

Penyusun
# DAFTAR ISI

KATA PENGANTAR ........................................................................................................... ii
DAFTAR ISI .................................................................................................................. iii
DAFTAR TABEL .............................................................................................................. iv
DAFTAR GAMBAR ......................................................................................................... vi

**BAB 1 KONSEP DASAR WEB** .................................................................................. 1
  1.1. Dasar Website .................................................................................................. 1
  1.2. Struktur Navigasi .......................................................................................... 2
  1.3. Tools Pembuatan Website ........................................................................... 4

**BAB 2 PENGENALAN HTML** ................................................................................. 10
  2.1. Pengertian HTML ....................................................................................... 11
  2.2. Struktur Dasar HTML .............................................................................. 12

**BAB 3 PENGENALAN PHP** ....................................................................................... 19
  3.1. Pengertian PHP .......................................................................................... 19
  3.2. Dasar-dasar PHP ........................................................................................ 20
  3.3. Variabel ....................................................................................................... 24
  3.4. Konstanta ..................................................................................................... 28
  3.5. Tipe Data ..................................................................................................... 30
  3.6. Operator ....................................................................................................... 36
  3.7. Seleksi Kondisi ........................................................................................... 43
  3.8. Perulangan ................................................................................................... 47
  3.8. Array ........................................................................................................... 50

**BAB 4 PENGENALAN JAVA SCRIPT** ................................................................. 56
  4.1. Pengertian Java Script ................................................................................. 56
  4.2. Penulisan dan Penggunaan Java Script ...................................................... 56
  4.3. Variabel ....................................................................................................... 58
4.4. Function ...........................................................................................................61
4.5. Seleksi Kondisi .................................................................................................63
4.6. Perulangan ..........................................................................................................67
4.7. Array ....................................................................................................................72

BAB V PENGENALAN CSS.............................................................................................80
  5.1. Pengertian CSS ...................................................................................................80
  5.2. Penggunaan CSS ..................................................................................................80
  5.3. Pengertian Selector, Property, dan Value .........................................................84
  5.4. Desain Halaman Web Dengan CSS .................................................................90

DAFTAR PUSTAKA ........................................................................................................102

BIODATA PENULIS ....................................................................................................103
DAFTAR TABEL

Tabel 2.1 Tag Dasar HTML ................................................................. 10
Tabel 2.2 Tag Form HTML ................................................................. 11
Tabel 2.3 Tag Tabel HTML ................................................................. 12
Tabel 2.4 Tag Kategori Image ............................................................. 14
Tabel 2.5 Tag kategori List ................................................................. 15
Tabel 3.1 Daftar Variabel Sistem PHP ............................................... 26
Tabel 3.2 Karakter Khusus ................................................................. 30
Tabel 3.3 Daftar Operator Aritmatika PHP ........................................ 33
Tabel 3.4 Daftar Operator Perbandingan PHP .................................... 37
Tabel 3.5 Daftar Operator Logika PHP ................................................ 38
Tabel 3.6 Daftar Operator Increment/Decrement PHP ....................... 39
DAFTAR GAMBAR

Gambar 1.1 Struktur Navigasi Linier .................................................. 2
Gambar 1.2 Struktur Navigasi Non Linier ........................................... 2
Gambar 1.3 Struktur Navigasi Hirarki ................................................. 3
Gambar 1.4 Struktur Navigasi Campuran ............................................. 3
Gambar 1.5 Download Xampp .............................................................. 5
Gambar 1.6 Instalasi Xampp ................................................................. 6
Gambar 1.7 Instalasi Xampp Pilih Komponen ....................................... 6
Gambar 1.8 Instalasi Xampp pilih folder .............................................. 7
Gambar 1.9 Pilih Instalasi Xampp Star Menu ....................................... 7
Gambar 1.10 Menjalankan Xampp ......................................................... 8
Gambar 2.1 Tampilan Contoh 1 ............................................................ 12
Gambar 2.2 Tampilan Contoh 2 ............................................................ 13
Gambar 2.3 Tampilan Contoh 3 ............................................................ 15
Gambar 2.4 Tampilan Contoh 4 ............................................................ 16
Gambar 2.5 Tampilan Contoh 5 ............................................................ 17
Gambar 3.1 Tampilan Contoh 6 ............................................................ 20
Gambar 3.2 Tampilan Contoh 7 ............................................................ 21
Gambar 3.3 Tampilan Contoh 8 ............................................................ 22
Gambar 3.4 Tampilan Contoh 9 ............................................................ 22
Gambar 3.5 Tampilan Contoh 10 ........................................................... 22
Gambar 3.6 Tampilan Contoh 11 ........................................................... 23
Gambar 3.7 Tampilan Contoh 12 ........................................................... 23
Gambar 3.8 Tampilan Contoh 13 ........................................................... 27
Gambar 3.9 Tampilan Contoh 14 ........................................................... 27
Gambar 3.10 Tampilan Contoh 15 ........................................................ 27
Gambar 3.11 Tampilan Contoh 16 ................................................. 29
Gambar 3.12 Tampilan Contoh 17 ............................................... 30
Gambar 3.13 Tampilan Contoh 18 ............................................... 30
Gambar 3.15 Tampilan Contoh 20 ............................................... 31
Gambar 3.16 Tampilan Contoh 21 ............................................... 32
Gambar 3.17 Tampilan Contoh 22 ............................................... 32
Gambar 3.18 Tampilan Contoh 23 ............................................... 33
Gambar 3.19 Tampilan Contoh 24 ............................................... 34
Gambar 3.20 Tampilan Contoh 25 ............................................... 34
Gambar 3.21 Tampilan Contoh 26 ............................................... 35
Gambar 3.22 Tampilan Contoh 27 ............................................... 35
Gambar 3.23 Tampilan Contoh 28 ............................................... 37
Gambar 3.24 Tampilan Contoh 29 ............................................... 37
Gambar 3.25 Tampilan Contoh 30 ............................................... 38
Gambar 3.26 Tampilan Contoh 31 ............................................... 38
Gambar 5.1 Tampilan Halaman Beranda .................................... 90
Gambar 5.2 Tampilan halaman Profil ........................................ 90
Gambar 5.3 Tampilan Halaman Kontak ........................................ 91
Gambar 5.4 Tampilan Halaman Login ......................................... 91
BAB 1

KONSEP DASAR WEB

deskripsi:

Membahas tentang web server pada xampp, mengetahui tata letak penyimpanan folder pada xampp dan pemahaman tentang lembar kerja editor serta membahas tentang struktur navigasi web.

Tujuan Pembelajaran:

Setelah melakukan pembelajaran ini, pembaca diharapkan mampu:

1. Melakukan instalasi web server xampp dan penyimpanan folder pada web server xampp
2. Melakukan instalasi tools editor notepad ++ atau macromedia dreamweaver
3. Memahami struktur navigasi website

1.1. Dasar Website


1.2. Struktur Navigasi

Struktur Navigasi merupakan “struktur atau alur dari suatu program yang merupakan rancangan hubungan (rantai kerja) dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen pembuatan Website” Henky Prihatna (2005). Menurut Petrus Andre (2014) Untuk Menentukan struktur navigasi merupakan hal yang sebaiknya dilakukan sebelum membuat suatu aplikasi. Ada 4 (empat) macam bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan web yaitu:

1. Struktur Navigasi Linier

Struktur navigasi linier hanya mempunyai satu rangkaian cerita yang berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau dua halaman sesudahnya. Contoh

Sumber: Petrus Andre (2014)

Gambar 1.1 Struktur Navigasi Linier
2. Struktur Navigasi Non-linier


![Gambar 1.2 Struktur Navigasi Non-Linier](image)

Sumber: Petrus Andre (2014)

3. Struktur Navigasi Hirarki

Struktur navigasi hirarki biasa disebut struktur bercabang, merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu. Tampilan pada menu pertama akan disebut sebagai Master Page (halaman utama pertama), halaman utama ini mempunyai halaman percabangan yang disebut Slave Page (halaman pendukung). Jika salah satu halaman pendukung dipilih atau diaktifkan, maka tampilan tersebut akan bernama Master Page (halaman utama kedua), dan seterusnya. Pada struktur navigasi ini tidak diperkenankan adanya tampilan secara linier. Contoh gambar

![Gambar 1.3 Struktur Navigasi Hirarki](image)
4. Struktur Navigasi Campuran

Struktur navigasi campuran merupakan gabungan dari ketiga struktur sebelumnya yaitu linier, non-linier dan hirarki. Struktur navigasi ini juga biasa disebut dengan struktur navigasi bebas. Struktur navigasi ini banyak digunakan dalam pembuatan website karena struktur ini dapat digunakan dalam pembuatan website sehingga dapat memberikan ke-interaksian yang lebih tinggi. Contoh gambar

![Diagram Struktur Navigasi Campuran](image)

Sumber: Petrus Andre (2014)

**Gambar 1.4 Struktur Navigasi Campuran**

1.3. Tools Pembuatan Website

Tools pendukung untuk pembuatan website yaitu:

A. Teks Editor

Untuk menuliskan kode dalam pemrograman dan desain halaman web, ada beberapa teks editor yaitu notepad, notepad ++, Macromedia dreamweaver, dan masih banyak lagi teks editor yang bisa digunakan untuk pemrograman dan desain halaman web. Dalam melakukan teknik manual disarankan untuk menggunakan Notepad++, karena memiliki beberapa kelebihan, misalnya fasilitas multi tab, penandaan sintaks untuk menghindari kesalahan penulisan kode, serta fitur pencarian yang kompleks. Tetapi untuk desain halaman website dengan HTML dan CSS disarankan menggunakan macromedia
dreamweaver karena memiliki kemampuan desain view, code vie, dan split view Berfungsi menampilkan gabungan antara Code View dan Desain View pada saat bersamaan.Jadi bisa langsung melihat perubahan pada saat mengubah htmlnya.

B. Web Browser

Web browser adalah sebuah software yang dapat digunakan untuk menampilkan tampilan halaman sebuah situs website. Software ini kini telah dikembangkan dengan menggunakan user interface grafis, sehingga pemakai data melakukan ‘point and click’ untuk pindah antar dokumen. Salah satu web browser yang sering digunakan yaitu Mozilla firefox, google chrome, opera, dan masih banyak lagi browser yang bisa digunakan untuk menjalankan atau menampilkan halaman web.

C. Web Server

Web server merupakan suatu alat komunikasi yang digunakan untuk melayani pengiriman sebuah dokumen web yang berfungsi tempat penyimpanan dokumen web salah satunya adalah web browser seperti explorer atau navigator berkomunikasi melalui jaringan (termasuk jaringan internet) dengan web server, menggunakan HTTP. browser akan mengirimkan request ke server untuk meminta dokumen tertentu atau layanan lain yang disediakan oleh server. Server memberikan dokumen satu layanannya jika tersedia juga dengan menggunakan protocol HTTP". Salah satu aplikasi server localhost serta yang paling banyak digunakan dan cukup familiar di kalangan web developer saat ini adalah XAMPP.

