



**MODUL PRAKTIKUM  
PEMROGRAMAN WEB  
CRK567**



**Disusun oleh:  
ADI WIDODO, S.Kom., MMSI**



**PROGRAM STUDI Rekam Medis  
FAKULTAS Ilmu – Ilmu Kesehatan  
UNIVERSITAS ESA UNGGUL  
2018/2019**



# DAFTAR ISI

DAFTAR ISI .....	II
RANCANGAN MODUL PRAKTIKUM PEMROGRAMAN WEB .....	VI
MODUL 1.....	1
REVIEW HTML 5 PART 1 (BASIC ATTRIBUT + ELEMENT) .....	1
1.1 Apa itu HTML.....	1
1.2 Contoh <i>Sintax Dasar</i> HTML 5 .....	1
1.3 Attribut HTML Images .....	2
1.4 Attribut HTML Link .....	2
1.5 Attribut HTML Headings .....	3
1.6 Elemen HTML <head> dan <title> .....	3
1.7 Elemen HTML untuk Format Teks .....	3
1.8 Elemen HTML untuk <i>Style</i> .....	4
1.9 Elemen HTML List .....	5
1.10 Latihan .....	6
MODUL 2.....	7
REVIEW HTML 5 PART 2 ( FORM + TABLE).....	7
2.1 HTML Form .....	7
2.1.1 Elemen <input> HTML Forms.....	7
2.2 HTML Tables .....	10
2.2.1 HTML Table dengan attribute Border dan Width .....	10
2.2.2 HTML Table dengan span row (menggabungkan beberapa baris) .....	11

2.2.3	HTML Table dengan span column (menggabungkan beberapa kolom) .....	11
2.2.4	HTML Table dengan caption / judul .....	11
2.2.5	HTML Table dengan background color .....	12
2.3	Latihan .....	13
<b>MODUL 3</b> .....		<b>14</b>
<b>CSS LAYOUT (DIV) + SIMPLE JAVASCRIPT (ALERT)</b> .....		<b>14</b>
3.1	Definisi CSS (Cascading Style Sheet) .....	14
3.2	Asal mula CSS .....	14
3.3	Cara Kerja CSS .....	14
3.4	<i>Linking dan Embedding</i> CSS .....	15
3.5	<i>Selector</i> ID dan Class pada CSS .....	16
3.6	Java Script Alert .....	18
3.7	Latihan Layouting CSS dan Javascript Alert .....	19
<b>MODUL 4</b> .....		<b>22</b>
<b>PHP PART 1 (PENGENALAN PHP + VARIABEL)</b> .....		<b>22</b>
4.1	Definisi PHP (Hypertext Preprocessor) .....	22
4.2	Sintax Dasar PHP .....	23
4.3	Penulisan Komentar di PHP .....	23
4.4	Deklarasi Variabel .....	24
4.5	Tipe Data Variabel pada PHP .....	24
4.6	Echo dan Print .....	26
4.7	Latihan .....	28

<b>MODUL 5</b> .....	29
<b>PHP PART 2 ( HTTP REQUEST )</b> .....	29
5.1 Apa itu HTTP?.....	29
5.2 Metode HTTP.....	29
5.3 Metode GET.....	29
5.4 Metode POST.....	30
5.5 Latihan .....	30
<b>MODUL 6</b> .....	32
<b>PHP PART 3</b> .....	32
<b>(PERCABANGAN , PERULANGAN, ARRAY)</b> .....	32
6.1 Percabangan.....	32
6.1.1 Pernyataan If.....	32
6.1.2 Pernyataan If..else.....	33
6.1.3 Pernyataan If..else.....	33
6.1.4 Pernyataan switch.....	34
6.2 Perulangan .....	35
6.2.1 Pernyataan while .....	35
6.2.2 Pernyataan do...while.....	35
6.2.3 Pernyataan for .....	36
6.2.4 Pernyataan foreach.....	36
6.3 Array .....	37
6.3.1 Tipe Array .....	37
6.3.2 <i>Indexed Array</i> .....	37
6.3.3 <i>Associative Array</i> .....	38
6.3.4 <i>Multidimensional Array</i> .....	39

6.3.5	Pengurutan <i>Array</i> .....	40
6.4	Latihan .....	42
<b>MODUL 7</b> .....		<b>46</b>
<b>PHP + MYSQL PART 1</b> .....		<b>46</b>
<b>(DATABASE CONNECTION + SELECT + LOGIN DENGAN SESSION)</b> .....		<b>46</b>
7.1	Sekilas Tentang My SQL .....	46
7.2	Membuat <i>Database</i> dan Tabel di MySQL .....	46
7.3	Sintax Koneksi PHP ke Database MySQL .....	48
7.4	Sintax PHP untuk Menampilkan Data dari Tabel .....	49
7.5	Session PHP untuk Login .....	50
7.6	Latihan .....	52
<b>MODUL 8</b> .....		<b>61</b>
<b>PHP + MYSQL PART 2</b> .....		<b>61</b>
<b>(INSERT + UPDATE + DELETE)</b> .....		<b>61</b>
8.1	Sintax PHP untuk Insert Data ke Tabel .....	61
8.2	Sintax PHP untuk Update Data ke Tabel .....	62
8.3	Sintax PHP untuk Delete Data ke Tabel .....	63
8.4	Latihan .....	64

## RANCANGAN MODUL PRAKTIKUM PEMROGRAMAN WEB

1. Review HTML 5 part 1 (Basic Attribute + Element)
2. Review HTML 5 part 2 ( Form + Basic Layout)
3. CSS + Simple Javascript (Alert)
4. PHP part 1 (Pengenalan PHP + Variabel)
5. PHP part 2 ( HTTP Request )
6. PHP Part 3 ( Controlling , Looping, Array)
7. PHP + MySQL part 1 (Database Connection + Select + Login dengan Session)
8. PHP + MySQL part 2 ( Insert + Update + Delete)



# MODUL 1

## REVIEW HTML 5 PART 1 (BASIC ATTRIBUT + ELEMENT)

### 1.1 Apa itu HTML

HTML adalah bahasa markup untuk menggambarkan dokumen web ( halaman web ) .

- HTML merupakan singkatan dari *Hyper Text Markup Language*
- Sebuah bahasa markup terdiri dari seperangkat tag markup
- Dokumen HTML dijelaskan oleh **tag HTML**
- Setiap tag HTML menjelaskan isi dokumen yang berbeda

Adapun versi dari HTML sekarang sudah mencapai versi ke -5 . Dari gambar 1.1 dapat kita lihat berbagai versi dari HTML dari pertama kali keluar hingga versi sekarang.

(Sumber :[http://www.w3schools.com/html/html\\_intro.asp](http://www.w3schools.com/html/html_intro.asp))

Version	Year
HTML	1991
HTML+	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2012

Gambar 1.1 Versi dari HTML hingga sekarang

### 1.2 Contoh *Sintax Dasar* HTML 5

Contoh sintax dasar dari *html 5* dapat dilihat pada gambar 1.2

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h1>My First Heading</h1>
5 <p>My first paragraph.</p>
6 </body>
7 </html>
```

Gambar 1.2 Contoh *sintax* dasar HTML 5



Penjelasan dari gambar 1.2 :

- Deklarasi `<!DOCTYPE html>` mendefinisikan tipe dokumen yakni HTML 5
- Teks di antara `<html>` dan `</html>` menjelaskan dokumen web
- Teks di antara `<body>` dan `</body>` menggambarkan isi halaman yang terlihat
- Teks di antara `<h1>` dan `</h1>` menggambarkan *heading* yang bertipe 1
- Teks di antara `<p>` dan `</p>` menjelaskan paragraf

Dengan menggunakan *tag-tag* tersebut maka *browser* dapat menampilkan dokumen berupa judul dan paragraf. Seperti yang dapat dilihat pada gambar 1.3



Gambar 1.3 *Output* dari *sintax* gambar 1.2

Biasakan menggunakan huruf kecil ketika menulis atribut dan element pada HTML.

### 1.3 Atribut HTML Images

Contoh cara penulisan attribute HTML Images dapat dilihat pada gambar 1.4

```

```

Gambar 1.4 Contoh penulisan *Attribut* HTML Images

### 1.4 Atribut HTML Link

Contoh cara penulisan attribute HTML Images dapat dilihat pada gambar 1.5

```
<a href="http://www.detik.com"> Situs Detik </a>
```

Gambar 1.5 Contoh Penulisan *Attribut* HTML Link



## 1.5 Atribut HTML Headings

Heading pada HTML terdiri dari <h1> sampai <h6>. Cara penulisannya dapat anda lihat pada gambar 1.6

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
```

Gambar 1.6 Contoh Penulisan *Atribut HTML Headings*

## 1.6 Elemen HTML <head> dan <title>

Elemen HTML <head> tidak ada hubungannya dengan judul HTML . HTML <head> hanya berisi meta data . Elemen HTML <head > ditempatkan antara tag <html> dan tag <body >. Sementara HTML <title> element adalah meta data yang mendefinisikan judul dokumen HTML dan tidak akan ditampilkan dalam dokumen. Namun, mungkin akan ditampilkan di salah satu tab browser . Contoh pemakaian elemen <head> dan <title> dapat dilihat pada gambar 1.7

```
<!DOCTYPE html>
<html>

<head>
  <title>My First HTML</title>
  <meta charset="UTF-8">
</head>

<body>
```

Gambar 1.7 Contoh Penulisan *elemen <head> dan <title>*

## 1.7 Elemen HTML untuk Format Teks

Elemen - elemen HTML yang digunakan untuk *formatting* teks dapat dilihat pada tabel 1.1

Tabel 1.1 Daftar Elemen *Formatting* Teks

Elemen	Fungsi
<b>.....</b>	Untuk menebalkan teks
<i>.....</i>	Untuk membuat teks terlihat miring
<u>.....</u>	Untuk menggarisbawahi teks
<s>.....</s>	Untuk memberikan coretan pada teks
<blink>.....</blink>	Membuat teks berkedip (khusus netscape)
<tt>.....</tt>	Menampilkan teks dalam format typewriter
<big>.....</big>	Untuk membesarkan teks
<small>.....</small>	Untuk mengecilkan teks
<sub>.....</sub>	Membuat teks subscript
<sup>.....</sup>	Membuat teks superscript
<em>.....</em>	Membuat teks yang di <i>emphasize</i> / ditekankan
<mark>.....</mark>	Membuat teks yang di <i>highlight</i>

## 1.8 Elemen HTML untuk *Style*

Untuk contoh pemanfaatan *style* – *style* yang ada pada HTML dapat anda lihat pada gambar 1.8 – gambar 1.12

```
<body style="background-color:lightgrey">
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
```

Gambar 1.8 *Style* pada Body Background Color

```
<!DOCTYPE html>
<html>
<body>
<h1 style="color:blue">This is a heading</h1>
<p style="color:red">This is a paragraph.</p>
</body>
</html>
```

Gambar 1.9 *Style color* pada Paragraf dan Heading

```
<!DOCTYPE html>
<html>
<body>
<h1 style="font-family:verdana">This is a heading</h1>
<p style="font-family:courier">This is a paragraph.</p>
</body>
</html>
```

Gambar 1.10 *Style* pada Paragraf dan Heading

```

<!DOCTYPE html>
<html>

<body>
  <h1 style="font-size:300%">This is a heading</h1>
  <p style="font-size:160%">This is a paragraph.</p>
</body>
</html>

```

Gambar 1.11 Style font size pada Paragraf dan Heading

```

<!DOCTYPE html>
<html>

<body>
  <h1 style="text-align:center">Centered Heading</h1>
  <p>This is a paragraph.</p>
</body>
</html>

```

Gambar 1.12 Style text align pada Paragraf dan Heading

Untuk elemen – elemen yang lain anda bisa diskusikan dengan *coass* yang berada di laboratorium komputer . Mengingat cukup banyak elemen – elemen yang berada di HTML cukup banyak.

## 1.9 Elemen HTML List

HTML List berfungsi untuk membuat daftar urutan dengan menggunakan penomoran maupun symbol. Di dalam HTML List. Terdapat 3 macam jenis list (daftar), yaitu:

### A. Ordered List

Ordered List berfungsi untuk membuat daftar list dimana tiap bagian akan diberi penomoran, tag diawali dengan `<ol>` dan diakhiri dengan `</ol>`. Untuk tiap bagian diawali dengan `<li>` dan diakhiri dengan `</li>`. Penomoran list secara default akan diawali dengan angka 1 / dengan **attribute type = "1"**.

Macam – macam attribute untuk penomoran pada Ordered List antara lain :

- Jika anda ingin format penomoran dengan huruf A, B, C maka attribute yang digunakan adalah **type = "A"**

- Jika anda ingin format penomoran dengan huruf i, ii, iii maka attribute yang digunakan adalah **type = "i"**
- Jika anda ingin format penomoran dengan huruf I, II, III maka attribute yang digunakan adalah **type = "I"**
- Jika anda ingin format penomoran dengan huruf a, b, c maka attribute yang digunakan adalah **type = "a"**

Contoh penulisan syntax *ordered list* dapat dilihat pada gambar 1.13

```
<ol>
  <li>Coffee</li>
  <li>Milk</li>
</ol>
```

Gambar 1.13 Contoh penggunaan *ordered list*

## B. Unordered List

Unordered List berfungsi untuk membuat daftar list dimana tiap bagian tidak akan diberi penomoran, tag diawali dengan `<ul>` dan diakhiri dengan `</ul>`. Untuk tiap bagian diawali dengan `<li>` dan diakhiri dengan `</li>`. Penomoran list secara default menggunakan lingkaran / dengan attribute **type = "circle"**.

### 1.10 Latihan

Buatlah sebuah blog sederhana menggunakan elemen-elemen dan attribute yang sudah ada di modul dan dipelajari dari *coass*.

Tambahkan gambar sebagai pemanis .

## MODUL 2

### REVIEW HTML 5 PART 2 ( FORM + TABLE)

#### 2.1 HTML Form

HTML Form digunakan untuk memberikan data ke sebuah server. Sebuah HTML Form dapat memiliki input – input element antara lain text field, checkbox, radio-button, submit button dan lain-lain. Sebuah form juga dapat mengandung select lists, textarea, fieldset, legend, dan element label.

Tag yang digunakan untuk membuat HTML Form adalah `<form>` .

```
<form>
.
input elements
</form>
```

Gambar 2.1 Tag `<form>`

##### 2.1.1 Elemen `<input>` HTML Forms

Hal yang terpenting dari element form adalah elemen `<input>` . elemen `<input>` digunakan untuk mengambil informasi dari user. Sebuah elemen input dapat dipakai dengan berbagai cara tergantung dari tipe attributnya . Sebuah elemen input dapat berupa text field, checkbox, password, radio button, submit button, dan lain - lain. Input type yang secara umum digunakan antara lain

##### A. Text Field

Input yang digunakan akan berupa kotak text

```
<form>
First name: <input type="text" name="firstname"><br>
Last name: <input type="text" name="lastname">
</form>
```

Gambar 2.2 Contoh Sintax Input Type Texfield



## B. Password Field

Password field adalah kotak yang ketika anda menuliskan password disitu maka huruf akan diganti dengan simbol lingkaran untuk melindungi tulisan password yang diketik.

```
<form>
Password: <input type="password" name="pwd">
</form>
```

Gambar 2.3 Contoh Sintax Input Type Password Field

## C. Radio Button

Digunakan untuk memilih inputan hanya satu pilihan saja. Contoh: Jenis kelamin.

```
<form>
<input type="radio" name="sex" value="male">Male<br>
<input type="radio" name="sex" value="female">Female
</form>
```

Gambar 2.4 Contoh Sintax Input Type Radio Button

## D. Drop Down List / Combo box

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

<form action="">
<select name="cars">
<option value="volvo">Volvo</option>
<option value="saab">Saab</option>
<option value="fiat">Fiat</option>
<option value="audi">Audi</option>
</select>
</form>

</body>
</html>
```

Gambar 2.5 Contoh Sintax Input Type Drop Down List

### E. Check box

Perbedaan dengan radio button adalah dapat mencentang / memilih lebih dari satu pilihan .

```
<form>  
<input type="checkbox" name="vehicle" value="Bike">I have a bike<br>  
<input type="checkbox" name="vehicle" value="Car">I have a car  
</form>
```

Gambar 2.6 Contoh Sintax Input Type Check Box

### F. Submit Button

`<input type="submit">` mendefinisikan sebuah submit button.

