

**BAHAN AJAR  
MATA KULIAH DESAIN & PEMROGRAMAN WEB**



**TUTORIAL DASAR PEMROGRAMAN PHP DAN MySQL  
(Disertai Contoh Aplikasi CRUD Sederhana)  
Edisi Pertama**

**Disusun Oleh:  
Joko Suntoro, M.Kom  
NIDN: 0631078903**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI  
UNIVERSITAS SEMARANG**

## DAFTAR ISI

DAFTAR ISI .....	ii
DAFTAR GAMBAR .....	iv
DAFTAR TABEL .....	vi
Bab 1 PENGENALAN PHP .....	7
1.1 Syntax Dasar PHP .....	7
1.2 PHP Case Sensitivity .....	8
Bab 2 VARIABEL, KONSTANTA DAN OPERATOR .....	10
2.1 Variabel .....	10
2.2 Konstanta .....	11
2.3 Operator .....	12
2.3.1 Operator Aritmatika .....	12
2.3.2 Operator Perbandingan .....	13
Bab 3 STRUKTUR KENDALI .....	15
3.1 Struktur Kendali <i>if</i> .....	15
3.2 Struktur Kendali <i>if-else</i> .....	15
3.3 Struktur Kendali <i>if-elseif</i> .....	16
3.4 Struktur Kendali <i>switch-case</i> .....	17
Bab 4 PENGULANGAN .....	18
4.1 While .....	18
4.2 Do-While .....	19
4.3 For .....	20

Bab 5	FORM PHP.....	22
5.1	Form Input Text dan Password .....	22
5.2	Form Input Combo Box.....	24
5.3	Form Input Check Box .....	25
5.4	Form Input Radio Button.....	26
5.5	Form Input Textarea .....	27
Bab 6	APLIKASI CRUD SEDERHANA (TEKNIK PROSEDURAL) .....	29
6.1	Pembuatan Database dan Tabel .....	29
6.2	Pembuatan Koneksi PHP-MySQL .....	29
6.3	Tambah Data Mahasiswa (Create) .....	30
6.4	Tampil Data Mahasiswa (Read) .....	32
6.5	Ubah Data Mahasiswa (Update) .....	33
6.6	Hapus Data Mahasiswa (Delete) .....	33
	DAFTAR PUSTAKA.....	36

## DAFTAR GAMBAR

Gambar 1.1. Syntax Dasar PHP .....	7
Gambar 1.2 Tampilan pada Browser hello.php .....	8
Gambar 1.3 Source Code caseSensitivity.php .....	9
Gambar 1.4 Tampilan pada Browser caseSensitivity.php .....	9
Gambar 2.1 Source Code variabel.php .....	10
Gambar 2.2 Tampilan pada Browser variabel.php .....	11
Gambar 2.3 Source Code konstanta.php .....	11
Gambar 2.4 Tampilan pada Browser konstanta.php .....	12
Gambar 2.5 Source Code operatorAritmatika.php .....	13
Gambar 2.6 Source Code operatorPerbandingan.php .....	14
Gambar 3.1 Source Code if.php .....	15
Gambar 3.2 Source Code if-else.php .....	16
Gambar 3.3 Source Code if-elseif.php .....	16
Gambar 3.4 Source Code switch.php .....	17
Gambar 4.1 Source Code while.php .....	18
Gambar 4.2 Tampilan pada Browser while.php .....	19
Gambar 4.3 Source Code do-while.php .....	19
Gambar 4.4 Tampilan pada Browser do-while.php .....	19
Gambar 4.5 Source Code do-while2.php .....	20
Gambar 4.6 Tampilan pada Browser do-while2.php .....	20
Gambar 4.7 Source Code for.php .....	20
Gambar 4.8 Tampilan pada Browser for.php .....	21
Gambar 5.1 Source Code textPassword.php .....	23
Gambar 5.2 Source Code proses-textPassword.php .....	23
Gambar 5.3 Tampilan Form Input Type Text dan Password .....	23
Gambar 5.4 Source Code combobox.php .....	24
Gambar 5.5 Tampilan Form Input Combo Box .....	25
Gambar 5.6 Source Code checkbox.php .....	25
Gambar 5.7 Tampilan Form Input Check Box .....	26
Gambar 5.8 Source Code radiobutton.php .....	26

Gambar 5.9 Tampilan Form Input Radio Button.....	27
Gambar 5.10 Source Code textarea.php .....	27
Gambar 5.11 Tampilan Form Textarea .....	28
Gambar 6.1 Tampilan Browse Mahasiswa.....	29
Gambar 6.2 Source Code koneksi.php .....	30
Gambar 6.3 Source Code form-tambah.php .....	31
Gambar 6.4 Source Code proses-tambah.php.....	31
Gambar 6.5 Source Code index.php.....	32
Gambar 6.6 Source Code form-edit.php .....	34
Gambar 6.7 Source Code proses-edit.php.....	35
Gambar 6.8 Source Code hapus.php .....	35

## DAFTAR TABEL

Tabel 2.1 Jenis Operator Aritmatika.....	12
Tabel 2.2 Jenis Operator Perbandingan .....	14
Tabel 5.1 Perbedaan Method POST dan GET .....	22
Tabel 6.1 Deskripsi Tabel Mahasiswa.....	29