Aplikasi XAMPP ini dibuat oleh Apache Friends dan installer-nya bisa langsung diunduh dari situs mereka. Isi aplikasinya juga sudah sangat komplit, antara lain:

- Apache
- MySQL
- PHP
- phpMyAdmin
• FileZilla FTP Server
• Tomcat
• XAMPP Control Panel

Cara Instalasi xampp yaitu:


![Gambar 1.5 Download xampp](image1.png)

2. Dobel klik file XAMPP yang baru saja download, nanti selanjutnya akan muncul tampilan seperti di bawah ini dan pilih next:

![Gambar 1.6 Instalasi xampp](image2.png)
3. Selanjutnya tampilan untuk memilih komponen ceklis semuanya seperti dibawah ini dan pilih next:

![Gambar 1.7 Instalasi xampp Pilih Komponen](image1)

4. Selanjutnya untuk memilih folder dan pilih folder bawaannya xampp seperti dibawah ini pilih Next:

![Gambar 1.8 Instalasi xampp Pilih Folder](image2)

5. Proses instalasi berjalan dan pilih Finish
6. Untuk menggunakan nya jalankan xampp yang telah di instal bisa membukanya melalui shortcut desktop atau star menu seperti gambar dibawah ini dan pilih Xampp Control Panel
Gambar 1.9 Pilih Aplikasi Xampp start menu

7. Tampilan ketika xampp pertama kali dijalankan apabila mau menggunakan pilih start

Gambar 1.10 Menjalankan xampp

8. Buka browser dan ketikan “localhost” maka akan tampil seperti dibawah ini
Gambar 1.11 Tampilan xampp

9. Selamat, anda berhasil menginstal aplikasi XAMPP. Sekarang komputer Anda sudah berfungsi seperti server dan bisa menjalankan aplikasi-aplikasi berbasis web.

Istilah Xampp diambil dari kata X yang berarti empat sistem operasi apapun seperti Windows, Linux, Mac Os dan Solaris. Sedangkan A diambil dari kata Apache, Kemudian M singkatan dari kata MySQL, kemudian huruf P singkatan dari PHP dan untuk huruf P yang terakhir singkatan dari kata Perl.
BAB 2
PENGENALAN HTML

Deskripsi:
Membahas tentang pengertian dan pengenalan dasar HTML, mendeklarasikan script tag dasar HTML pada editor dan menampilkan hasil pada web browser yang digunakan.

Tujuan Pembelajaran:
Setelah melakukan pembelajaran ini, pembaca diharapkan mampu:
1. Memahami Dasar – Dasar HTML
2. Menuliskan script tag dasar HTML pada editor
3. Menampilkan hasil HTML tersebut pada web browser

2.1. Pengenalan HTML
Dalam perancangan sebuah website pada dasarnya adalah kumpulan dari pada hyperlink yang menuju pada sebuah alamat atau situs dengan menggunakan sebuah bahasa HTML(Hyper Tekt Markup Language) (Agung Baitul Hikmah dkk, 2015:1).

Hypertext Markup Language (HTML) adalah sebuah bahasa untuk menampilkan konten di web. Dan HTML juga merupakan bahasa pemrograman yang bebas artinya pengembangan dari bahasa yang dapat dikembangkan bersama secara global (Edy Winarno ST, 2014:1)

Dalam penggunaannya sebuah bahasa HTML tidak dapat bekerja sendiri untuk membuat halaman tampilan seperti membuat form, menyiapkan gambar, tabel, maupun teks tulisan yang menarik pada web browser. Akan tetapi setiap browser memiliki perbedaan dalam menampilkan script HTML yang dibuat bisa jadi apa yang ditampilkan di browser yang satu akan berbeda dengan browser yang lain sesuai dengan dukungan script dari browser yang kita gunakan. untuk itu sebuah bahasa HTML memerlukan penggabungan beberapa script lainnya seperti CSS (Cascading Style Sheets), Java Script maupun script lainnya.
Uji Pemahaman Mandiri

1. Lakukan Uji Coba Penerapan Kode HTML ke sejumlah Web Browser apakah setiap kode HTML Kompatible dengan setiap browser
2. Lakukan Uji Coba Penggabungan Kode HTML dengan Script lainnya seperti CSS dan Java Script

2.2. Struktur Dasar HTML

Dalam mempelajari sebuah Bahasa HTML kita hendaknya mengetahui struktur dasar dari Bahasa HTML ada istilah tag berupa tanda <tagname> isi konten</tagname> berikut contoh penggunaan tag dasar HTML berdasarkan kategori sesuai dengan aturan penulisannya.(Refsnes Data, 2019)

A. Kategori Basic HTML

<table>
<thead>
<tr>
<th>Tag</th>
<th>Kegunaan</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;!DOCTYPE &gt;</td>
<td>Mendefinisikan tipe Dokumen</td>
</tr>
<tr>
<td>&lt;html&gt;</td>
<td>Mendefinisikan HTML dokumen</td>
</tr>
<tr>
<td>&lt;head&gt;</td>
<td>Mendefinisikan informasi dokumen</td>
</tr>
<tr>
<td>&lt;title&gt;</td>
<td>Mendefinisikan judul dokumen</td>
</tr>
<tr>
<td>&lt;body&gt;</td>
<td>Mendefinisikan body dokumen</td>
</tr>
<tr>
<td>&lt;h1&gt; to &lt;h6&gt;</td>
<td>Mendefinisikan heading</td>
</tr>
<tr>
<td>&lt;p&gt;</td>
<td>Mendefinisikan paragrap</td>
</tr>
<tr>
<td>&lt;br&gt;</td>
<td>Mendefinisikan baris</td>
</tr>
</tbody>
</table>

Contoh Kode Program:

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title>Ini adalah judul</title>
5. <body>
6. 
7. <h1>Ini adalah heading</h1>
8. <p>Ini adalah paragraph.</p>
9. `<br>` Ini adalah baris.

10. 

11. `</body>`

12. `</head>`

13. `</html>`

Hasilnya:

![Hasil】

Gambar 2.1 Tampilan Contoh 1

B. Kategori *Form* dan *Input*

<table>
<thead>
<tr>
<th>Tag</th>
<th>Kegunaan</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;form&gt;</code></td>
<td>Mendefinisikan Form dan Input</td>
</tr>
<tr>
<td><code>&lt;input&gt;</code></td>
<td>Mendefinisikan input kontrol</td>
</tr>
<tr>
<td><code>&lt;textarea&gt;</code></td>
<td>Mendefinisikan multiline input kontrol (textarea)</td>
</tr>
<tr>
<td><code>&lt;button&gt;</code></td>
<td>Mendefinisikan button Click</td>
</tr>
<tr>
<td><code>&lt;select&gt;</code></td>
<td>Mendefinisikan drop-down list</td>
</tr>
<tr>
<td><code>&lt;optgroup&gt;</code></td>
<td>Mendefinisikan group untuk options drop-down list</td>
</tr>
<tr>
<td><code>&lt;option&gt;</code></td>
<td>Mendefinisikan option drop-down list</td>
</tr>
<tr>
<td><code>&lt;label&gt;</code></td>
<td>Menedfiniskan label untuk <code>&lt;input&gt;</code> element</td>
</tr>
<tr>
<td><code>&lt;fieldset&gt;</code></td>
<td>Group element untuk form</td>
</tr>
<tr>
<td><code>&lt;legend&gt;</code></td>
<td>Menedfiniskan kalimat untuk <code>&lt;fieldset&gt;</code> element</td>
</tr>
</tbody>
</table>
Contoh Kode Program:

1. `<!DOCTYPE html>`
2. `<html>`
3. `<body>`
4. `<h2>`Contoh HTML Forms`</h2>`
5. `<form action="/action_page.php">`
6. Nama Depan:<br>
7. `<input type="text" name="firstname" value="Agung">`
8. `<br>`
9. Nama Belakang:<br>
10. `<input type="text" name="lastname" value="Baitul">`
11. `<br>`
12. `<input type="submit" value="Proses">`
13. `</form>`
14.
15. `</body>`
16. `</html>`

Hasilnya:

![Contoh HTML Forms](localHost/contoh1.html)

**Gambar 2.2 Tampilan Contoh 2**
C. Kategori Tabel

### Tabel 2.3 Tag Tabel HTML

<table>
<thead>
<tr>
<th>Tag</th>
<th>Kegunaan</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;tabel&gt;</td>
<td>Mendefinisikan tabel</td>
</tr>
<tr>
<td>&lt;caption&gt;</td>
<td>Mendefinisikan tabel caption</td>
</tr>
<tr>
<td>&lt;th&gt;</td>
<td>Mendefinisikan header cell pada tabel</td>
</tr>
<tr>
<td>&lt;tr&gt;</td>
<td>Mendefinisikan baris pada tabel</td>
</tr>
<tr>
<td>&lt;td&gt;</td>
<td>Mendefinisikan kolom pada tabel</td>
</tr>
<tr>
<td>&lt;thead&gt;</td>
<td>Groups header content pada tabel</td>
</tr>
<tr>
<td>&lt;tbody&gt;</td>
<td>Groups body content pada tabel</td>
</tr>
<tr>
<td>&lt;tfoot&gt;</td>
<td>Groups footer content pada tabel</td>
</tr>
<tr>
<td>&lt;col&gt;</td>
<td>Spasi kolom pada tabel</td>
</tr>
<tr>
<td>&lt;colgroup&gt;</td>
<td>Spasi kolom pada group tabel</td>
</tr>
</tbody>
</table>

Contoh Kode Program:

```html
1. <!DOCTYPE html>
2. <html>
3. <head>
4. </head>
5. <body>
6. <h2>Contoh HTML Table</h2>
7. <table border='1'>
8. <tr>
9. <th>No</th>
10. <th>NIM</th>
11. <th>Nama Mahasiswa</th>
12. </tr>
13. <tr>
14. <td>01</td>
15. <td>1209868</td>
16. <td>Agung Baitul Hikmah</td>
17. </tr>
18. <tr>
19. <td>02</td>
20. <td>1209869</td>
21. <td>Asha Putri Nur Hikmah</td>
```
Hasilnya:

Gambar 2.3 Tampilan Contoh 3

D. Kategori Image

Tabel 2.4 Tag Kategori Image

<table>
<thead>
<tr>
<th>Tag</th>
<th>Kegunaan</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;img&gt;</code></td>
<td>Mendefiniskan gambar</td>
</tr>
<tr>
<td><code>&lt;map&gt;</code></td>
<td>Mendefiniskan peta gambar</td>
</tr>
<tr>
<td><code>&lt;area&gt;</code></td>
<td>Mendefiniskan area di dalam peta-gambar</td>
</tr>
</tbody>
</table>

Contoh Kode Program:

1. `<!DOCTYPE html>`
2. `<html>`
3. `<body>`
4. `<h2>`Contoh Menampilkan Image</h2`
5. `<p>`Contoh Menampilkan Gambar dengan lebar 500 pixel dan tinggi 600 pixel:</p`
6. `<img src='img_girl.jpg' width='500' height='600'>`
7. `</body>`
8. `</html>`
E. Kategori *Link*

<table>
<thead>
<tr>
<th>Tag</th>
<th>Kegunaan</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;a&gt;</code></td>
<td>Mendefinisikan hyperlink</td>
</tr>
<tr>
<td><code>&lt;link&gt;</code></td>
<td>Mendefinisikan hubungan antara dokumen dan sumber daya eksternal (paling sering digunakan untuk menautkan ke style sheet)</td>
</tr>
</tbody>
</table>

**Contoh Kode Program:**

1. `<!DOCTYPE html>`
2. `<html>`
3. `<body>`
4. `<h2>Contoh Link</h2>`
5. `<p>Silakan Download Materi HTML disini:</p>`
6. `<a href="https://www.agungbaitul83.blogspot.com">Download Link</a>`
7. `</body>`
8. `</html>`

Hasilnya:

![Contoh Link](https://www.agungbaitul83.blogspot.com)

Gambar 2.4 Tampilan Contoh 4

F. Kategori *List*

<table>
<thead>
<tr>
<th>Tag</th>
<th>Kegunaan</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;ul&gt;</code></td>
<td>Mendefinisikan unordered list</td>
</tr>
<tr>
<td><code>&lt;ol&gt;</code></td>
<td>Mendefiniskan ordered list</td>
</tr>
<tr>
<td>Tag</td>
<td>Deskripsi</td>
</tr>
<tr>
<td>-------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td><code>&lt;li&gt;</code></td>
<td>Mendefiniskan list item</td>
</tr>
<tr>
<td><code>&lt;dl&gt;</code></td>
<td>Mendefiniskan description list</td>
</tr>
<tr>
<td><code>&lt;dt&gt;</code></td>
<td>Mendefiniskan istilah/nama pada description list</td>
</tr>
<tr>
<td><code>&lt;dd&gt;</code></td>
<td>Mendefiniskan pada deskripsi istilah/nama pada description list</td>
</tr>
</tbody>
</table>