```
<form name="input" action="demo_form_action.asp" method="get">  
Username: <input type="text" name="user">  
<input type="submit" value="Submit">  
</form>
```

Gambar 2.7 Contoh Sintax Input Type Submit Button dalam Sebuah Form

### G. Text Area

Text Area seperti text field namun bedanya adalah text area dapat terdiri dari beberapa line/baris dan kolom

```
<textarea rows="4" cols="50">  
At w3schools.com you will learn how to make a website. We offer free tutorials in all web  
development technologies.  
</textarea>
```

Gambar 2.8 Contoh Sintax Input Text Area dalam Sebuah Form



## 2.2 HTML Tables

```
<table style="width:100%">
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Gambar 2.9 Contoh Sintax Penggunaan <table> </table>

Tabel didefinisikan dengan tag **<table>**. Tabel terdiri dari **tabel rows / baris** dengan tag **<tr>**. Dan **tabel column / kolom** dengan tag **<td>**. Tabel row / baris dapat juga terdiri dari **table headings** dengan tag **<th>**.

### 2.2.1 HTML Table dengan attribute Border dan Width

Penggunaan border pada gambar 2.10, digunakan untuk memberikan garis di sekitar tabel sedangkan style="width:100%" menunjukkan lebar tabel tersebut adalah 100% dari lebar pixel layar monitor yang digunakan saat membuka browser.

```
<table border="1" style="width:100%">
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Gambar 2.10 Contoh Sintax Penggunaan Style Attribute width pada Table

### 2.2.2 HTML Table dengan span row (menggabungkan beberapa baris)

Gunakan attribute rowspan = "<banyak baris>" untuk menggabungkan beberapa baris dari tabel. Contohnya dapat dilihat pada gambar 2.11

```
<table style="width:100%">
<tr>
<th>First Name:</th>
<td>Bill Gates</td>
</tr>
<tr>
<th rowspan="2">Telephone:</th>
<td>555 77 854</td>
</tr>
<tr>
<td>555 77 855</td>
</tr>
</table>
```

Gambar 2.11 Contoh Sintax Penggunaan Rowspan

### 2.2.3 HTML Table dengan span column (menggabungkan beberapa kolom)

Gunakan attribute colspan = "<banyak kolom>" untuk menggabungkan beberapa baris dari tabel. Contohnya dapat dilihat pada gambar 2.12

```
<table style="width:100%">
<tr>
<th>Name</th>
<th colspan="2">Telephone</th>
</tr>
<tr>
<td>Bill Gates</td>
<td>555 77 854</td>
<td>555 77 855</td>
</tr>
</table>
```

Gambar 2.12 Contoh Sintax Penggunaan Colspan

### 2.2.4 HTML Table dengan caption / judul

Gunakan tag <caption ....></caption> untuk membuat judul pada tabel. Tag <caption..></caption> harus dimasukkan setelah tag <table>. Contohnya dapat dilihat pada gambar 2.13

```
<table style="width:100%">
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$50</td>
  </tr>
</table>
```

Gambar 2.13 Contoh Sintax Penggunaan Caption

### 2.2.5 HTML Table dengan background color

Penggunaan attribute bgcolor dapat menggunakan bilangan hexa atau dengan nama warna yang sudah umum seperti "Red", "Yellow", "Green", "Light Green", dan lain-lain. Contoh penggunaannya dapat dilihat pada gambar 2.14

```
<table bgcolor="#00FF00">
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

Gambar 2.14 Contoh Sintax Penggunaan attribute bgcolor

### 2.3 Latihan

Mintalah pada koas anda latihan dengan menggunakan element dan attribute dari form dan table !!



## MODUL 3

### CSS Layout (DIV) + SIMPLE JAVASCRIPT (ALERT)

#### 3.1 Definisi CSS (Cascading Style Sheet)

CSS adalah salah satu bahasa pemrograman desain web (*style sheet languages*) yang mengontrol format tampilan sebuah halaman web yang ditulis dengan bahasa penanda (*markup language*). Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang bahasa pemrograman CSS bisa diaplikasikan untuk segala dokumen XML, termasuk SVG dan XUL.

#### 3.2 Asal mula CSS

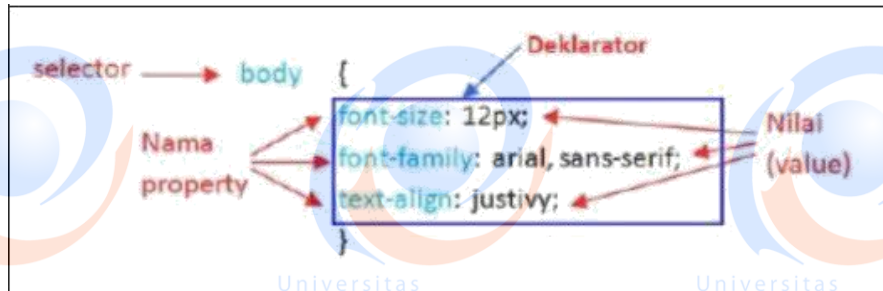
Tim Berners – Lee ( Penemu *World Wide Web* ) merancang CSS untuk diterapkan pada halaman web. Pada awalnya terdapat tag-tag style seperti <font>, <strong> serta beberapa tag style yang lain berupa tag-tag yang langsung terpresentasi pada halaman web itu sendiri. Dalam hal ini, CSS berfungsi untuk membedakan konten dari halaman web dengan pemberi tampilan dari halaman web itu sendiri.

#### 3.3 Cara Kerja CSS

CSS bekerja dengan menggunakan dua buah elemen penting untuk pemformatan tampilan, diantaranya elemen selector dan deklarator. Dua buah elemen ini berfungsi sebagai penentu format tampilan dan lainnya menempatkan format tampilan.

Deklarator berisi beberapa perintah – perintah CSS untuk menentukan format tampilan dari sebuah elemen pada halaman web. Sedangkan selector adalah sebuah perintah lanjutan dari deklarator dan berfungsi menempatkan format tampilan dari deklarator pada halaman web.





Gambar 3.1 Cara Kerja Selector dan Deklarator

Aturan dasar untuk CSS antara lain :

1. Setiap statement harus mempunyai selector dan deklartor. Deklarator harus dibuat setelah selector dan diapit 2 buah kurung kurawal (buka dan tutup).
2. Properti dari deklarator dipisahkan menggunakan titik koma (;)
3. Setiap property mempunyai nama dan diikuti oleh titik dua dan selanjutnya nilai (*value*) dari properti tersebut. Terdapat banyak tipe dari nilai ini, tetapi setiap properti hanya bisa mengambil beberapa value seperti yang telah diatur pada spesifikasi CSS. Spesifikasi ini akan memberitahu nilai yang mungkin diletakkan pada setiap property
4. Beberapa properti bisa menampung beberapa nilai, seperti pada contoh pada gambar 3.1 yakni font-family, nilai baru harus dipisah dengan koma dan spasi.
5. Beberapa nilai mempunyai ukuran tertentu seperti pt (point), px(pixel), seperti pada font-size. Anda tidak dapat member spasi antara nilai dan ukurannya seperti 45pt / 45px.
6. Spasi dapat digunakan untuk memudahkan anda mengubah atau memperbahuri file CSS anda.

### 3.4 *Linking dan Embedding CSS*

Ada beberapa cara untuk memasukkan CSS ke dalam satu dokumen HTML antara lain :

1. Menyisipkan CSS langsung ke dalam tag HTML

Contoh dapat dilihat pada gambar 3.2

```
<h2 style="color:red;background:black;">This is a red heading with a black background</h2>
```

Gambar 3.2 *Inlince* CSS ke dalam tag HTML

## 2. Mengembed CSS ke HTML

Contoh dapat dilihat pada gambar 3.3

```
<style media="screen" type="text/css">
Add style rules here
</style>
```

Gambar 3.3 *Emmbedding* CSS ke HTML

## 3. Mehubungkan file CSS yang terpisah ke HTML (*Linked CSS*)

Contoh dapat dilihat pada gambar 3.4

```
<link rel="stylesheet" type="text/css" href="mystyles.css" media="screen" />
```

Gambar 3.4 *Linking* CSS ke HTML

## 4. Mengimpor file CSS di dalam CSS

Contoh dapat dilihat pada gambar 3.5

```
@import "newstyles.css";
```

Gambar 3.5 Impor file CSS

Untuk praktikum saat ini kita cukup memakai cara nomer 1 sampai dengan nomer 3.

### 3.5 *Selector* ID dan Class pada CSS

Untuk *selector* id pada css ditandai dengan tanda #(pagar) contoh penulisan dapat dilihat pada gambar 3.6



```
#Tulisan
{
color : red;
font-family : Arial;
}
```

Gambar 3.6 Contoh Selector ID

Adapun cara pemanggilan untuk ID diatas dapat dilihat pada gambar 3.7

```
<body>
<p id="Tulisan"> Tes </p>
</body>
```

Gambar 3.7 Contoh Pemanggilan CSS Selector ID

Sedangkan untuk *selector* class pada css ditandai dengan tanda .(dot) contoh penulisan dapat dilihat pada gambar 3.8

```
.Center
{
Background-image : Url („Namafile.jpg“)
Background-position : Center Center;
}
```

Gambar 3.8 Contoh Selector Class

Adapun cara pemanggilan untuk ID diatas dapat dilihat pada gambar 3.9

```
<body class="Center">
</body>
```

Gambar 3.9 Contoh Pemanggilan CSS Selector Class

Masih ada banyak sebenarnya jenis-jenis *selector* pada CSS namun untuk praktikum saat ini anda cukup mengetahui selector jenis Id dan Class.

### 3.6 Java Script Alert

Javascript adalah bahasa script yang populer di internet dan dapat bekerja di sebagian besar penjelajah web browser seperti Internet Explorer (IE), Mozilla Firefox, Netscape, opera dan web browser lainnya. Kode javascript biasa dituliskan dalam bentuk fungsi (Function) yang ditaruh di bagian dalam tag <head> yang dibuka dengan tag <script language=" javascript">

Sedangkan alert adalah salah satu komponen dari javascript untuk menampilkan sebuah kotak pesan.

Contoh penulisan sintax dasar Javascript untuk penulisan alert dapat dilihat pada gambar 3.10

```
<head>
<title>Latihan Modul 2</title>
<script language="javascript">
function cobaAlert()
{
  alert ("Halo Dunia .. !");
}
</script>
</head>
```

Gambar 3.10 Contoh Penulisan Sintax Alert untuk Javascript

Setelah function tadi dibuat maka function tadi dapat dipanggil melalui salah satu komponen HTML. Sebagai contoh dapat dilihat pada gambar 3.11 tentang cara memanggil fungsi cobaAlert() lewat komponen button dan hasilnya ketika button kirim ditekan maka akan muncul pesan seperti pada gambar 3.12

```
<input type="button" value="Kirim" onClick="cobaAlert()"/>
```

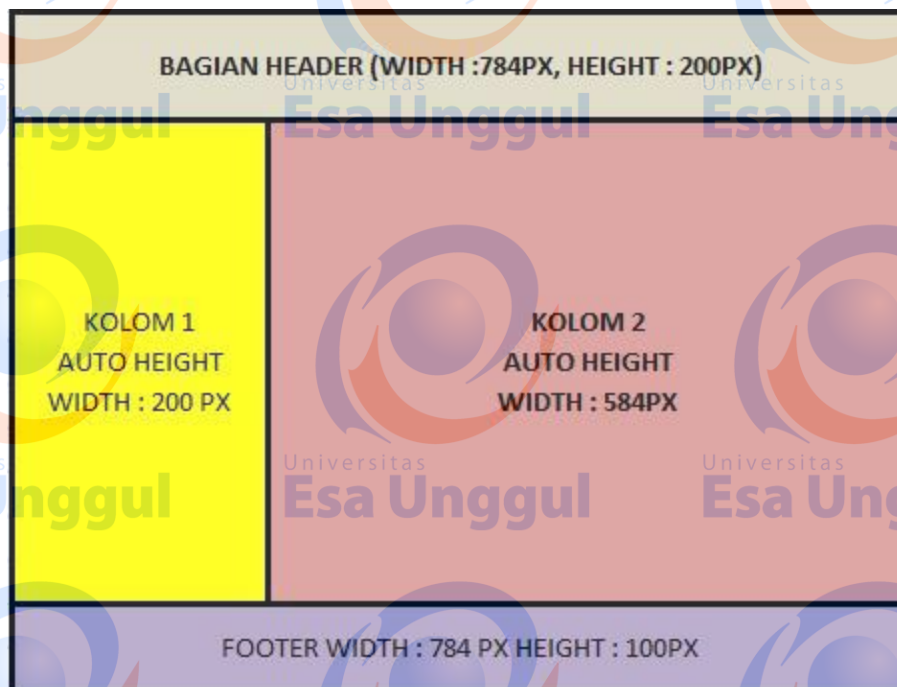
Gambar 3.11 Contoh Pemanggilan fungsi Alert untuk Javascript



Gambar 3.12 Alert yang tampil

### 3.7 Latihan Layouting CSS dan Javascript Alert

Desainlah sebuah web dengan layout seperti gambar berikut.



Gambar 3.13 Desain Layout

Untuk tampilan website sendiri akan berusaha untuk menduplikasi website seperti gambar 3.14. Tanpa menggunakan tabel (murni hanya menggunakan tag <div> dan selector id )



Gambar 3.14 Tampilan Website nya

Untuk background header dapat anda download di :  
<http://192.168.100.3/images/Header.jpg>

Sedangkan untuk gambar yang menunjukkan ujian praktikum dapat anda download di : <http://192.168.100.3/UP.jpg>.

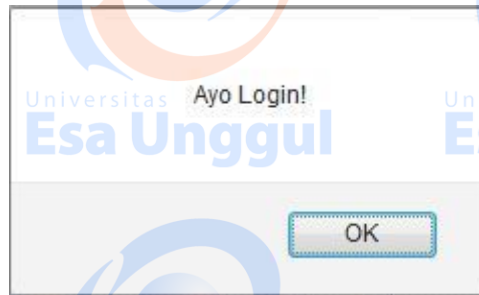
Aturlah gambar ujian praktikum tadi sedemikian hingga seperti pada gambar 3.14

Untuk CSS Layout nya silahkan dapat anda lihat script nya pada gambar 3.15 .  
 Tanyakanlah pada coass anda maksud dari script pada gambar 3.15 itu.  
 Gunakanlah **linked CSS** seperti pada gambar 3.4 untuk membuat CSSnya. Untuk gambar di header tarulah pada folder **images** untuk dijadikan background dari header. Tulisan © gunakan fungsi &copy;

```
1 #layout {
2     float: none;
3     height: auto;
4     width: 784px;
5 }
6 #header {
7     float: left;
8     height: 150px;
9     width: 784px;
10    background-image: url(images/header.jpg);
11 }
12 #kolom1 {
13     float: left;
14     height: auto;
15     width: 200px;
16     background-color : #ffffff;
17 }
18 #kolom2 {
19     float: left;
20     height: auto;
21     width: 584px;
22     background-color : #ffffff;
23 }
24 #footer {
25     float: left;
26     height: 50px;
27     width: 784px;
28     text-align: center;
29     background-color : blue;
30 }
```

Gambar 3.15 CSS Untuk mengatur Layout web

Jika tombol login ditekan maka akan menampilkan alert seperti pada gambar 3.16. Tanyakan pada coass anda mengenai script nya.



Gambar 3.16 Alert ketika button login ditekan



## MODUL 4

### PHP PART 1 (PENGENALAN PHP + VARIABEL)

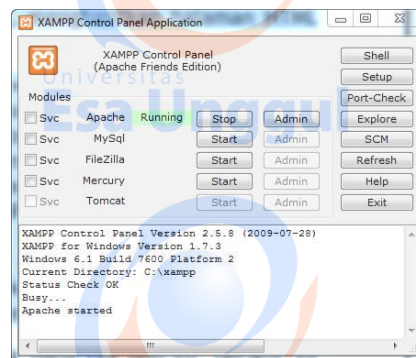
#### 4.1 Definisi PHP (Hypertext Preprocessor)

PHP (PHP: Hypertext Preprocessor) merupakan bahasa pemrograman yang mampu berjalan di server-side (berjalan pada server web atau CGI), berbeda dengan javascript yang normalnya berjalan pada client-side (berjalan pada browser). PHP merupakan turunan C, Java dan Perl, yang dibuat dengan bahasa C. PHP bertujuan melakukan preprocessing, menghasilkan halaman HTML yang dinamis untuk tiap client.