## Bab 1

### PENGENALAN PHP

#### 1.1 Syntax Dasar PHP

Ketika PHP mem-parsing file, akan dicari tag pembuka **<?php** dan tag penutup **?>**. Parsing dengan cara ini memungkinkan PHP dapat disematkan dalam beberapa dokumen yang berbeda, karena segala sesuatu di luar sepasang tag pembuka dan penutup diabaikan oleh parser PHP. Berikut adalah syntax dasar pemrograman PHP:

- Syntax awal untuk membuat program PHP adalah diawali dengan **<?php** dan diakhiri dengan syntax **?>**.
- Untuk menampilkan tulisan dalam browser, maka kita menggunakan syntax **echo ".....";**.
- Untuk memberikan keterangan atau komentar pada PHP (komentar tidak akan muncul dalam browser) kita menggunakan tanda berikut:

**// Komentar**

**# Komentar**

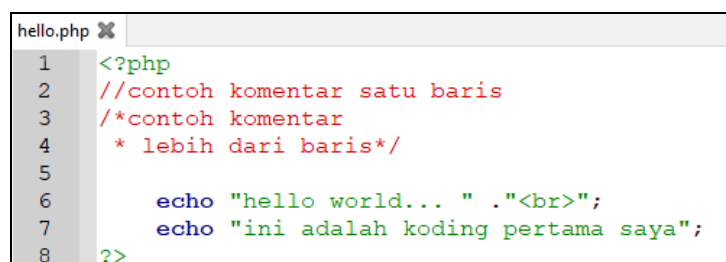
**/\* Komentar \*/**

- Penamaan file pemrograman PHP berekstensi **.php**

Contoh penulisan syntax dasar PHP dapat dilihat pada Gambar 1.1, simpan dengan nama file `hello.php` dan letakkan ke dalam folder `.../xampp/htdocs/belajar/hello.php`. Ketika koding tersebut dijalankan dalam browser (`http://localhost/belajar/hello.php`) akan terlihat seperti Gambar 1.2.

Nama file: `hello.php`

Lokasi: `.../htdocs/belajar/hello.php`

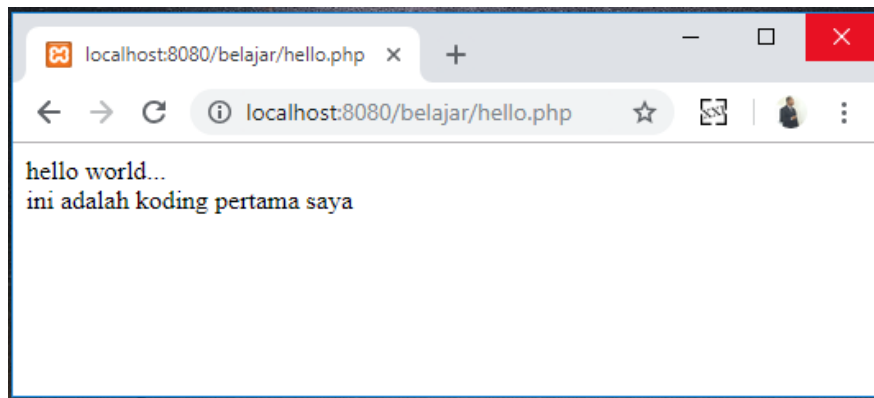


```
hello.php ✕
1  <?php
2  //contoh komentar satu baris
3  /*contoh komentar
4   * lebih dari baris*/
5
6      echo "hello world... " . "<br>";
7      echo "ini adalah koding pertama saya";
8  ?>
```

Gambar 1.1. Syntax Dasar PHP

Penjelasan baris kode:

- Baris 1, merupakan tag pembuka, untuk mengawali syntax PHP
- Baris 2, komentar untuk 1 (satu) baris
- Baris 3-4, komentar untuk dua atau lebih baris
- Baris 6-7, tulisan yang akan ditampilkan pada browser, pada baris 6 terdapat syntax `<br>`, syntax tersebut berfungsi untuk ganti baris
- Baris 8, merupakan tag penutup, untuk mengakhiri syntax PHP



Gambar 1.2 Tampilan pada Browser hello.php

## 1.2 PHP Case Sensitivity

Dalam PHP, semua keyword (seperti: if, else, while, echo), classes dan functions TIDAK case-sensitive. Namun semua **penamaan variabel** (variabel akan dibahas pada Bab 2) case-sensitive. Source code case-sensitivity pada PHP dapat dilihat pada Gambar 1.3 dan tampilan pada browser dapat dilihat pada Gambar 1.4.

Pada Gambar 1.4 terlihat terdapat *error* ketika source code caseSensitivity.php dijalankan dalam browser. *Error* tersebut terletak pada baris ke-5 dan baris ke-6, dimana variabel \$Hobi dan \$HOBI yang dibuat pada source code caseSensitivity.php (lihat Gambar 1.3) tidak terdefinisi. Penamaan variabel pada PHP bersifat case sensitif, terlihat bahwa penamaan variabel \$hobi berbeda dengan variabel \$Hobi dan \$HOBI.

