

# Dasar Pemograman Web



HTML



JS



CSS



**Penyusun :**

Miftah Farid Adiwisastra, S.T, M.Kom

Agung Baitul Hikmah, S.Kom, M.Kom

Ai ilah Warnilah, S.T, M.Kom



Penerbit CV. SARNU UNTUNG

**Dasar Pemograman Web**

**Hak Cipta© Miftah Farid Adiwiasta,S.T,M.Kom; Agung Baitul Hikmah,S.Kom,M.Kom;  
dan Ai ilah Warnilah,S.T,M.Kom 2019**

Penulis:

**Miftah Farid Adiwiasta,S.T,M.Kom**

**Agung Baitul Hikmah,S.Kom,M.Kom**

**Ai ilah Warnilah,S.T,M.Kom**

ISBN : 978-602-5650-54-3

Desain sampul dan tata letak:

Yahya Abdulloh

Penerbit:

CV. Sarnu Untung

Redaksi:

Jalan R.Suprpto, Gg.Pringgondani, RT 07, RW 21,

Purwodadi-Grobogan, Jawa Tengah,58111

No. HP 085726280111

Email: [ntoeng87@yahoo.co.id](mailto:ntoeng87@yahoo.co.id)

**Anggota IKAPI** (No. 146/JTE/2015)

Cetakan pertama, Oktober 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara

Apapun tanpa ijin tertulis dari penerbit

## **KATA PENGANTAR**

Dengan mengucapkan Puji syukur kehadiran 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 aja 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**

Web dapat diartikan “sekumpulan halaman yang berupa laman yang berisi informasi dalam bentuk data digital baik itu berupa text, gambar, animasi, video, dan audio lainnya yang disediakan melalui jalur koneksi internet”. Abdullah (2015) Halaman website biasanya berupa dokumen yang ditulis dalam format Hyper Text Markup Language (HTML), yang bisa diakses melalui HTTP. HTTP adalah suatu protocol yang menyampaikan berbagai informasi dari server website untuk ditampilkan kepada user atau pemakai melalui web browser”. Pada awalnya, koneksi internet hanya bisa menampilkan teks saja namun sejak tahun 1990, Timothy Berners-Lee, mengajukan protokol tentang cara penampilan informasi melalui media internet. Kemudian oleh CERN (Laboratorium Fisika Partikel di Swiss) proposal tersebut mendapat respon positif alias ditindaklanjuti dengan mengembangkan World Wide Web.

World Wide Web atau WWW adalah suatu program yang ditemukan oleh Tim Bernes-Lee pada tahun 1991”. Awalnya Bernes-Lee hanya ingin menemukan cara untuk menyusun arsiparsip risetnya. Pada tahun 1989, Bernes-Lee membuat pengajuan untuk proyek pembuatan hiperteks global, kemudian pada bulan Oktober 1990, “Warning Wera Wanua” sudah dapat dijalankan dalam lingkungan CERN (Pusat Penelitian Fisika Partikel 8 Eropa)

Pada musim panas pada tahun 1991, World Wide Web atau WWW secara resmi digunakan secara luas pada jaringan internet.

## 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

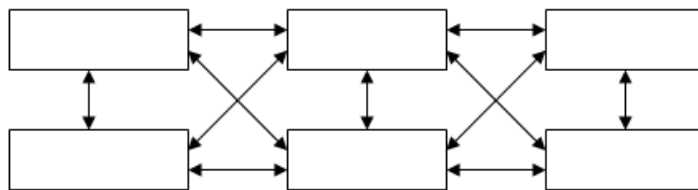


Sumber: Petrus Andre (2014)

**Gambar 1.1 Struktur Navigasi Linier**

## 2. Struktur Navigasi Non-linier

Struktur navigasi non-linier atau struktur tidak berurut merupakan pengembangan dari struktur navigasi linier. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur nonlinier ini berbeda dengan percabangan pada struktur hirarki, karena pada percabangan nonlinier ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada Master Page dan Slave Page. Contoh gambar

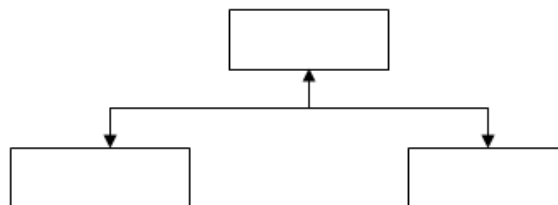


Sumber: Petrus Andre (2014)

**Gambar 1.2 Struktur Navigasi Non-Linier**

## 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

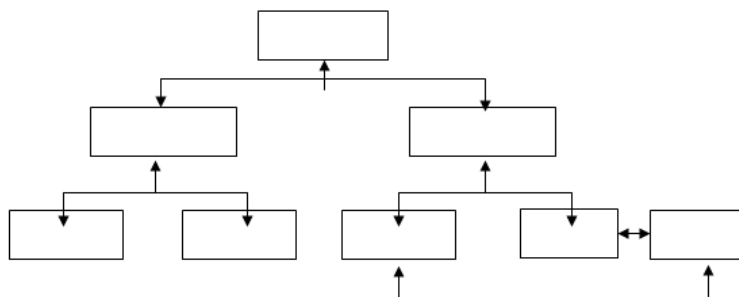


Sumber: Petrus Andre (2014)

**Gambar 1.3 Struktur Navigasi Hirarki**

#### 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



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.

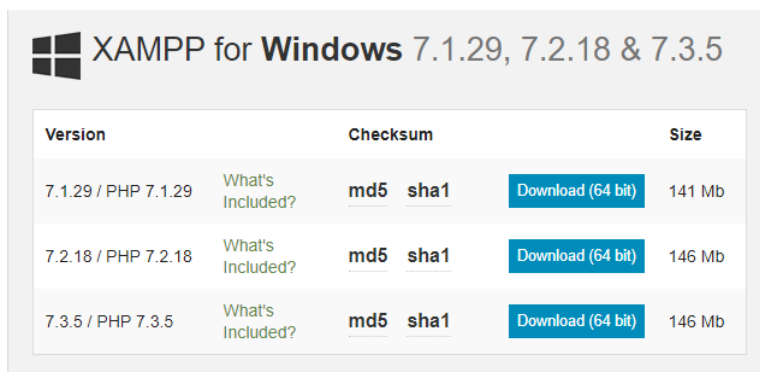
Aplikasi XAMPP ini dibuat oleh Apache Friends dan installer-nya bisa langsung diunduh dari situs mereka. Isi aplikasinya juga sudah sangat komplit, antara lain:

- Apache
- MySQL
- PHP
- phpMyAdmin

- FileZilla FTP Server
- Tomcat
- XAMPP Control Panel

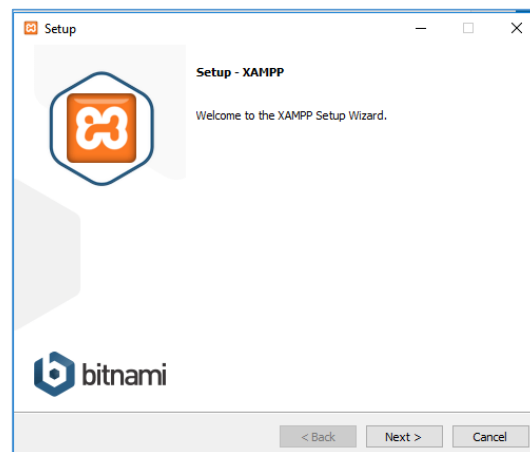
Cara Instalasi xampp yaitu:

1. Download aplikasi XAMPP pada web resminya <https://www.apachefriends.org/download.html> . Pilih versi terbaru



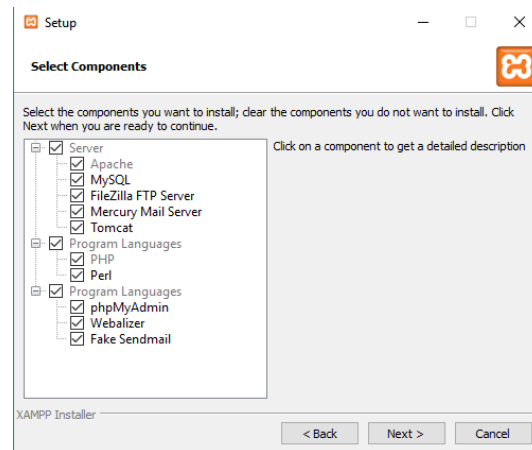
**Gambar 1.5 Download xampp**

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



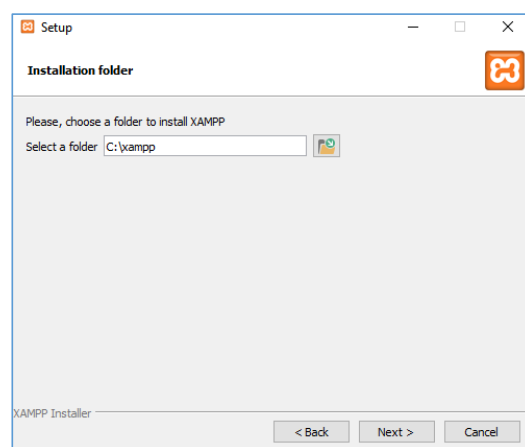
**Gambar 1.6 Instalasi xampp**

3. Selanjutnya tampilan untuk memilih komponen ceklis semuanya seperti dibawah ini dan pilih next:



**Gambar 1.7 Instalasi xampp Pilih Komponen**

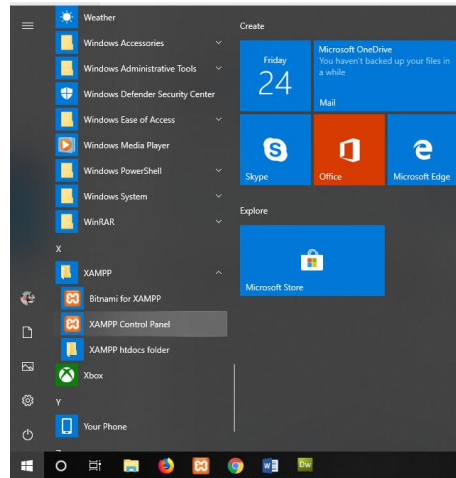
4. Selanjutnya untuk memilih folder dan pilih folder bawaannya xampp seperti dibawah ini pilih Next:



**Gambar 1.8 Instalasi xampp Pilih Folder**

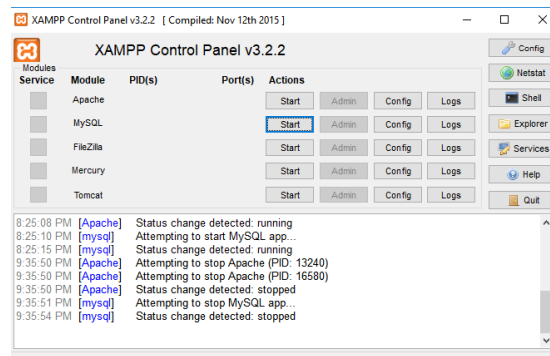
5. Proses instalasi berjalan dan pilih Finish
6. Untuk menggunakan nya jalankan xampp yang telah di instal bisa membukanya melalui shortcut desktop atau star menu seperti gambar dibawah ini dan pilih Xampp Control Panel





**Gambar 1.9 Pilih Aplikasi Xampp star menu**

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



**Gambar 1.10 Menjalankan xampp**

8. Buka browser dan ketikkan "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*, menyisipkan 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*

**Tabel 2.1 Tag Dasar HTML**

Tag	Kegunaan
<code>&lt;!DOCTYPE&gt;</code>	Mendefinisikan tipe Dokumen
<code>&lt;html&gt;</code>	Mendefinisikan HTML dokumen
<code>&lt;head&gt;</code>	Mendfinisikan informasi dokumen
<code>&lt;title&gt;</code>	Mendefinisikan judul dokumen
<code>&lt;body&gt;</code>	Mendefinisikan body dokumen
<code>&lt;h1&gt; to &lt;h6&gt;</code>	Mendefinisikan heading
<code>&lt;p&gt;</code>	Mendefinisikan paragraf
<code>&lt;br&gt;</code>	Mendefinisikan baris