Sebelum memulai membuat website dengan menggunakan PHP, kita membutuhkan web server, dan nantinya untuk menyimpan data tiap user secara terstruktur, kita membutuhkan database. Umumnya PHP dipaketkan dengan Apache (web server) dan MySQL (database), paket tersebut sering disebut dengan AMP.

Beberapa contoh AMP adalah XAMPP, WAMPServer, VertrigoServ, LAMP, UniServer (Uniform Server), Server2Go dan WAMP.

Adapun untuk praktikum di laboratorium komputer saat ini kita menggunakan XAMPP. Gambar 4.1 menunjukkan bagaimana Web Server dinyalakan di XAMPP.



Gambar 4.1 Apache Server dinyalakan

Untuk seterusnya penamaan *file* menggunakan *extension* .php dan peletakan file akan ditaruh ke dalam folder /htdocs yang akan dijelaskan lebih lanjut oleh *co-asisten*.

## 4.2 Sintax Dasar PHP

Skrip PHP dimulai dengan `<?php` dan diakhiri dengan `?>` seperti pada gambar 4.2

```
<?php
// PHP code goes here
?>
```

Gambar 4.2 Sintax Dasar PHP

Skrip PHP juga dapat disisipkan ke dalam dokumen yang mengandung tag HTML, seperti ditunjukkan pada gambar 4.3.

```
<!DOCTYPE html>
<html>
<body>
<h1>My first PHP page</h1>
<?php
echo "Hello World!";
?>
</body>
</html>
```

Gambar 4.3 Sintax Dasar PHP yang Disisipkan ke dalam HTML

## 4.3 Penulisan Komentar di PHP

Komentar berguna untuk proses dokumentasi dan dipakai untuk mendeskripsikan skrip PHP yang ditulis. Gambar 4.4 menunjukkan penulisan komentar.

```
<!DOCTYPE html>
<html>
<body>
<?php
// This is a single line comment
# This is also a single line comment
/*
Universitas
This is a multiple lines comment block
that spans over more than
one line
*/
?>
</body>
</html>
```

Gambar 4.4 Sintax Dasar PHP untuk Komentar



## 4.4 Deklarasi Variabel

Untuk pendeklarasian variabel pada PHP diawali dengan tanda \$ contohnya dapat dilihat pada gambar 4.5

```
<?php
$txt="Hello world!";
$x=5;
$y=10.5;
?>
```

Gambar 4.5 Penulisan Deklarasi Variabel.

## 4.5 Tipe Data Variabel pada PHP

### A. String

String merupakan sekumpulan karakter, seperti "Hello world!", contoh dapat dilihat pada gambar 4.6

```
<?php
$x = "Hello world!";
echo $x;
echo "<br>";
$x = 'Hello world!';
echo $x;
?>
```

Gambar 4.6 Penulisan Tipe Data String

### B. Integer

String merupakan sekumpulan bilangan tanpa koma (non decimal), aturan untuk tipe data integer :

1. Harus minimal 1 digit (0 – 9).
2. Tanpa mengandung koma atau tanda kurung.
3. Tidak mengandung titik desimal.
4. Bisa bilangan positif atau negatif
5. Dispesifikasikan menjadi 3 format : decimal (10-based), hexadecimal (16-based diawali dengan 0x) atau octal (8-based diawali dengan 0)

Contoh penulisan tipe data integer dapat dilihat pada gambar 4.7, var\_dump() merupakan fungsi untuk mengembalikan tipe data dari variabel.

```

<?php
$x = 5985;
var_dump($x);
echo "<br>";
$x = -345; // negative number
var_dump($x);
echo "<br>";
$x = 0x8C; // hexadecimal number
var_dump($x);
echo "<br>";
$x = 047; // octal number
var_dump($x);
?>

```

Gambar 4.7 Penulisan Tipe Data Integer.

### C. Floating Point

Floating Point merupakan bilangan dengan menggunakan desimal atau bilangan dengan eksponensial. Contoh penulisan dapat dilihat pada gambar 4.9.

```

<?php
$x = 10.365;
var_dump($x);
echo "<br>";
$x = 2.4e3;
var_dump($x);
echo "<br>";
$x = 8E-5;
var_dump($x);
?>

```

Gambar 4.8 Penulisan Tipe Data Floating Point.

### D. Boolean

Boolean hanya digunakan untuk menentukan *true* atau *false*. Contoh penulisan dapat dilihat pada gambar 4.9.

```

$x=true;
$y=false;

```

Gambar 4.8 Penulisan Tipe Data Boolean.

### E. Array

Sebuah array menyimpan sekumpulan nilai yang bertipe data sama . Contoh penulisan dapat dilihat pada gambar 4.10.

```
<?php
$cars=array("Volvo","BMW","Toyota");
var_dump($cars);
?>
```

Gambar 4.10 Penulisan Tipe Data Array.

## F. Object

Untuk tipe data ini dipakai untuk menyimpan informasi yang diberikan ke dalam suatu *class*. Anda akan mempelajari ini lebih lanjut jika anda mendalami Pemrograman Berbasis Objek. Contoh penulisan dapat dilihat pada gambar 4.11.

```
<?php
class Car
{
    var $color;
    function Car($color="green") {
        $this->color = $color;
    }
    function what_color() {
        return $this->color;
    }
}
?>
```

Gambar 4.11 Penulisan Tipe Data Object.

## G. Null

Untuk penulisan tipe data null dapat dilihat pada gambar 4.12

```
<?php
$x="Hello world!";
$x=null;
var_dump($x);
?>
```

Gambar 4.12 Penulisan Tipe Data Null.

## 4.6 Echo dan Print

Ada beberapa perbedaan antara echo dan print :

- echo – dapat mengeluarkan satu atau lebih string.
- print – hanya mengeluarkan satu string, dan selalu mengembalikan 1.

Echo adalah bahasa konstruksi , dan dapat digunakan dengan atau tanpa tanda kurung : echo atau echo()

Echo digunakan untuk menampilkan tulisan seperti ditunjukkan pada gambar 4.13 yang dapat disisipkan tag HTML, atau menampilkan variabel seperti ditunjukkan pada gambar 4.14

```
<?php
echo "<h2>PHP is fun!</h2>";
echo "Hello world!<br>";
echo "I'm about to learn PHP!<br>";
echo "This", " string", " was", " made", " with multiple parameters.";
?>
```

Gambar 4.13 Echo untuk Menampilkan String

```
<?php
$txt1="Learn PHP";
$txt2="W3Schools.com";
$cars=array("Volvo","BMW","Toyota");

echo $txt1;
echo "<br>";
echo "Study PHP at $txt2";
echo "My car is a {$cars[0]}";
?>
```

Gambar 4.14 Echo untuk Menampilkan Variabel

Print juga adalah bahasa konstruksi , dan dapat digunakan dengan atau tanpa tanda kurung : print atau print()

Print digunakan untuk menampilkan tulisan seperti ditunjukkan pada gambar 4.15 dan teristimewa dapat digunakan untuk menuliskan syntax HTML di dalamnya, atau menampilkan variabel seperti ditunjukkan pada gambar 4.16

```
<?php
print "<h2>PHP is fun!</h2>";
print "Hello world!<br>";
print "I'm about to learn PHP!";
?>
```

Gambar 4.15 Print untuk Menampilkan String

```
<?php
$txt1="Learn PHP";
$txt2="W3Schools.com";
$cars=array("Volvo","BMW","Toyota");

print $txt1;
print "<br>";
print "Study PHP at $txt2";
print "My car is a {$cars[0]}";
?>
```

Gambar 4.16 Print untuk Menampilkan Variabel

#### 4.7 Latihan

Modifikasilah latihan pada modul 3 sehingga tag HTML yang digunakan bisa masuk ke dalam tag PHP.

Gunakan ekstensi file .php dalam memberi nama file, jangan lupa untuk menaruh file kedalam folder .htdocs.



## MODUL 5

### PHP PART 2 ( HTTP REQUEST )

#### 5.1 Apa itu HTTP?

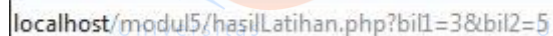
HTTP yakni Hypertext Transfer Protocol didesain untuk komunikasi antara *client* dan *server*. HTTP bekerja sebagai protocol dari respon permintaan antara *client* dan *server*. Sebuah web browser dapat bertindak sebagai *client* dan sebuah aplikasi komputer yang meng-*hosting* sebuah web dapat bertindak sebagai *server*.

#### 5.2 Metode HTTP

Ada 2 metode HTTP yang pada umumnya digunakan untuk melakukan respon permintaan antara *client* dan *server* yakni :

1. **GET** - Untuk *request* data dari sumber yang spesifik.
2. **POST** – Untuk *submit* data yang diproses ke sumber yang spesifik.

#### 5.3 Metode GET



```
localhost/modul5/hasilLatihan.php?bil1=3&bil2=5
```

Gambar 5.1 Contoh pemakaian Metode Get

Untuk penulisan method get maka hasil yang diterima akan ditampilkan pada web browser dengan tanda ?, sedangkan jika ada lebih dari 1 data yang *direquest* maka pemisahannya akan dihubungkan dengan tanda & . Contoh penerapan metode GET dapat dilihat pada gambar 5.1

Beberapa catatan tentang GET :

- Permintaan melalui metode GET dapat di-*cache*
- Permintaan melalui metode GET akan disimpan dalam riwayat browser
- Permintaan melalui metode GET dapat di *bookmark*
- Permintaan melalui metode GET tidak boleh digunakan untuk data-data yang sensitif
- Memiliki batasan panjang data yang *direquest*

- Digunakan hanya untuk mengambil data

## 5.4 Metode POST

Berbeda dengan metode GET, untuk metode POST tidak ada hasil dari data yang *direquest* yang ditampilkan ke browser. Contoh penerapan dapat dilihat pada gambar 5.2.



localhost/modul5/hasilLatihan.php

Gambar 5.2 Contoh pemakaian Metode POST

Beberapa catatan tentang metode POST :

- Permintaan POST tidak pernah disimpan dalam cache
- Permintaan POST tidak disimpan tetap dalam riwayat browser
- Permintaan POST tidak dapat *dibookmark*
- Permintaan POST tidak memiliki batasan pada panjang data

## 5.5 Latihan

1. Nyalakan XAMPP dan APACHE pada computer praktikum anda kemudian buatlah sebuah form dengan format tampilan seperti ini



Bilangan 1 :   
Bilangan 2 :

Gambar 5.3 Format Form Latihan

Ketiklah script seperti pada gambar 5.4 dan simpanlah dengan nama formLatihan.php pada folder .htdocs

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <form name="input" action="hasilLatihan.php" method="get">
5 Bilangan 1 : <input type="text" name="bil1"> </br>
6 Bilangan 2 : <input type="text" name="bil2">
7 </br>
8 <input type="submit" value="Hitung">
9 </form>
10 </body>
11 </html>
12
13
14
```

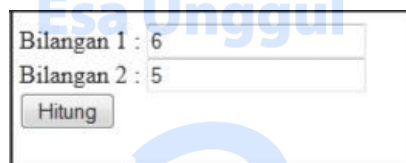
Gambar 5.4 Script Desain Form

Kemudian ketiklah script seperti pada gambar 5.5 dan simpanlah dengan nama hasilLatihan.php pada folder .htdocs

```
1 <?php
2 $hasilJumlah = ($_GET["bil1"]+$_GET["bil1"]);
3 $hasilKurang = ($_GET["bil1"]-$_GET["bil2"]);
4
5 echo "Hasil Penjumlahan : " . $hasilJumlah;
6 echo "<br/>";
7 echo "Hasil Pengurangan : " . $hasilKurang;
8
9 ?>
```

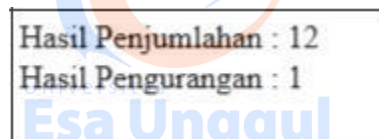
Gambar 5.5 Script hasilLatihan.php

Jika sudah jalan kan url : <http://localhost/formLatihan.php> pada web browser anda kemudian isikan bilangan pada text field bilangan 1 dan bilangan 2 seperti pada gambar 5.6. Jika sudah terisi maka tekan lah tombol hitung dan hasilnya dapat anda lihat pada gambar 5.7.



Bilangan 1 : 6  
Bilangan 2 : 5  
Hitung

Gambar 5.6 Contoh Isian Form pada formLatihan.php



Hasil Penjumlahan : 12  
Hasil Pengurangan : 1

Gambar 5.7 Hasil pada hasilLatihan.php

Amati web browser anda apa yang tertera disitu dan tanyakan pada koas anda

2. Tambahkan juga untuk hasil perkalian dan penjumlahan

3. Ubahlah Methode HTTP nya dari GET ke POST (Tanyakan pada Coass anda bagaimana caranya) dan amatilah hasilnya

## MODUL 6 PHP PART 3

### (PERCABANGAN , PERULANGAN, ARRAY)

#### 6.1 Percabangan

Dalam PHP terdapat beberapa pernyataan untuk percabangan antara lain :

- Pernyataan dengan `if` – digunakan untuk mengeksekusi beberapa baris kode jika satu kondisi yang spesifik bernilai *true*
- Pernyataan `if..else` - digunakan untuk mengeksekusi beberapa baris kode jika satu kondisi di yang spesifik bernilai *true* dan kondisi yang lain bernilai *false*
- Pernyataan `if..elseif...else` – digunakan untuk mengeksekusi beberapa baris blok yang bernilai *true*
- Pernyataan `switch` - memilih satu dari beberapa baris blok untuk dieksekusi jika bernilai *true*

##### 6.1.1 Pernyataan If

Pernyataan `if` digunakan untuk mengeksekusi sejumlah kode hanya jika dalam satu kondisi yang secara spesifik bernilai *true*.

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.1 dan 6.2

```
if (condition) {  
    code to be executed if condition is true;  
}
```

Gambar 6.1 Syntax If



```
<?php
$t=date("H");

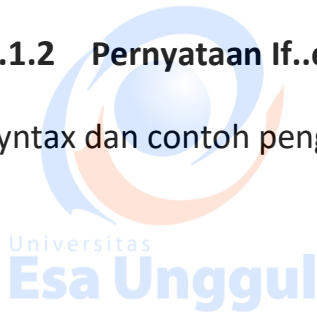
if ($t<"20") {
    echo "Have a good day!";
}
?>
```



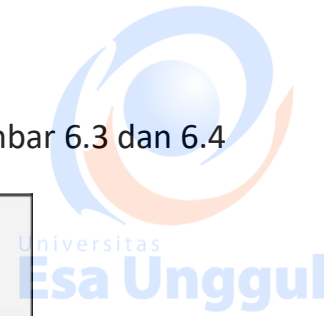
Gambar 6.2 Contoh penggunaan syntax If

### 6.1.2 Pernyataan If..else

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.3 dan 6.4



```
if (condition) {
    code to be executed if condition is true;
} else {
    code to be executed if condition is false;
}
```



Gambar 6.3 Syntax If..else



```
<?php
$t=date("H");

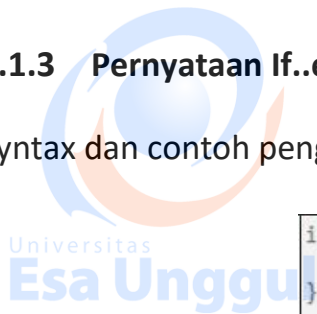
if ($t<"20") {
    echo "Have a good day!";
} else {
    echo "Have a good night!";
}
?>
```



Gambar 6.4 Contoh penggunaan syntax If....else

### 6.1.3 Pernyataan If..elseif..else

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.5 dan 6.6



```
if (condition) {
    code to be executed if condition is true;
} elseif (condition) {
    code to be executed if condition is true;
} else {
    code to be executed if condition is false;
}
```



Gambar 6.5 Syntax If..elseif..else





```

<?php
$t=date("H");

if ($t<"10") {
    echo "Have a good morning!";
} elseif ($t<"20") {
    echo "Have a good day!";
} else {
    echo "Have a good night!";
}
?>

```

Gambar 6.6 Contoh penggunaan syntax If..elseif..else

#### 6.1.4 Pernyataan switch

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.7 dan 6.8

```

switch (n) {
    case label1:
        code to be executed if n=label1;
        break;
    case label2:
        code to be executed if n=label2;
        break;
    case label3:
        code to be executed if n=label3;
        break;
    ...
    default:
        code to be executed if n is different from all labels;
}

```

Gambar 6.7 Syntax switch