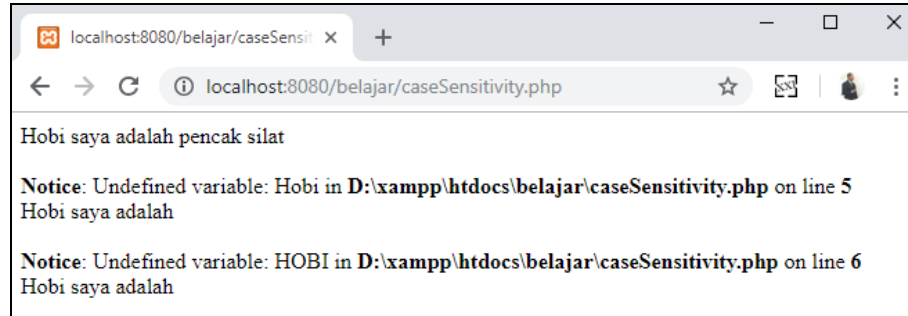
Nama File: caseSensitivity.php

Lokasi: ../htdocs/belajar/ caseSensitivity.php



```
1 <?php
2     $hobi = "pencak silat";
3
4     echo "Hobi saya adalah " . $hobi. "<br>";
5     ECHO "Hobi saya adalah " . $Hobi. "<br>";
6     Echo "Hobi saya adalah " . $HOBI. "<br>";
7 ?>
```

Gambar 1.3 Source Code caseSensitivity.php



Gambar 1.4 Tampilan pada Browser caseSensitivity.php

## Bab 2

### VARIABEL, KONSTANTA DAN OPERATOR

#### 2.1 Variabel

Variabel pada PHP digunakan untuk tempat penyimpanan data sementara. Variabel diawali dengan tanda \$ (dolar). Variabel bersifat case sensitif, variabel dapat memiliki nama pendek (seperti \$x, \$y, \$z, dll) atau nama yang lebih deskriptif (seperti \$usia, \$pekerjaan, \$hobi, dll). Nilai pada variabel dapat diubah-ubah.

Contoh penggunaan variabel dapat dilihat pada Gambar 2.1 dan 2.2, dan berikut adalah aturan-aturan dalam pembuatan variabel:

- Variabel diawali dengan tanda \$ (dolar), kemudian diikuti oleh nama variabel
- Nama variabel harus diawali dengan huruf atau *underscore* (garis bawah)
- Nama variabel tidak boleh diawali dengan angka
- Nama variabel bersifat case sensitif (\$usia dan \$Usia adalah dua variabel yang berbeda)

Nama File: variabel.php

Lokasi: ../htdocs/belajar/variabel.php

```
1  <?php
2      $nama="Joko Suntoro";
3      $jur="Teknik Informatika";
4      $kampus="Universitas Semarang";
5
6      echo "Halo, nama saya adalah <b>" . $nama. "</b><br>";
7      echo "saya dulu pernah belajar di " . $kampus. ", jurusan" . $jur;
8  ?>
```

Gambar 2.1 Source Code variabel.php

Penjelasan baris kode dari file variabel.php:

- Baris 1, tag pembuka PHP
- Baris 2-4, pembuatan variabel, dengan nama variabel: nama, jur, kampus. Masing-masing nama variabel tersebut mempunyai nilai Joko Suntoro, Teknik Informatika dan Universitas Semarang
- Baris 6-7, menampilkan nilai dari variabel pada browser
- Baris 8, tag penutup PHP



Gambar 2.2 Tampilan pada Browser variabel.php

## 2.2 Konstanta

Konstanta seperti variabel, namun nilai yang sudah didefinisikan pada konstanta tidak dapat diubah-ubah. Nama konstanta dimulai dengan huruf atau garis bawah (tidak ada tanda \$ sebelum nama konstanta). Contoh penggunaan konstanta dapat dilihat pada Gambar 2.3 dan Gambar 2.4. Pembuatan konstanta diawali dengan fungsi *define()*. Syntax penulisan konstanta adalah sebagai berikut:

*define(name, value)*

keterangan parameter:

- name: nama konstanta
- value: nilai dari konstanta

Nama File: konstanta.php

Lokasi: ../htdocs/belajar/konstanta.php

```

1  <?php
2      define("koding", "pemrograman PHP");
3      define("suasana", "mudah dan menyenangkan");
4      define("Dosen", "Joko Suntoro");
5
6      echo "Belajar bahasa " .koding. "<br>";
7      echo "sangat " .suasana. "<br>";
8      echo "apalagi diajar oleh pak " .Dosen;
9  ?>

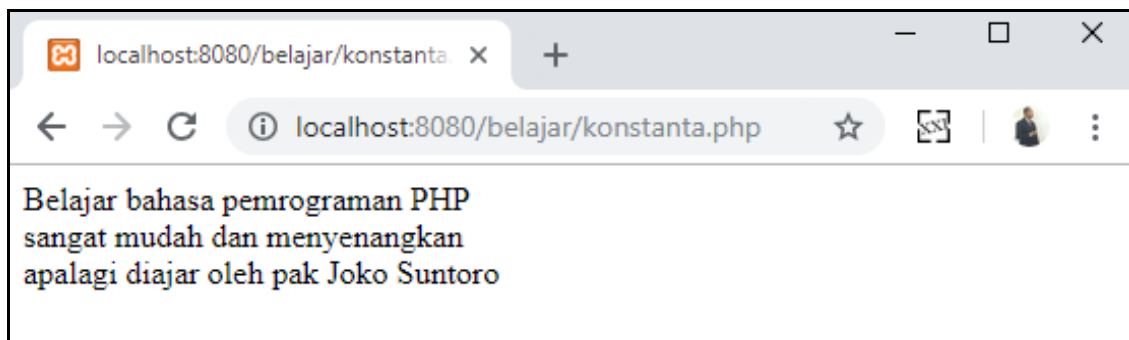
```