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.</br>
10.
11. </body>
12. </head>
13. </html>

```

Hasilnya:



**Gambar 2.1 Tampilan Contoh 1**

B. Kategori *Form* dan *Input*

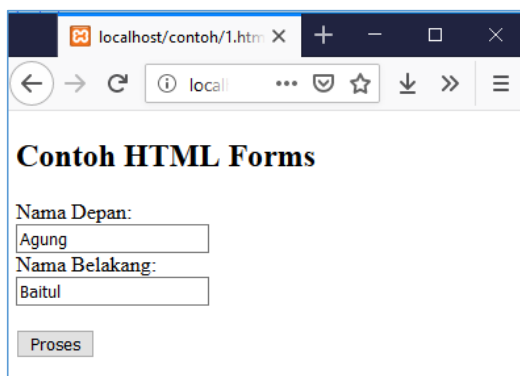
**Tabel 2.2 Tag Form HTML**

Tag	Kegunaan
<form>	Mendefinisikan Form dan Input
<input>	Mendefinisikan input kontrol
<textarea>	Mendefinisikan multiline input kontrol (text area)
<button>	Mendefinisikan button Click
<select>	Mendefinisikan drop-down list
<optgroup>	Mendefinisikan group untuk options drop-down list
<option>	Mendefinisikan option drop-down list
<label>	Mendefinisikan label untuk <input> element
<fieldset>	Group element untuk form
<legend>	Mendefinisikan kalimat untuk <fieldset> element

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><br>
12.  <input type="submit" value="Proses">
13. </form>
14.
15. </body>
16. </html>
```

Hasilnya:



**Gambar 2.2 Tampilan Contoh 2**

### C. Kategori Tabel

**Tabel 2.3 Tag Tabel HTML**

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

Contoh Kode Program:

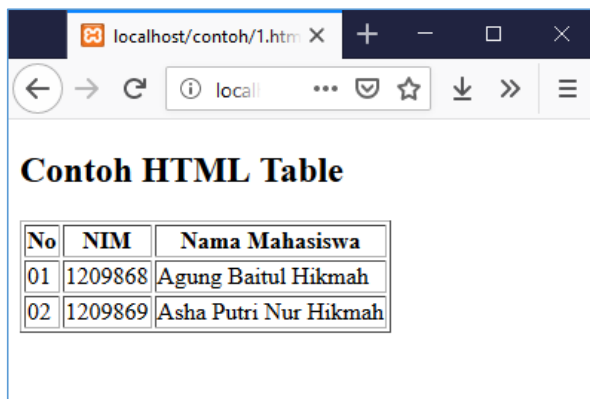
```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. </head>
5. <body>
6. <h2>Contoh HTML Table</h2>
7. <table border = 1>
8.   <tr>
9.     <th>No</th>
10.    <th>NIM</th>
11.    <th>Nama Mahasiswa</th>
12.  </tr>
13.  <tr>
14.    <td>01</td>
15.    <td>1209868</td>
16.    <td>Agung Baitul Hikmah</td>
17. 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

```

22. </tr>
23. </table>
24. </body>
25. </html>

```

Hasilnya:



**Gambar 2.3 Tampilan Contoh 3**

D. Kategori *Image*

**Tabel 2.4 Tag Kategori Image**

Tag	Kegunaan
<img>	Mendefinisikan gambar
<map>	Mendefinisikan peta gambar
<area>	Mendefinisikan area di dalam peta-gambar

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. 
7. </body>
8. </html>

```



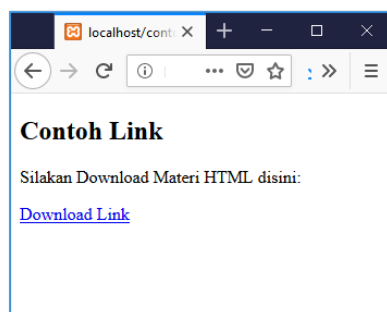
#### E. Kategori *Link*

Tag	Kegunaan
<a>	Mendefinisikan hyperlink
<link>	Mendefinisikan hubungan antara dokumen dan sumber daya eksternal (paling sering digunakan untuk menautkan ke style sheet)

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.
9. </html>
```

Hasilnya:



**Gambar 2.4 Tampilan Contoh 4**

#### F. Kategori *List*

**Tabel 2.5 Tag Kategori List**

Tag	Kegunaan
<ul>	Mendefinisikan unordered list
<ol>	Mendefinisikan ordered list

<li>	Mendefinisikan list item
<dl>	Mendefinisikan description list
<dt>	Mendefinisikan istilah/nama pada description list
<dd>	Mendefinisikan pada deskripsi istilah/nama pada description list

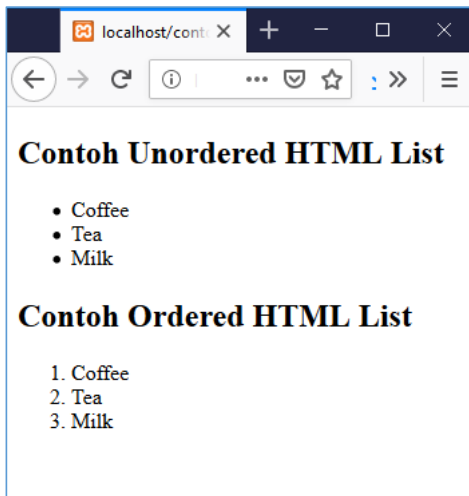
Contoh kode program:

```

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:

#### Data Karyawan PT. Maju Lancar

No	Nama Karyawan	Jenis Kelamin	Alamat Karyawan
01	Salha Suci Nur Hikmah	Perempuan	Jln. Tanuwijaya No.4 Tasikmalaya
02	Siti Rohayati	Perempuan	Jln. Mangkubumi Indah No.8 Ciamis
03	Asep Mulyana	Laki-Laki	Jln. Ciendog Kavling 05 Banjar
04	Silvia Suherman	Perempuan	Jln. Merdeka Barat No.5 Garut Selatan

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. <br><br>
15. <input type="submit">
16. </form>
17.
18. </body>
19. </html>
```

## 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** adalah bahasa pemrograman script server-side yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum ([wikipedia](http://wikipedia)). PHP dikembangkan pada tahun 1995 oleh **Rasmus Lerdorf**, dan sekarang dikelola oleh **The PHP Group**. Situs resmi PHP beralamat di <http://www.php.net>.

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

website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll. Saat ini PHP adalah singkatan dari **PHP: Hypertext Preprocessor**, sebuah kepanjangan *rekursif*, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: **PHP: Hypertext Preprocessor**. PHP dapat digunakan dengan gratis (free) dan bersifat *Open Source*. PHP dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *GNU General Public License (GPL)* yang biasa digunakan untuk proyek *Open Source*. Kemudahan dan kepopuleran **PHP** sudah menjadi standar bagi programmer web di seluruh dunia. Menurut [wikipedia](#) pada februari 2014, sekitar 82% dari web server di dunia menggunakan PHP. PHP juga menjadi dasar dari *aplikasi CMS (Content Management System)* populer seperti *Joomla*, *Drupal*, dan *WordPress*.

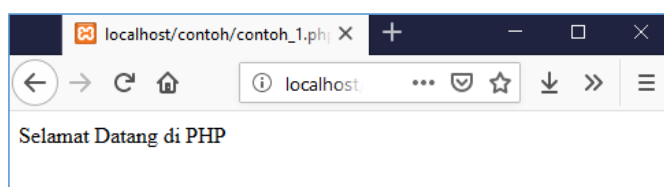
### 3.2. Dasar-dasar PHP

#### A. Penulisan Syntax PHP

Untuk penulisan syntax nya, PHP ditandai dengan membuat tag pembuka `<?php` dan diakhiri dengan dengan tag penutup `?>`. syntax PHP dapat disipkan pada bagian-bagian HTML. Kemudian diakhir setiap baris syntax php harid ditutup dengan tanda titik koma (;) contoh penulisan syntax php yang benar.

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

Outputnya :



**Gambar 3.1 Tampilan Contoh 6**

## B. Fungsi-fungsi Dasar PHP

PHP memiliki beberapa fungsi dasar untuk mencetak teks ke layar:

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

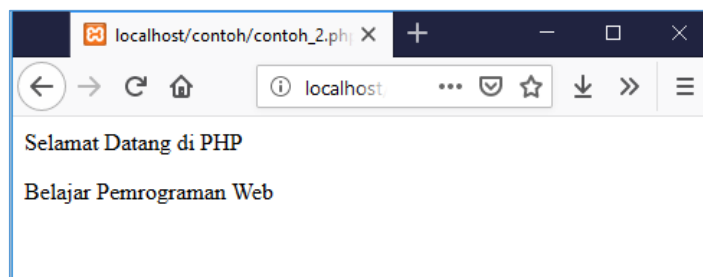
Fungsi-fungsi ini akan sering kita gunakan dalam pemrograman PHP. Perbedaan dari ketiga fungsi tersebut yaitu:

### 1. Fungsi `echo()`

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

```
1. <?php
2. echo "Selamat Datang di PHP";
3. echo "<p>";
4. echo ("Belajar Pemrograman Web");
5. ?>
```

Maka hasilnya :



**Gambar 3.2 Tampilan Contoh 7**

Fungsi `echo()` tidak akan mengembalikan apa-apa setelah dieksekusi. Dia hanya bertugas menampilkan teks saja.

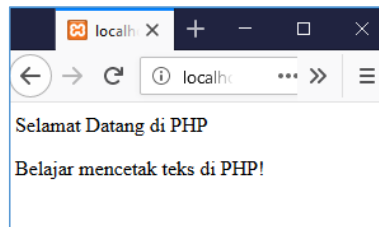
### 2. Fungsi `print()`

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

Contoh :

```
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**

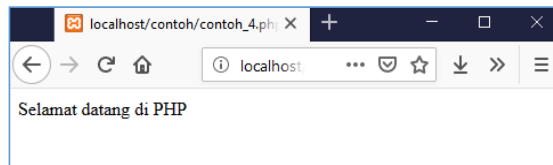
Perbedaannya dengan `echo()`:

- 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 :

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

Maka hasilnya :



**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:

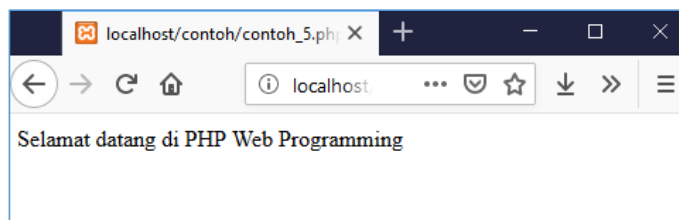


**Gambar 3.5 Tampilan Contoh 10**

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**



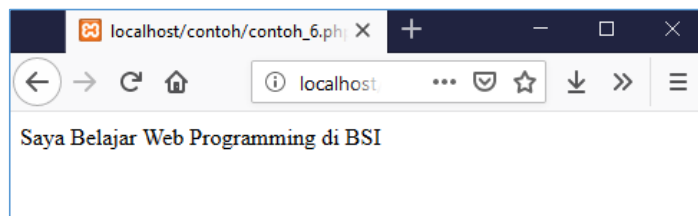
### 3. Fungsi printf()

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

Contoh format string (%s):

```
1. <?php
2. $teks= "BSI";
3. printf("Saya Belajar Web Programming di %s", $teks);
4. ?>
```

Hasilnya :



**Gambar 3.7 Tampilan Contoh 12**

Contoh menggunakan fungsi echo():

```
1. <?php
2. $teks= "BSI";
3. echo "Saya Belajar Web Programming di $teks";
4. ?>
```

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

Untuk membuat halaman web, sebenarnya PHP bukanlah bahasa pemrograman yang wajib digunakan. Kita bisa saja membuat website hanya menggunakan HTML saja. Web yang dihasilkan dengan HTML (dan CSS) ini dikenal dengan website statis, dimana konten dan halaman web bersifat tetap. Sebagai perbandingan, website dinamis yang bisa dibuat menggunakan PHP adalah situs web yang bisa menyesuaikan tampilan konten tergantung situasi. Website dinamis juga bisa menyimpan data ke dalam database, membuat halaman yang berubah-ubah sesuai input dari *user*, memproses form. Untuk pembuatan web, kode **PHP** biasanya di sisipkan kedalam dokumen HTML. Dengan PHP halaman HTML menjadi lebih powerful karena PHP yang menggenerate kode HTML secara dinamis. Karena fitur inilah PHP disebut juga sebagai **Scripting Language** atau bahasa pemrograman **script**.