```

<?php
$favcolor="red";
switch ($favcolor) {
    case "red":
        echo "Your favorite color is red!";
        break;
    case "blue":
        echo "Your favorite color is blue!";
        break;
    case "green":
        echo "Your favorite color is green!";
        break;
    default:
        echo "Your favorite color is neither red, blue, or green!";
}
?>

```

Gambar 6.8 Contoh penggunaan syntax switch

## 6.2 Perulangan

Dalam PHP terdapat pula beberapa pernyataan untuk perulangan antara lain :

- Pernyataan `while` - perulangan yang dilakukan dalam blok kode selama kondisi bernilai `true`.
- Pernyataan `do..while` - perulangan yang dilakukan dalam blok kode minimal sekali dan kemudian diulangi lagi jika kondisi bernilai `true`.
- Pernyataan `for` - perulangan yang dilakukan dalam baris kode dengan batasan jumlah tertentu.
- Pernyataan `foreach` - perulangan yang dilakukan untuk blok kode dari setiap elemen yang ada di `array`.

### 6.2.1 Pernyataan `while`

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.9 dan 6.10

```
while (condition is true) {  
    code to be executed;  
}
```

Gambar 6.9 Syntax `while`

```
<?php  
$x=1;  
  
while($x<=5) {  
    echo "The number is: $x <br>";  
    $x++;  
}  
?>
```

Gambar 6.10 Contoh penggunaan syntax `while`

### 6.2.2 Pernyataan `do...while`

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.11 dan 6.12

```
do {  
    code to be executed;  
} while (condition is true);
```

Gambar 6.11 Syntax do..while

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

Gambar 6.12 Contoh penggunaan syntax do..while

### 6.2.3 Pernyataan for

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.13 dan 6.14

```
for (init counter; test counter; increment counter) {  
    code to be executed;  
}
```

Gambar 6.13 Syntax for

```
<?php  
for ($x=0; $x<=10; $x++) {  
    echo "The number is: $x <br>";  
}  
?>
```

Gambar 6.14 Contoh penggunaan syntax for

### 6.2.4 Pernyataan foreach

Syntax dan contoh penggunaannya dapat dilihat pada gambar 6.15 dan 6.16

```
foreach ($array as $value) {  
    code to be executed;  
}
```

Gambar 6.13 Syntax foreach

```

<?php
$colors = array("red","green","blue","yellow");

foreach ($colors as $value) {
    echo "$value <br>";
}
?>

```

Gambar 6.14 Contoh penggunaan syntax foreach

## 6.3 Array

Array adalah sebuah variabel yang bersifat spesial dimana dapat menampung lebih dari satu nilai dalam satu waktu.

### 6.3.1 Tipe Array

Di dalam PHP terdapat tiga tipe array, antara lain :

- Indexed arrays -> array dengan index numeric
- Associative arrays -> array dengan pemberian nama key
- Multidimensional arrays -> array yang terdiri dari satu atau lebih array.

Oleh karena itu nantinya akan dibahas satu persatu secara singkat bagaimana cara mengcreate array dan mencetak array tersebut berdasarkan ketiga tipe array tadi.

### 6.3.2 Indexed Array

Ada dua cara untuk menghasilkan *indexed array* yang dapat dilihat pada gambar 6.15 dan 6.16. Untuk cara pada gambar 6.15 nomer index dari array \$cars otomatis dimulai dari 0.

```

$cars=array("Volvo", "BMW", "Toyota");

```

Gambar 6.15 Cara pertama menghasilkan *indexed array*

```

$cars[0]="Volvo";
$cars[1]="BMW";
$cars[2]="Toyota";

```

Gambar 6.16 Cara kedua menghasilkan *indexed array*

Sedangkan untuk mencetak *indexed array* bisa dengan cara memanggil indexnya langsung seperti pada gambar 6.17 atau dengan memakai bantuan perulangan seperti pada gambar 6.18

```
<?php
$scars=array("Volvo","BMW","Toyota");
echo "I like " . $scars[0] . ", " . $scars[1] . " and " . $scars[2] . ".";
?>
```

Gambar 6.17 Mencetak *indexed array* dengan cara memanggil index

```
<?php
$scars=array("Volvo","BMW","Toyota");
$arrlength=count($scars);

for($x=0;$x<$arrlength;$x++) {
    echo $scars[$x];
    echo "<br>";
}
?>
```

Gambar 6.18 Mencetak *indexed array* dengan perulangan

Untuk mengetahui jumlah elemen / panjang *indexed array* maka kita bisa menggunakan fungsi `count()` yang contoh penggunaannya dapat dilihat pada gambar 6.19.

```
<?php
$scars=array("Volvo","BMW","Toyota");
echo count($scars);
?>
```

Gambar 6.19 Contoh penggunaan fungsi `count()`.

### 6.3.3 Associative Array

Seperti *indexed array*, ada dua cara pula untuk menghasilkan array yang *associative* seperti dapat dilihat pada gambar 6.20 dan 6.21.



```
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");
```

Gambar 6.20 Cara pertama menghasilkan *associative array*

```
$age['Peter']="35";  
$age['Ben']="37";  
$age['Joe']="43";
```

Gambar 6.21 Cara kedua menghasilkan *associative array*

Sedangkan untuk mencetak *associative array* bisa dengan cara memanggil index key nya langsung seperti pada gambar 6.22 atau dengan memakai bantuan perulangan seperti pada gambar 6.23.

```
<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
echo "Peter is " . $age['Peter'] . " years old."  
>>
```

Gambar 6.22 Mencetak *associative array* dengan cara memanggil index

```
<?php  
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");  
  
foreach($age as $x=>$x_value) {  
    echo "Key=" . $x . ", Value=" . $x_value;  
    echo "<br>";  
}  
>>
```

Gambar 6.23 Mencetak *associative array* dengan perulangan.

### 6.3.4 *Multidimensional Array*

Untuk modul ini untuk sementara kita batasi hanya sampai array 2 dimensi .

Contoh untuk memasukan data dari tabel 6.1 ke dalam array multi dimensi (2 dimensi) maka digunakanlah cara pada gambar 6.24

Name	Stock	Sold
Volvo	22	18
BMW	15	13
Saab	5	2
Land Rover	17	15

Tabel 6.1 Data Penjualan dan Stok Mobil

```
$cars = array
(
    array("Volvo",22,18),
    array("BMW",15,13),
    array("Saab",5,2),
    array("Land Rover",17,15)
);
```

Gambar 6.24 Contoh cara menginput data tabel 6.1 ke dalam array 2 dimensi

Sedangkan untuk mencetak hasil dari array \$cars, dapat digunakan dengan cara memanggil indexnya secara langsung seperti pada gambar 6.25 atau dengan bantuan perulangan seperti pada gambar 6.26.

```
<?php
echo $cars[0][0].": In stock: ".$cars[0][1].", sold: ".$cars[0][2]."<br>";
echo $cars[1][0].": In stock: ".$cars[1][1].", sold: ".$cars[1][2]."<br>";
echo $cars[2][0].": In stock: ".$cars[2][1].", sold: ".$cars[2][2]."<br>";
echo $cars[3][0].": In stock: ".$cars[3][1].", sold: ".$cars[3][2]."<br>";
?>
```

Gambar 6.25 Contoh cara mencetak array 2 dimensi dengan cara memanggil index

```
<?php
for ($row = 0; $row < 4; $row++) {
    echo "<p><b>Row number $row</b></p>";
    echo "<ul>";
    for ($col = 0; $col < 3; $col++) {
        echo "<li>".$cars[$row][$col]."</li>";
    }
    echo "</ul>";
}
?>
```

Gambar 6.26 Mencetak associative array 2 dimensi dengan perulangan.

### 6.3.5 Pengurutan Array

Ada beberapa cara di PHP untuk melakukan pengurutan pada array antara lain :

- sort() – mengurutkan array secara *ascending* (dari kecil ke besar)
- rsort() - mengurutkan array secara *descending* (dari besar ke kecil)
- asort() - mengurutkan associative arrays in *ascending* (dari kecil ke besar), berdasarkan nilai

- ksort() - mengurutkan associative arrays secara *ascending* (dari kecil ke besar), berdasarkan *index key*.
- arsort() - mengurutkan associative arrays secara *descending* (dari besar ke kecil), berdasarkan nilai.
- krsort() - mengurutkan associative arrays secara *descending* (dari besar ke kecil)), berdasarkan *index key*.

Adapun keistimewaan dari fungsi pengurutan di PHP adalah mampu mengurutkan tidak hanya angka namun juga huruf alphabet.

Contoh penggunaan dari fungsi pengurutan seperti yang sudah disebutkan di atas dapat dilihat mulai gambar 6.27 sampai dengan gambar 6.xx

```
<?php
$cars=array("Volvo","BMW","Toyota");
sort($cars);
?>
```

Gambar 6.27 Contoh penggunaan fungsi sort().

```
<?php
$numbers=array(4,6,2,22,11);
rsort($numbers);
?>
```

Gambar 6.28 Contoh penggunaan fungsi rsort().

```
<?php
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");
asort($age);
?>
```

Gambar 6.29 Contoh penggunaan fungsi asort().

```
<?php
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");
ksort($age);
?>
```

Gambar 6.30 Contoh penggunaan fungsi ksort().

```
<?php
$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");
arsort($age);
?>
```

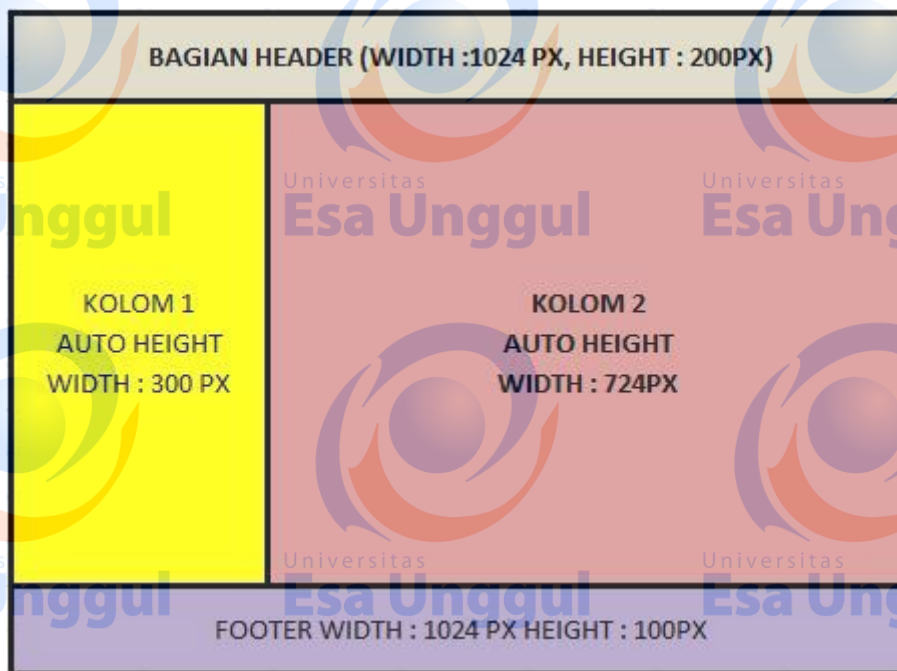
Gambar 6.31 Contoh penggunaan fungsi arsort().

```
<?php
$page=array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
ksort($page);
?>
```

Gambar 6.31 Contoh penggunaan fungsi krsort().

## 6.4 Latihan

Untuk latihan kali ini , layout dari website dapat dilihat pada gambar 6.32



Gambar 6.32 Layout

Untuk mengakomodasi layout pada gambar 6.32 buatlah file dengan nama latihan6.css dengan script seperti pada gambar 6.33 berikut