Gambar 2.3 Source Code konstanta.php

Penjelasan baris kode dari file konstanta.php:

- Baris 1, tag pembuka PHP
- Baris 2-4, pembuatan konstanta, dengan nama konstanta: koding, suasana, dosen

- Baris 6-7, menampilkan nilai dari konstanta di browser
- Baris 8, tag penutup PHP



Gambar 2.4 Tampilan pada Browser konstanta.php

## 2.3 Operator

### 2.3.1 Operator Aritmatika

Operator aritmatika digunakan untuk penghitungan aritmatika, seperti penjumlahan, pengurangan, perkalian, pembagian, dll. Sama halnya pada aturan aritmatika, operator ini memperhatikan urutan prioritasnya, contohnya operator pangkat/akar dikerjakan terlebih dahulu sebelum operator perkalian/pembagian, operator perkalian/pembagian dikerjakan terlebih dahulu sebelum operator penjumlahan/pengurangan.

Table 2.1 menunjukkan lambang dan nama operator aritmatika. Contoh penerapan operator aritmatika PHP dapat dilihat pada Gambar 2.5.

Tabel 2.1 Jenis Operator Aritmatika

Operator	Nama	Contoh	Hasil
+	penjumlahan	$\$x + \$y$	menjumlahkan $\$x$ dengan $\$y$
-	pengurangan	$\$x - \$y$	mengurangi $\$x$ dengan $\$y$
*	perkalian	$\$x * \$y$	mengalikan $\$x$ dengan $\$y$
/	pembagian	$\$x / \$y$	membagi $\$x$ dengan $\$y$
%	modulus	$\$x \% \$y$	sisa hasil pembagaian antara $\$x$ dan $\$y$
pow	pangkat	pow( $\$x, \$y$ )	$\$x$ pangkat $\$y$ (pada PHP versi 5.5 ke bawah)
**	pangkat	$\$x ** \$y$	$\$x$ pangkat $\$y$ (pada PHP versi 5.6)

Nama File: operatorAritmatika.php

Lokasi: ../htdocs/belajar/operatorAritmatika.php

```
1  <?php
2  $x = 5;
3  $y = 3;
4  # operator penjumlahan
5  $jumlah=$x+$y;
6  //operator pengurangan
7  $kurang=$x-$y;
8  /* operator perkalian */
9  $kali=$x*$y;
10 # operator pembagian
11 $bagi=$x/$y;
12 // operator modulus
13 $mod=$x%$y;
14 /* operator pangkat */
15 $pangkat=pow($x,$y);
16
17 // tampilan data pada browser
18 echo "angka 1 = " . $x. "<br>";
19 echo "angka 2 = " . $y. "<hr>";
20 echo "angka 1 + angka 2 = " . $jumlah. "<br>";
21 echo "angka 1 - angka 2 = " . $kurang. "<br>";
22 echo "angka 1 x angka 2 = " . $kali. "<br>";
23 echo "angka 1 : angka 2 = " . $bagi. "<br>";
24 echo "sis hasil bagi angka 1 dan angka 2 adalah " . $mod. "<br>";
25 echo "angka 1 ^ angka 2 = " . $pangkat. "<br>";
26 ?>
```

Gambar 2.5 Source Code operatorAritmatika.php

### 2.3.2 Operator Perbandingan

Operator perbandingan digunakan untuk membandingkan dua nilai. Tabel 2.2 menunjukkan jenis operator perbandingan dan Gambar 2.6 adalah contoh source code penerapan operator perbandingan pada PHP.

Tabel 2.2 Jenis Operator Perbandingan

Operator	Nama	Contoh	Hasil
==	equal	\$x == \$y	bernilai true jika \$x sama dengan \$y
!=	not equal	\$x != \$y	bernilai true jika \$x tidak sama dengan \$y
<>	not equal	\$x <> \$y	bernilai true jika \$x tidak sama dengan \$y
>	lebih besar	\$x > \$y	bernilai true jika \$x lebih besar daripada \$y
<	lebih kecil	\$x < \$y	bernilai true jika \$x lebih kecil daripada \$y
>=	lebih besar sama dengan	\$x >= \$y	bernilai true jika \$x lebih besar atau sama dengan \$y
<=	lebih kecil sama dengan	\$x <= \$y	bernilai true jika \$x lebih kecil atau sama dengan \$y

Nama File: operatorPerbandingan.php

Lokasi: .../htdocs/belajar/operatorPerbandingan.php

```

1  <?php
2  $a = 15;
3  $b = 5;
4
5  echo "nilai a = " . $a. "<br>";
6  echo "nilai b = " . $b. "<hr>";
7
8  if ($a > $b) {
9      echo "a lebih besar daripada b";
10 } elseif ($a < $b) {
11     echo "a lebih kecil daripada b";
12 } else {
13     echo "a sama dengan b";
14 }
15 ?>

```

Gambar 2.6 Source Code operatorPerbandingan.php

## Bab 3

### STRUKTUR KENDALI

Struktur kendali adalah elemen terpenting dalam PHP. Struktur kendali berfungsi untuk mengontrol jalannya eksekusi program. Jika pernyataan tersebut bernilai benar, maka PHP akan mengeksekusi pernyataan tersebut, sedangkan jika pernyataan tersebut bernilai salah, maka PHP akan mengabaikannya.