## 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 :

```
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:

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

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

**Parse error:** syntax error, unexpected '2' (T\_LNUMBER), expecting variable (T\_VARIABLE)

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

```
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:

```
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 menggunakannya 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 variabel tersebut.

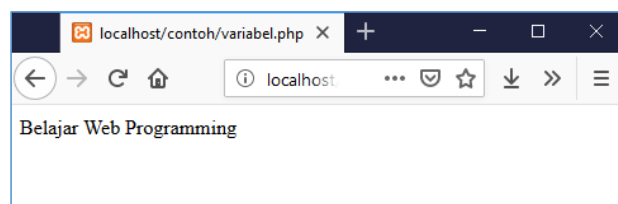
## B. Memberikan Nilai

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 :*

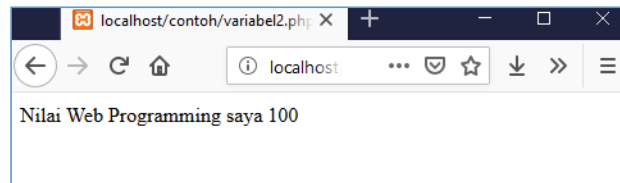


### Gambar 3.8 Tampilan Contoh 13

- Contoh variabel tipe data integer:

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

Hasilnya:

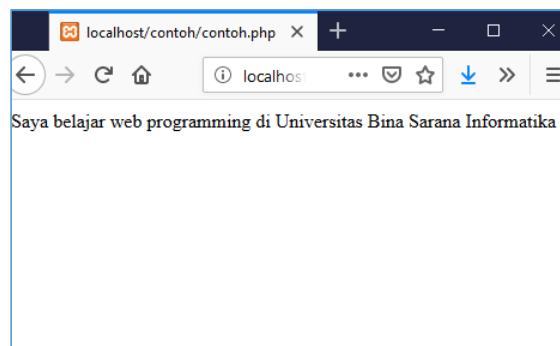


### Gambar 3.9 Tampilan Contoh 14

- Contoh variabel tipe data string

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

Hasilnya:



### Gambar 3.10 Tampilan Contoh 15

#### 5. 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**

<b>Nama Variabel</b>	<b>Keterangan</b>
\$GLOBALS	Variabel tersedia dalam lingkup global
\$_SERVER	Informasi Server dan eksekusi
\$_GET	Varibel HTTP GET
\$_POST	Variabel HTTP POST
\$_FILES	Variabel Upload File HTTP
\$_REQUEST	Variabel Permintaan HTTP
\$_SESSION	Variabel Sesi
\$_ENV	Ruang lingkup Variabel
\$_COOKIE	Cookie HTTP
\$php_errormsg	Pesan Kesalahan sebelumnya
\$HTTP_RAW_POST_DATA	POST data Mentah
\$http_response_header	Header Respon HTTP
\$argc	Jumlah argument yang diteruskan ke script
\$argv	Array argument yang diteruskan ke script

Selain daftar varibel sistem PHP diatas ungkin 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`

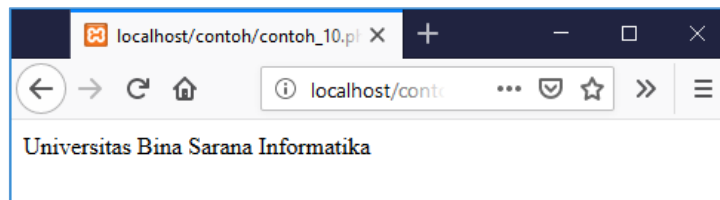
bentuk umum:

`const nama_konstanta = nilai konstanta;`

contoh :

```
1 <?php
2 const kampus ="Universitas Bina Sarana Informatika";
3 echo kampus;
4 ?>
```

Hasilnya:

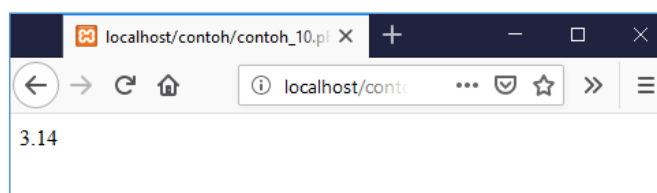


**Gambar 3.11 Tampilan Contoh 16**

Contoh :

```
1 <?php
2 const x = 3.14;
3 echo x;
4 ?>
```

Hasilnya:



**Gambar 3.12 Tampilan Contoh 17**

b. Menggunakan fungsi `define`

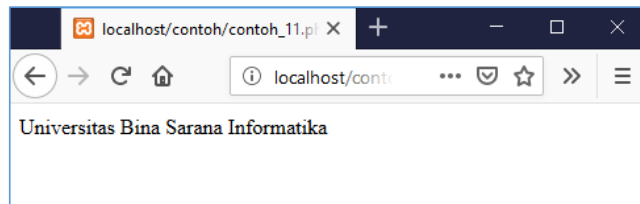
Bentuk umum:

`define (nama_konstanta,nilai konstanta);`

contoh:

```
1 <?php
2 define("kampus","Universitas Bina Sarana Informatika");
3 echo kampus;
4 ?>
```

Hasilnya:

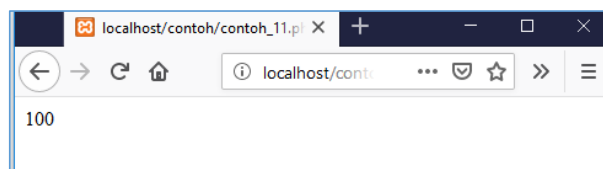


**Gambar 3.13 Tampilan Contoh 18**

Contoh 2:

```
1 <?php
2 define("nilai",100);
3 echo nilai;
4 ?>
```

Hasilnya:



**Gambar 3.14 Tampilan Contoh 19**

### 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. Pemrograman PHP mendukung

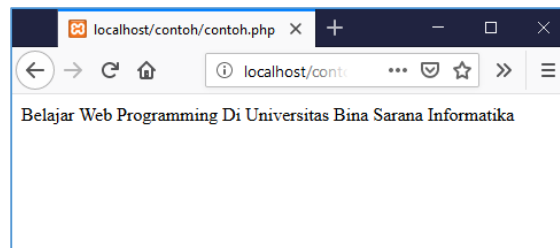


dua tipe dalam penulisan tipe data string yaitu diapit petik tunggal(single quoted) dan diapit petik ganda (double quoted).

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

```
1 <?php
2 $a = 'Belajar Web Programming ';
3 $b = 'Di Universitas Bina Sarana Informatika';
4 echo $a;
5 echo $b;
6 ?>
```

Hasilnya:

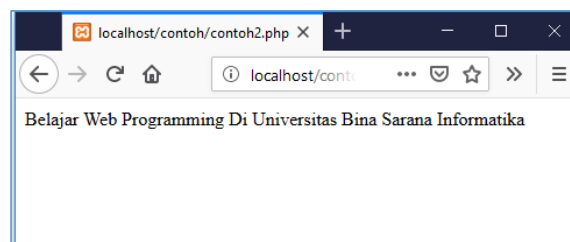


**Gambar 3.15 Tampilan Contoh 20**

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

```
1 <?php
2 $a = "Belajar Web Programming ";
3 $b = "Di Universitas Bina Sarana Informatika";
4 echo $a;
5 echo $b;
6 ?>
```

Hasilnya akan sama akan sama dengan diapit petik tunggal :



**Gambar 3.16 Tampilan Contoh 21**

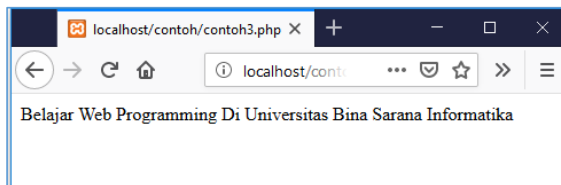
8. Contoh tipe data string diapit petik tunggal dan ganda:

```

1 <?php
2 $a = 'Belajar Web Programming ';
3 $b = "Di Universitas Bina Sarana Informatika";
4 echo $a;
5 echo $b;
6 ?>

```

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**

Cara Penulisan	Karakter Yang Ditampilkan
\"	Karakter tanda petik dua (")
\n	Karakter newline (garis baru)
\r	Karakter carriage return
\t	Karakter Tab
\\	Karkter backslash
\\$	Karakter dollar sign (\$)
\{	Karakter pembuka kurung kurawal ({)
\}	Karakter penutup kurung kurawal (})
\[	Karakter pembuka kurung

	siku ( )
\	Karakter penutup kurung siku ( )
\0 sampai \777	Karakter ASCII menggunakan nilai oktal
\x0 sampai \xFF	Karakter ASCII menggunakan nilai heksadesimal

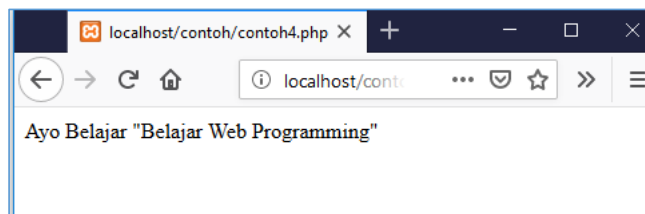
Contoh menampilkan karakter tanda petik dua

```

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

```

Maka hasilnya :



**Gambar 3.18 Tampilan Contoh 23**

#### b. Tipe Data Integer

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

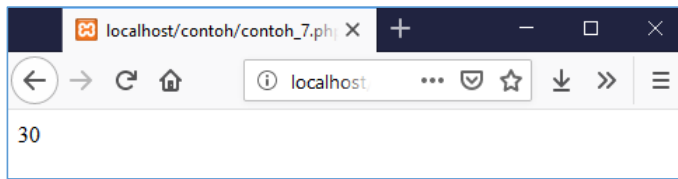
9. Contoh:

```

1 <?php
2 $x= 30;
3 echo $x;
4 ?>

```

Hasilnya:

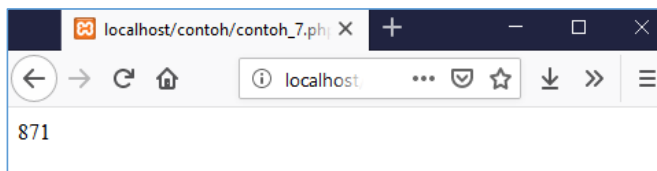


**Gambar 3.19 Tampilan Contoh 24**

10. Contoh penjumlahan

```
1 <?php
2 $x= 20;
3 $y= 851;
4 $hasil = $x + $y;
5 echo $hasil;
6 ?>
```

Maka hasilnya:



**Gambar 3.20 Tampilan Contoh 25**

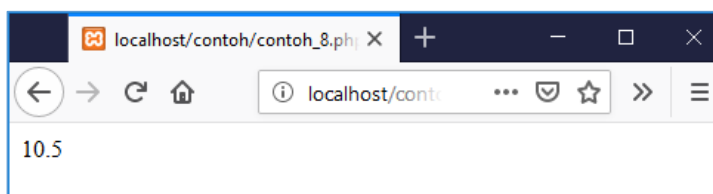
c. Float

Tipe data float yaitu tipe data yang memperpresentasikan 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:

```
1 <?php
2 $x= 10.5;
3 echo $x;
4 ?>
```

Hasilnya:



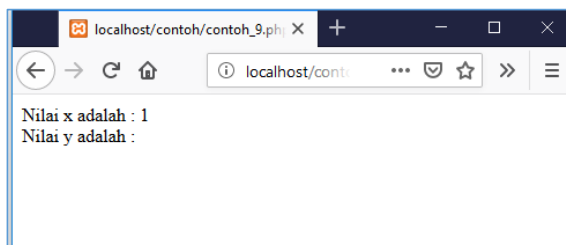
**Gambar 3.21 Tampilan Contoh 26**

d. Boolean

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

```
1 <?php
2 $x= true;
3 $y= false;
4 echo "Nilai x adalah : $x";
5 echo "<br>";
6 echo "Nilai y adalah : $y";
7 ?>
```

Hasilnya:



**Gambar 3.22 Tampilan Contoh 27**

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**

Operator	Keterangan
+	Operator untuk penjumlahan
-	Operator untuk pengurangan
*	Operator untuk perkalian
/	Operator untuk pembagian
%	Operator untuk mengetahui sisa dari pembagian (modulus)

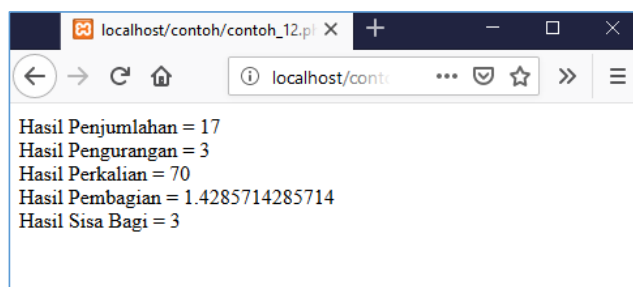
Contoh penggunaan operator:

```

1  <?php
2  $x = 10;
3  $y = 7;
4
5  $penjumlahan = $x + $y;
6  $pengurangan = $x - $y;
7  $perkalian   = $x * $y;
8  $pembagian   = $x / $y;
9  $sisa_bagi   = $x % $y;
10
11 echo "Hasil Penjumlahan = ".$penjumlahan."<br>";
12 echo "Hasil Pengurangan = ".$pengurangan."<br>";
13 echo "Hasil Perkalian   = ".$perkalian."<br>";
14 echo "Hasil Pembagian   = ".$pembagian."<br>";
15 echo "Hasil Sisa Bagi   = ".$sisa_bagi."<br>";
16
17 ?>

```

Hasilnya:



**Gambar 3.23 Tampilan Contoh 28**

#### d. Operator Penugasan (Assignment)

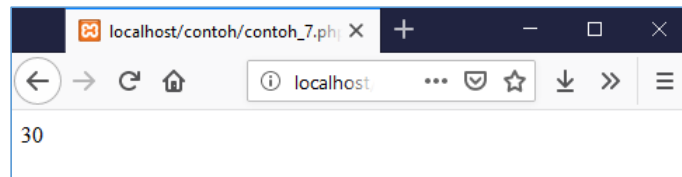
Operator yang digunakan untuk mengisi nilai ke dalam suatu variabel. Operator penugasan dalam PHP yaitu = (sama dengan). Contoh

11.  $x = y$  sama dengan  $y = x$

contoh koding:

```
1 <?php
2 $x= 30;
3 echo $x;
4 ?>
```

Hasilnya:



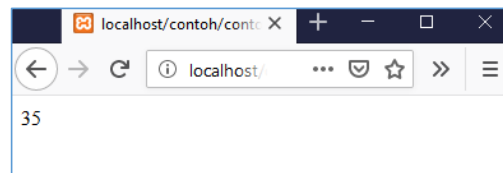
**Gambar 3.24 Tampilan Contoh 29**

12.  $x += y$  sama dengan  $x = x + y$

contoh:

```
1 <?php
2 $x = 5;
3 $x += 30;
4
5 echo $x;
6 ?>
```

Hasilnya:



**Gambar 3.25 Tampilan Contoh 30**

13.  $x -= y$  sama dengan  $x = x - y$

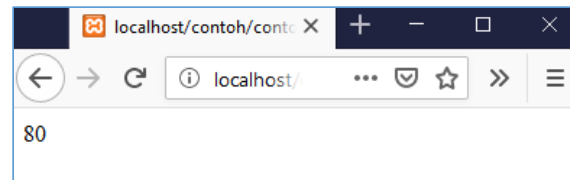
contoh:

```

1  <?php
2  $x = 100;
3  $x -= 20;
4
5  echo $x;
6  ?>

```

Hasilnya:



**Gambar 3.26 Tampilan Contoh 31**

14.  $x *= y$  sama dengan  $x = x * y$

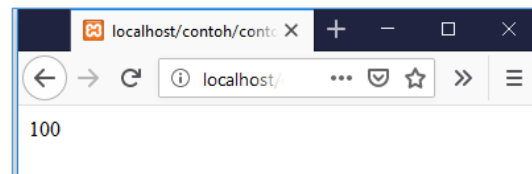
contoh:

```

1  <?php
2  $x = 25;
3  $x *= 4;
4
5  echo $x;
6  ?>

```

Hasilnya:



15.  $x /= y$  sama dengan  $x = x / y$

contoh:

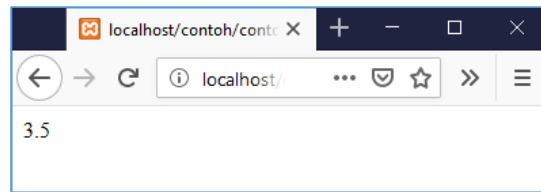
```

1  <?php
2  $x = 7;
3  $x /= 2;
4
5  echo $x;
6  ?>

```

Hasilnya:



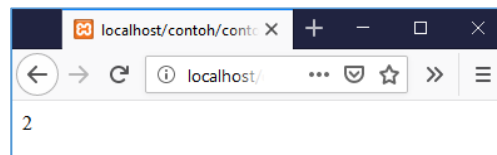


16.  $x \% y$  sama dengan  $x = x \% y$

contoh:

```
1 <?php
2 $x = 20;
3 $x %= 3;
4
5 echo $x;
6 ?>
```

Hasilnya:



#### e. Operator Pembandingan(Relasional)

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

**Tabel 3.4 Daftar Operator Perbandingan**

Operator	Keterangan
$\$x = \$y$	Menghasilkan nilai true jika \$x sama dengan \$y
$\$x === \$y$	Menghasilkan nilai true jika \$x sama dengan \$y, serta \$x dan \$y bertipe sama
$\$x \neq \$y$	Menghasilkan nilai true jika \$x tidak sama dengan \$y
$\$x < > \$y$	Menghasilkan nilai true jika \$x tidak sama

	dengan \$y
$\$x \neq \$y$	Menghasilkan nilai true jika \$x tidak sama dengan \$y, serta \$x dan \$y tidak bertipe sama
$\$x > \$y$	Menghasilkan nilai true jika \$x lebih besar dari \$y
$\$x < \$y$	Menghasilkan nilai true jika \$x lebih kecil dari \$y
$\geq$	Menghasilkan nilai true jika \$x lebih besar atau sama dengan \$y
$\leq$	Menghasilkan nilai true jika \$x lebih kecil atau sama dengan \$y

Contoh penggunaan operator perbandingan

```

1  <?php
2  $x = 20;
3  $y = 10;
4  echo "<pre>";
5  echo "Hasil dari $x == $y = " . ($x==$y) . "<br>";
6  echo "Hasil dari $x === $y = " . ($x=== $y) . "<br>";
7  echo "Hasil dari $x != $y = " . ($x!= $y) . "<br>";
8  echo "Hasil dari $x <> $y = " . ($x<>$y) . "<br>";
9  echo "Hasil dari $x !== $y = " . ($x!== $y) . "<br>";
10 echo "Hasil dari $x > $y = " . ($x > $y) . "<br>";
11 echo "Hasil dari $x < $y = " . ($x < $y) . "<br>";
12 echo "Hasil dari $x >= $y = " . ($x >= $y) . "<br>";
13 echo "Hasil dari $x <= $y = " . ($x <= $y) . "<br>";
14
15 ?>

```

Hasilnya:

```

Hasil dari 20 == 10 =
Hasil dari 20 === 10 =
Hasil dari 20 != 10 =1
Hasil dari 20 <> 10 =1
Hasil dari 20 !== 10 =1
Hasil dari 20 > 10 =1
Hasil dari 20 < 10 =
Hasil dari 20 >= 10 =1
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

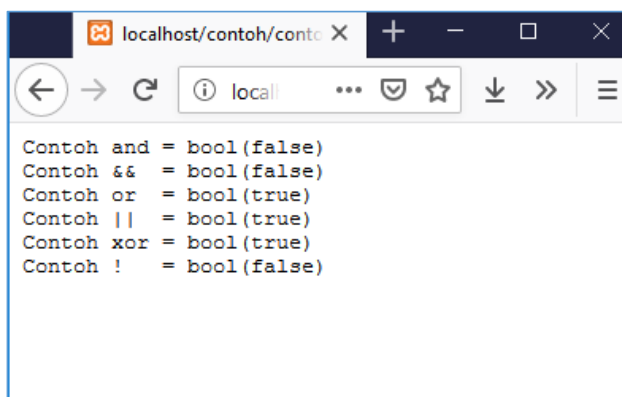
**Tabel 3.5 Daftar Operator Logika PHP**

<b>Nama Operator</b>	<b>Contoh</b>	<b>Keterangan</b>
AND	<code>\$x &amp;&amp; \$y</code>	Logika AND Menghasilkan TRUE jika \$x dan \$y bernilai TRUE
AND	<code>\$x and \$y</code>	Logika AND Menghasilkan TRUE jika \$x dan \$y bernilai TRUE
Or	<code>\$x or \$y</code>	Logika or Menghasilkan TRUE jika salah satu \$x atau \$y bernilai TRUE
<code>  </code>	<code>\$x    \$y</code>	Logika or Menghasilkan TRUE jika salah satu \$x atau \$y bernilai TRUE
xor	<code>\$x xor \$y</code>	Logika or Menghasilkan TRUE jika salah satu \$x atau \$y bernilai TRUE tetapi bukan keduanya
!	<code>!\$x</code>	Menghasilkan TRUE jika \$x = False

Contoh penggunaan :

```
1 <?php
2 $x = true;
3 $y = false;
4
5 echo"<pre>";
6 echo"Contoh and = "; var_dump ($x and $y );
7 echo"Contoh && = "; var_dump ($x && $y );
8 echo"Contoh or = "; var_dump ($x or $y );
9 echo"Contoh || = "; var_dump ($x || $y );
10 echo"Contoh xor = "; var_dump ($x xor $y );
11 echo"Contoh ! = "; var_dump (!$x);
12
13 ?>
```

Hasilnya :



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

Operator	Nama	Keterangan
++\$x	Pre Increment	
\$++	Post increment	
--\$x	Pre decrement	

\$x--	Post decrement	
-------	-------------------	--

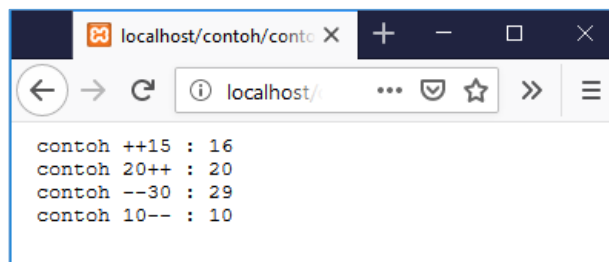
Contoh:

```

1  <?php
2  $a = 15;
3  $b = 20;
4  $c = 30;
5  $d = 10;
6  echo "<pre>";
7  echo " contoh ++$a : ".$a."<br>";
8  echo " contoh $b++ : ".$b."<br>";
9  echo " contoh --$c : ".$c."<br>";
10 echo " contoh $d-- : ".$d."<br>";
11
12 ?>

```

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:

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

Bentuk umum:

```

If (kondisi) {
    pernyataan;
}

```

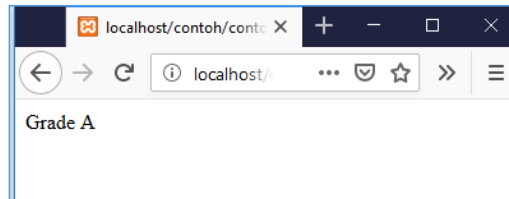
Contoh :

```

1 <?php
2 $nilai=90;
3
4 if($nilai>=80){
5     echo "Grade A";
6 }
7 ?>

```

Hasilnya:



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

Bentuk umum

```

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

```

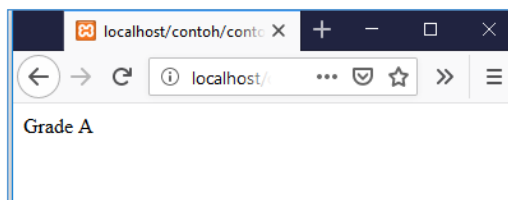
Contoh:

```

1 <?php
2 $nilai=90;
3
4 if($nilai>=80){
5     echo "Grade A";
6 }else{
7     echo "Grade B";
8 }
9 ?>

```

Hasilnya:



- c. Pernyataan if...elseif yaitu mengeksekusi kondisi yang berbeda dari dua kondisi atau lebih.