```
1 #layout {
2     float: none;
3     height: auto;
4     width: 1024px;
5 }
6 #header {
7     float: left;
8     height: 200px;
9     width: 1024px;
10    background-image: url(images/header.jpg);
11    background-repeat: no-repeat;
12    background-size: 1024px;
13 }
14 #kolom1 {
15     line-height: 30px;
16     background-color: #eeeeee;
17     height: auto;
18     width: 300px;
19     float: left;
20 }
21 }
22 #kolom2 {
23     width: 724px;
24     float: left;
25     background-color: #ffeee;
26 }
27 #footer {
28     background-color: black;
29     color: white;
30     clear: both;
31     text-align: center;
32 }
33 }
```

Gambar 6.33 Script latihan6.css

Kemudian buatlah file dengan nama latihan6.php , sehingga tampilan web bisa seperti pada gambar 6.34



Gambar 6.34 Tampilan dari latihan6.php jika dilihat di browser

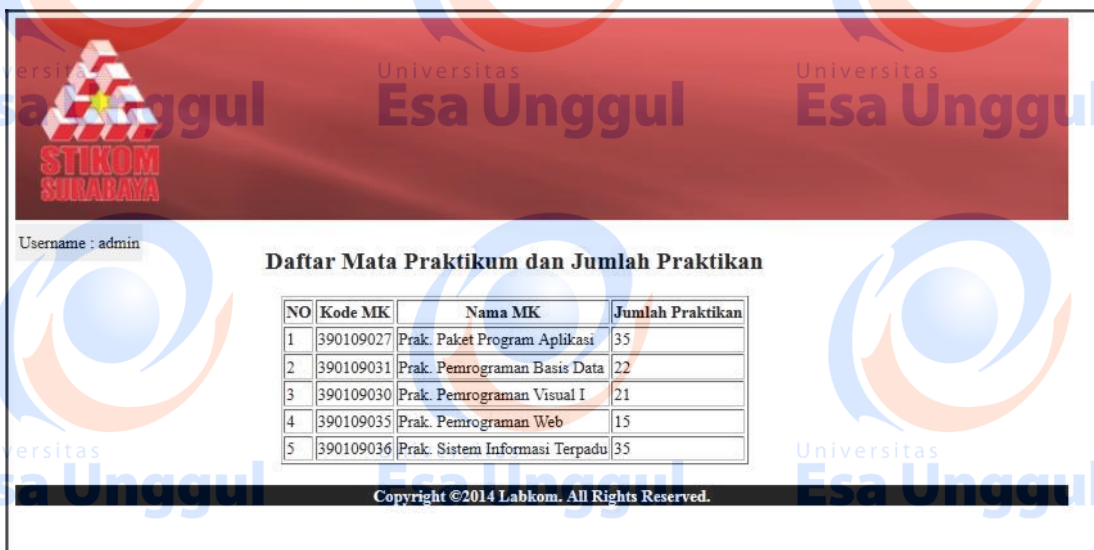


Untuk background header dapat anda download di :  
<http://192.168.100.3/images/Header.jpg>

Sedangkan untuk gambar yang menunjukkan ujian praktikum dapat anda download di : <http://192.168.100.3/UP.jpg>.

Kemudian username melakukan login ke dalam form username dan password, untuk diarahkan ke loginLabkom.php

Adapun jika username adalah “admin” dan password adalah “admin” maka tampilan dari loginLabkom.php akan menjadi seperti pada gambar 6.35



Username : admin

**Daftar Mata Praktikum dan Jumlah Praktikan**

NO	Kode MK	Nama MK	Jumlah Praktikan
1	390109027	Prak. Paket Program Aplikasi	35
2	390109031	Prak. Pemrograman Basis Data	22
3	390109030	Prak. Pemrograman Visual I	21
4	390109035	Prak. Pemrograman Web	15
5	390109036	Prak. Sistem Informasi Terpadu	35

Copyright ©2014 Labkom. All Rights Reserved.

Gambar 6.35 Tampilan jika username dan password adalah admin pada loginLabkom.php

Namun jika username salah maka tampilan akan menjadi seperti pada gambar 6.36



Gambar 6.36 Tampilan jika username dan password bukan admin pada loginLabkom.php

Untuk data mata praktikum dan jumlah praktikum nya harus ditampilkan dari array (silahkan pilih dari ketiga tipe array yang sudah dijelaskan) yang berasal dari tabel 6.2 berikut ini

Tabel 6.2 Data MK dan Jumlah Mahasiswa

NO	KODE MK	NAMA MK	JUMLAH MAHASISWA
1	390109027	Prak. Paket Program Aplikasi	35
2	390109031	Prak. Pemrograman Basis Data	22
3	390109030	Prak. Pemrograman Visual I	21
4	390109035	Prak. Pemrograman Web	15
5	390109036	Prak. Sistem Informasi Terpadu	5



MODUL 7  
PHP + MYSQL PART 1  
(DATABASE CONNECTION + SELECT + LOGIN DENGAN SESSION)

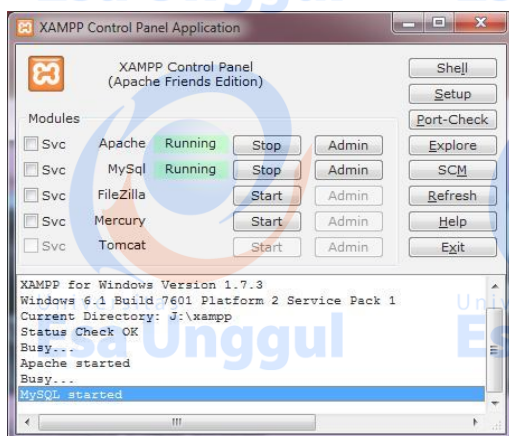
### 7.1 Sekilas Tentang My SQL

My SQL merupakan RDBMS (*Relational Database Management System*) yang diciptakan pada tahun 1979 oleh Michael "Monty" Widenius, seorang *programmer* komputer asal Swedia. Monty mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi *low-level ISAM database engine* dengan *indexing*.

Pada saat itu Monty bekerja pada perusahaan bernama TcX di Swedia, (Sumber : <http://id.wikipedia.org/wiki/MySQL>). Adapun keistimewaan dari PHP dan MySQL adalah keduanya merupakan bahasa pemrograman web dan RDBMS yang *cross-platform* (dapat dijalankan pada sistem operasi Linux dan Windows).

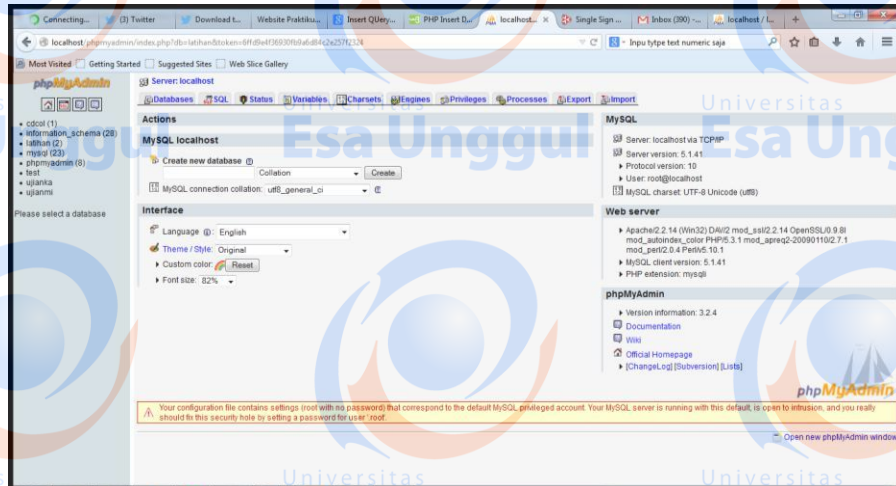
### 7.2 Membuat *Database* dan Tabel di MySQL

Karena untuk praktikum pemrograman Web saat ini menggunakan XAMPP, maka untuk masuk ke dalam area pembuatan *database*, kita harus menyalakan terlebih dahulu MySQL service dari XAMPP, seperti pada gambar 7.1.



Gambar 7.1 Xampp Control Panel jika MySQL Service Dinyalakan

Setelah service dinyalakan kemudian bukalah *web browser* dan ketikkan URL <http://localhost/phpmyadmin> pada text *url address* sehingga muncul tampilan seperti pada gambar 7.2

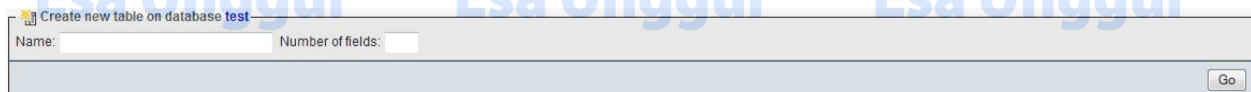


Gambar 7.2 Tampilan dari PHPMyAdmin di *web browser*

Kemudian untuk membuat *database*, anda bisa melakukannya dengan mengisi nama *database* pada kotak teks yang di atasnya terdapat tulisan **Create new database** dan kemudian tekanlah tombol **Create**.

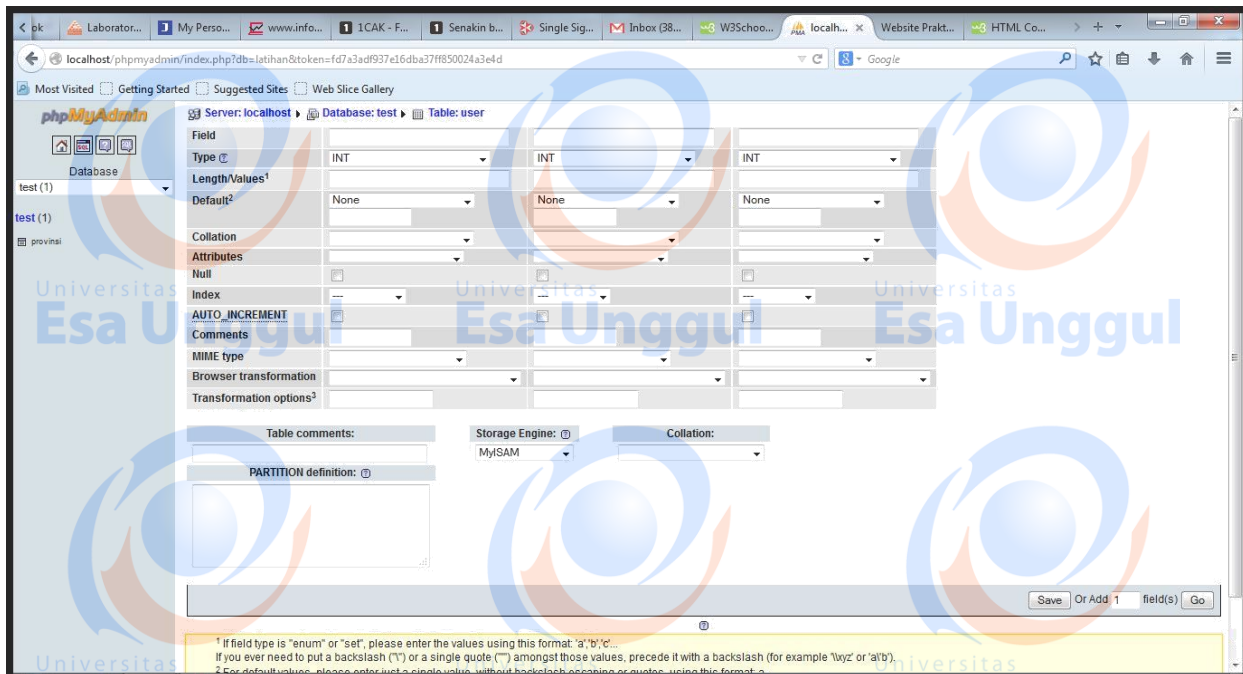
Sesudah itu untuk membuat tabel anda bisa melakukan klik kiri pada nama *database* yang sudah anda buat yang tertera di sebelah kiri layar. Kemudian di kotak teks yang di atasnya terdapat tulisan **Create new table on database <nama database>**, buatlah tabel beserta jumlah kolom yang terdapat di situ lalu tekan tombol **Go**. Jangan kuatir apabila salah menginputkan jumlah kolom, karena anda masih bisa menambah atau mengurangi jumlah kolom sesuai dengan kebutuhan.

Beberapa langkah untuk membuat tabel dapat anda lihat pada gambar 7.3 hingga gambar 7.4



Gambar 7.3 Bagian untuk mengisi nama tabel dan jumlah kolom





Gambar 7.4 Tampilan untuk membuat kolom pada tabel

Apabila sudah mendefinisikan tabel pada tampilan seperti pada gambar 7.4 maka tekan lah tombol **Save**.

### 7.3 Sintax Koneksi PHP ke Database MySQL

Seperti yang sudah disebutkan pada bagian 7.1, selain keistimewaan dari PHP dan MySQL yang *cross-plaftrom* terdapat keistimewaan lain yang juga ada di antara keduanya, yakni keduanya saling berelasi dalam syntax untuk melakukan koneksi atau dengan kata lain PHP memiliki syntax-syntax khusus yang langsung terintegrasi dengan MySQL.

```

1  <?php
2
3  //Membuat database MySQL
4  $nama_db = "latihan";
5  $server = "localhost";
6  $userserver = "root";
7  $password = "";
8
9  $sambung = mysql_connect($server,$userserver,$password);
10
11 mysql_select_db($nama_db,$sambung) or die("Koneksi ke $nama_db gagal");
12
13  ?>
14

```

Gambar 7.5 Syntax untuk ke database MySQL



Untuk melakukan koneksi ke *database* MySQL, perhatikan kode syntax pada gambar 7.5, pada gambar 7.5 terdapat bagian syntax yang bernama **mysql\_connect()**, yang mana parameter yang dibutuhkan adalah nama *server* yang direpresentasikan dengan variabel `$server`, kemudian *username* yang terdapat pada server MySQL yang direpresentasikan dengan variabel `$userserver` dan yang terakhir adalah *password* dari user. name tersebut yang direpresentasikan dengan variabel `$password`.

Kemudian terdapat pula bagian syntax yang bernama **mysql\_select\_db()**, yang mana parameter yang dibutuhkan antara lain adalah nama *database* yang direpresentasikan dengan variabel `$nama_db`, dan parameter berikutnya adalah pengalamatan koneksi dengan **mysql\_connect()** yang direpresentasikan dengan variabel `$sambung`.

Sedangkan syntax **die("koneksi ke \$nama\_db gagal")** digunakan untuk mematikan proses keseluruhan dan mencetak tulisan "koneksi ke <nama database> gagal"

Biasanya syntax dari gambar 7.5 akan disimpan ke dalam 1 file tersendiri untuk kemudian di *include* kan dalam file **<nama file>.php** untuk mengambil koneksi.

#### 7.4 Sintax PHP untuk Menampilkan Data dari Tabel

```
<h2> Daftar Mata Praktikum dan Jumlah Praktikan </h2>
<?php
$stampil = "select * from mk";
$sq_buat = mysql_query($stampil);
?>

<table border=1>
<tr>
<th>No.</th>
<th>Kode</th>
<th>Mata Praktikum</th>
<th>Jumlah Praktikan</th>
</tr>
<?php
$х = 1;
while($row = mysql_fetch_array($sq_buat)){
echo "<tr><td align='right'>".$х."</td>";
echo "<td align='right'>".$row['kode_mk']."</td>";
echo "<td align='left'>".$row['nama_mk']."</td>";
echo "<td align='right'>".$row['jumlah_praktikan']."</td></tr>";
$х++;
}
?>

</table>
```

Gambar 7.6 Contoh syntax untuk menginputkan data ke dalam tabel

Perhatikan contoh penggalan syntax pada gambar 7.6 !! Contoh syntax pada gambar 7.6 adalah penggalan syntax untuk menampilkan seluruh isi dari tabel MK ke dalam tiap baris dari sebuah tabel di HTML. Adapun untuk menampilkan seluruh isi tabel MK digunakan perulangan while dan **mysql\_fetch\_array ()**, yang mana fungsi **mysql\_fetch\_array**, menerima parameter dari **mysql\_query** yang direpresentasikan dengan variabel **\$q\_buat**.

Selain dengan menggunakan **mysql\_fetch\_array()**. Cara lain untuk menampilkan data dari sebuah tabel di *database* untuk ditampilkan ke HTML bisa dengan menggunakan **mysql\_fetch\_row()**. Adapun untuk contoh syntaxnya dapat dilihat pada gambar 7.7.

```
<h2> Daftar Mata Praktikum dan Jumlah Praktikan </h2>
<?php
$tampil = "select * from mk";
$q_buat = mysql_query($tampil);
?>
<table border=1>
<tr>
<th>No.</th>
<th>Kode</th>
<th>Mata Praktikum</th>
<th>Jumlah Praktikan</th>
</tr>
<?php
$x = 1;
while($row = mysql_fetch_row($q_buat)){
    echo("<tr><td align='right'>".$x."</td>");
    echo("<td align='right'>".$row[0]."</td>");
    echo("<td align='left'>".$row[1]."</td>");
    echo("<td align='right'>".$row[2]."</td></tr>");
    $x++;
}
?>
</table>
```

Gambar 7.7 Contoh penggunaan **mysql\_fetch\_row** untuk menampilkan data ke dalam tabel

## 7.5 Session PHP untuk Login

Fungsi session dari PHP diawali dengan fungsi **session\_start()**. Variabel session di PHP termasuk ke dalam variabel global : **\$\_SESSION**.

Contoh penggunaan PHP Session dapat dilihat pada gambar 7.8.

```
<?php
// Start the session
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// Set session variables
$_SESSION["favcolor"] = "green";
$_SESSION["favanimal"] = "cat";
echo "Session variables are set.";
?>

</body>
</html>
```

Gambar 7.8 Contoh penggunaan session di PHP

Fungsi session biasanya digunakan untuk fungsi login, dimana *username* dan password diambil dari database. Contoh sintax dari fungsi login dapat dilihat pada gambar 7.9

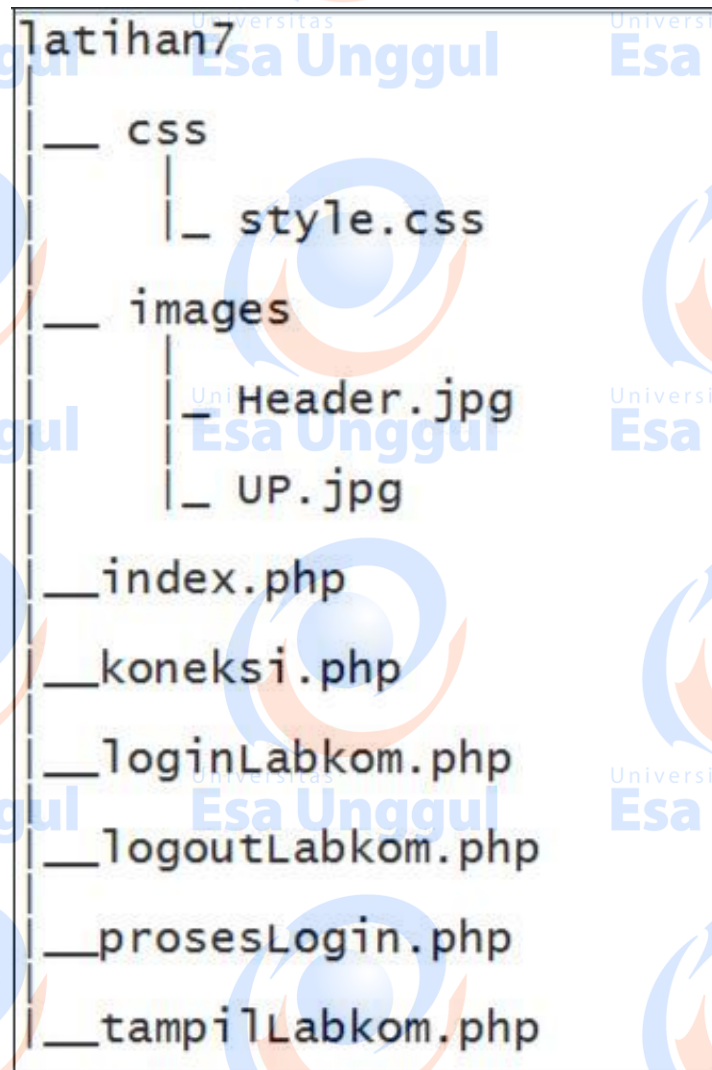
```
1 <?php
2 // Start the session
3 session_start();
4 include ("koneksi.php");
5 $user = $_POST['username'];
6 $pass = $_POST['password'];
7
8 $stampil = "select count(kode_user) from user WHERE nama_user = \"\".$user.\"\" AND password_user = \"\".$pass.\"\"";
9 $q_buat = mysql_query($stampil,$sambung);
10 $jml = mysql_fetch_row($q_buat);
11
12 if ($jml[0] >= 1){
13     $_SESSION["username"] = $user;
14     $_SESSION["password"] = $pass;
15     echo "<script>>window.location.href = \"http://localhost/latmodul7/tampilLabkom.php\" </script>";
16 }
17 elseif ($jml[0] < 1){
18     echo "<script>alert('Username atau password anda salah coba anda inputkan lagi')</script>";
19     echo "<script>>window.location.href = \"http://localhost/latmodul7/index.php\" </script>";
20 }
21 ?>
22
```

Gambar 7.9 Contoh penggunaan session di PHP untuk login

Pada gambar 7.9, di *line* nomer 15,18 dan 19 perlu diperhatikan bahwa PHP dapat digabungkan dengan javascript dengan terlebih dahulu menggunakan **echo** sebagai media untuk menuliskan script javascript. Fungsi javascript **window.location.href** digunakan untuk memindahkan lokasi halaman web (URL) ke URL tertentu. Adapun untuk menghapus session digunakan dua fungsi yakni **session\_unset()** untuk menghilangkan variabel **\$\_SESSION** dan berikutnya adalah **session\_destroy()**

## 7.6 Latihan

Untuk latihan pada modul 7 berikut ini, buatlah sebuah folder di htdocs dengan nama latihan7, kemudian susunan file-file dan folder tambahan (css dan images) yang akan kita buat dapat dilihat pada gambar 7.10.



Gambar 7.10 Skema file dengan ekstensi .php dan folder css dan images pada folder latihan7