#### 3.1 Struktur Kendali *if*

Pernyataan *if* akan menjalankan beberapa kode jika satu kondisi benar. Gambar 3.1 menunjukkan contoh penggunaan struktur kendali *if*. Jika waktu saat ini kurang dari 24, maka akan ditampilkan “Tetap semangat, semoga tetap dalam perdjoeangan”.

Nama file: if.php

Lokasi: .../htdocs/belajar/if.php

```
1 <?php
2     $t = date("H");
3     if($t < 24) {
4         echo "Tetap semangat, semoga tetap dalam perdjoeangan";
5     }
6     ?>
```

Gambar 3.1 Source Code if.php

#### 3.2 Struktur Kendali *if-else*

Pernyataan *if-else* menjalankan beberapa kode jika kondisi benar dan kode lain jika kondisi itu salah. Contoh penggunaan struktur kendali *if-else* dapat dilihat pada Gambar 3.2. jika variabel user terisi dengan ftik dan variabel pass terisi dengan usm, maka akan ditampilkan tulisan “Selamat datang di FTIK USM”, namun jika variabel user dan pass tidak terisi dengan ftik dan usm, maka akan ditampilkan tulisan “User dan Pass yang Anda masukkan salah”.

Nama File: if-else.php

Lokasi: .../htdocs/belajar/if-else.php

```

1  <?php
2      $user = "ftik";
3      $pass = "usm";
4
5      if($user == "ftik" and $pass == "usm") {
6          echo "Selamat datang di FTIK USM";
7      } else {
8          echo "User dan Pass yang Anda masukkan salah";
9      }
10  ?>

```

Gambar 3.2 Source Code if-else.php

### 3.3 Struktur Kendali *if-elseif*

Struktur kendali *if-elseif* akan menjalankan beberapa kode berbeda untuk lebih dari dua kondisi. Gambar 3.3 menunjukkan contoh penggunaan struktur kendali *if-elseif*.

Nama File: if-elseif.php

Lokasi: .../htdocs/belajar/if-elseif.php

```

1  <?php
2      date_default_timezone_set('Asia/Jakarta');
3      $t = date("H");
4
5      echo "Saat ini pukul " . $t. "<br>";
6
7      if ($t < "10") {
8          echo "Selamat pagi";
9      } elseif ($t < "15") {
10         echo "Selamat siang";
11     } elseif ($t < "20") {
12         echo "Selamat sore";
13     } else {
14         echo "Selamat malam";
15     }
16  ?>

```

Gambar 3.3 Source Code if-elseif.php

Penjelasan baris kode file if-elseif.php:

- Baris 1, tag pembuka PHP
- Baris 2, function bawaan PHP untuk setting waktu bagian Indonesia (WIB)
- Baris 3, pembuatan variabel *t*, dengan value waktu real time saat browser dibuka
- Baris 5, menampilkan text dalam browser
- Baris 7-8, jika waktu kurang dari 10, maka akan tampil “Selamat pagi” di browser



- Baris 9-10, jika waktu kurang dari 15, maka akan tampil “Selamat siang” di browser
- Baris 11-12, jika waktu kurang dari 20, maka akan tampil “Selamat sore” di browser
- Baris 13-14, jika waktu lebih dari 20, maka akan tampil “Selamat malam” di browser

### 3.4 Struktur Kendali *switch-case*

Struktur kendali *switch-case* hampir mirip dengan struktur kendali *if-else*. *Switch-case* digunakan ketika akan dibandingkan variabel dan ekspresi yang sama, namun dengan banyak nilai yang berbeda. Gambar 3.4 menunjukkan contoh penggunaan struktur kendali *switch-case*.

Nama File: switch.php

Lokasi: .../htdocs/belajar/switch.php

```

1  <?php
2      $lantai = 2;
3
4      switch ($lantai) {
5          case "1":
6              echo "Anda akan menuju ke lantai 1";
7              break;
8          case "2":
9              echo "Anda akan menuju ke lantai 2";
10             break;
11         case "3":
12             echo "Anda akan menuju ke lantai 3";
13             break;
14         default:
15             echo "Silakan pilih lantai yang akan Anda tuju";
16     }
17 ?>

```

Gambar 3.4 Source Code switch.php

## Bab 4

### PENGULANGAN

Pengulangan sering disebut juga dengan looping, yaitu suatu proses pengulangan nilai, dimana nilai-nilai tersebut telah disetting dengan aturan-aturan tertentu. Contoh penerapan pengulangan adalah pembuatan halaman website (pagination), looping query array, dan lain-lain. Berikut adalah jenis-jenis pengulangan pada PHP:

#### 4.1 While

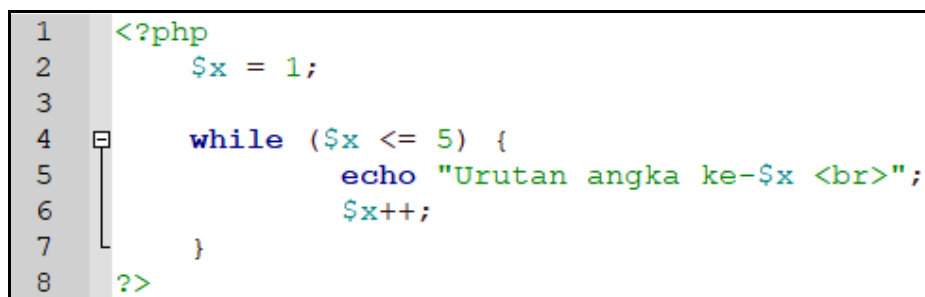
Pengulangan while adalah jenis pengulangan sederhana dalam PHP. Pengulangan while akan menjalankan pernyataan berulang kali, selama kondisi yang ditentukan bernilai TRUE (benar). Syntax penulisan pengulangan while adalah sebagai berikut:

```
while (kondisi bernilai benar) {  
    pernyataan dijalankan;  
}
```

Contoh penggunaan while dapat dilihat pada Gambar 4.1, dan Gambar 4.2 menunjukkan tampilan source code while ketika dijalankan di browser. Kita akan menampilkan angka 1 sampai dengan 5, langkah awal yang kita lakukan adalah menambahkan variabel \$x dengan value sama dengan 1 (\$x=1), kemudian kita ulang/looping \$x tersebut sebanyak kurang dari sama dengan 5 (\$x <= 5). Value \$x akan meningkat sejumlah 1 ketika looping berjalan (\$x++)

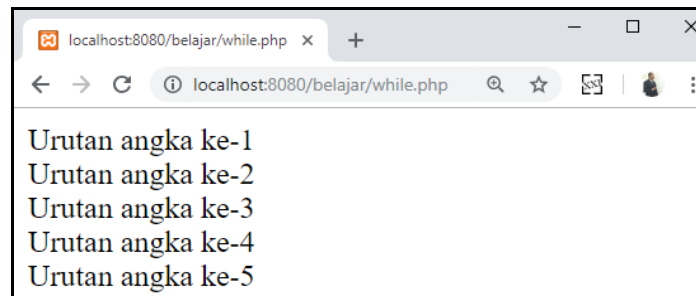
Nama File: while.php

Lokasi: .../htdocs/belajar/while.php



```
1 <?php  
2     $x = 1;  
3  
4     while ($x <= 5) {  
5         echo "Urutan angka ke-$x <br>";  
6         $x++;  
7     }  
8 ?>
```

Gambar 4.1 Source Code while.php



Gambar 4.2 Tampilan pada Browser while.php

## 4.2 Do-While

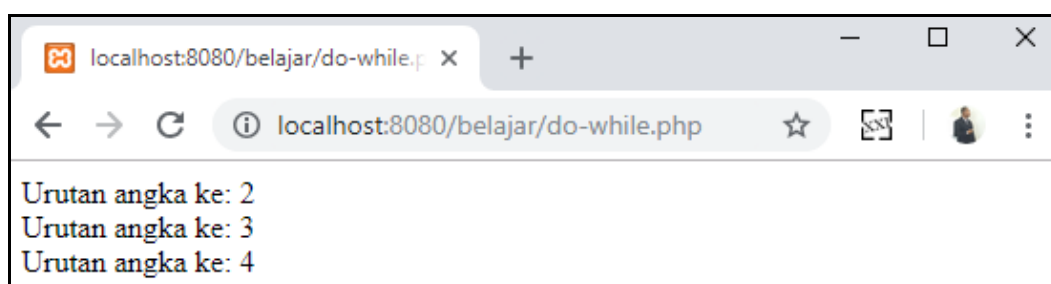
Pengulangan do-while hampir sama dengan pengulangan while, kecuali ekspresi kebenaran dicek pada akhir setiap iterasi, bukan di awal. Sehingga pada pengulangan ini akan tetap melakukan looping minimal sekali, meskipun kondisi tidak terpenuhi. Gambar 4.3 adalah contoh penggunaan do-while ketika kondisi terpenuhi, sedangkan Gambar 4.5 adalah contoh penggunaan do-while ketika kondisi tidak terpenuhi. Gambar 4.4 dan 4.6 adalah tampilan masing-masing kondisi ketika dijalankan di browser.