Contoh kode program:

```html
1. <!DOCTYPE html>
2. <html>
3. <body>
4. <h2>Contoh Unordered HTML List</h2>
5. <ul>
6. <li>Coffee</li>
7. <li>Tea</li>
8. <li>Milk</li>
9. </ul>
10. <h2>Contoh Ordered HTML List</h2>
11. <ol>
12. <li>Coffee</li>
13. <li>Tea</li>
14. <li>Milk</li>
15. </ol>
16. </body>
17. </html>
```

Hasilnya:
Gambar 2.5 Tampilan Contoh 5

Uji Pemahaman Mandiri
1. Buatlah kode program untuk tampilan form dibawah ini:

<table>
<thead>
<tr>
<th>No</th>
<th>Nama Karyawan</th>
<th>Jenis Kelamin</th>
<th>Alamat Karyawan</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Salha Sucu Nur Hikmah</td>
<td>Perempuan</td>
<td>Jln. Tanawijaya No.4 Tasikmalaya</td>
</tr>
<tr>
<td>02</td>
<td>Siti Rohayati</td>
<td>Perempuan</td>
<td>Jln. Mangkubumi Indah No.8 Ciamis</td>
</tr>
<tr>
<td>03</td>
<td>Asep Mulyana</td>
<td>Laki-Laki</td>
<td>Jln. Ciendog Kaving 05 Banjar</td>
</tr>
<tr>
<td>04</td>
<td>Silvia Suherman</td>
<td>Perempuan</td>
<td>Jln. Merdeka Barat No.5 Garut Selatan</td>
</tr>
</tbody>
</table>

2. Buatlah tampilan form berdasarkan kode program dibawah ini:

1. <!DOCTYPE html>
2. <html>
3. 
4. <body>
5. <h2>Menu Makanan Nasi Padang Teu Wareg</h2>
6. <p>Silakan Pilih Menu Di Bawah ini:</p>
7. <form action="/action_page.php">
8.  <select name="menu">
9.   <option value="rendang">Rendang</option>
10.  <option value="cincang">Cincang</option>
11.  <option value="tongkol">Ikan Tongkol</option>
12.  <option value="telur">Telur Dadar</option>
13. </select>
14. </form>
BAB 3
PENGENALAN PHP

Deskripsi:
Membahas tentang pengenalan dasar PHP, mendeklarasikan tag dasar PHP pada editor dan menampilkan hasil pada web browser yang digunakan.

Tujuan Pembelajaran:
Setelah melakukan pembelajaran ini, pembaca diharapkan mampu:
1. Menuliskan script tag dasar PHP pada editor notepad ++
2. Menampilkan hasil tag dasar PHP tersebut pada web browser

3.1. Pengertian PHP


PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client). Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya, PHP digunakan untuk membuat

**3.2. Dasar-dasar PHP**

A. Penulisan Syntax PHP


1. `<?php`
2. `echo 'Selamat Datang di PHP';`
3. `?>`

Outputnya :

[![Gambar 3.1 Tampilan Contoh 6](image)](image)
B. Fungsi-fungsi Dasar PHP

PHP memiliki beberapa fungsi dasar untuk mencetak teks ke layar:

- fungsi `echo();`
- fungsi `print();`
- fungsi `printf()`.

Fungsi-fungsi ini akan sering kita gunakan dalam pemrograman PHP. Perbedaan dari ketiga fungsi tersebut yaitu:

1. Fungsi `echo()`

   Fungsi `echo()` adalah fungsi untuk menampilkan teks ke layar. Fungsi ini dapat digunakan dengan tanda kurung maupun tanpa tanda kurung. Contoh:

   ```php
   <?php
   echo "Selamat Datang di PHP";
   echo "<p>";
   echo ("Belajar Pemrograman Web");
   ?>
   
   Maka hasilnya:
   
   Gambar 3.2 Tampilan Contoh 7
   Fungsi `echo()` tidak akan mengembalikan apa-apa setelah dieksekusi. Dia hanya bertugas menampilkan teks saja.

2. Fungsi `print()`
Fungsi `print()` sama seperti fungsi `echo()`. Dia juga digunakan untuk menampilkan teks ke layar. Fungsi `print()` juga bisa digunakan tanpa tanda kurung.

Contoh :

```php
1. <?php
2. print "Selamat Datang di PHP";
3. print "<p>";
4. print ("Belajar mencetak teks di PHP!");
5. ?>
```

Maka hasilnya :

![Gambar 3.3 Tampilan Contoh 8](image)

Perbedaannya dengan `echo()`:

- Fungsi `print()` akan selalu mengembalikan nilai 1 saat dieksekusi, sedangkan `echo()` tidak mengembalikan apa-apa.
- Fungsi `print()` hanya boleh diberikan satu parameter saja, sedangkan `echo()` boleh lebih dari satu.

Contoh fungsi `print()` dengan satu parameter :

```php
1. <?php
2. print "Selamat datang di PHP";
3. ?>
```

Maka hasilnya :

22
Gambar 3.4 Tampilan Contoh 9

Contoh fungsi `print()` dengan dua parameter:

1. `<?php`
2. `print 'Selamat datang di PHP', 'Web Programming';`
3. `?>`

Maka akan terjadi error seperti ini:

Gambar 3.5 Tampilan Contoh 10

Contoh fungsi `echo()` dengan dua parameter:

1. `<?php`
2. `echo 'Selamat datang di PHP', 'Web Programming';`
3. `?>`

Maka hasilnya:

Gambar 3.6 Tampilan Contoh 11

23
3. Fungsi printf()

Fungsi `printf()` adalah fungsi untuk memformat teks atau string. Fungsi ini akan mengembalikan panjang dari teks saat dieksekusi.

Contoh format string (%s):

```
<?php
    $teks = 'BSI';
    printf('Saya Belajar Web Programming di %s', $teks);
?>
```

Hasilnya :

![Gambar 3.7 Tampilan Contoh 12](image)

Contoh menggunakan fungsi `echo()`:

```
<?php
    $teks = 'BSI';
    echo 'Saya Belajar Web Programming di $teks';
?>
```

Maka hasilnya akan sama dengan menggunakan fungsi `printf()`

Selain simbol `%s` ada juga simbol:
- `%d` untuk bilangan desimal (integer);
- `%f` untuk pecahan (float);
- `%b` untuk boolean.
Dari ketiga fungsi echo(), print(), dan printf() mempunyai fungsi yang sama yaitu untuk mencetak teks ke layar

C. Fungsi PHP Dalam Pemrograman Web


3.3. Variabel

A. Penulisan Variabel

Dalam penulisan variable dalam PHP ada beberapa aturan yaitu:

1. Diawali dengan tanda dollar ($) diikuti dengan nama variable
   Contoh penulisan variable yang benar :

   ```php
   1. <?php
   2. $a;
   3. $nama;
   4. ?>
   ```

2. Nama variabel harus dimulai dengan huruf atau karakter garis bawah
   Contoh penulisan variable yang benar :
1. <?php
2. $_a;
3. $nama_lengkap;
4. ?>

3. Nama variabel tidak dapat dimulai dengan angka
   Contoh penulisan variable yang salah:

   ```php
   1. <?php
   2. $2a;
   3. ?>
   ```

   Maka ketika dijalankan muncul error nama variable tidak terdeteksi seperti dibawah ini:

   ```
   Parse error: syntax error, unexpected '2' (T_NUMBER), expecting variable (T_VARIABLE)
   ```

4. Nama variabel hanya dapat berisi karakter alfanumerik dan garis bawah (A-z, 0-9, dan _)

   ```php
   1. <?php
   2. $alamat1;
   3. $alamat2;
   4. $nama_lengkap;
   5. ?>
   ```

5. Bersifat Case sensitive
   PHP membedakan variabel yang ditulis dengan huruf besar dan kecil (bersifat case sensitif), sehingga $nama tidak sama dengan $Nama dan $NAMA, ketiganya akan dianggap sebagai variabel yang berbeda.
   Contoh penulisan yang benar:

   ```php
   1. <?php
   2. $nama;
   3. $Nama;
   4. $NAMA;
   5. ?>
   ```
Variable diatas walaupun nama variable nya sama tetapi hasil outputnya akan berbeda karena variable tersebut memang berbeda. Untuk menghindari kesalahan program yang dikarenakan salah merujuk variabel, disarankan menggunakan huruf kecil untuk seluruh nama variabel.

6. Tidak memerlukan deklarasi terlebih dahulu
   Variabel dalam PHP tidak perlu dideklarasikan terlebih dahulu, bisa langsung menggunakan tanpa dideklarasikan terlebih dahulu

7. Variabel dalam PHP tidak bertipe
   PHP termasuk jenis bahasa pemrograman yang variabelnya tidak terikat pada sebuah tipe tertentu, setiap variabel bisa di isi dengan nilai apa saja dan secara otomatis PHP mengkonversi variabel ke tipe data yang benar sesuai dengan isi dari varibel tersebut.

B. Memberikan Nilai

   Untuk memberikan nilai kepada sebuah variabel, PHP menggunakan tanda sama dengan (=). Operator ‘sama dengan’ ini dikenal dengan istilah Assignment Operators.

   - Contoh menampilkan nilai variabel

```
1. <?php
2. $x = "Belajar Web Programming";
3. echo $x;
4. ?>
```

Hasilnya :

```
Belajar Web Programming
```

Contoh variabel tipe data integer:

1. `<php
2.   $nilai = 100;
3.   echo "Nilai Web Programming saya $nilai";
4.   ?>

Hasilnya:

```
Nilai Web Programming saya 100
```

Contoh variabel tipe data string

1. `<php
2.   $teks = "Universitas Bina Sarana Informatika";
3.   echo "Saya belajar web programming di $.teks."<br>";
4.   ?>

Hasilnya:

```
Saya belajar web programming di Universitas Bina Sarana Informatika
```

Variabel Sistem PHP (Predefined Variables)

Variabel Sistem PHP (Predefined Variables) yaitu variabel yang telah didefinisikan secara sistem oleh PHP. Ada beberapa contoh variabel sistem PHP

**Tabel 3.1 Daftar Variabel Sistem PHP**
<table>
<thead>
<tr>
<th>Nama Variabel</th>
<th>Keterangan</th>
</tr>
</thead>
<tbody>
<tr>
<td>$GLOBALS</td>
<td>Variabel tersedia dalam lingkup global</td>
</tr>
<tr>
<td>$_SERVER</td>
<td>Informasi Server dan eksekusi</td>
</tr>
<tr>
<td>$_GET</td>
<td>Variabel HTTP GET</td>
</tr>
<tr>
<td>$_POST</td>
<td>Variabel HTTP POST</td>
</tr>
<tr>
<td>$_FILES</td>
<td>Variabel Upload File HTTP</td>
</tr>
<tr>
<td>$_REQUEST</td>
<td>Variabel Permintaan HTTP</td>
</tr>
<tr>
<td>$_SESSION</td>
<td>Variabel Sesi</td>
</tr>
<tr>
<td>$_ENV</td>
<td>Ruang lingkup Variabel</td>
</tr>
<tr>
<td>$_COOKIE</td>
<td>Cookie HTTP</td>
</tr>
<tr>
<td>$php_errormsg</td>
<td>Pesan Kesalahan sebelumnya</td>
</tr>
<tr>
<td>$HTTP_RAW_POST_DATA</td>
<td>POST data Mentah</td>
</tr>
<tr>
<td>$http_response_header</td>
<td>Header Respon HTTP</td>
</tr>
<tr>
<td>$argc</td>
<td>Jumlah argument yang diteruskan ke script</td>
</tr>
<tr>
<td>$argv</td>
<td>Array argument yang diteruskan ke script</td>
</tr>
</tbody>
</table>

Selain daftar varibel sistem PHP diatas unggki masih terdapat beberapa variabel lain tergantung jenis web server, versi PHP yang digunakan. Sebaiknya menghindari dalam penamaan variabel yang sudah terdapat dalam table diatas.

3.4. Konstanta
Yaitu pengenal yang sifatnya konstan atau tetap, nilai nya tidak dapat diubah selama proses eksekusi program. Cara mendefinisikan konstanta dalam PHP ada dua cara yaitu:

a. Menggunakan keyword const

   bentuk umum:
   ```php
   const nama_konstanta = nilai konstanta;
   ```

   contoh :
   ```php
   <?php
   const kampus = "Universitas Bina Sarana Informatika";
   echo kampus;
   ?>
   ```

   Hasilnya:
   ![Gambar 3.11 Tampilan Contoh 16](attachment:contoh16.png)

   Contoh :
   ```php
   <?php
   const x = 3.14;
   echo x;
   ?>
   ```

   Hasilnya:
   ![Gambar 3.12 Tampilan Contoh 17](attachment:contoh17.png)

b. Menggunakan fungsi define

   Bentuk umum:
   ```php
   define (nama_konstanta,nilai konstanta);
   ```
3.5. Tipe Data

Variabel atau konstanta dapat menyimpan data dari tipe data yang berbeda sehingga sebuah variabel atau konstanta akan memiliki tipe data tertentu. Pemrograman PHP mendukung tipe data sebagai berikut:

a. Tipe data String

Yaitu tipe data yang mempresentasikan data yang berupa teks atau kumpulan karakter. Pemograman PHP mendukung
dua tipe dalam penulisan tipe data string yaitu diapit petik tunggal (single quoted) dan diapit petik ganda (double quoted).

6. Contoh tipe data string diapit petik tunggal (single quoted):

```php
<?php
    $a = 'Belajar Web Programming';
    $b = 'Di Universitas Bina Sarana Informatika';
    echo $a;
    echo $b;
?>
```

Hasilnya:

![Gambar 3.15 Tampilan Contoh 20](image)

7. Contoh tipe data string diapit petik ganda (double quoted):

```php
<?php
    $a = "Belajar Web Programming";
    $b = "Di Universitas Bina Sarana Informatika";
    echo $a;
    echo $b;
?>
```

Hasilnya akan sama akan sama dengan diapit petik tunggal:

![Gambar 3.16 Tampilan Contoh 21](image)

8. Contoh tipe data string diapit petik tunggal dan ganda:
Maka hasilnya:

Gambar 3.17 Tampilan Contoh 22

Dari ketiga contoh diatas tidak ada perbedaan dengan menggunakan tanda petik tunggal ataupun ganda. Namun untuk menampilkan karakter khusus seperti tanda petik (‘), tanda dollar ($), dan tanda-tanda khusus lainnya harus menggunakan backslash (\). Berikut table dalam penggunaan karakter khusus untuk tanda petik ganda (double quoted).

**Tabel 3.2 Karakter Khusus**

<table>
<thead>
<tr>
<th>Cara Penulisan</th>
<th>Karakter Yang Ditampilkan</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;</td>
<td>Karakter tanda petik dua (&quot;)</td>
</tr>
<tr>
<td>\n</td>
<td>Karakter newline (garis baru)</td>
</tr>
<tr>
<td>\r</td>
<td>Karakter carriage return</td>
</tr>
<tr>
<td>\t</td>
<td>Karakter Tab</td>
</tr>
<tr>
<td>\</td>
<td>Karakter backslash</td>
</tr>
<tr>
<td>$</td>
<td>Karakter dollar sign ($)</td>
</tr>
<tr>
<td>\i</td>
<td>Karakter pembuka kurung kurawal ()</td>
</tr>
<tr>
<td>\1</td>
<td>Karakter penutup kurung kurawal ()</td>
</tr>
<tr>
<td>[</td>
<td>Karakter pembuka kurung</td>
</tr>
</tbody>
</table>
Karakter penutup kurung siku ()

\ 0 sampai \777  Karakter ASCII menggunakan nilai oktal

\x0 sampai \xFF  Karakter ASCII menggunakan nilai heksadesimal

Contoh menampilkan karakter tanda petik dua

```php
<?php
    $x = "Ayo Belajar "Belajar Web Programming" ";
    echo $x;
?>
```

Maka hasilnya:

![](image)

**Gambar 3.18 Tampilan Contoh 23**

b. Tipe Data Integer

Tipe data integer yaitu tipe data yang memperesentasikan bilangan non desimal atau bilangan bulat seperti 1,2,-3,10,100.

9. Contoh:

```php
<?php
    $x= 30;
    echo $x;
?>
```

Hasilnya:
Gambar 3.19 Tampilan Contoh 24

10. Contoh penjumlahan

```php
<?php
    $x = 20;
    $y = 51;
    $hasil = $x + $y;
    echo $hasil;
?>
```

Maka hasilnya:

10

Gambar 3.20 Tampilan Contoh 25

c. Float

Tipe data float yaitu tipe data yang memperentesasikan nilai numeric dalam bentuk angka desimal dibelakang koma. Tipe data float berada pada rentang 1.7E-308 dan 1.7E+308 dengan 15 digit akurasi

Contoh:

```php
<?php
    $x = 10.5;
    echo $x;
?>
```

Hasilnya:

10.5
d. Boolean

Tipe data yang hanya memiliki 2 kemungkinan yaitu benar(true) atau salah(false). Contoh:

```php
<?php
$x = true;
y = false;
echo "Nilai x adalah : " . $x;
echo "\n" . $y;
?>
```

Hasilnya:

Gambar 3.21 Tampilan Contoh 26

Nilai x=1 dianggap benar (true)
Nilai y= nilainya salah (false) tanpa output karena kalau ditulisnya “0” maka dianggap string. Tanda <br> sebagai baris agar tampilan
Tipe data Boolean sering digunakan dalam pengujian if bersyarat, untuk lebih jelasnya contoh tentang Boolean nanti akan dibahas dalam bab selanjutnya.

### 3.6. Operator

Dalam PHP operator dikategorikan beberapa kelompok yaitu:

c. Operator Aritmatika

Operator Aritmatika dalam PHP digunakan untuk melakukan operasi aritmatika atau perhitungan. Berikut daftar operator aritmatika dalam PHP

**Tabel 3.3 Daftar Operator Aritmatika PHP**

36
<table>
<thead>
<tr>
<th>Operator</th>
<th>Keterangan</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Operator untuk penjumlahan</td>
</tr>
<tr>
<td>-</td>
<td>Operator untuk pengurangan</td>
</tr>
<tr>
<td>*</td>
<td>Operator untuk perkalian</td>
</tr>
<tr>
<td>/</td>
<td>Operator untuk pembagian</td>
</tr>
<tr>
<td>%</td>
<td>Operator untuk mengetahui sisa dari pembagian (modulus)</td>
</tr>
</tbody>
</table>

Contoh penggunaan operator:

```php
<?php
    $x = 5;
    $y = 7;
    $penjumlahan = $x + $y;
    $pengurangan = $x - $y;
    $perkalian = $x * $y;
    $pembagian = $x / $y;
    $sisa_bagi = $x % $y;

    echo "Hasil Penjumlahan = ", $penjumlahan, ",<br>
    echo "Hasil Pengurangan = ", $pengurangan, ",<br>
    echo "Hasil Perkalian = ", $perkalian, ",<br>
    echo "Hasil Pembagian = ", $pembagian, ",<br>
    echo "Hasil Sisa Bagi = ", $sisa_bagi, ",<br>
?>
```

Hasilnya:

Gambar 3.23 Tampilan Contoh 28

d. Operator Penugasan (Assignement)

Operator yang digunakan untuk mengisi nilai ke dalam suatu variabel. Operator penugasan dalam PHP yaitu = (sama dengan). Contoh
11. \( x = y \) sama dengan \( y = x \)
    contoh koding:
    
    ```php
    <?php
    $x= 30;
    echo $x;
    ?>
    ```
    
    Hasilnya:
    
    ![Gambar 3.24 Tampilan Contoh 29](image)

12. \( x += y \) sama dengan \( x = x + y \)
    contoh:
    
    ```php
    <?php
    $x = 5;
    $x += 30;
    echo $x;
    ?>
    ```
    
    Hasilnya:
    
    ![Gambar 3.25 Tampilan Contoh 30](image)

13. \( x -= y \) sama dengan \( x = x - y \)
    contoh:
14. \( x * = y \) sama dengan \( x = x \times y \)

contoh:

```php
<?php
$x = 25;
$x *= 4;
echo $x;
?>
```

Hasilnya:

![Hasil contoh 14]

15. \( x /= y \) sama dengan \( x = x / y \)

contoh:

```php
<?php
$x = 7;
$x /= 2;
echo $x;
?>
```

Hasilnya:
16. \( x \% = y \) sama dengan \( x = x \% y \) 

contoh:

```
<?php
$x = 20;
$x %= 5;
echo $x;
```

Hasilnya:

```
2
```

e. Operator Pembanding (Relasional)

Operator yang digunakan untuk membandingkan dua nilai numerik atau string. Berikut daftar operator perbandingan.

**Tabel 3.4 Daftar Operator Perbandingan**

<table>
<thead>
<tr>
<th>Operator</th>
<th>Keterangan</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x = = y</td>
<td>Menghasilkan nilai true jika $x sama dengan $y</td>
</tr>
<tr>
<td>$x = = = y</td>
<td>Menghasilkan nilai true jika $x sama dengan $y, serta $x dan $y bertipe sama</td>
</tr>
<tr>
<td>$x != y</td>
<td>Menghasilkan nilai true jika $x tidak sama dengan $y</td>
</tr>
<tr>
<td>$x &lt; &gt; y</td>
<td>Menghasilkan nilai true jika $x tidak sama dengan $y</td>
</tr>
<tr>
<td>Operator</td>
<td>Deskripsi</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>$x != = y$</td>
<td>Menghasilkan nilai true jika $x$ tidak sama dengan $y$, serta $x$ dan $y$ tidak bertipe sama</td>
</tr>
<tr>
<td>$x &gt; y$</td>
<td>Menghasilkan nilai true jika $x$ lebih besar dari $y$</td>
</tr>
<tr>
<td>$x &lt; y$</td>
<td>Menghasilkan nilai true jika $x$ lebih kecil dari $y$</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Menghasilkan nilai true jika $x$ lebih besar atau sama dengan $y$</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Menghasilkan nilai true jika $x$ lebih kecil atau sama dengan $y$</td>
</tr>
</tbody>
</table>

Contoh penggunaan operator perbandingan

```php
<?php
  $x = 20;
  $y = 10;
  echo "Hasil dari $x == $y = " . ($x == $y) . "\n";
  echo "Hasil dari $x >= $y = " . ($x >= $y) . "\n";
  echo "Hasil dari $x <= $y = " . ($x <= $y) . "\n";
  echo "Hasil dari $x < $y = " . ($x < $y) . "\n";
  echo "Hasil dari $x > $y = " . ($x > $y) . "\n";
?>
```

Hasilnya:

```
Hasil dari 20 == 10 =
Hasil dari 20 >= 10 =
Hasil dari 20 != 10 =1
Hasil dari 20 <= 10 =
Hasil dari 20 < 10 =
Hasil dari 20 > 10 =
Hasil dari 20 >= 10 =
Hasil dari 20 <= 10 =
```
Untuk yang nilainya true nilai nya 1, dan yang false nilainya kosong. Operator perbandingan banyak digunakan dalam seleksi kondisi atau pemilihan (if –else), dan pengulangan(for, while, dan do-while).

f. Operator Logika

Yaitu operator untuk mengoperasikan dua buah operand yang bertipe Boolean. Hasilnya akan bertipe Boolean yaitu true atau false. Daftar operator logika dalam PHP

**Tabel 3.5 Daftar Operator Logika PHP**

<table>
<thead>
<tr>
<th>Nama Operator</th>
<th>Contoh</th>
<th>Keterangan</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND</td>
<td>$x &amp;&amp; $y</td>
<td>Logika AND Menghasilkan TRUE jika $x dan $y bernilai TRUE</td>
</tr>
<tr>
<td>AND</td>
<td>$x and $y</td>
<td>Logika AND Menghasilkan TRUE jika $x dan $y bernilai TRUE</td>
</tr>
<tr>
<td>Or</td>
<td>$x or $y</td>
<td>Logika or Menghasilkan TRUE jika salah satu $x atau $y bernilai TRUE</td>
</tr>
<tr>
<td></td>
<td>$x</td>
<td></td>
</tr>
<tr>
<td>xor</td>
<td>$x xor $y</td>
<td>Logika or Menghasilkan TRUE jika salah satu $x atau $y bernilai TRUE tetapi bukan keduanya</td>
</tr>
<tr>
<td>!</td>
<td>!$x</td>
<td>Menghasilkan TRUE jika $x = False</td>
</tr>
</tbody>
</table>
Contoh penggunaan:

```php
<?php
$x = true;
y = false;
echo "x = ", var_dumps ($x or $y);
$y = false;
echo "y = ", var_dumps ($x and $y);
$y = true;
echo "x and = ", var_dumps ($x and $y);
$x = false;
echo "x or = ", var_dumps ($x or $y);
$x = true;
echo "x xor = ", var_dumps ($x xor $y);
?>
```

Hasilnya:

```
Contoh and = bool(false)
Contoh or = bool(false)
Contoh xor = bool(true)
Contoh || = bool(true)
Contoh ? = bool(false)
```

**g. Operator Increment Decrement**

Operator Increment (kenaikan) digunakan untuk meningkatkan nilai variabel. Operator Decrement (Penurunan) digunakan untuk mengurangi nilai variabel. **Tabel 3.6 Daftar operator Increment/Decrement**

<table>
<thead>
<tr>
<th>Operator</th>
<th>Nama</th>
<th>Keterangan</th>
</tr>
</thead>
<tbody>
<tr>
<td>++$x</td>
<td>Pre Increment</td>
<td></td>
</tr>
<tr>
<td>$++</td>
<td>Post increment</td>
<td></td>
</tr>
<tr>
<td>--$x</td>
<td>Pre decrement</td>
<td></td>
</tr>
</tbody>
</table>
Contoh:

```php
<?php
$a = 15;
$b = 30;
$c = 30;
$d = 10;
echo "<pre>");
echo " $a + $a : ", $a + $a, "\n";
echo " $c + $c : ", $c + $c, "\n";
echo " $b - $b : ", $b - $b, "\n";
echo " $d - $d : ", $d - $d, "\n";
?>
```

Hasilnya:

contoh +15 : 16
contoh 20++ : 20
contoh --30 : 29
contoh 10-- : 10

### 3.7. Seleksi Kondisi

Seleksi kondisi dalam PHP untuk melakukan tindakan kondisi yang berbeda menentukan pernyataan mana yang akan di eksekusi sesuai kondisi yang didefinisikan.

Dalam PHP, seleksi kondisi dapat dilakukan dengan pernyataan berikut:

a. Pernyataan if... yaitu mengeksekusi beberapa kondisi jika satu syarat benar

Bentuk umum:

```php
If (kondisi) {
    pernyataan;
}
```

Contoh:
b. Pernyataan if…else yaitu mengeksekusi beberapa kondisi jika suatu kondisi benar dan kondisi lain jika kondisi itu salah

Bentuk umum

```php
if (kondisi) {
    pernyataan1;
} else {
    pernyataan2;
}
```

Contoh:

```php
<?php

if(nilai>=80){
    echo "Grade A";
} else{
    echo "Grade B";
}
?>
```

Hasilnya:

```
Grade A
```
c. Pernyataan if…elseif yaitu mengeksekusi kondisi yang berbeda dari dua kondisi atau lebih.

Bentuk umum:
```java
if (kondisi1) {
    pernyataan1;
}else if (kondisi2){
    pernyataan2;
}else if (kondisi3){
    pernyataan3;
}else {
    pernyataan4;
}
```

Contoh kode program:
```php
<?php
    $nilai=73;
    if($nilai>=80) {
        echo "Grade A";
    } elseif ($nilai>=75) {
        echo "Grade B";
    } elseif ($nilai>=65) {
        echo "Grade C";
    } elseif ($nilai>=45) {
        echo "Grade D";
    } else {
        echo "Grade E";
    }
?>
```

Hasilnya:

Grade C

d. Pernyataan Switch yaitu memilih salah satu kondisi dari beberapa kondisi yang akan dieksekusi program akan mencari nilai konstanta yang cocok yang terakandung didalam variabel.
Bentuk umum:
switch (variabel) {
    case nilai1:
        pernyataan1;
        break;
    case nilai2:
        pernyataan2;
        break;
    case nilai3:
        pernyataan3;
        break;
        ....
    default:
        pernyataan4;
}

Contoh kode program:

```php
<?php
    $prodi="Teknik Informatika";
    switch ($prodi) {
        case "Sistem Informasi":
            echo "Saya kuliah mengambil program studi Sistem Informasi";
            break;
        case "Teknik Informatika";
            echo "Saya kuliah mengambil program studi Teknik Informatika";
            break;
        case "Teknik Komputer";
            echo "Saya kuliah mengambil program studi Teknik Komputer";
            break;
        default:
            echo "Saya kuliah mengambil program studi Teknik Informatika";
    }
?>
```
3.8. Perulangan

Perulangan atau loop berfungsi sebagai instruksi program untuk mengulang beberapa baris perintah. Dalam pemrograman PHP ada beberapa jenis perulangan yaitu:

a. Perulangan for

Bentuk umum:

```php
for (inisialisasi; ekspresi; increment/decrement) {
    pernyataan1;
}
```

Contoh kode:

```php
for ($i=1; $i<=10; $i++)
{
    echo "$i:

echo "Kuliah...? BSI Aja !!";
    echo "$i";
}
?>
```
Hasilnya:

![Console output](image)

b. Perulangan while

Bentuk umum:

```plaintext
while (ekspresi)
{
    pernyataan;
    ....
}
```

Contoh kode:

```php
<?php
$x = 1;
while ($x <= 10) {
    echo "Bilangan ke-$x: 
    " . "$x"
    echo "<br>
    "$x++;
}
?>
```
c. Perulangan do while

Bentuk umum:

```php
    do{
        pernyataan;
    }while(ekspresi);
```

Contoh kode:

```php
<?php
    $x=1;
    do{
        echo"Bilangan Ke : $x"
        echo "<br>";
        $x++;
    }while ($x<=5);
?>
```

Hasilnya:
d. Perulangan foreach

Bentuk umum:
   ```php
   foreach (array as $value
       pernyataan;
   }
   ```

Contoh kode:
   ```php
   <?php
   $matkul = array("Web Programming","Logika Algoritma","PTIK","APS1");
   foreach ($matkul as $value) {
       echo "$value <br>
   }
   ?>
   ```

Hasilnya:
   ![Hasil kode]

3.9. Array

Array adalah variabel khusus atau pengenal didalam program yang berfungsi menampung lebih dari satu nilai yang berkaitan serta memiliki hubungan. Dalam PHP fungsi untuk membuat array yaitu dengan perintah : array()

Bentuk umum:
   ```php
   $nama_variabel = array (  
       Key1 => value1  
       Key2 => value2  
       Key3 => value3  
   )
   ```
Dalam Pemrograman PHP ada 3 jenis Array yaitu:
1. Array berindex yaitu array dengan index numerik atau diindex berdasarkan angka pada umumnya dimulai dari angka 0.

Contoh:

```php
<?php
$mobil = array("Honda","Chevrolet","Toyota","Nissan");
echo "Saya punya mobil ", $mobil[0], ", ", $mobil[1], ", ", $mobil[2], ", ", $mobil[3];
?>
```

Hasilnya:

```
Saya punya mobil Honda, Chevrolet, Toyota dan Nissan.
```

Contoh array berindex menggunakan perulangan for:

```php
<?php
$mobil = array("Honda","Chevrolet","Toyota","Nissan");
$jumlah = count($mobil);
for ($i=0; $i<$jumlah; $i++)
{
    echo $mobil[$i];
    echo "\n";
}
?>
```
2. Array Asosiatif

Yaitu array yang diindex berdasarkan kunci tertentu.

Contoh:

```php
<?php
    $mahasiswa = array(
        "nim"=>"18082317",
        "nama"=>"Dzimar Rauhullah",
        "alamat"=>"Tasikmalaya"
    );

    echo "NIM : $mahasiswa[nim]<br>";
    echo "Nama : $mahasiswa[nama]<br>";
    echo "Alamat : $mahasiswa[alamat]<br>";
?>
```

Hasilnya:

```
NIM : 18082317
Nama : Dzimar Rauhullah
Alamat : Tasikmalaya
```
Contoh array berindex menggunakan perulangan foreach:

```php
<?php
$mahasiswa = array(
    "nim" => "18082317",
    "name" => "Dzimar Rauhillaah",
    "alamat" => "Tasikmalaya"
);
foreach ($mahasiswa as $x => $y) {
    echo "$x : $y 
    echo "<br>
}
?>
```

Hasilnya:

```
nim : 18082317
name : Dzimar Rauhillaah
alamat : Tasikmalaya
```

3. Array multidimensi
Yaitu array yang berisi dua atau lebih array. Dalam pemrograman PHP ada beberapa array multidimensi yaitu Array dua dimensi, array tiga dimensi, dan array 3v dimensi atau lebih.

Contoh array 2 dimensi:

```php
<?php
$mobil = array(
    array("Toyota","Yaris","Avanza"),
    array("Honda","Jazz","Mobilio"),
    array("Nissan","March","Evalia")
);
echo "Jenis Mobil ".mobil[0][0]." : ".mobil[0][1]." dan ".mobil[0][2]."<br>";
echo "Jenis Mobil ".mobil[1][0]." : ".mobil[1][1]." dan ".mobil[1][2]."<br>";
echo "Jenis Mobil ".mobil[2][0]." : ".mobil[2][1]." dan ".mobil[2][2]."<br>
?>
```
Hasilnya:

<table>
<thead>
<tr>
<th>Jenis Mobil Toyota</th>
<th>Yaris dan Avanza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jenis Mobil Honda</td>
<td>Jazz dan Mobilio</td>
</tr>
<tr>
<td>Jenis Mobil Nissan</td>
<td>March dan Evalia</td>
</tr>
</tbody>
</table>
Deskripsi:
Membahas tentang pengenalan Java Script, penggunaan java script pada HTML, penggunaan java script pada PHP, pembuatan alert, pembuatan pop up box editor dan menampilkan hasil pada web browser yang digunakan.

Tujuan Pembelajaran:
Setelah melakukan pembelajaran ini, pembaca diharapkan mampu:
1. Menggunakan java script pada HTML
2. Menggunakan java script pada PHP
3. Menampilkan hasil tersebut pada web browser

4.1. Pengertian Java Script

A. Fungsi Java Script dalam

4.2. Penulisan dan Penggunaan Java Script

Untuk menjalankan JavaScript menggunakan aplikasi text editor seperti notepad, notepad++ atau menggunakan aplikasi macromedia dreamweaver dan sebuah web browser seperti Google Chrome atau Mozilla Firefox untuk menampilkannya.

A. Penulisan Java Script

Cara penulisan JavaScript mirip dengan penulisan bahasa pemograman web lainnya seperti PHP dan CSS, yakni dengan menysipkan kode JavaScript di dalam HTML. Meletakkan kode program JavaScript bisa diletakkan dibagian manapun dalam HTML, selama berada di dalam tag <script>. Ada beberapa posisi penulisan kode program java script yaitu:

1. Diawal HTML yaitu tag <script> diletakkan didalam tag <head>

Contoh:

```html
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title>Belajar Tag JavaScript</title>
5. <script>
6. alert("Selamat Datang!!");
7. </script>
8. </head>
9. <body>
10. </body>
```
12. <p>Saya Belajar JavaScript Di</p>
13. <p>Universitas Bina Sarana Informatika</p>
14.
15. </body>
16. </html>

Hasilnya ketika pertama kali dijalankan:

![Image of the output when the code is executed for the first time]

Hasilnya ketika diklik ok:

![Image of the output when the button is clicked]

Jadi disimpulkan bahwa pemrosesan program dimulai dari atas ke bawah halaman web maka kode tersebut akan dijalankan (diproses) sebelum ‘isi’ website ditampilkan.

2. Ditengah HTML yaitu tag <script> diletakkan didalam tag <body>

Contoh:
1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<title>Belajar Tag JavaScript</title>`
5. `<body>`
6. `<p>Saya Belajar JavaScript Di</p>`
7. `<div id="tes"></div>`
8. `<script>`
9. `document.getElementById("tes").innerHTML = "UBSI";`
10. `</script>`
11. `</body>`
12. `</html>`

Hasilnya:

![Hasil JavaScript](image)

Pada tag `<p>` ="tes"> </p> untuk menampilkan hasil dari javascript
3. Diakhir HTML yaitu tag `<script>` diletakkan sebelum tag akhir </html>

Contoh:

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<title>Belajar Tag JavaScript</title>`
5. `<body>`
6. `<p>Saya Belajar JavaScript Di</p>`
7. `<div id="tes"></div>`
8. `<script>`
9. `document.getElementById("tes").innerHTML = "UBSI";`
10. `</script>`
11. `</body>`
12. `</html>`
Hasilnya:

Saya Belajar JavaScript Di
Universitas Bina Sarana Informatika

Dari ketiga contoh diatas mengenai peletakan kode java script sesuai dengan kebutuhan.

4.3. Variabel

Didalam JavaScript variabel sebagai penampung untuk menyimpan nilai data. variabel dapat bertipe Angka (Number), String, Boolean, atau yang lainya serta tidak perlu mendeklarasikan jenis tipe data. Untuk mendeklarasikan variabel dalam javascript menggunakan keyword var. Aturan penamaan variabel dalam javascript yakni sama dengan aturan pembuatan identifier yaitu:

- Karakter pertama harus diawali dengan huruf, underscore (_) atau tanda dollar ($)
- Karakter kedua dan seterusnya bisa ditambahkan dengan huruf, angka, underscore (_) atau tanda dollar ($).

Contoh pendeklarasian variabel tipe numeric atau angka:
4. <h2>Contoh variabel</h2>

7. <p>Jika x=10 dan y=12</p>
8. <p>Berapa hasil dari x + y ?</p>
9. 
10. <p id="demo"></p>
11. <script>
12. var x=10;
13. var y=12;
14. var hasil=x+y;
15. document.getElementById("demo").innerHTML =
16. "Jawaban Hasil dari x + y adalah : " + hasil;
17. 
18. </body>
19. </html>

Pada kode baris ke 12,13,dan 14 adalah contoh pendeklarasian variabel pada java script, untuk kode baris 16 pada kode + hasil yaitu memanggil variabel atau menampilkan isi dari variabel yang akan ditampilkan pada tag <p id="demo"></p>. Maka hasilnya:

<table>
<thead>
<tr>
<th>Contoh variabel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jika x=10 dan y=12</td>
</tr>
<tr>
<td>berapa hasil dari x + y ?</td>
</tr>
<tr>
<td>Jawaban Hasil dari x + y adalah : 22</td>
</tr>
</tbody>
</table>

Selanjutnya contoh pendeklarasian variabel dengan tipe string:

1. <!DOCTYPE html>
2. <html>
3. <body>
4. <h2>Contoh variabel</h2>
Penjelasan kode diatas untuk kode baris 11 adalah pendeklarasian variabel, sedangkan untuk kode baris 12 dan 13 adalah memberikan nilai pada variabel teks1, teks2. Pemanggilan variabel pada javascript pada kode baris 14 dan 15 serta ditampilkan hasilnya pada kode baris 7 dan 8 seperti dibawah ini hasilnya:

4.4. **Fungsi (Function)**

Fungsi(function) dalam javascript adalah blok kode yang dirancang untuk melakukan tugas tertentu. Fungsi berjalan jika ada yang memanggilnya. Fungsi dalam JavaScript didefinisikan dengan kata kunci function, diikuti oleh nama fungsi, diikuti oleh tanda kurung (). Nama fungsi dapat berisi huruf, angka, garis bawah, dan tanda dolar (aturan yang sama dengan variabel). Tanda kurung dapat menyertakan nama parameter yang dipisahkan dengan koma:
Kode yang akan dieksekusi dalam fungsi ditempatkan di dalam kurung kurawal: 

Bentuk umum fungsi dalam Javascript:
```
function nama(parameter1, parameter2, parameter3) {
  // code yang akan dieksekusi
}
```
Parameter fungsi tercantum di dalam tanda kurung () dalam definisi fungsi. Argumen fungsi adalah nilai yang diterima oleh fungsi ketika dipanggil. Di dalam fungsi, argumen (parameter) berperilaku sebagai variabel lokal.

Contoh penggunaan fungsi menggunakan return:
```
1. <!DOCTYPE html>
2. <html>
3. <body>
4. <p>Contoh Fungsi Dalam JavaScript</p>
5. <p id="demo"></p>
6. <script>
7. var x = Perkalian(10, 2);
8. document.getElementById("demo").innerHTML = x;
9. function Perkalian(y, z) {
10.   return y * z;
11. }
12. </script>
13. </body>
14. </html>
```

Penjelasan untuk kode baris 10 yaitu fungsi yang akan dipanggil dan ditampung pada variabel x, dan pada kode baris 14 sebagai fungsi untuk mengembalikan nilai dari variabel y, dan z, Saat JavaScript mencapai pernyataan return, fungsi tersebut akan berhenti dieksekusi. Jika fungsi dipanggil dari pernyataan, JavaScript akan "kembali" untuk
mengeksekusi kode setelah pernyataan pemanggilan. Fungsi sering menghitung nilai balik. Nilai kembali "dikembalikan" kembali ke "pemanggil":
maka hasilnya seperti dibawah ini:

Contoh penggunaan fungsi dengan variabel lokal:

```html
1. <!DOCTYPE html>
2. <html>
3. <body>
4.
5. <p id="demo"></p>
6.
7. <script>
8. myKampus();
9.
10. function myKampus() {
11. var teks = "Universitas Bina Sarana Informatika";
12. document.getElementById("demo").innerHTML = teks;
13. }
14. </script>
15.
16. </body>
17. </html>
```

Penjelasan untuk kode baris 8 yaitu deklarasi nama fungsi nya, untuk kode baris 11 deklarasi dan pemebrian nilai variabel lokal teks. Variabel lokal hanya dikenali di dalam fungsinya, variabel dengan nama yang sama dapat digunakan dalam fungsi yang berbeda. Variabel lokal dibuat ketika fungsi dimulai, dan dihapus ketika fungsi selesai. Maka hasilnya:
4.5. Seleksi Kondisi

Seleksi kondisi digunakan untuk dalam JavaScript ada beberapa seleksi kondisi yaitu:

A. Seleksi Kondisi if

Digunakan untuk menentukan blok kode yang akan dieksekusi jika kondisi yang ditentukan benar.

Bentuk Umum:

```javascript
if (kondisi) {
    kode dieksekusi jika kondisinya benar;
}
```

Dalam penulisan if nya menggunakan huruf kecil (if) jangan menggunakan huruf kapital (IF).

Contoh kode program:

```html
<!DOCTYPE html>
<html>
<body>
<p id="demo"></p>
<script>
var teks = "UBSI";
if (teks == "UBSI") {
    document.getElementById("demo").innerHTML = "Universitas Bina Sarana Informatika";
}
</script>
</body>
</html>
```
Kode program diatas tidak akan menampilkan “Universitas Bina Sarana Informatika” jika variabel teks tidak sama dengan UBSI. Karena variabel teks sama dengan UBSI maka tampil seperti dibawah ini:

B. Seleksi Kondisi if else

Digunakan untuk menentukan blok kode yang akan dieksekusi, jika kondisi if tidak terpenuhi yaitu dengan menambahkan perintah else.

Bentuk umum:

```plaintext
if (kondisi) {
    kode dieksekusi jika kondisinya benar;
}
else {
    kode dieksekusi jika kondisinya salah;
}
```

contoh kode program:

```html
1. <!DOCTYPE html>
2. <html>
3. <body>
4. 
5. <p id="demo"></p>
6. 
7. <script>
8. var angka = 13;
9. if (angka%2 == 0) {
10.    document.getElementById("demo").innerHTML =
11.    "Bilangan Genap";
12. } else {
13.    document.getElementById("demo").innerHTML =
14.    "Bilangan Ganjil";
15. }
16. }
```
Dari contoh diatas jika sisa hasil bagi % (mod) variabel angka dengan 2 adalah 0, maka jalankan TRUE “Bilangan Genap”. Jika tidak maka jalankan FALSE “ Bilangan Ganjil “. Kondisi angka%2==0 hanya akan bernilai TRUE jika nilai dari variabel angka bisa habis dibagi 2, yang menandakan angka adalah bilangan genap, selain itu dapat dipastikan bahwa angka adalah bilangan ganjil karena nilai angka=13 berarti bilangan ganjil maka hasilnya:

C. Seleksi Kondisi if else if
Digunakan untuk menentukan kondisi baru untuk dieksekusi jika kondisi lama salah.
Bentuk umum:

```java
if (kondisi1) {
    kode dieksekusi jika kondisi1 benar;
} else if (kondisi2) {
    kode dieksekusi jika kondisi2 benar;
} else {
    kode dieksekusi jika kondisi1 dan kondisi2 salah;
}
```

Contoh kode program:

1. ```html```
2. ```html```
Dari contoh diatas akan menghasilkan output:

D. Seleksi kondisi switch case
Pernyataan yang digunakan untuk memilih salah satu dari banyak blok kode program yang akan dieksekusi
Bentuk umum:
```javascript
switch(ekspresi) {
    case x:
        \textit{Kode blok program yang akan dieksekusi}
        break;
    case y:
        \textit{Kode blok program yang akan dieksekusi}
        break;
    default:
        \textit{Kode blok program yang akan dieksekusi}
```
**Kode blok program yang akan dieksekusi**

} 

cara kerjanya:
Ekspresi dievaluasi Kemudian nilai ekspresi dibandingkan dengan nilai setiap kasus(case x, y) .Jika ada kecocokan, blok kode program terkait dijalankan.

Contoh program:

```
1. <!DOCTYPE html>
2. <html>
3. <body>
4. 
5. <h2>Contoh Switch</h2>
6. 
7. <p id="demo"></p>
8. 
9. <script>
10. var x = 1;
11. 
12. switch (x) {
13.   case 1:
14.     text = 'Angka satu';
15.     break;
16.   case 2:
17.     text = 'Angka Dua';
18.     break;
19.   default:
20.     text = 'Tidak ada keterangan';
21. }
22. document.getElementById('demo').innerHTML = text;
23. </script>
24. 
25. </body>
26. </html>
```

Dari contoh diatas akan menghasilkan output:
4.6. Perulangan

Looping (perulangan) yaitu intruksi untuk membentuk aksi kerja secara berulang pada blok yang sama, dimana didalam blok yang akan diulang ada terdapat statement atau pernyataan. JavaScript mendukung beberapa perulangan yaitu:

a. Perulangan for

Bentuk umum:

```javascript
for (inisialisasi ekspresi; kondisi perulangan; ekspresi penambahan) {
    // pernyataan yang akan dikesekusi
}
```

contoh kode program:

```html
<!DOCTYPE html>
<html>
<body>
<h2>Contoh Perulangan For</h2>
<p id="demo"></p>
<script>
var text = "";
var i;
for (i = 0; i < 10; i++) {
    text += "Angka " + i + "<br>";
}
document.getElementById("demo").innerHTML = text;
</script>
</body>
</html>
```
Hasil outputnya:

![Contoh Perulangan For](image)

Dari contoh diatas penjelasannya:

i=0 sebagai inisialisai ekspresi yaitu mulainya dari 0 dan
i<10 Berulang sebanyak 10 kali dimulai dari 0
i++ sebagai ekspresi peubah atau penambahan sebanyak i+1

b. Perulangan for in

Perulangan for in digunakan untuk memanggil properti dari objek

Contoh kode program:

```html
<!DOCTYPE html>
<html>
<body>
<h2>Contoh Perulangan For In</h2>
<p id="demo"></p>
<script>
var txt = "";
var mahasiswa = {Nama:"Dzimar Rauhillah", Alamat:"Tasikmalaya", Usia:17};
var x;
for (x in mahasiswa) {
  txt += mahasiswa[x] + " ";
}
document.getElementById("demo").innerHTML = txt;
</script>
</body>
</html>
```
Hasil outputnya:

```
Contoh Perulangan For In
Dinata Rashillah Tasikmalaya 17
```

c. Perulangan for of

Perulangan for of digunakan melalui nilai objek yang dapat diubah

Contoh kode program:

```html
1. <DOCTYPE html>
2. <html>
3. <body>
4. 
5. <h2>Contoh Perulangan for of</h2>
6. 
7. 
8. <p id="demo"></p>
9. 
10. <script>
11. var kota = ['Tasikmalaya', 'Bandung', 'Jakarta'];
12. var x;
13. 
14. for (x of kota) {
15.    document.write(x + "<br>");
16. }
17. </script>
18. 
19. </body>
20. </html>
```
Hasil outputnya:

Contoh Perulangan for of

Tasikmalaya
Bandung
Jakarta

d. Perulangan while

Perulangan while adalah untuk melakukan tugas berulang selama pernyataan sesuai dengan kondisi atau bernilai benar

Bentuk umum:

```javascript
while (kondisi) {
    // pernyataan yang akan di eksekusi
}
```

contoh kode program:

```html
<!DOCTYPE html>
<html>
<body>

<h2>Contoh perulangan while</h2>
<p id="demo"></p>

<script>
var text = "

</script>
</body>
</html>
```


```javascript
11. var i = 0;
12. while (i < 10) {
13.   text += "<br>Angka " + i;
14.   i++;
15. }
16. document.getElementById("demo").innerHTML = text;
17. </script>
18.
19. </body>
20. </html>

Hasil outputnya:

![Contoh perulangan while](image)

Dari contoh diatas penjelasannya:
i=0 sebagai inisialisai ekspresi yaitu mulainya dari 0 dan
i<10 Berulang sebanyak 10 kali dimulai dari 0
i++ sebagai ekspresi peubah atau penambahan sebanyak i+1

e. Perulangan do while
Perulangan do while yaitu perulangan yang mengeksekusi
pernyataan tertentu hingga kondisi pemeriksaan penyataan
tersebut bernilai false . Kondisi akan diperiksa setelah
pernyataan dieksekusi/dijalankan dengan hasil penyataan
tersebut yang dijalankan sekurangnya sekali. sedangkan while
akan mencek kondisi di depan atau awal (sbelum mengulang).
Bentuk umum:

---

74
do {
    // pernyataan yang akan dieksekusi
}
while (kondisi);

contoh kode program

```html
<!DOCTYPE html>
<html>
<body>
<h2>Contoh perulangan do while</h2>
<p id="demo"></p>
<script>
var text = "";
var i = 0;
do {
    text += "<br>Angka " + i;
i++;
} while (i < 10);
document.getElementById("demo").innerHTML = text;
</script>
</body>
</html>
```

Hasil outputnya:
Perulangan do/while akan mengecek kondisi di belakang (sesudah mengulang)

4.7. Array

Array adalah variabel khusus, yang dapat menampung lebih dari satu nilai pada suatu waktu. Array dapat menyimpan banyak nilai dengan satu nama variabel, dan dapat mengakses nilai variabel tersebut dengan nomor indeks.

Bentuk umum:

```
var nama_array = [item1, item2, ...];
```

Contoh kode program:

```html
<!DOCTYPE html>
<html>
<body>
<h2>Contoh JavaScript Array</h2>
<p id="demo"></p>
<script>
var mahasiswa = ['Jafar', 'Irwan', 'Rifki'];
document.getElementById("demo").innerHTML = mahasiswa;
</script>
</body>
</html>
```
Hasil outputnya:

Contoh kode program mengakses elemen array dengan merujuk ke nomor index:

```html
<!DOCTYPE html>
<html>
<body>
<h2>Contoh JavaScript Arrays</h2>
<p id="demo"></p>
<script>
var mahasiswa = ['Jafar', 'Irwan', 'Rifki'];
document.getElementById('demo').innerHTML = mahasiswa[0];
</script>
</body>
</html>
```

Hasil outputnya:
Disebutkan indexnya yg ingin ditampilkan contoh diatas mahasiswa[0] yang dipanggil urutan pertama yaitu akan tampil “Jafar”, kalau mau panggil selanjutnya maka mahasiswa[1], [2], dan seterusnya
BAB 5
PENGENALAN CSS

Deskripsi:
Membahas tentang pengenalan CSS, penggunaan CSS, sintak CSS dan desain halaman web dengan CSS serta menampilkan hasil pada web browser yang digunakan.

Tujuan Pembelajaran:
Setelah melakukan pembelajaran ini, pembaca diharapkan mampu:
1. Membuat CSS pada halaman web
2. Desain halaman web dengan CSS
3. Menampilkan hasil tersebut pada web browser

5.1. Pengertian CSS

CSS adalah singkatan dari Cascading Style Sheets yaitu kumpulan kode program yang digunakan untuk mendesain atau mempercantik tampilan halaman HTML. Menjelaskan bagaimana elemen-elemen HTML ditampilkan di layar. Dengan CSS kita bisa mengubah desain dari text, warna, gambar dan latar belakang dari (hampir) semua kode tag HTML sehingga dapat mengontrol tata letak beberapa halaman web sekaligus. CSS biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi. HTML ditujukan untuk membuat struktur, atau konten dari halaman web. Sedangkan CSS digunakan untuk tampilan dari halaman web tersebut. Istilahnya, “HTML for content, CSS for Presentation”.

5.2. Penggunaan CSS

Dalam penggunaan CSS terdapat beberapa cara untuk menginput kode CSS ke dalam HTML, yaitu:
A. Inline Style

Metode Inline Style adalah cara menginput kode CSS langsung ke dalam tag HTML dengan menggunakan atribut style.

Contoh penggunaannya:

1. <DOCTYPE html>
2. <html>
3. <head>
4. <title>Contoh Inline Style CSS</title>
5. </head>
6. <body>
7. <h2 style="background-color:blue; color:white">Universitas Bina Sarana Informatika</h2>
8. </body>
9. </html>

Baris kode no.07 diatas menyisipkan atribut style pada tag <h2>, nilai dari atribut style ini adalah kode CSS yang ingin diterapkan. Maka hasilnya seperti dibawah ini:

![Contoh Inline Style CSS](image)

Tidak disarankan untuk penggunaan CSS yang langsung digabungkan dengan HTML seperti kode diatas karena tidak memenuhi tujuan dibuatnya CSS agar desain terpisah dengan konten.

B. Internal Style Sheets

Metode Internal Style Sheets, atau disebut juga Embedded Style Sheets digunakan untuk memisahkan kode CSS dari tag HTML namun tetap dalam satu halaman HTML. Atribut style yang sebelumnya berada di
dalam tag, dikumpulkan pada pada sebuah tag <style>. Tag style ini harus berada pada bagian <head> dari halaman HTML.
Contoh penggunaan nya:

```html
1. <DOCTYPE html>
2. <html>
3. <head>
4. <style>
5.  p {
6.    color: #000099;
7.    text-align: center;
8.  }
9. </style>
10. </head>
11. <body>
12. <p>Selamat Datang Di</p>
13. <p>Universitas Bina Sarana Informatika</p>
14. </body>
15. </html>
```

Contoh metode internal style sheets diatas sudah jauh lebih baik daripada inline style, karena sudah memisahkan CSS dari HTML. Seluruh kode CSS akan berada pada tag head dari HTML. Namun kekurangan menggunakan internal style sheets, jika kita memiliki beberapa halaman dengan style yang sama, maka kita harus membuat kode CSS pada masing-masing halaman tersebut. Hal ini dapat diatasi dengan menggunakan metode external style sheets.

C. External Style Sheets.

Metode External Style Sheets digunakan untuk membuat kode CSS tersebut kedalam sebuah file tersendiri yang terpisah sepenuhnya dari halaman HTML. Setiap halaman yang membutuhkan kode CSS, tinggal dipanggil file CSS tersebut.

Contoh penggunaan nya
Membuat file style.css (untuk nama bebas yang paling penting ekstensi nya harus.css)

```css
1. p {
2.   color: #FFFFFF;
3.   text-align: center;
4.   background-color: #000066;
5. }
```

Kemudian membuat file html nya dengan nama contoh1.html:

```html
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <link rel="stylesheet" type="text/css" href="style.css">
5. </head>
6. <body>
7. <p>Selamat Datang Di</p>
8. <p>Universitas Bina Sarana Informatika</p>
9. </body>
10. </html>
```

Pada metode link **external style sheets** ini, kita menggunakan atribut `href` pada tag `<link>`, yang akan berisi alamat dari halaman CSS, dalam hal ini `style.css`. dalam penyimpanannya file css harus satu folder dengan file html apabila berbeda folder maka disesuaikan dengan alamat folder nya.

Hasil outputnya:

![Output HTML](image)

Dari ketiga jenis cara input CSS ke dalam halaman HTML, yang **paling direkomendasikan** adalah **metode external style sheets**, baik
menggunakan @import maupun dengan tag <link>. Karena dengan menggunakan kode CSS yang dipisahkan, seluruh halaman web dapat menggunakan file CSS yang sama, dan jika kita ingin mengubah seluruh tampilan halaman website, kita hanya butuh mengubah 1 file CSS saja.

5.3. Pengertian Selector, Properti, dan Value

A. Selector

Selector yaitu tag yang menghubungkan tag html dengan tag CSS yang digunakan untuk mencari bagian web yang ingin dimanipulasi atau yang ingin di-style. Misalnya: “cari seluruh tag <p>”, atau “cari seluruh tag HTML yang memiliki atribut class="warning"” atau “cari seluruh link yang ada di dalam tag <p>”.

Selector paling dasar dari CSS adalah tag dari HTML itu sendiri, misalnya: tag p, i, h1, li, dll. Selector didalam CSS dapat menjadi kompleks tergantung kebutuhannya. Mengenai selector, akan kita bahas secara lebih detail dalam tutorial-tutorial CSS selanjutnya.

Selektor CSS digunakan untuk memilih elemen HTML berdasarkan nama elemen, id, kelas, atribut, dan lainnya.

Contoh:

```html
1. p {
2.   text-align: center;
3.   color: blue;
4. }
```

semua elemen <p> pada kode diatas akan ditampilkan sejajar ditengah, dengan warna teks biru.