```

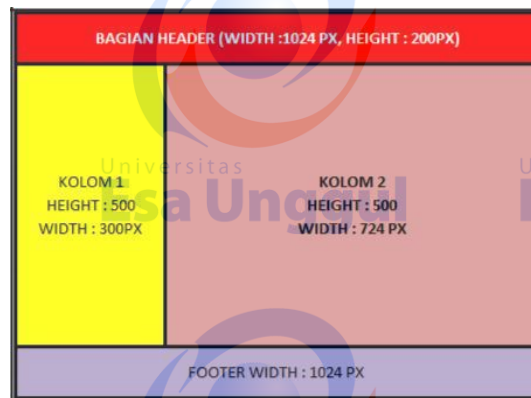
1  #layout {
2      float: none;
3      height: auto;
4      width: 1024px;
5  }
6  #header {
7      float: left;
8      height: 200px;
9      width: 1024px;
10     background-image: url(../images/Header.jpg);
11     background-repeat: no-repeat;
12     background-size: 1024px;
13 }
14 #kolom1 {
15     line-height: 30px;
16     background-color: #99FF99;
17     height: 500px;
18     width: 300px;
19     float: left;
20     ...
21 }
22 #kolom2 {
23     width: 724px;
24     float: left;
25     height: 500px;
26     background-color: #FFFD66;
27 }
28 #footer {
29     background-color: black;
30     color: white;
31     clear: both;
32     text-align: center;
33     width: 1024px;
34     height: auto;
35 }

```

Gambar 7.11 Script style.css

Buatlah sebuah *script* CSS dengan nama *style.css* dengan isi seperti pada gambar 7.11 simpanlah pada folder *css*.

Hasil dari *script* tersebut akan digunakan untuk menghasilkan *layout* website seperti yang ditunjukkan pada gambar 7.12



Gambar 7.12 Layout web dari CSS

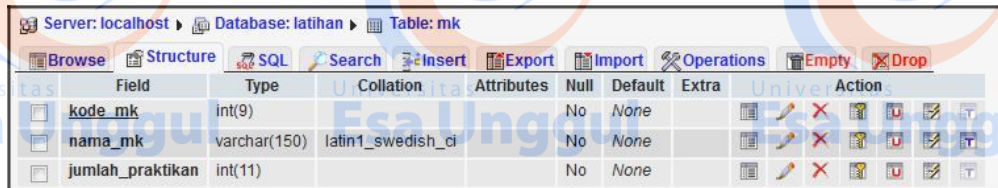


Untuk background header dapat anda download di :  
<http://192.168.100.3/images/Header.jpg>

Sedangkan untuk gambar yang menunjukkan ujian praktikum dapat anda download di : <http://192.168.100.3/UP.jpg>.

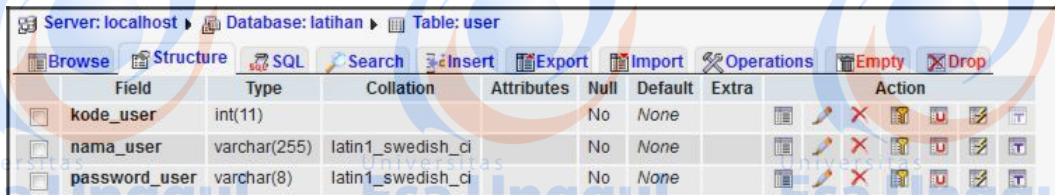
Simpanlah kedua gambar tadi ke dalam folder images.

Kemudian nyalakan *service* Apache dan MySQL dari XAMPP komputer anda, lalu masuklah ke PHPMyAdmin. Buatlah database dengan nama **latihan** dan kemudian buatlah 2 buah tabel yakni **user** dan **MK** dengan format tabel seperti pada gambar 7.13 dan 7.14.



Field	Type	Collation	Attributes	Null	Default	Extra	Action
kode_mk	int(9)			No	None		
nama_mk	varchar(150)	latin1_swedish_ci		No	None		
jumlah_praktikan	int(11)			No	None		

Gambar 7.13 Struktur tabel MK



Field	Type	Collation	Attributes	Null	Default	Extra	Action
kode_user	int(11)			No	None		
nama_user	varchar(255)	latin1_swedish_ci		No	None		
password_user	varchar(8)	latin1_swedish_ci		No	None		

Gambar 7.14 Struktur tabel User

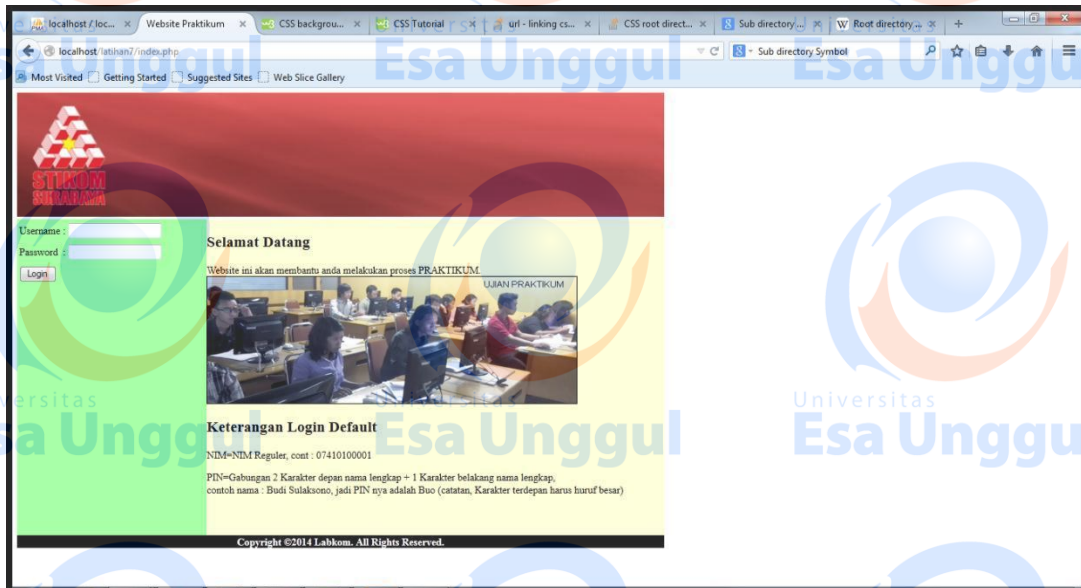
Setelah membuat kedua tabel yakni MK dan User, kemudian isikan data-data pada gambar 7.15 ke tabel MK dan data pada gambar 7.16 ke dalam tabel User

Gambar 7.15 Isi data tabel MK

kode_user	nama_user	password_user
1	admin	admin
2	biasa	biasa

Gambar 7.16 Isi data tabel User

Usai mengisi tabel, buatlah file dengan nama **index.php** di folder latihan7 dengan tampilan seperti pada gambar 7.17, lalu buatlah file dengan nama **koneksi.php** juga di folder latihan7 yang *script* nya dapat dilihat pada gambar 7.5.



Gambar 7.17 Tampilan index.php

Adapun *script* dari index.php dapat anda lihat pada gambar 7.18

```

1 <!--DOCTYPE
2 <html>
3 <head>
4 <title>Website Praktikum </title>
5
6 <link href="/css/style.css" rel="stylesheet" type="text/css" />
7 </head>
8
9 <body >
10 <div id="layout">
11 <div id="header"> </div>
12 <div id="kolom1">
13
14 <form action="proseslogin.php" method="post">
15 <table width="100%">
16 <tr>
17 <td>username </td> <td> </td> <input type="text" name="username"> </td>
18 </tr>
19 <tr>
20 <td>password </td> <td> </td> <input type="text" name="password"> </td>
21 </tr>
22 <tr>
23 <td colspan="3" style="text-align: center;>
24 <input type="submit" value="login"> </td> <td colspan="3" style="text-align: center;> </td>
25 </tr>
26 </table>
27 </form>
28 </div>
29
30 <div id="kolom2">
31 <td> Selamat Datang </td>
32 Website ini akan membantu anda melakukan proses PRAKTIKUM.
33 
34 <td> Keterangan Login Default </td>
35 <p> NIM=NIM Reguler, cont : 0741010001 </p>
36 <p>
37 PIN=Gabungan 2 Karakter depan nama lengkap + 1 Karakter belakang nama lengkap. <br/>
38 contoh name : Budi Sulaksana, jadi PIN nya adalah Bus (gabutan, Karakter terdepan harus huruf besar) </p>
39 </div>
40
41 <div id="Footer"><p> Copyright &copy;2014 Labkom. All Rights Reserved. </p>
42 </div>
43 </div>
44 </body>
45 </html>

```

Gambar 7.18 Script index.php

Di sini *user* akan melakukan login dengan mengisi terlebih dahulu *text field username* dan password, apabila sudah terisi dan *user* menekan tombol login maka tampilan akan diarahkan ke tampilLabkom.php, namun sebelum ke tampilLabkon.php diadakan pengecekan *login user* dari pengisian *username* dan password pada tabel User, jika cocok maka boleh masuk ke tampilLabkom.php namun jika tidak akan dikembalikan ke index.php. Proses pengecekan login user dibuat pada file prosesLogin.php yang scriptnya dapat dilihat pada gambar 7.19

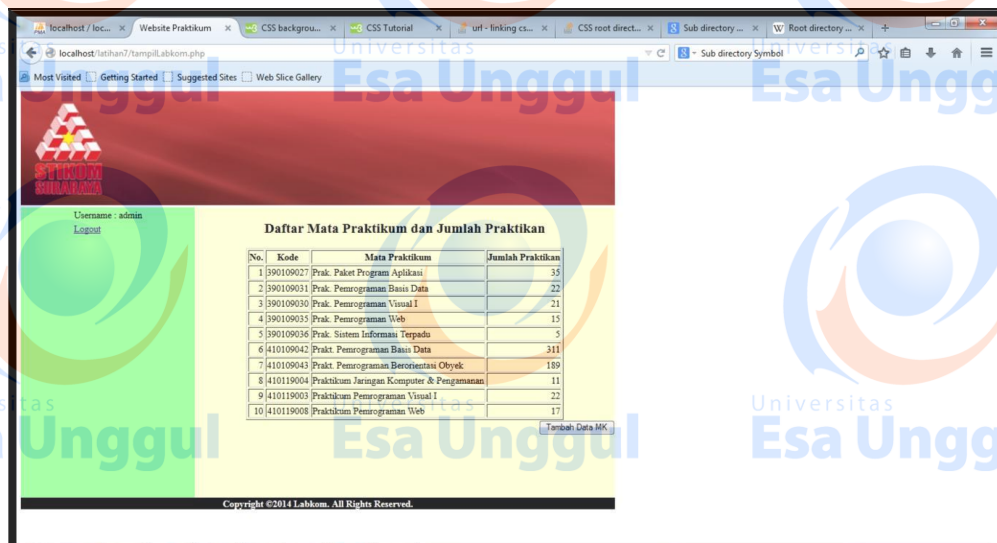
Tampilan dari tampilLabkom.php dapat dilihat pada gambar 7.20, sedangkan *scriptnya* dapat dilihat pada gambar 7.21 dan gambar 7.22

```

1 <?php
2 // Start the session
3 session_start();
4 include ("koneksi.php");
5
6 $user = $_POST['username'];
7 $pass = $_POST['password'];
8
9 $stampil = "select count(kode_user)from user:WHERE nama_user = \"\".$user.\"\" AND:password_user = \"\".$pass.\"\"";
10 $q_buat = mysql_query($stampil,$sambung);
11 $jml = mysql_fetch_row($q_buat);
12 if( ($jml[0] < 1) && (!ISSET($_SESSION["username"]) && !ISSET($_SESSION["password"])))
13 {
14 echo "<script>alert('Anda Harus Login Terlebih Dahulu')</script>";
15 echo "<script>>window.location.href = \"http://localhost/latihan8/index.php\" </script>";
16 }
17 if ($jml[0] >= 1){
18     $_SESSION["username"] = $user;
19     $_SESSION["password"] = $pass;
20     echo "<script>>window.location.href = \"http://localhost/latihan8/tampilLabkom.php\" </script>";
21 }
22 elseif ($jml[0] < 1){
23     echo "<script>alert('Username atau password anda salah coba anda inputkan lagi')</script>";
24     echo "<script>>window.location.href = \"http://localhost/latihan8/index.php\" </script>";
25 }
26 }
27 ?>

```

Gambar 7.19 Script prosesLogin.php



Gambar 7.20 Tampilan tampilLabkom.php



```

1 <?php
2 session_start();
3 include ("koneksi.php");
4 if($_SESSION["username"] == "")
5 {
6     echo "<script>alert('Anda Harus Login Terlebih Dahulu')</script>";
7     echo "<script>>window.location.href = \"http://localhost/latmodul7/index.php\" </script>";
8 }
9 }
10 <?>
11 <!DOCTYPE
12 <html>
13 <head>
14 <title>Website Praktikum </title>
15 <link href="/css/style.css" rel="stylesheet" type="text/css" />
16 </head>
17 <body>
18 <div id="layout" align="center">
19 <div id="header"></div>
20 <div id="kolom1">
21 <table>
22 <tr>
23 <td>Username </td> <td> :</td><td> <?php echo $_SESSION["username"];
24 <?> </td>
25 </tr>
26 <tr>
27 <td> <a href="logoutLabkom.php">Logout</a>
28 </td>
29 </tr>
30 </table>
31 </div>
32 </div>
33 <div id="kolom2">
34 <div align="center">
35 <h2> Daftar Mata Praktikum dan Jumlah Praktikan </h2>

```

Gambar 7.21 Script tampilLabkom.php

```

39 <?php
40 $tampil = "select * from mk";
41 $q_buat = mysql_query($tampil);
42 <?>
43 <table border=1>
44 <tr>
45 <th>No.</th>
46 <th>Kode</th>
47 <th>Mata Praktikum</th>
48 <th>Jumlah Praktikan</th>
49 </tr>
50 <tr>
51 <td>
52 <?php
53 $x = 1;
54 while($row = mysql_fetch_array($q_buat)){
55     echo "<tr><td align='right'>".$x."</td>";
56     echo "<td align='right'>".$row['kode_mk']."</td>";
57     echo "<td align='left'>".$row['nama_mk']."</td>";
58     echo "<td align='right'>".$row['jumlah_praktikan']."</td></tr>";
59     $x++;
60 }
61 <?>
62 </table>
63 <form action="inputMK.php" method="post">
64 <div align="right"> <input type="submit" value="Tambah Data MK"> </div>
65 </form>
66 </div>
67 </div>
68 <br>
69 </div>
70 <div id="footer"><b> Copyright &copy;2014 Labkom. All Rights Reserved. </b>
71 </div>
72 </div>
73 </body>
74 </html>

```

Gambar 7.22 Lanjutan script tampilLabkom.php



Di tampilLabkom.php apabila *hyperlink* logout ditekan maka akan terjadi proses kembali ke halaman index.php, untuk itu jika *hyperlink* logout ditekan maka akan diarahkan terlebih dahulu ke logoutLabkom.php, adapun *script* logoutLabkom.php dapat dilihat pada gambar 7.23