Nama File: do-while.php

Lokasi: .../htdocs/belajar/do-while.php

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

Gambar 4.3 Source Code do-while.php



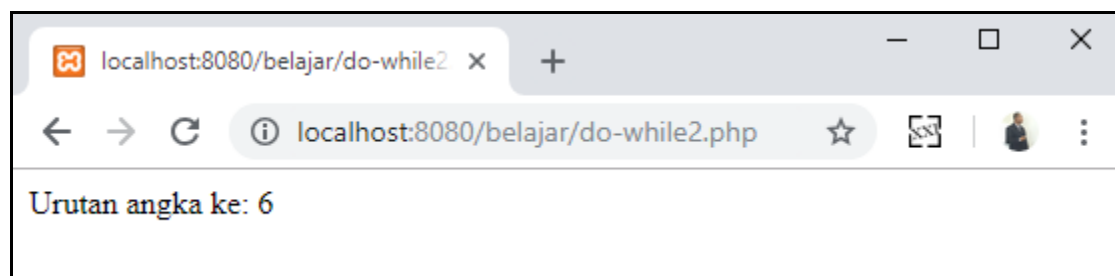
Gambar 4.4 Tampilan pada Browser do-while.php

Nama File: do-while2.php

Lokasi: .../htdocs/belajar/do-while2.php

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

Gambar 4.5 Source Code do-while2.php



Gambar 4.6 Tampilan pada Browser do-while2.php

### 4.3 For

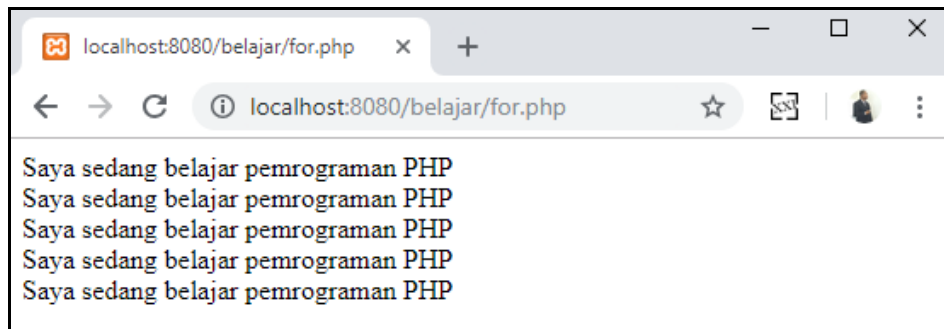
Pengulangan for adalah pengulangan paling kompleks dalam PHP. Pengulangan for digunakan ketika kita sudah mengetahui sebelumnya berapa kali pernyataan harus diulang/dijalankan. Gambar 4.6 menunjukkan source code pengulangan for dan Gambar 4.7 adalah tampilan pada browser ketika source code dijalankan.

Nama File: for.php

Lokasi: .../htdocs/belajar/for/php

```
1 <?php
2 for ($i=1; $i<=5; $i++) {
3     echo "Saya sedang belajar pemrograman PHP <br>";
4 }
5 ?>
```

Gambar 4.7 Source Code for.php



Gambar 4.8 Tampilan pada Browser for.php

## Bab 5

### FORM PHP

Form digunakan untuk melakukan pengiriman data dari halaman klien ke server untuk diolah dan akan ditampilkan kembali ke halaman klien jika diperlukan. Pengiriman data pada form PHP dilakukan dengan dua cara yaitu menggunakan method `$_POST` dan method `$_GET`. Tabel 5.1 menunjukkan perbedaan method POST dan GET, namun secara umum kedua method tersebut memiliki fungsi yang sama yaitu mengirimkan nilai pada variabel dan mengambil nilai pada variabel.

Tabel 5.1 Perbedaan Method POST dan GET

POST	GET
Tidak menampilkan nilai variabel pada URL	Menampilkan nilai variabel pada URL
Lebih aman (nilai variabel tidak ditampilkan)	Kurang aman (nilai variabel ditampilkan)
Panjang string tidak dibatasi	Panjang string dibatasi 2.047 karakter
Pengambilan data menggunakan <code>\$_POST</code>	Pengambilan data menggunakan <code>\$_GET</code>
Digunakan untuk pengiriman data private	Digunakan untuk pengiriman data public

Form biasanya terdiri dari satu atau lebih inputan, berikut adalah jenis-jenis inputan pada form PHP:

#### 5.1 Form Input Text dan Password

Input type text dan password mempunyai fungsi yang sama, yaitu berfungsi sebagai kolom isian dari user berupa text. Dengan perbedaan untuk type password, text isian pada kolom tidak ditampilkan pada browser, namun diganti dengan karakter khusus. Gambar 5.1 menunjukkan contoh penggunaan input text dan password, ketika nanti tombol “Proses” diklik, maka akan diproses file bernama proses-textPassword.php. Source code proses-textPassword.php dapat dilihat pada Gambar 5.2, sedangkan Gambar 5.3 menunjukkan tampilan pada browser input type text dan password.

Nama File: textPassword.php

Lokasi: ../htdocs/belajar-form/textPassword.php

```
1 <html>
2 <form action="proses-textPassword.php" method="post">
3     Nama: <input type="text" name="txtNama"><br>
4     Password: <input type="password" name="txtPass"><br>
5     <input type="submit" name="proses" value="Proses">
6 </form>
7 </html>
```