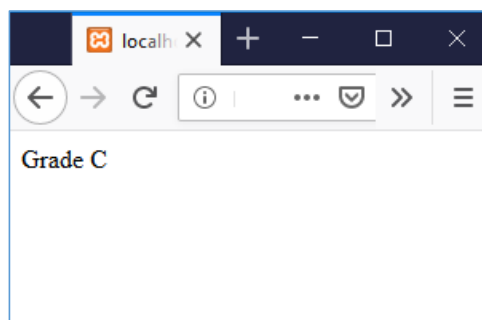
Bentuk umum:

```
if (kondisi1) {  
    pernyataan1;  
}elseif (kondisi2){  
    pernyataan2;  
}elseif (kondisi3){  
    pernyataan3;  
}else {  
    pernyataan4;  
}
```

Contoh kode program:

```
1 <?php  
2 $nilai=72;  
3  
4 if($nilai>=80){  
5     echo "Grade A";  
6 }elseif ($nilai>=75){  
7     echo "Grade B";  
8 }elseif ($nilai>=65){  
9     echo "Grade C";  
10 }elseif ($nilai>=45){  
11     echo "Grade D";  
12 }else {  
13     echo "Grade E";  
14 }  
15 ?>
```

Hasilnya:



- 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:

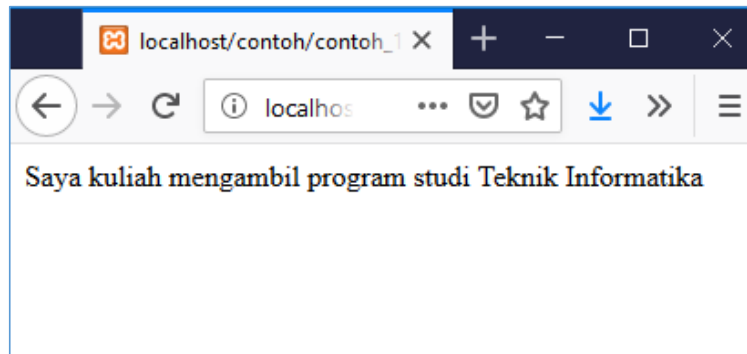
```
switch (variabel) {  
    case nilai1:  
        pernyataan1;  
        break;  
    case nilai2:  
        pernyataan2;  
        break;  
    case nilai3:  
        pernyataan3;  
        break;  
    .....  
    default:  
        pernyataan4;  
}
```

Contoh kode program:

```
1  <?php  
2  $prodi="Teknik Informatika";  
3  
4  switch ($prodi){  
5      case "Sistem Informasi":  
6          echo "Saya kuliah mengambil program studi Sistem Informasi";  
7          break;  
8      case "Teknik Informatika":  
9          echo "Saya kuliah mengambil program studi Teknik Informatika";  
10         break;  
11     case "Teknik Komputer":  
12         echo "Saya kuliah mengambil program studi Teknik Komputer";  
13         break;  
14     default:  
15         echo "Saya kuliah mengambil program studi Teknik Informatika";  
16     }  
17  ?>
```



Hasilnya:



### 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

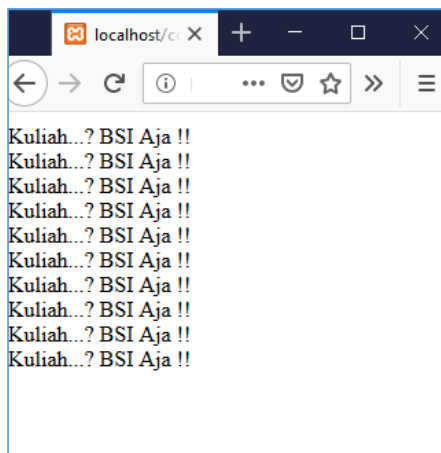
Bentuk umum:

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

Contoh kode:

```
1 <?php
2 for ($x=1; $x<=10; $x++)
3 {
4     echo"Kuliah...? BSI Aja !!";
5     echo"<br>";
6 }
7
8 ?>
```

Hasilnya:



b. Perulangan while

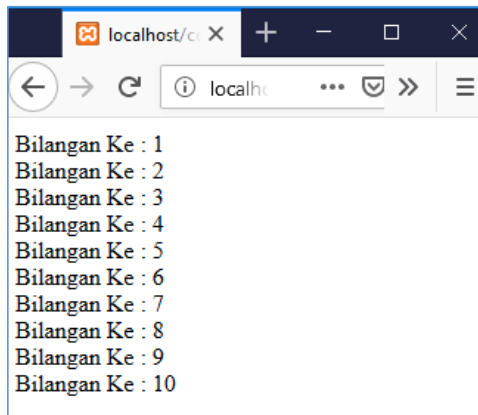
Bentuk umum:

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

Contoh kode:

```
1  <?php
2  $x=1;
3  while ($x <= 10)
4  {
5      echo"Bilangan Ke : $x";
6      echo "<br>";
7      $x++;
8  }
9
10 ?>
```

Hasilnya:



c. Perulangan do while

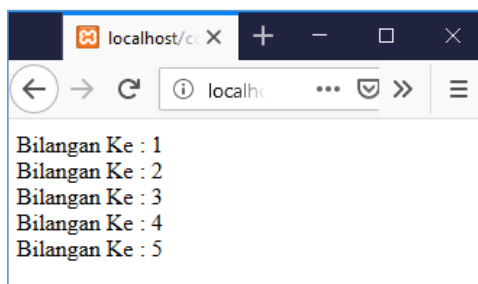
Bentuk umum:

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

Contoh kode:

```
1  <?php
2  $x=1;
3  do{
4      echo"Bilangan Ke : $x";
5      echo "<br>";
6      $x++;
7  }while ($x<=5);
8
9  ?>
```

Hasilnya:



d. Perulangan foreach

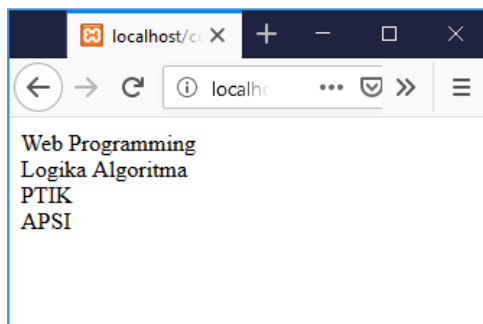
Bentuk umum:

```
foreach (array as $value{  
    pernyataan;  
}
```

Contoh kode:

```
1 <?php  
2  
3 $matkul = array("Web Programming", "Logika Algoritma", "PTIK", "APSI");  
4 foreach ($matkul as $value){  
5     echo "$value <br>";  
6 }  
7  
8 ?>
```

Hasilnya:



### 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:

```
$nama_variabel = array (  
    Key1 => value1  
    Key2 => value2  
    Key3 => value3
```

```
Key4 => value4
.....
)
```

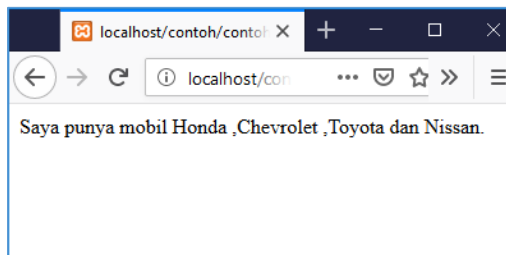
Dalam Pemrograman PHP ada 3 jenis Array yaitu:

1. Array berindex yaitu array dengan index numerik atau diindex berdasarkan angka pada umumnya dimulai dari angka 0.

Contoh:

```
1 <?php
2
3 $mobil = array("Honda","Chevrolet","Toyota","Nissan");
4 echo "Saya punya mobil ".$mobil[0]." ,".$mobil[1]." ,".$mobil[2]." dan ".$mobil[3].".";
5
6 ?>
```

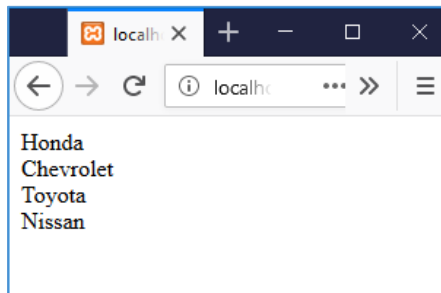
Hasilnya:



Contoh array berindex menggunakan perulangan for:

```
1 <?php
2
3 $mobil = array("Honda","Chevrolet","Toyota","Nissan");
4 $jumlah = count($mobil);
5
6 for ($x=0; $x<$jumlah; $x++)
7 {
8     echo $mobil[$x];
9     echo "<br>";
10 }
11
12 ?>
```

Hasilnya:



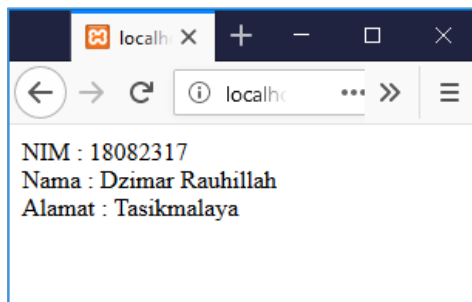
## 2. Array Asosiatif

Yaitu array yang diindex berdasarkan kunci tertentu.

Contoh:

```
1 <?php
2
3 $mahasiswa = array(
4     "nim"=>"18082317",
5     "nama"=>"Dzimar Rauhillah",
6     "alamat"=>"Tasikmalaya"
7 );
8 echo "NIM      : $mahasiswa[nim]<br>";
9 echo "Nama     : $mahasiswa[nama]<br>";
10 echo "Alamat  : $mahasiswa[alamat]<br>";
11
12 ?>
```

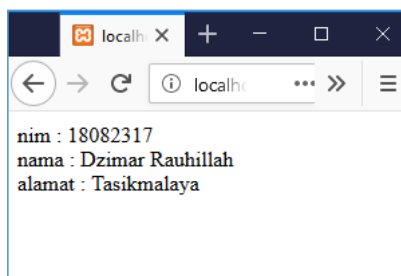
Hasilnya:



Contoh array berindex menggunakan perulangan foreach:

```
1 <?php
2
3 $mahasiswa = array(
4     "nim"=>"18082317",
5     "nama"=>"Dzimar Rauhillah",
6     "alamat"=>"Tasikmalaya"
7 );
8 foreach ($mahasiswa as $x => $y)
9 {
10     echo "$x : $y ";
11     echo "<br>";
12 }
13
14 ?>
```

Hasilnya:



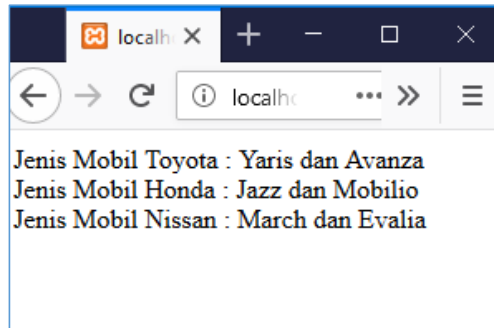
### 3. Array multidimensi

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:

```
1. <?php
2.
3. $mobil= array(
4.     array("Toyota","Yaris","Avanza"),
5.     array("Honda","Jazz","Mobilio"),
6.     array("Nissan","March","Evalia")
7. );
8. echo "Jenis Mobil ".$mobil[0][0]. " : ".$mobil[0][1]. " dan ".$mobil[0][2]. "<br>";
9. echo "Jenis Mobil ".$mobil[1][0]. " : ".$mobil[1][1]. " dan ".$mobil[1][2]. "<br>";
10. echo "Jenis Mobil ".$mobil[2][0]. " : ".$mobil[2][1]. " dan ".$mobil[2][2]. "<br>";
11. ?>
```

Hasilnya:





## **BAB 4**

### **PENGENALAN JAVA SCRIPT**

#### **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**

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. Client Side Programming Language adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome dan Mozilla Firefox. Java Script digunakan untuk membuat interaksi atau menambah fitur web dinamis kedalam sebuah web. Untuk membuat halaman web, tidak harus menggunakan *JavaScript*. Namun, saat ini *JavaScript* hadir dalam hampir setiap halaman web modern. Sehingga *JavaScript* adalah salah satu bahasa pemrograman web yang wajib di kuasai. Java Script adalah salah satu dari 3 bahasa yang harus di kuasai oleh pengembang web yaitu HTML untuk menentukan konten web, CSS untuk mengatur tampilan halaman web, dan CSS untuk interaksi (behavior). Java Script bukan hanya digunakan untuk membuat halaman web tetapi juga digunakan dalam program aplikasi desktop dan server seperti Node.js adalah yang paling dikenal. Beberapa database, seperti MongoDB dan CouchDB, juga menggunakan JavaScript sebagai bahasa pemrogramannya.

JavaScript adalah bahasa yang sama sekali berbeda, baik dalam konsep maupun desain. JavaScript ditemukan oleh Brendan Eich pada tahun 1995, dan menjadi standar ECMA pada tahun 1997. ECMA-262 adalah nama resmi standar. ECMAScript adalah nama resmi bahasa tersebut. Dalam mempelajari *JavaScript*, sebaiknya telah menguasai dasar-dasar HTML, dan bisa membuat halaman web sederhana menggunakan HTML. Pengetahuan tentang CSS dan PHP juga akan membantu, walaupun tidak diharuskan.