```

1 <?php
2 // Start the session
3 session_start(); // Must start a session before destroying it
4
5 if (isset($_SESSION))
6 {
7     unset($_SESSION);
8     session_unset();
9     session_destroy();
10 }
11 echo "<script>>window.location.href = \"http://localhost/latihan7/index.php\" </script>";
12 ?>
13

```

Gambar 7.23 *script* logoutLabkom.php

Apabila semua *file* sudah dibuat *script* nya dan gambar-gambar pendukung sudah ditaruh pada folder, coba lah anda cek dulu apakah susunan *file* dan folder sudah seperti pada gambar 7.24 atau belum?



Gambar 7.24 Hasil pembuatan file untuk localhost/latihan7

Setelah menaruh file sesuai dengan susunan pada gambar 7.11 dan gambar 7.24, lakukan hal-hal sebagai berikut :

1. Ketikkan `localhost/latihan7/prosesLogin.php` dan cobalah amati apabila masih bisa tampil tanpa melakukan login berarti masih ada yang salah pada *script* anda, maka cobalah perbaiki !!
2. Ketikkan `localhost/latihan7/tampilLabkom.php` dan cobalah amati apabila masih bisa tampil tanpa melakukan login berarti masih ada yang salah pada *script* anda, maka cobalah perbaiki !!
3. Jalankan program di *web browser* anda dan ketikkan `localhost/latihan7/index.php` pada URL address dan cobalah lakukan login

## MODUL 8

### PHP + MYSQL PART 2

#### (INSERT + UPDATE + DELETE)

#### 8.1 Sintax PHP untuk Insert Data ke Tabel

Untuk melakukan *insert* ke dalam tabel , perhatikan contoh sintax pada gambar 8.1

```
1 <?php
2 // Start the session
3 session_start();
4 include ("koneksi.php");
5 if($_SESSION["username"] == "")
6 {
7     echo "<script>alert('Anda Harus Login Terlebih Dahulu')</script>";
8     echo "<script>>window.location.href = \"http://localhost/latmodul7/index.php\" </script>";
9 }
10 $kode = $_GET['txtKodeMK'];
11 $nama = $_GET['txtNamaMK'];
12 $jml = $_GET['txtJumlah'];
13
14 $hitung = "SELECT COUNT(KODE_MK) FROM MK WHERE KODE_MK = '". $kode. "'";
15 $q_hitung = mysql_query($hitung);
16 $jumlah = 0;
17 while($sarr = mysql_fetch_row($q_hitung))
18 {
19     $jumlah = $sarr[0];
20 }
21 if ($jumlah > 0)
22 {
23     echo "<script>alert('Data Dengan kode mk ". $kode. " sudah ada di tabel MK')</script>";
24     echo "<script>>window.location.href = \"http://localhost/latihan8/tampilLabkom.php\" </script>";
25 }
26 else
27 {
28     $masuk= "INSERT INTO MK VALUES ('". $kode. "', '". $nama. "', '". $jml. "')";
29     $q_buat = mysql_query($masuk, $sambung);
30
31     echo "<script>alert('Input Data Berhasil')</script>";
32     echo "<script>>window.location.href = \"http://localhost/latihan8/tampilLabkom.php\" </script>";
33 }
34 }
35 >>
```

Gambar 8.1 Contoh sintax untuk menginputkan data ke dalam tabel

Seperti yang telah dijelaskan pada modul ke 7 pada bagian 7.4, file yang berisi informasi untuk koneksi ke database bisa di *include* contohnya pada gambar 8.1, yakni pada bagian syntax **include("koneksi.php")**. Setelah itu hal yang harus dilakukan adalah membuat query insert pada untuk menjadi parameter yang diinputkan ke dalam **mysql\_query(<query>)** yang direpresentasikan dengan variabel \$masuk.

Sedangkan nilai yang diterima berasal dari form yang diperoleh dari HTTP request, untuk contoh pada gambar 8.1, nilai-nilai yang diinputkan ke dalam tabel MK diterima dari metode \$\_GET yang berasal dari form.

Penjelasan untuk baris kode nomer 14-34 adalah sebelum proses *insert* ke dalam tabel MK dilakukan pengecekan dulu apakah kode mk yang berasal dari variabel \$kode sudah pernah diinputkan sebelumnya ke dalam tabel MK, jika tidak maka proses *insert* dapat terus dilakukan dan halaman akan berpindah ke tampilLabkom.php, namun jika tidak maka proses *insert* ke tabel MK tidak dapat dilakukan dan halaman akan juga kembali ke tampilLabkom.php.

Sedangkan baris kode nomer 1-9 adalah penggunaan session untuk mencegah agar user tidak memaksa masuk ke dalam localhost/<nama file.php> sebelum login terlebih dahulu.

## 8.2 Sintax PHP untuk Update Data ke Tabel

Untuk melakukan *update* data ke dalam tabel , perhatikan contoh sintax pada gambar 8.2

```
1 <?php
2 // Start the session
3 session_start();
4 include ("koneksi.php");
5 if($_SESSION["username"] == "")
6 {
7 echo "<script>alert('Anda Harus Login Terlebih Dahulu')</script>";
8 echo "<script>>window.location.href = \"http://localhost/latmodule8/index.php\" </script>";
9 }
10 $kodeMK = $_GET['txtKodeMK'];
11 $namaMK = $_GET['txtNamaMK'];
12 $jumlah = $_GET['txtJumlah'];
13
14 $ubah= "UPDATE mk
15 SET nama_mk = '$namaMK.',
16 jumlah_praktikan = '$jumlah.'
17 WHERE kode_mk = '$kodeMK.'";
18 $q_upd = mysql_query($ubah,$sambung);
19
20 echo "<script>alert('Ubah Data Kode MK = '$kodeMK.'" Berhasil')</script>";
21 echo "<script>>window.location.href = \"http://localhost/latihan8/tampilLabkom.php\" </script>";
22
23 ?>
24
```

Gambar 8.2 Contoh sintax untuk mengupdate data ke dalam tabel

Baris kode pada nomer 3-9 digunakan untuk melakukan pengecekan apakah *user* sudah melakukan login atau belum, jika belum maka user akan dipaksa untuk masuk ke halaman index.php



Selanjutnya untuk baris kode nomer 10-21 adalah setelah variabel \$kodeMK, \$namaMK dan \$jumlah diisi nilai yang diperoleh dari inputan melalui form dari metode \$\_GET , maka langkah selanjutnya adalah membuat variabel \$ubah berisi sintax query update ke tabel MK untuk dieksekusi dengan **mysql\_query(\$ubah,\$sambung)**. Selanjutnya ditampilkan sebuah pesan melalui alert Javascript bahwa pengubahan data dari kode mk yang berasal dari variabel \$kodeMK berhasil dan halaman web akan diarahkan ke tampilLabkon.php

### 8.3 Sintax PHP untuk Delete Data ke Tabel

```
1 <?php
2 // Start the session
3 session_start();
4 include ("koneksi.php");
5 if($_SESSION["username"] == "")
6 {
7     echo "<script>alert('Anda Harus Login Terlebih Dahulu')</script>";
8     echo "<script>>window.location.href = \"http://localhost/latmodul8/index.php\" </script>";
9 }
10 $kode = $_GET['kodeMK'];
11 $hapus= "DELETE FROM MK WHERE kode_mk = '". $kode. "'";
12 $q_buat = mysql_query($hapus, $sambung);
13
14 echo "<script>alert('Hapus Data Dengan Kode MK = ". $kode. " Berhasil')</script>";
15 echo "<script>>window.location.href = \"http://localhost/latihan8/tampilLabkom.php\" </script>";
16 ?>
17
```

Gambar 8.3 Contoh sintax untuk mengdelete data ke dalam tabel

Untuk melakukan *delete* data ke dalam tabel , perhatikan contoh sintax pada gambar 8.3

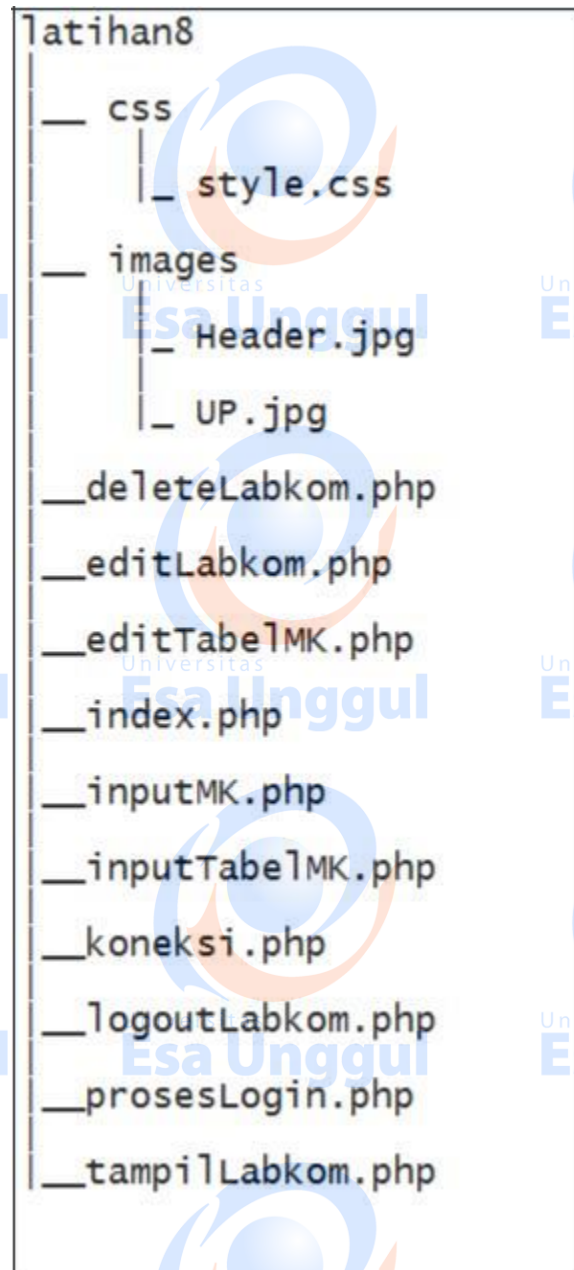
Pada gambar 8.3, baris kode nomer 3-9 digunakan untuk melakukan pengecekan apakah *user* sudah melakukan login atau belum, jika belum maka user akan dipaksa untuk masuk ke halaman index.php Selanjutnya untuk baris kode nomer 10-15 adalah setelah variabel \$kodeMK, diisi nilai yang diperoleh dari inputan melalui form dari metode \$\_GET , maka langkah selanjutnya adalah membuat variabel \$hapus berisi sintax query *delete* ke tabel MK yang setelah itu perintah tersebut dieksekusi dengan **mysql\_query(\$hapus,\$sambung)**.

Selanjutnya ditampilkan sebuah pesan melalui alert Javascript bahwa penghapusan data dari kode mk yang berasal dari variabel \$kodeMK berhasil dan halaman web akan diarahkan ke tampilLabkon.php



## 8.4 Latihan

Untuk latihan pada modul 8 berikut ini, buatlah sebuah folder di htdocs dengan nama latihan8, kemudian susunan file-file dan folder tambahan (css dan images) yang akan kita buat dapat dilihat pada gambar 8.4.



Gambar 8.4 Skema file dengan ekstensi .php dan folder css dan images pada folder latihan8

```

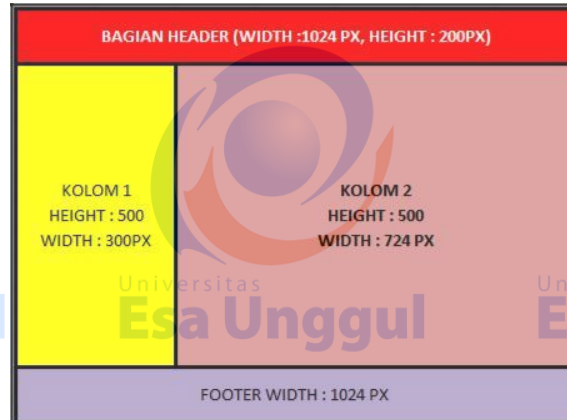
1  #layout {
2      float: none;
3      height: auto;
4      width: 1024px;
5  }
6  #header {
7      float: left;
8      height: 200px;
9      width: 1024px;
10     background-image: url(../images/Header.jpg);
11     background-repeat: no-repeat;
12     background-size: 1024px;
13 }
14 #kolom1 {
15     line-height:30px;
16     background-color:#99FF99;
17     height:500px;
18     width:300px;
19     float:left;
20 }
21 }
22 #kolom2{
23     width:724px;
24     float:left;
25     height:500px;
26     background-color:#FFFD6;
27 }
28 #footer {
29     background-color:black;
30     color:white;
31     clear:both;
32     text-align:center;
33     width: 1024px;
34     height : auto;
35 }

```

Gambar 8.5 *Script* style.css pada folder css

Buatlah sebuah *script* CSS dengan nama style.css dengan isi seperti pada gambar 8.5 simpanlah pada folder css

Hasil dari *script* tersebut akan digunakan untuk menghasilkan *layout* website seperti yang ditunjukkan pada gambar 8.6



Gambar 8.6 *Layout* web dari CSS

Untuk background header dapat anda download di :  
<http://192.168.100.3/images/Header.jpg>

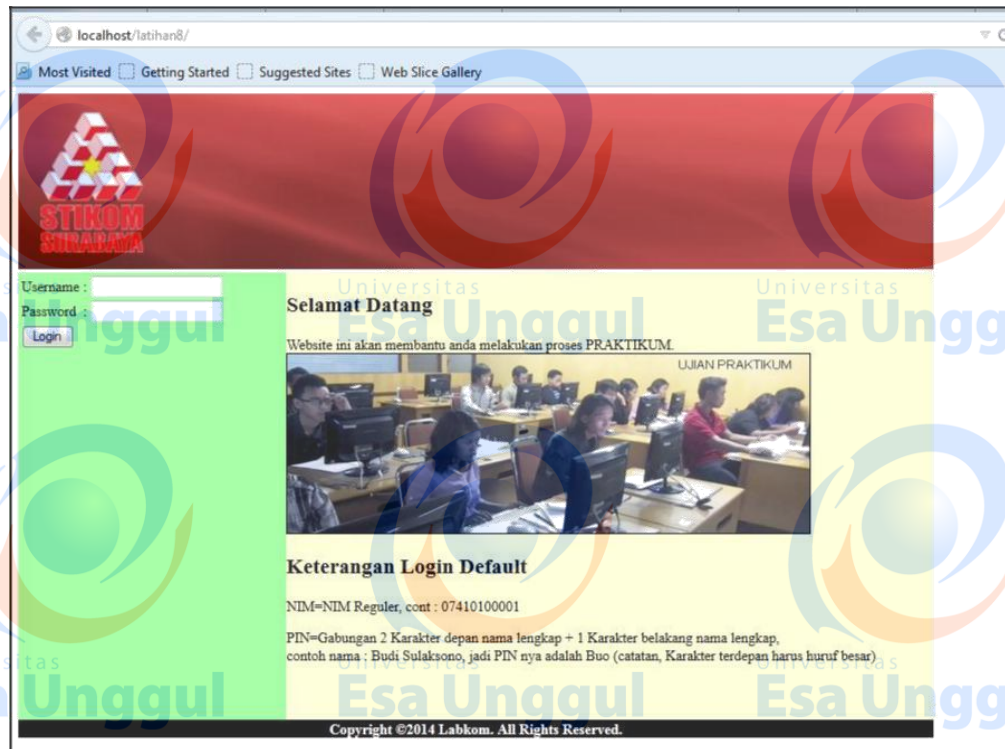
Sedangkan untuk gambar yang menunjukkan ujian praktikum dapat anda download di : <http://192.168.100.3/UP.jpg>.

Simpanlah kedua gambar tadi ke dalam folder images.

Langkah selanjutnya adalah membuat *database* dengan nama latihan dan mengisi dengan *database* latihan tersebut dengan tabel user dan mk yang dibuat sesuai dengan cara pada latihan modul 7. Berikutnya jangan lupa untuk mengisi dengan data yang ada di gambar

Setelah anda membuat *database* dan mengisi tabel dengan data yang ada pada gambar 7.15 dan 7.16 pada modul 8.

Usai mengisi tabel buatlah file dengan nama **index.php** di folder latihan8 dengan tampilan pada browser seperti pada gambar 8.7, lalu buatlah file dengan nama **koneksi.php** juga di folder latihan8 yang *script* nya dapat dilihat pada gambar 8.8.



Gambar 8.7 Tampilan dari localhost/latihan8/index.php