Bagian-bagian dari selector yaitu:

1. Id selector

Id selector digunakan sebagai atribut id dari elemen HTML untuk memilih elemen tertentu penamaan nya harus unik di dalam satu halaman, jadi id selector digunakan untuk memilih satu elemen unik untuk memilih elemen dengan id tertentu, penulisannya menggunakan hash (#), diikuti oleh id elemen.

Contoh penulisannya:
Contoh penerapan dalam program:

```
1. <DOCTYPE html>
2. <html>
3. <head>
4. <style>
5. #baris1 {
6.   text-align: center;
7.   color: blue;
8. }
9. </style>
10. </head>
11. <body>
12. 
13. <p id="para1">Selamat Pagi</p>
14. <p>Selamat Siang</p>
15. 
16. </body>
17. </html>
```

Untuk kata “selamat pagi” warna nya akan berwarna biru karena dipanggil selector id="baris1". Sedangkan untuk kata “selamat siang” tidak akan berubah walaupun sama <p> tetapi tidak memanggil id nya. Hasil tampilan nya :

![Hasil Tampilan](image)

2. Class Selector
Class selector memilih elemen dengan atribut kelas tertentu. Untuk memilih elemen dengan kelas tertentu, ditulis dengan karakter titik (.), Diikuti dengan nama kelas.

Contoh penulisannya:

```html
1. .posisi {
2.  text-align: center;
3.  color: blue;
4. }
```

Contoh penerapan dalam program:

```html
1. <!DOCTYPE html>
2. <html>
3.  <head>
4.   <style>
5.     .posisi {
6.       text-align: center;
7.       color: blue;
8.     }
9.   </style>
10. </head>
11. <body>
12.  
13.   <h3 class="posisi">Universitas Bina Sarana Informatika</h3>
14.   <p class="posisi">Program Studi Sistem Informasi</p>
15.  
16. </body>
17. </html>
```

Untuk contoh diatas posisi nya akan sama tampil ditengah karena memanggil class="posisi" tetapi tetapi tampilan bentuk teks nya kan berbeda yaitu <h3> dan yang satu lagi <p>. sperti dibawah ini tampilannya:
3. Group Selector

Group selector digunakan untuk mengelompokkan selector yang memiliki nilai yang sama.

Contoh sebelum dikelompokkan:

```html
1. h1 {
2.   text-align: center;
3.   color: blue;
4. }
5.
6. h2 {
7.   text-align: center;
8.   color: blue;
9. }
10.
11. p {
12.   text-align: center;
13.   color: blue;
14. }
```

Akan menjadi seperti dibawah ini setelah dikelompokkan:

```html
1. h1, h2, p {
2.   text-align: center;
3.   color: blue;
4. }
```

Contoh penerapannya:

```html
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <style>
```

87
Universitas Bina Sarana Informatika
Fakultas Teknologi Informasi
Program Studi Sistem Informasi

B. Property

Property CSS adalah *jenis style*, atau elemen apa yang akan diubah dari sebuah tag HTML. CSS memiliki puluhan property yang dapat digunakan agar menampilkan hasil akhir yang kita inginkan. Hampir semua property dalam CSS dapat dipakai untuk seluruh selector. Jika selector digunakan misalnya untuk “mencari seluruh tag `<p>`“, maka property adalah “*efek apa yang ingin dimanipulasi dari tag p tersebut*“, seperti ukuran text, warna text, jenis fontnya.
Contoh:

```css
p {
  text-align: center;
  color: blue;
}
```

Dari kode diatas kata “text-align” dan “color” disebut dengan property

**C. Value**

Value CSS adalah nilai dari property. Misalkan untuk property background-color yang digunakan untuk mengubah warna latar belakang dari sebuah selector, value atau nilainya dapat berupa red, blue, black, atau white.

```css
p {
  text-align: center;
  color: blue;
}
```

Dari kode diatas kata “center” dan “blue” disebut dengan value

Untuk lebih ringkasnya tentang selector, property, dan value seperti gambar dibawah ini:
5.4. Desain Halaman Web dengan CSS

Sebelum membuat desain halaman web buat dulu folder di htdoc dengan nama desain_web untuk menyimpan semua filenya. Seperti gambar dibawah ini:

Gambar

A. Desain CSS

Desain CSS ini adalah sebagai tahapan pertama untuk desain halaman web sebelum desain halaman lain nya. Buat file style.css kode nya seperti dibawah ini:

1. @charset "utf-8";
2. *
3. box-sizing: border-box;
4. }
5. body {
6. font-family: Arial;
7. padding: 10px;
8. background: #f1f1f1;
9. }
10. input[type=text], select, textarea {
11. width: 100%;
12. padding: 12px;
13. border: 1px solid #ccc;
14. border-radius: 4px;
15. resize: vertical;
16. }
17. label {
18. padding: 12px 12px 12px 0;
19. display: inline-block;
20. }
input[type=submit] {
  background-color: #4CAF50;
  color: white;
  padding: 12px 20px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
  float: right;
}

input[type=submit]:hover {
  background-color: #45a049;
}

.container {
  border-radius: 5px;
  background-color: #f2f2f2;
  padding: 20px;
}

.col-25 {
  float: left;
  width: 25%;
  margin-top: 6px;
}

.col-75 {
  float: left;
  width: 75%;
  margin-top: 6px;
}

.header {
  padding: 30px;
  text-align: center;
  background: white;
}

.header h1 {
  font-size: 50px;
}

.topnav {
}
overflow: hidden;

background-color: #333;
}

.topnav a {
  float: left;
  display: block;
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}

.topnav a:hover {
  background-color: #ddd;
  color: black;
}

.leftcolumn {
  float: left;
  width: 75%;
}

.rightcolumn {
  float: left;
  width: 25%;
  background-color: #f1f1f1;
  padding-left: 20px;
}

.fakeimg {
  background-color: #aaa;
  width: 100%;
  padding: 20px;
}

.card {
B. Desain halaman header

Untuk desain halaman header nya buat file dengan nama `header.php`. kode nya seperti dibawah ini:

```
  <div class="header">
```

Jangan lupa simpan di folder `htdocs/desain_web` yang sudah dibuat sebelumnya
C. Desain Halaman Menu
Membuat halaman menu buat file dengan nama menu.php. kode nya seperti dibawah ini:

```html
1. <div class="topnav">
2. <a href="index.php">Beranda</a>
3. <a href="profil.php">Profil</a>
4. <a href="kontak.php">Kontak</a>
5. <a href="login.php" style="float:right">Login</a>
6. </div>
```

D. Desain Halaman Footer
Membuat halaman footer buat file dengan nama footer.php. kode nya seperti dibawah ini:

```html
1. <div class="footer">
2. <h2>Copyright 2019 Universitas Bina Sarana Informatika</h2>
3. </div>
```

E. Desain Halaman Konten_kanan
Membuat halaman konten kanan buat file dengan nama konten_kanan.php. kode nya seperti dibawah ini:

```html
1. <div class="rightcolumn">
2. <div class="card">
3. <h2>Tentang Kami</h2>
4. <div class="fakeimg" style="height:100px;">Image</div>
5. <p>Ini adalah penjelasan tentang website</p>
6. </div>
7. <div class="card">
8. <h3>Follow Me</h3>
9. <p>Instagram</p>
```
F. Desain Halaman Beranda
Membuat halaman beranda sebagai halaman pertama ketika web diakses. buat file dengan nama index.php. kode nya seperti dibawah ini:

```php
1. <p>Facebook</p>
2. </div>
3. </div>

F. Desain Halaman Beranda
Membuat halaman beranda sebagai halaman pertama ketika web diakses. buat file dengan nama index.php. kode nya seperti dibawah ini:

```
G. Desain Halaman Profil

Membuat halaman profil. buat file dengan nama `profil.php`. kode nya seperti dibawah ini:

```html
<!DOCTYPE html>
<html>
<head>
    <title>Profil Website Layout</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
<?php
include ('header.php');
include ('menu.php');
?>
<div class="row">
    <div class="leftcolumn">
        <div class="card">
            <h2>Profil Website</h2>
            <h5>Tasikmalaya, 17 Agustus 2019</h5>
            <div class="fakeimg" style="height:200px;">Image</div>
            <p>ini adalah deskripsi gambar nya atau deskripsi tentang profil websitenya</p>
        </div>
    </div>
<?php
include ('konten_kanan.php');
?>
</div>
<?php
include ('footer.php');
?>
</body>
</html>
```
**H. Desain Halaman Kontak**

Membuat halaman kontak buat file dengan nama `kontak.php`. Kodenya seperti dibawah ini:

```html
<!DOCTYPE html>
<html>
<head>
<title>Kontak Website Layout</title>
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
<?php
include ('header.php');
include ('menu.php');
?>
<div class="row">
<div class="leftcolumn">
<div class="card">
<h2>Hubungi Kami</h2>
<form action="/action_page.php">
<div class="row">
<div class="col-25">
<label for="fname">Nama Lengkap</label>
</div>
<div class="col-75">
<input type="text" id="fname" name="nama" placeholder="Nama Lengkap..">
</div>
</div>
</form>
</div>
</div>
</div>
</body>
</html>
```
I. Desain Halaman Login

Membuat halaman login buat file dengan nama login.php. Kode nya seperti dibawah ini:

```html
<!DOCTYPE html>
<html>
<head>
<title>Login Website Layout</title>
<link rel="stylesheet" type="text/css" href="style.css"/>
</head>
<form>
<div class="row">
    <div class="col-25">
        <label for="subject">Pesan</label>
    </div>
    <div class="col-75">
        <textarea id="subject" name="pesan" placeholder="Tulis Pesan.." style="height:200px"></textarea>
    </div>
</div>
<input type="submit" value="kirim">
</form>
</html>
```
<?php
include ('header.php');
include ('menu.php');
?>

<div class="row">
  <div class="leftcolumn">
    <div class="card">
      <h2>Login Form</h2>
      <div class="container">
        <form action="/action_page.php">
          <div class="row">
            <div class="col-25">
              <label for="fname">Username</label>
            </div>
            <div class="col-75">
              <input type="text" id="fname" name="username" placeholder="username..">
            </div>
          </div>
          <div class="row">
            <div class="col-25">
              <label for="lname">Password</label>
            </div>
            <div class="col-75">
              <input type="text" id="lname" name="password" placeholder="Password...">
            </div>
          </div>
          <div class="row">
            <input type="submit" value="Login">
          </div>
        </form>
      </div>
    </div>
  </div>
</div>

<?php
include ('konten_kanan.php');
?>
Setelah selesai semua desain halamannya pastikan semua file tersimpan dalam satu folder yaitu folder htdoc/desain_web. Lihat gambar dibawah ini:

Gambar File Desain Halaman web

Apabila sudah tersimpan semuanya. Cek hasil tampilan nya buka browsernya dan ketikkan link http://localhost/desain_web/ maka tampilan pertamanya atau tampilan halaman beranda seperti dibawah ini:
Gambar 5.1 Tampilan Halaman Beranda
Selanjutnya klik profil, maka tampilan nya seperti dibawah ini:

Gambar 5.2 Tampilan Halaman Profil
Selanjutnya klik kontak, maka tampilan nya seperti dibawah ini
Gambar 5.3 Tampilan Halaman Kontak

Selanjutnya klik login, maka tampilan nya seperti dibawah ini:

Gambar 5.4 Tampilan Login

Apabila tampilan nya sama seperti diatas berarti sudah selesai, silahkan dimodifikasi sesuai dengan kebutuhan.
DAFTAR PUSTAKA

Abdullah, Rohi. 2015. Web Programing is Easy. Jakarta: Elek Media Komputindo


Dunia Ilmu komputer. Diambil dari https://www.duniailkom.com/


BIODATA PENULIS