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 pemrograman web lainnya seperti PHP dan CSS, yakni dengan menyisipkan 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:

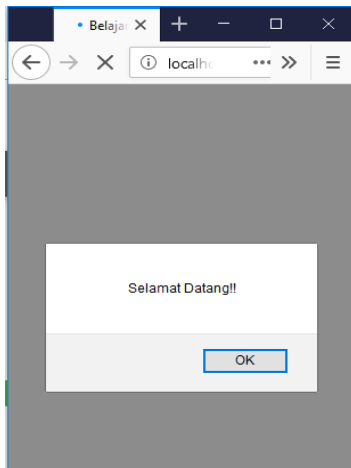
1. Diawal HTML yaitu tag <script> diletakkan didalam tag <head>

Contoh:

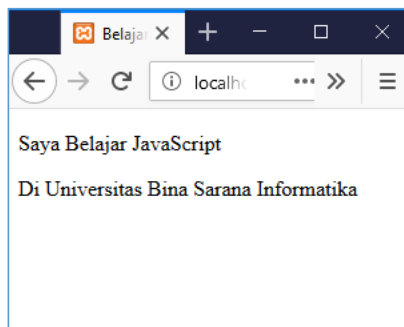
```
1. <!DOCTYPE html>
2. <html>
3. <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
4. <head>
5. <title>Belajar Tag JavaScript</title>
6.
7. <script>
8.   alert("Selamat Datang!!");
9. </script>
10. </head>
11. <body>
```

```
12. <p>Saya Belajar JavaScript Di</p>
13. <p>Universitas Bina Sarana Informatika</p>
14.
15. </body>
16. </html>
```

Hasilnya ketika pertama kali dijalankan:



Hasilnya ketika diklik ok:



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

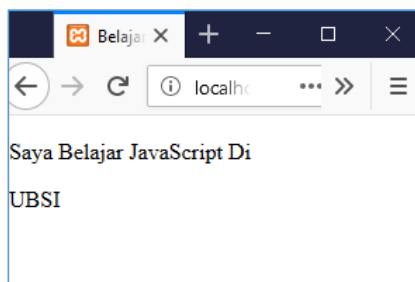
2. Ditengah HTML yaitu tag <script> diletakkan didalam tag <body>  
Contoh:

```

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

```

Hasilnya:



Pada tag <p ="test"></p> untuk menampilkan hasil dari javascript

3. Diakhir HTML yaitu tag <script> diletakkan sebelum tag akhir </html>

Contoh:

```

1. <!DOCTYPE html>
2. <html>
3. <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
4. <head>
5. <title>Belajar Tag JavaScript</title>
6.
7. </head>
8. <body>
9. <p>Saya Belajar JavaScript Di</p>
10. <div id="tes"></div>
11.

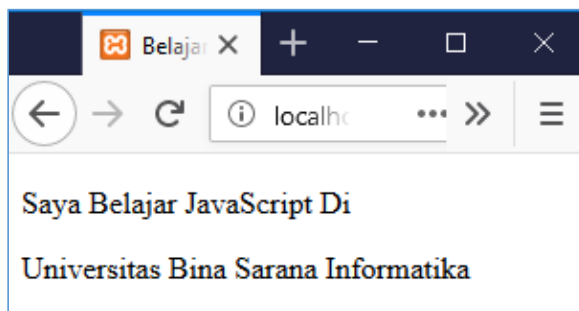
```

```

12. </body>
13.
14. <script>
15.   var a=document.getElementById("tes");
16.   a.innerHTML="<p>Universitas Bina Sarana Informatika</p>";
17. </script>
18. </html>

```

Hasilnya:



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 lainnya 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:

```

1. <!DOCTYPE html>
2. <html>
3. <body>

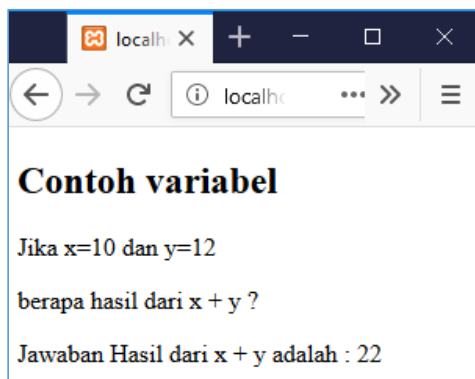
```

```

4.
5. <h2>Contoh variabel</h2>
6.
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:



Selanjutnya contoh pendeklarasian variabel dengan tipe string:

```

1. <!DOCTYPE html>
2. <html>
3. <body>
4.
5. <h2>Contoh variabel</h2>

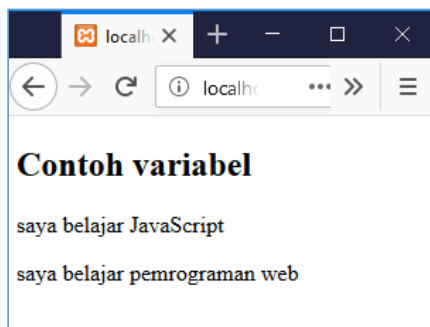
```

```

6.
7. <p id="tampil1"></p>
8. <p id="tampil2"></p>
9.
10. <script>
11. var teks1, teks2;
12. teks1 ="saya belajar JavaScript";
13. teks2= "saya belajar pemrograman web";
14. document.getElementById("tampil1").innerHTML = teks1;
15. document.getElementById("tampil2").innerHTML = teks2;
16. </script>
17.
18. </body>
19. </html>

```

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:

(parameter1, parameter2, ...)

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:

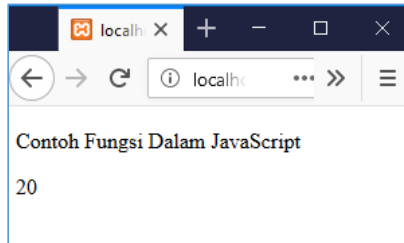
```
1. <!DOCTYPE html>  
2. <html>  
3. <body>  
4.  
5. <p>Contoh Fungsi Dalam JavaScript</p>  
6.  
7. <p id="demo"></p>  
8.  
9. <script>  
10. var x = Perkalian(10, 2);  
11. document.getElementById("demo").innerHTML = x;  
12.  
13. function Perkalian(y, z) {  
14.     return y * z;  
15. }  
16. </script>  
17.  
18. </body>  
19. </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":

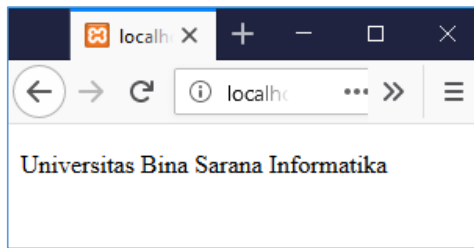
maka hasilnya seperti dibawah ini:



Contoh penggunaan fungsi dengan variabel lokal:

```
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 pemberian 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:

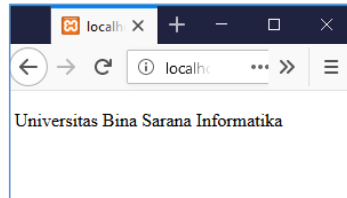
```
if (kondisi) {  
    kode dieksekusi jika kondisinya benar;  
}
```

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

Contoh kode program:

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

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

contoh kode program:

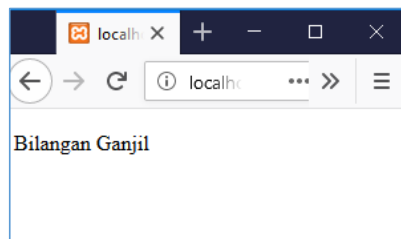
```
1. <!DOCTYPE html>  
2. <html>  
3. <body>  
4.  
5. <p id="demo"></p>  
6.  
7. <script>  
8.   var angka =13;  
9.   if (angka%2== 0)  
10.  {  
11.    document.getElementById("demo").innerHTML =  
12.    "Bilangan Genap";  
13.  }else {  
14.    document.getElementById("demo").innerHTML =  
15.    "Bilangan Ganjil";  
16.  }
```

```

17. </script>
18.
19. </body>
20. </html>

```

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:

```

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. <!DOCTYPE html>
2. <html>

```

```

3. <body>
4.
5. <p id="demo"></p>
6.
7. <script>
8.   var nilai =70;
9.   if (nilai>=80)
10.  {
11.    document.getElementById("demo").innerHTML =
12.    "Grade A";
13.  }else if(nilai>=70){
14.    document.getElementById("demo").innerHTML =
15.    "Grade B";
16.  }else{
17.    document.getElementById("demo").innerHTML =
18.    "Grade C";
19.  }
20. </script>
21.
22. /body>
23. </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:

```

switch(ekspresi) {
  case x:
    Kode blok program yang akan dieksekusi
    break;
  case y:
    Kode blok program yang akan dieksekusi
    break;
  default:

```

*Kode blok program yang akan dieksekusi*

}

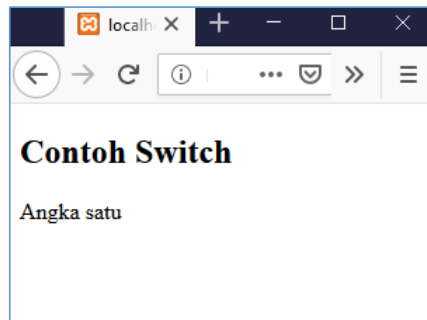
cara kerjanya:

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:

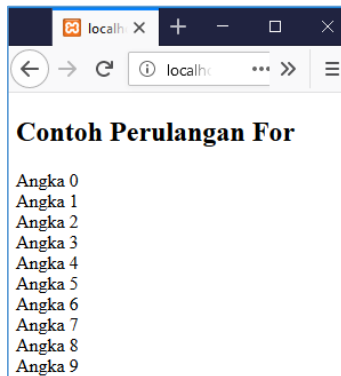
```
for (inisialisasi ekspresi; kondisi perulangan; ekspresi penambahan) {  
    // pernyataan yang akan dieksekusi  
}
```

contoh kode program:

```
1. <!DOCTYPE html>  
2. <html>  
3. <body>  
4.  
5. <h2>Contoh Perulangan For</h2>  
6.  
7. <p id="demo"></p>  
8.  
9. <script>  
10. var text = "";  
11. var i;  
12. for (i = 0; i < 10; i++) {  
13.     text += "Angka " + i + "<br>";  
14. }  
15. document.getElementById("demo").innerHTML = text;  
16. </script>  
17.
```

```
18. </body>
19. </html>
```

Hasil outputnya:



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:

```
1. <!DOCTYPE html>
2. <html>
3. <body>
4.
5. <h2>Contoh Perulangan For In</h2>
6.
7.
8. <p id="demo"></p>
9.
10. <script>
11. var txt = "";
12. var mahasiswa = {Nama:"Dzimar Rauhillah", Alamat:"Tasikmalaya", Usia:17};
13. var x;
14. for (x in mahasiswa) {
15.   txt += mahasiswa[x] + " ";
16. }
17. document.getElementById("demo").innerHTML = txt;
18. </script>
```



```
19.  
20. </body>  
21. </html>
```

Hasil outputnya:



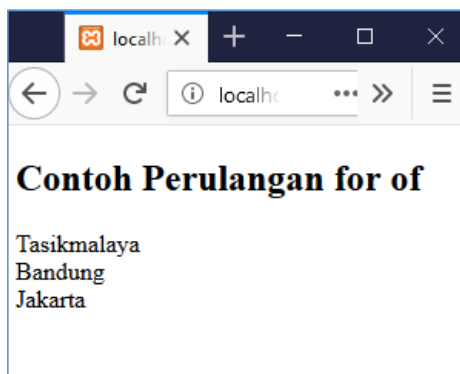
c. Perulangan for of

Perulangan for of digunakan melalui nilai objek yang dapat diubah

Contoh kode program:

```
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:



d. Perulangan while

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

Bentuk umum:

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

contoh kode program:

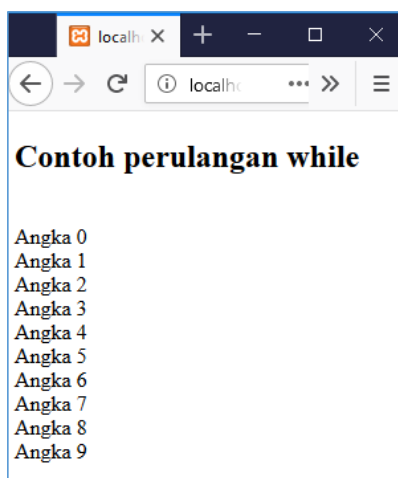
```
1. <!DOCTYPE html>  
2. <html>  
3. <body>  
4.  
5. <h2>Contoh perulangan while</h2>  
6.  
7. <p id="demo"></p>  
8.  
9. <script>  
10. var text = "";
```

```

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:



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

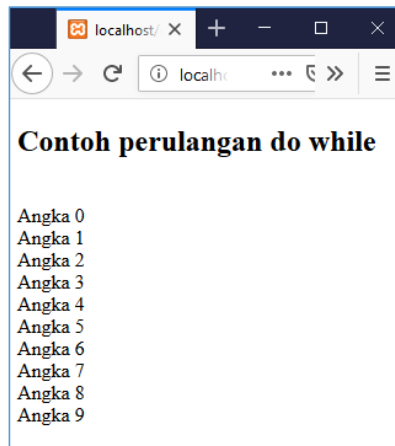
Perulangan do while yaitu **perulangan** yang mengeksekusi pernyataan tertentu hingga kondisi pemeriksaan pernyataan tersebut bernilai false . Kondisi akan diperiksa setelah pernyataan dieksekusi/dijalankan dengan hasil pernyataan tersebut yang dijalankan sekurangnya sekali. sedangkan while akan mengecek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