```

1 <?php
2
3 $nama_db = "latihan";
4 $server = "localhost";
5 $userserver = "root";
6 $password = "";
7
8 $sambung = mysql_connect($server,$userserver,$password);
9
10 mysql_select_db($nama_db,$sambung) or die("Koneksi ke $nama_db gagal");
11
12 >>
13

```

Gambar 8.8 koneksi.php

Sedangkan *script* untuk index.php dapat dilihat pada gambar 8.9 dan 8.10





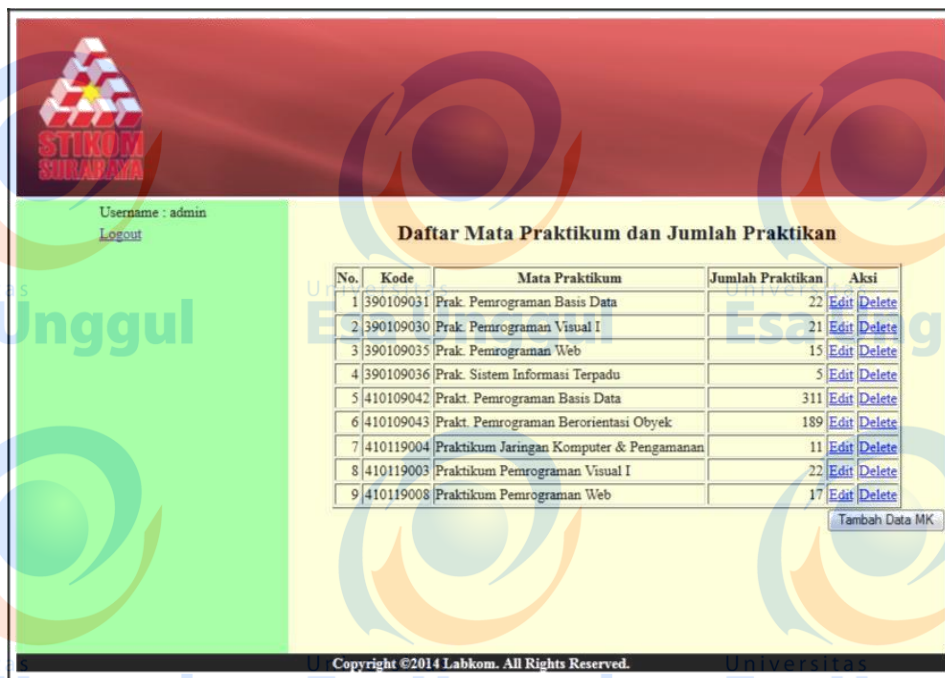
```
1 <?php
2 // Start the session
3 session_start();
4 include ("koneksi.php");
5
6 $user = $_POST['username'];
7 $pass = $_POST['password'];
8
9 $stampil = "select count(kode_user) from user WHERE nama_user = \"\".$user.\"\" AND password_user = \"\".$pass.\"\"";
10 $q_buat = mysql_query($stampil,$sambung);
11 $jml = mysql_fetch_row($q_buat);
12 if( ($jml[0] < 1) && (!ISSET($_SESSION["username"]) && !ISSET($_SESSION["password"])))
13 {
14 echo "<script>alert('Anda Harus Login Terlebih Dahulu!')</script>";
15 echo "<script>>window.location.href = \"http://localhost/latihan8/index.php\" </script>";
16 }
17 if ($jml[0] >= 1){
18 $_SESSION["username"] = $user;
19 $_SESSION["password"] = $pass;
20 echo "<script>>window.location.href = \"http://localhost/latihan8/tampilLabkom.php\" </script>";
21 }
22 elseif ($jml[0] < 1){
23 echo "<script>alert('Username atau password anda salah coba anda inputkan lagi')</script>";
24 echo "<script>>window.location.href = \"http://localhost/latihan8/index.php\" </script>";
25 }
26
27 ?>
```

Gambar 8.11 Script dari prosesLogin.php

Setelah user mengisi *username* serta *password* yang diketikan pada *textfield* maka berikutnya user menekan tombol Login yang akan diarahkan ke script **prosesLogin.php**, yang *scriptnya* dapat dilihat pada gambar 8.11

Usai proses login dilakukan halaman web akan berpindah ke *tampilLabkom.php*, dengan tampilan seperti pada gambar 8.12, sedangkan *scriptnya* dapat dilihat pada gambar 8.13 dan gambar 8.14





Gambar 8.12 Tampilan dari tampilLabkom.php

```

1  <?php
2  session_start();
3  include ("koneksi.php");
4  if($_SESSION["username"] == "")
5  {
6  echo "<script>alert('Anda Harus Login Terlebih Dahulu')</script>";
7  echo "<script>>window.location.href = '<http://localhost/labmodul8/index.php'</script>";
8  }
9  }
10 }
11 </DOCTYPE>
12 <html>
13 <head>
14 <title>Website Praktikum </title>
15 <link href="..<css/style.css" rel="stylesheet" type="text/css" />
16 </head>
17 <body>
18 <div id="layout" align="center">
19 <div id="header"></div>
20 <div id="kolom1">
21 <table>
22 <tr>
23 <td>
24 <td>Username </td> <td> </td><td> <?php echo $_SESSION["username"];
25 <td> </td>
26 </tr>
27 <tr>
28 <td> <a href="logoutLabkom.php">Logout</a>
29 </tr>
30 </table>
31 </div>
32 <div id="kolom2">
33 <div align="center">
34 <h2> Daftar Mata Praktikum dan Jumlah Praktikan </h2>
35 <?php
36 $stampil = "select * from mk";
37 $sql_buat = mysql_query($stampil);
38 }

```

Gambar 8.12 Script dari tampilLabkom.php

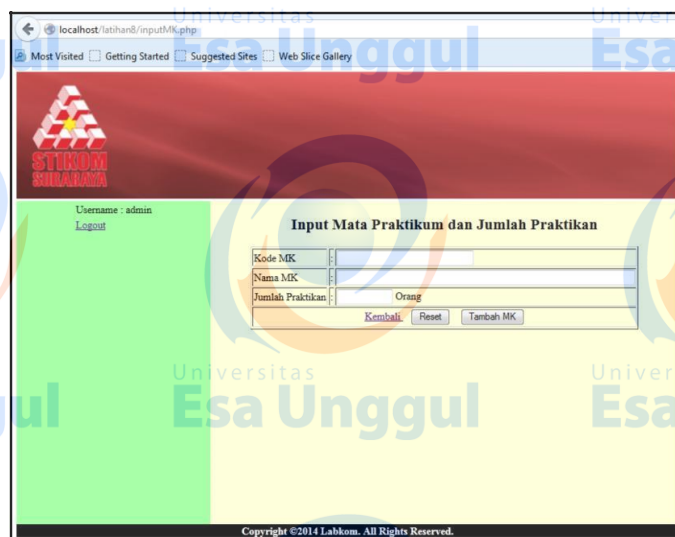
```

42 <table border=1>
43 <tr>
44 <th>No.</th>
45 <th>Kode</th>
46 <th>Mata Praktikum</th>
47 <th>Jumlah Praktikan</th>
48 <th colspan = "2">Aksi</th>
49 </tr>
50 <?php
51 $x = 1;
52 while($row = mysql_fetch_array($q_buat)){
53     echo ("<tr><td align=\\"right\\">". $x."</td>");
54     echo ("<td align=\\"right\\">". $row['kode_mk']."</td>");
55     echo ("<td align=\\"left\\">". $row['nama_mk']."</td>");
56     echo ("<td align=\\"right\\">". $row['jumlah_praktikan']."</td>");
57     echo ("<td align=\\"right\\"><a href=\\"editLabkom.php?kodeMK=". $row['kode_mk'].">Edit </a> </td>");
58     echo ("<td align=\\"right\\"><a href=\\"deleteTabelMK.php?kodeMK=". $row['kode_mk'].">Delete </a> </td>");
59     echo"</tr>";
60
61     $x++;
62 }
63 ?>
64 </table>
65
66 <form action="inputMK.php" method="post">
67 <div align = "right"> <input type="submit" value="Tambah Data MK"> </div>
68 </form>
69 </div>
70 </br>
71 </div>
72 <div id="footer"><b> Copyright &copy;2014 Labkom. All Rights Reserved. </b>
73 </div>
74 </div>
75 </body>
76 </html>
77

```

Gambar 8.13 Lanjutan script dari tampilLabkom.php

Apabila tombol “Tambah Data MK” ditekan maka halaman web akan merujuk ke inputMK.php dengan tampilan seperti pada gambar 8.14 dan untuk script nya dapat dilihat pada gambar 8.15 dan gambar 8.16



Gambar 8.14 Tampilan dari localhost/latihan8/inputMK.php

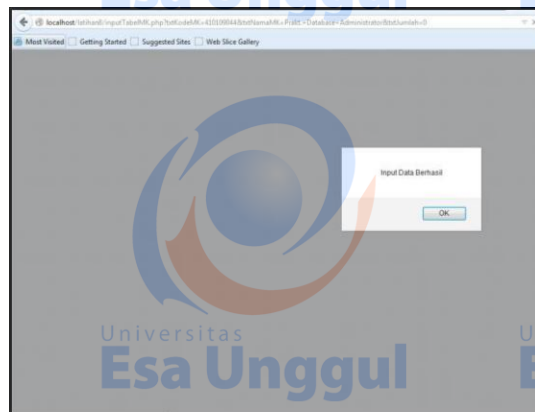




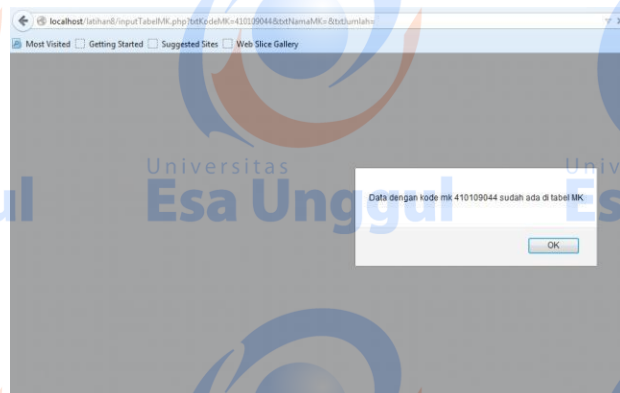
Perlu diketahui bahwa untuk hyperlink “kembali” pada halaman inputMK.php digunakan untuk mengembalikan halaman ke tampilLabkom.php, sedangkan untuk tombol “reset”, digunakan untuk menghapus isian dari semua *textfield* yang ada di halaman inputMK.php.

Apabila tombol “Tambah MK” ditekan maka proses berlanjut ke inputTabelMK.php untuk menampilkan *alert* yang berisi pesan data MK berhasil disimpan dan selanjutnya halaman akan kembali ke tampilLabkom.php. Untuk *script* berikut penjelasannya dapat anda lihat pada gambar 8.1 dan bagian 8.1 pada awal modul 8 ini.

Sedangkan untuk gambar *alert* berisi data MK yang sudah ada serta *alert* yang berisi pesan bahwa data MK berhasil diinputkan dapat dilihat pada gambar 8.16 dan gambar 8.17



Gambar 8.16 *Alert* untuk input data MK yang berhasil



Gambar 8.17 *Alert* untuk input data MK yang sudah ada / ada duplikasi



Kemudian pada tampilLabkom.php, apabila *hyperlink* “edit” ditekan maka halaman web akan merujuk ke editLabkom.php, yang tampilan pada *browser* dapat dilihat pada gambar 8.18 dan untuk *script* nya dapat dilihat pada gambar 8.19, gambar 8.20 dan gambar 8.21.



Gambar 8.18 Tampilan dari editLabkom.php





```
71     </br>
72 </div>
73
74 <div id="footer"><b> Copyright &copy;2014 Labkom. All Rights Reserved. </b>
75 </div>
76 </div>
77 </body>
78 </html>
79
```

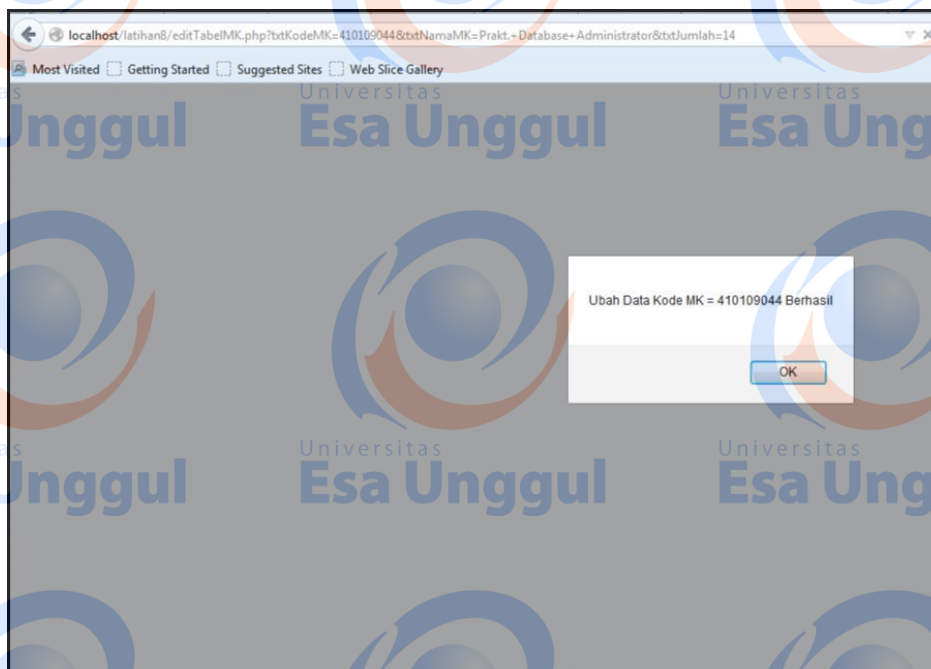
Gambar 8.21 Bagian ketiga dari *script* editLabkom.php

Jika tombol “edit MK” ditekan maka proses akan berlanjut ke editTabelMK.php dimana *script* dan penjelasannya dapat anda lihat pada gambar 8.2.

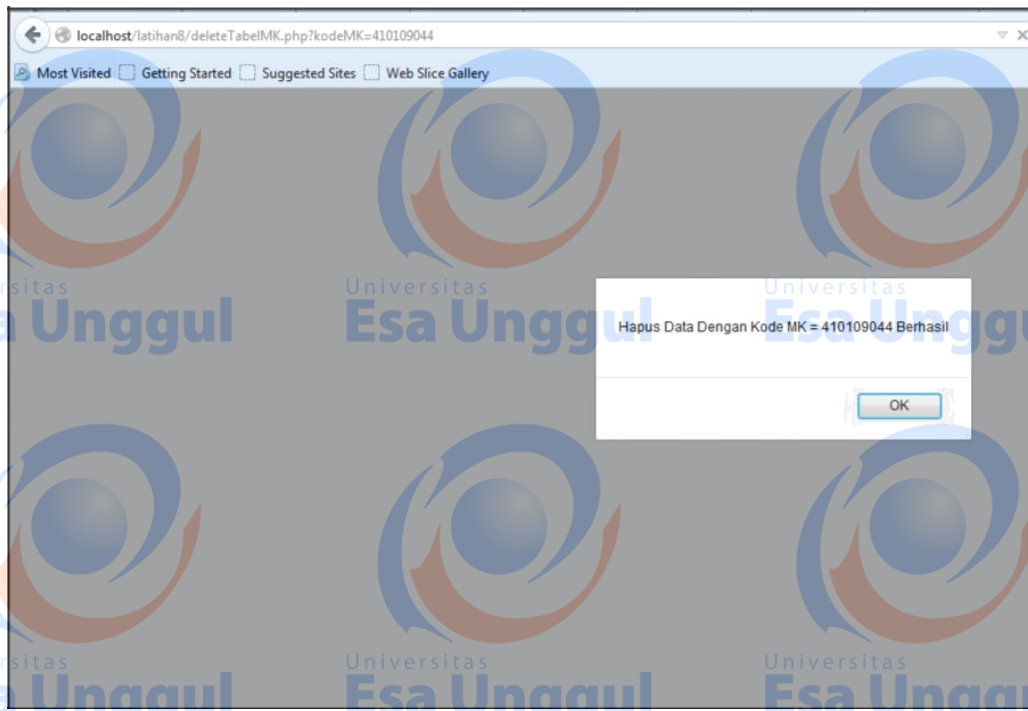
Untuk *alert* yang menandakan bahwa proses *update* data dari tabel MK berhasil, dapat dilihat pada gambar 8.22.

Untuk link “delete” pada tampilLabkom.php pada gambar 8.12, akan merujuk ke deleteTabelMK.php.

Untuk *script* deleteTabelMK.php dapat anda lihat pada gambar 8.3 berikut dengan penjelasannya. Sedangkan untuk *alert* yang menandakan bahwa proses hapus data dari tabel MK berhasil dapat dilihat pada gambar 8.23.



Gambar 8.22 *Alert* berhasil mengubah data dari tabel MK



Gambar 8.22 Alert berhasil menghapus data dari tabel MK