Bentuk umum:

```
do {  
    // pernyataan yang akan dieksekusi  
}  
while (kondisi);  
contoh kode program
```

```
1. <!DOCTYPE html>  
2. <html>  
3. <body>  
4.  
5. <h2>Contoh perulangan do while</h2>  
6.  
7. <p id="demo"></p>  
8.  
9. <script>  
10. var text = ""  
11. var i = 0;  
12.  
13. do {  
14.   text += "<br>Angka " + i;  
15.   i++;  
16. }  
17. while (i < 10);  
18.  
19. document.getElementById("demo").innerHTML = text;  
20. </script>  
21.  
22. </body>  
23. </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:

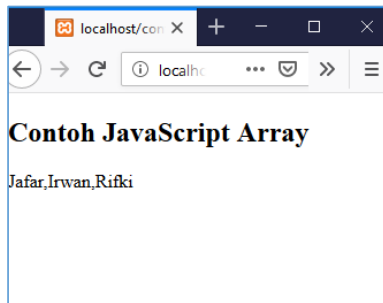
```
var nama_array = [item1, item2, ...];
```

Contoh kode program:

```
1. <!DOCTYPE html>
2. <html>
3. <body>
4.
5. <h2>Contoh JavaScript Array</h2>
6.
7. <p id="demo"></p>
8.
9. <script>
10. var mahasiswa = ["Jafar", "Irwan", "Rifki"];
11. document.getElementById("demo").innerHTML = mahasiswa;
12. </script>
13.
14. </body>
```

15. `</html>`

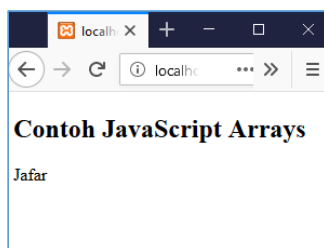
Hasil outputnya :



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

```
1. <!DOCTYPE html>
2. <html>
3. <body>
4.
5. <h2>Contoh JavaScript Arrays</h2>
6.
7.
8. <p id="demo"></p>
9.
10. <script>
11. var mahasiswa = ["Jafar", "Irwan", "Rifki"];
12. document.getElementById("demo").innerHTML = mahasiswa[0];
13. </script>
14.
15. </body>
16. </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:

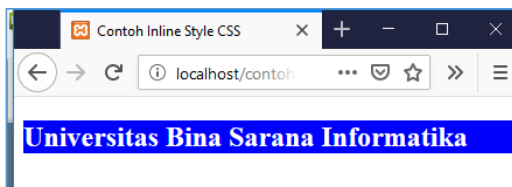
## A. Inline Style

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" >
8.     Universitas Bina Sarana Informatika
9.   </h2>
10. </body>
11. </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:



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:

```
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.
13. <p>Selamat Datang Di</p>
14. <p>Universitas Bina Sarana Informatika</p>
15.
16. </body>
17. </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

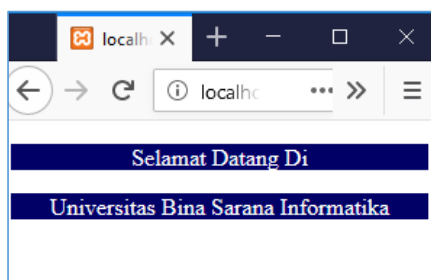
```
1. p {  
2.   color: #FFFFFF;  
3.   text-align: center;  
4.   background-color:#000066;  
5. }
```

Kemudian membuat file html nya dengan nama contoh1.html:

```
1. <!DOCTYPE html>  
2. <html>  
3. <head>  
4. <link rel="stylesheet" type="text/css" href="style.css">  
5. </head>  
6. <body>  
7.  
8. <p>Selamat Datang Di</p>  
9. <p>Universitas Bina Sarana Informatika</p>  
10.  
11. </body>  
12. </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:



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:

```
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.

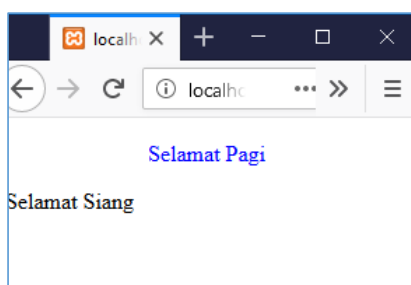
Contoh penulisan nya:

```
1. #baris1 {  
2.   text-align: center;  
3.   color: blue;  
4. }
```

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 :



## 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.

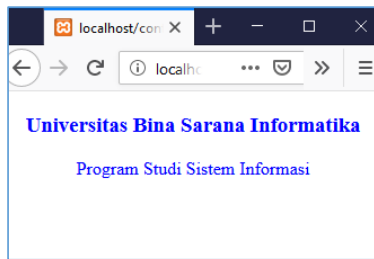
Contoh penulisannya:

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

Contoh penerapan dalam program:

```
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:

```
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:

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

Contoh penerapannya:

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



```

5. h1, h2, p {
6.   text-align: center;
7.   color: blue;
8. }
9. </style>
10. </head>
11. <body>
12.
13. <h1>Universitas Bina Sarana Informatika</h1>
14. <h2>Fakultas Teknologi Informasi</h2>
15. <p>Program Studi Sistem Informasi</p>
16.
17. </body>
18. </html>

```

Maka tampilannya seperti dibawah ini semuanya sama tampil posisinya ditengah:



## 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:

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

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

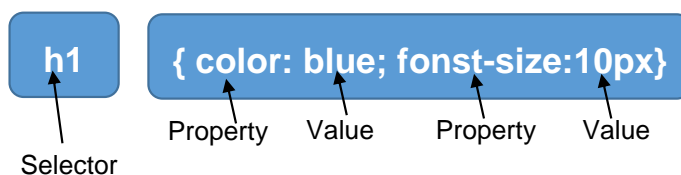
### C. Value

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.

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

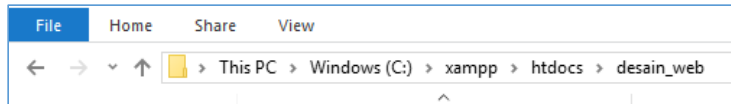
Dari kode diatas kata “center” dan “blue” disebut dengan value

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. * {
4.   box-sizing: border-box;
5. }
6.
7. body {
8.   font-family: Arial;
9.   padding: 10px;
10.  background: #f1f1f1;
11. }
12.
13. input[type=text], select, textarea {
14.   width: 100%;
15.   padding: 12px;
16.   border: 1px solid #ccc;
17.   border-radius: 4px;
18.   resize: vertical;
19. }
20.
21. label {
22.   padding: 12px 12px 12px 0;
23.   display: inline-block;
24. }
25.
```

```
26. input[type=submit] {
27.   background-color: #4CAF50;
28.   color: white;
29.   padding: 12px 20px;
30.   border: none;
31.   border-radius: 4px;
32.   cursor: pointer;
33.   float: right;
34. }
35.
36. input[type=submit]:hover {
37.   background-color: #45a049;
38. }
39.
40. .container {
41.   border-radius: 5px;
42.   background-color: #f2f2f2;
43.   padding: 20px;
44. }
45.
46. .col-25 {
47.   float: left;
48.   width: 25%;
49.   margin-top: 6px;
50. }
51.
52. .col-75 {
53.   float: left;
54.   width: 75%;
55.   margin-top: 6px;
56. }
57. .header {
58.   padding: 30px;
59.   text-align: center;
60.   background: white;
61. }
62.
63. .header h1 {
64.   font-size: 50px;
65. }
66.
67.
68. .topnav {
```

```

69.  overflow: hidden;
70.  background-color: #333;
71. }
72.
73.
74. .topnav a {
75.  float: left;
76.  display: block;
77.  color: #f2f2f2;
78.  text-align: center;
79.  padding: 14px 16px;
80.  text-decoration: none;
81. }
82.
83.
84. .topnav a:hover {
85.  background-color: #ddd;
86.  color: black;
87. }
88.
89.
90. .leftcolumn {
91.  float: left;
92.  width: 75%;
93. }
94.
95.
96. .rightcolumn {
97.  float: left;
98.  width: 25%;
99.  background-color: #f1f1f1;
100.  padding-left: 20px;
101. }
102.
103.
104. .fakeimg {
105.  background-color: #aaa;
106.  width: 100%;
107.  padding: 20px;
108. }
109.
110.
111. .card {

```

```

112.     background-color: white;
113.     padding: 20px;
114.     margin-top: 20px;
115. }
116.
117.
118. .row:after {
119.     content: "";
120.     display: table;
121.     clear: both;
122. }
123.
124.
125. .footer {
126.     padding: 20px;
127.     text-align: center;
128.     background: #ddd;
129.     margin-top: 20px;
130. }
131.
132. @media screen and (max-width: 800px) {
133.     .leftcolumn, .rightcolumn {
134.         width: 100%;
135.         padding: 0;
136.     }
137. }
138.
139.
140. @media screen and (max-width: 400px) {
141.     .topnav a {
142.         float: none;
143.         width: 100%;
144.     }
145. }

```

Jangan lupa simpan di folder **htdocs/desain\_web** yang sudah dibuat sebelumnya

## B. Desain halaman header

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

```

1. <div class="header">

```

2. `<h2>UNIVERSITAS BINA SARANA INFORMATIKA</h2>`
3. `<p>Program Studi Sistem Informasi</p>`
4. `</div>`

### C. Desain Halaman Menu

Membuat halaman menu buat file dengan nama **menu.php**. kode nya seperti dibawah ini:

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:

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:

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>`

```

10. <p>Facebook</p>
11. </div>
12. </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:

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title>Desain Website</title>
5. <link rel="stylesheet" type="text/css" href="style.css">
6. </head>
7. <body>
8.
9. <?php
10. include ('header.php');
11. include ('menu.php');
12. ?>
13.
14. <div class="row">
15. <div class="leftcolumn">
16. <div class="card">
17. <h2>Selamat Datang</h2>
18. <p>Ini adalah halaman beranda website silahkan pilih menu untuk mengakses halaman web</p>
19. <div class="fakeimg" style="height:200px;">Gambar</div>
20. <p>deskripsi gambar nya</p>
21.
22. </div>
23. </div>
24. <?php
25. include ('konten_kanan.php');
26. ?>
27.
28. </div>
29.
30. <?php
31. include ('footer.php');

```



```

32. ?>
33.
34. </body>
35. </html>

```

## G. Desain Halaman Profil

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

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title>Profil Website Layout</title>
5. <link rel="stylesheet" type="text/css" href="style.css">
6. </head>
7. <body>
8.
9. <?php
10. include ('header.php');
11. include ('menu.php');
12. ?>
13.
14. <div class="row">
15. <div class="leftcolumn">
16. <div class="card">
17. <h2>Profil Website</h2>
18. <h5>Tasikmalaya, 17 Agustus 2019</h5>
19. <div class="fakeimg" style="height:200px;">Image</div>
20. <p>ini adalah deskripsi gambar nya atau deskripsi tentang profil websitenya</p>
21. </div>
22. </div>
23. <?php
24. include ('konten_kanan.php');
25. ?>
26.
27. </div>
28.
29. <?php
30. include ('footer.php');
31. ?>
32.

```

```
33. </body>
34. </html>
```

## H. Desain Halaman Kontak

Membuat halaman kontak buat file dengan nama **kontak.php**. kode nya seperti dibawah ini:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title>Kontak Website Layout</title>
5. <link rel="stylesheet" type="text/css" href="style.css">
6. </head>
7. <body>
8.
9. <?php
10. include ('header.php');
11. include ('menu.php');
12. ?>
13.
14. <div class="row">
15. <div class="leftcolumn">
16. <div class="card">
17. <h2>Hubungi Kami</h2>
18. <div class="container">
19. <form action="/action_page.php">
20. <div class="row">
21. <div class="col-25">
22. <label for="fname">Nama Lengkap</label>
23. </div>
24. <div class="col-75">
25. <input type="text" id="fname" name="nama" placeholder="Nama Lengkap..">
26. </div>
27. </div>
28. <div class="row">
29. <div class="col-25">
30. <label for="lname">No.Hp</label>
31. </div>
32. <div class="col-75">
```

```

33. <input type="text" id="lname" name="hp" placeholder="No.Hp..">
34. </div>
35. </div>
36. <div class="row">
37. <div class="col-25">
38. <label for="subject">Pesan</label>
39. </div>
40. <div class="col-75">
41. <textarea id="subject" name="pesan" placeholder="Tulis Pesan.." style="height:200px">
    </textarea>
42. </div>
43. </div>
44. <div class="row">
45. <input type="submit" value="kirim" >
46. </form>
47. </div>
48. </div>
49. </div>
50. <?php
51. include ('konten_kanan.php');
52. ?>
53.
54. </div>
55.
56. <?php
57. include ('footer.php');
58. ?>
59.
60. </body>
61. </html>

```

## I. Desain Halaman Login

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

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title>Login Website Layout</title>
5. <link rel="stylesheet" type="text/css" href="style.css">
6. </head>

```

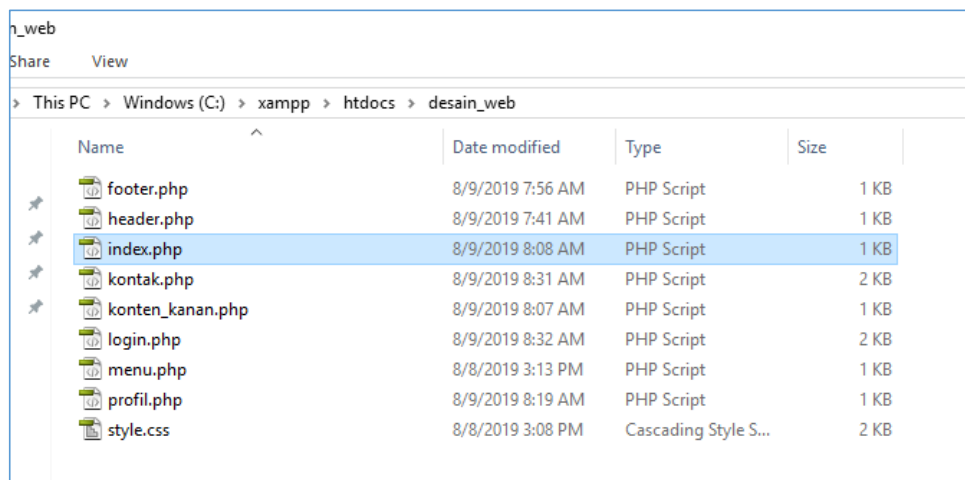
```

7. <body>
8.
9. <?php
10. include ('header.php');
11. include ('menu.php');
12. ?>
13.
14. <div class="row">
15. <div class="leftcolumn">
16. <div class="card">
17. <h2>Login Form</h2>
18. <div class="container">
19. <form action="/action_page.php">
20. <div class="row">
21. <div class="col-25">
22. <label for="fname">Username</label>
23. </div>
24. <div class="col-75">
25. <input type="text" id="fname" name="username" placeholder="username..">
26. </div>
27. </div>
28. <div class="row">
29. <div class="col-25">
30. <label for="lname">Passowrd</label>
31. </div>
32. <div class="col-75">
33. <input type="text" id="lname" name="password" placeholder="Password...">
34. </div>
35. </div>
36. <div class="row">
37. <input type="submit" value="Login">
38. </div>
39. </form>
40. </div>
41. </div>
42. </div>
43. <?php
44. include ('konten_kanan.php');
45. ?>
46.
47. </div>
48.
49. <?php

```

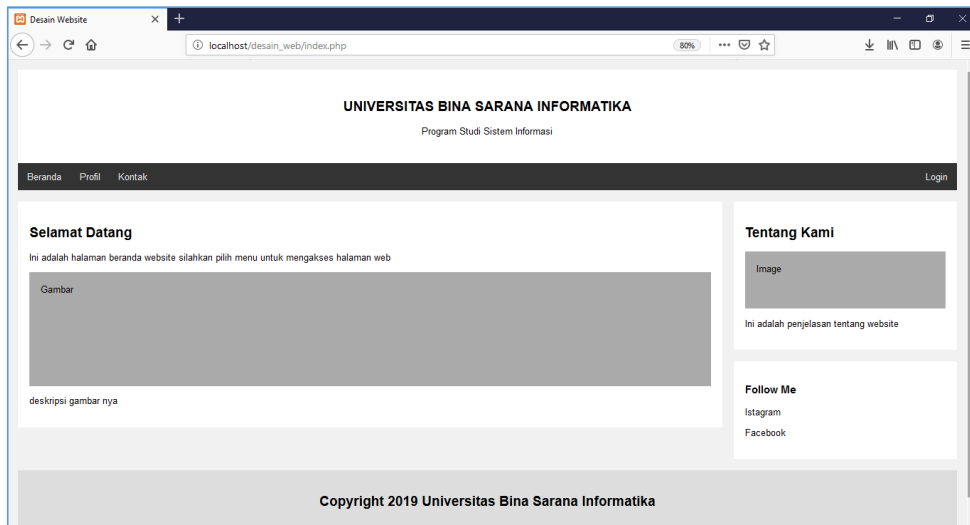
```
50. include ('footer.php');
51. ?>
52.
53. </body>
54. </html>
```

Setelah selesai semua desain halamannya pastikan semua file tersimpan dalam satu folder yaitu folder htdoc/desain\_web. Lihat gambar dibawah ini:



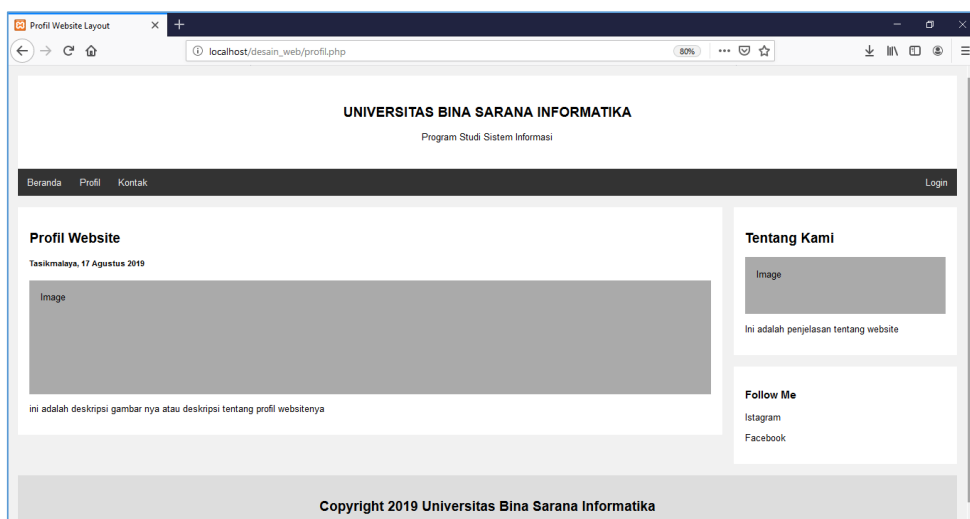
**Gambar File Desain Halaman web**

Apabila sudah tersimpan semuanya. Cek hasil tampilan nya buka browsernya dan ketikkan link [http://localhost/desain\\_web/](http://localhost/desain_web/) maka tampilan pertamanya atau tampilan halaman beranda seperti dibawah ini:



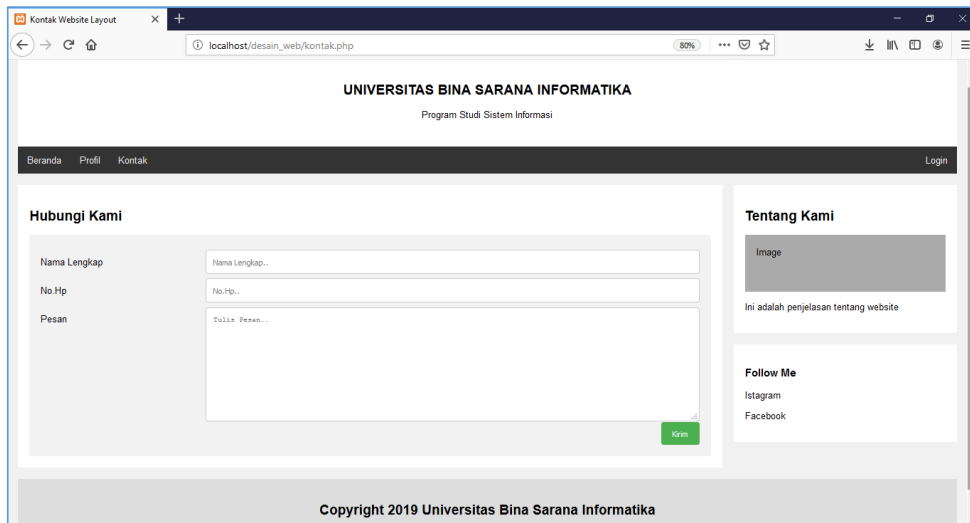
**Gambar 5.1 Tampilan Halaman Beranda**

Selanjutnya klik profil, maka tampilan nya seperti dibawah ini:



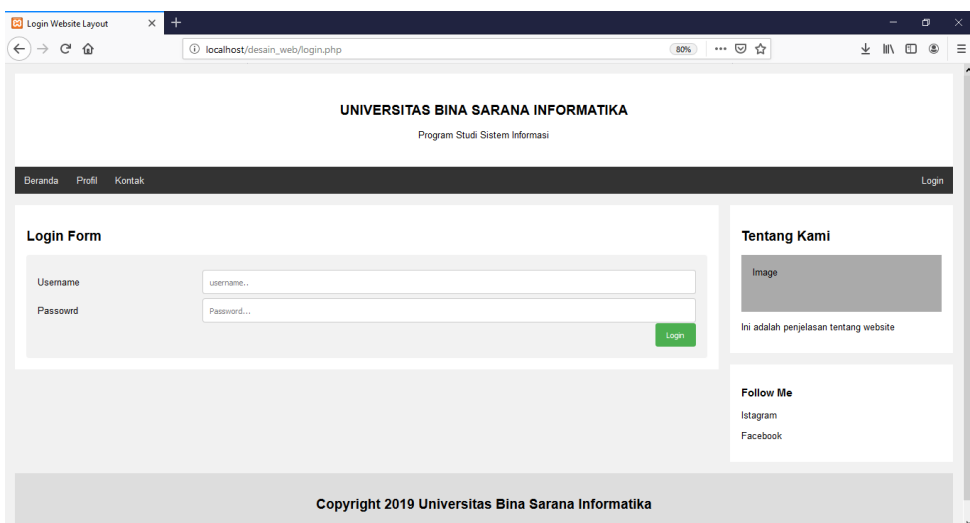
**Gambar5.2 Tampilan Halaman Profil**

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
- Agung Baitul Hikmah; Deddy Supriadi; Tuti Alawiyah. (2015). *Cara Cepat Membangun Website dari Nol: Studi Kasus: Web Dealer Motor*. Yogyakarta: Penerbit Andi.
- Edy Winarno ST, M. E. A. Z. S. D. C. (2014). *Pemrograman Web Berbasis HTML 5, PHP, Dan JavaScript*. Elex Media Komputindo.
- Dunia Ilmu komputer. Diambil dari <https://www.dunailkom.com/>
- Henky Prihatna. 2005. Kiat praktis menjadi web master professional. PT.Elexmedia komputindo. Jakarta.
- Petrus Andre. 2004. Macam-macam Struktur Navigasi Pada Website Diambil dari <https://www.andre.web.id/struktur-navigasi-website/>
- Raharjo. Budi, "Belajar Pemrograman Web", 1st ed, Bandung: MODULA, 2011.
- Refsnes Data. (2019). HTML5 Tutorial. Diambil dari <https://www.w3schools.com/html/>
- Sidik. Betha, "Pemrograman Web dengan PHP", 2nd ed, Bandung: INFORMATIKA, 2014.



## BIODATA PENULIS



**Miftah Farid Adiwisastra, S.T, M.Kom** lahir di Tasikmalaya 10 Juni 1986 telah menyelesaikan S1 tahun 2012 di Universitas BSI mengambil Jurusan Teknik Informatika Setelah itu melanjutkan Program Studi Magister Ilmu Komputer (S2) di STMIK Nusa Mandiri dan lulus pada tahun 2015. Saat ini menjadi Dosen Tetap di Universitas Bina Sarana Informatika Kampus Kota Tasikmalaya sejak 2013 sampai sekarang



**Agung Baitul Hikmah, S.Kom, M.Kom** lahir di Tasikmalaya 19 Agustus 1983 telah menyelesaikan S1 tahun 2009 di STMIK Nusa Mandiri , mengambil Jurusan Sistem Informasi, Setelah itu melanjutkan Program Studi Magister Ilmu Komputer (S2) di STMIK Nusa Mandiri dan lulus pada tahun 2013. Karya Buku yang sudah diterbitkan diantaranya buku berjudul “Cara Cepat Membangun Website dari Nol: Studi Kasus : Web Dealer Motor” pada Tahun 2015 dan Saat ini menjadi Dosen Tetap di Universitas Bina Sarana Informatika Kampus Kota Tasikmalaya sejak 2008 sampai sekarang



**Ai ilah Warnilah , S.T., M.Kom.** Lahir di Tasikmalaya, 18 Juni 1982. Telah menyelesaikan Program S1 tahun 2012 di Universitas Siliwangi Tasikmalaya, mengambil jurusan Teknik Informatika Setelah itu melanjutkan Program Studi Magister Ilmu Komputer (S2) di STMIK Nusa Mandiri dan lulus pada tahun 2015. Saat ini menjadi Dosen Tetap di Universitas Bina Sarana Informatika Kampus Kota Tasikmalaya sejak 2009 sampai sekarang