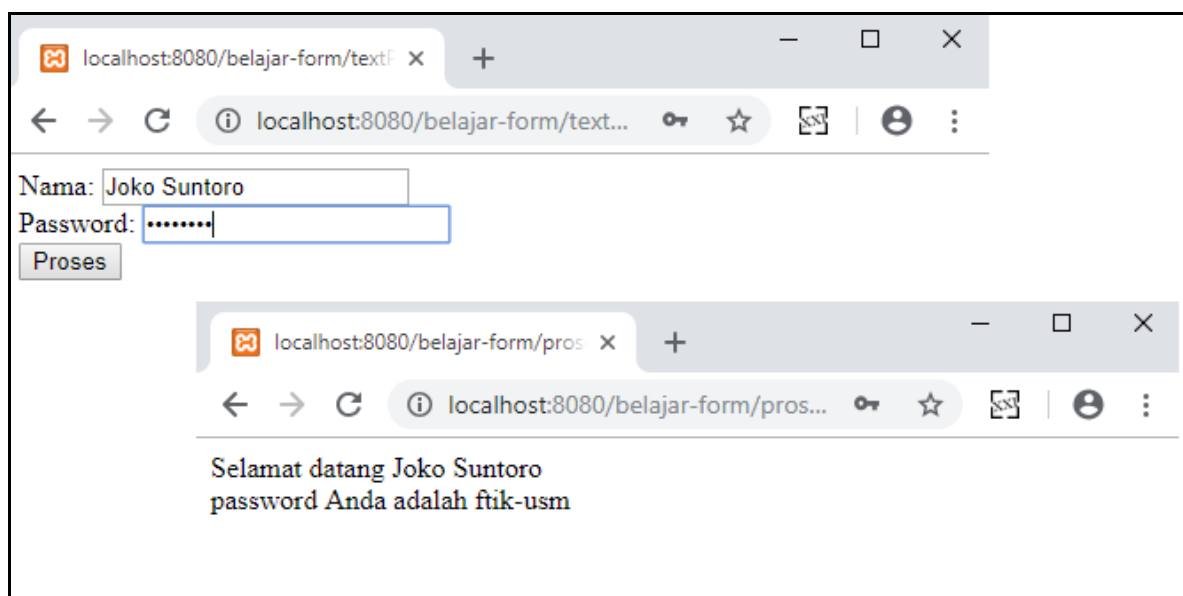
Gambar 5.1 Source Code textPassword.php

Nama File: proses-textPassword.php

Lokasi: ../htdocs/belajar-form/proses-textPassword.php

```
1 <?php
2     $nama = $_POST['txtNama'];
3     $password = $_POST['txtPass'];
4
5     echo "Selamat datang " . $nama . "<br>";
6     echo "password Anda adalah " . $password;
7 ?>
```

Gambar 5.2 Source Code proses-textPassword.php



Gambar 5.3 Tampilan Form Input Type Text dan Password

## 5.2 Form Input Combo Box

Form input combo box adalah sebuah elemen form berbentuk kotak yang apabila diklik akan ditampilkan data dalam bentuk drop down list (menu pilihan ke bawah). Tag yang digunakan untuk membuat combo box pada PHP adalah tag *select*, berbeda dengan input type text atau password, pada input type combo box dibutuhkan tag pembuka *<select>* dan tag penutup *</select>*. Diantara tag pembuka *<select>* dan tag penutup *</select>* disisipkan element *option*, element *option* berguna untuk pemberian nilai data.

Contoh source code penggunaan combo box pada PHP dapat dilihat pada Gambar 5.4. Sedangkan tampilan pada browser penggunaan combo box dapat dilihat pada Gambar 5.5.

Nama File: combobox.php

Lokasi File: .../htdocs/belajar-form/combobox.php

```
1 <html>
2 <form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>"
3   <h3>Penjumlahan atau Pengurangan</h3>
4   <div>Nilai A: <input type="text" name="a"></div>
5   <div>Nilai B: <input type="text" name="b"></div>
6   <div>
7     Pilih Metode:
8     <select name="penghitungan">
9       <option value='++' selected>--Silakan Pilih--</option>
10      <option value='tambah'>Penjumlahan</option>
11      <option value='kurang'>Pengurangan</option>
12    </select>
13  </div>
14  <div><input type="submit" name="proses" value="PROSES"></div>
15 </form>
16 </html>
17
18 <?php
19 error_reporting(0);
20 $hitung = $_POST['penghitungan'];
21 $a = $_POST['a'];
22 $b = $_POST['b'];
23
24 if ($hitung == 'tambah') {
25   $hasil = $a + $b;
26   $ket = "penjumlahan";
27 } elseif ($hitung == 'kurang') {
28   $hasil = $a - $b;
29   $ket = "pengurangan";
30 } else {
31   echo "Anda belum memilih metode <br>";
32 }
33
34 echo "Nilai A = " . $a. "<br>";
35 echo "Nilai B = " . $b. "<br>";
36 echo "Hasil $ket kedua nilai tersebut adalah " . $hasil;
37 ?>
```

Gambar 5.4 Source Code combobox.php



localhost:8080/belajar-form/com x +

localhost:8080/belajar-form/combobox.php ☆

### Penjumlahan atau Pengurangan

Nilai A:

Nilai B:

Pilih Metode:

Nilai A = 5  
 Nilai B = 3  
 Hasil penjumlahan kedua nilai tersebut adalah 8

Gambar 5.5 Tampilan Form Input Combo Box

### 5.3 Form Input Check Box

Form input check box digunakan untuk membuat check box atau kotak isian yang diisi dengan cara menceklis kotak tersebut. Check box biasa digunakan untuk memilih pilihan dengan lebih dari satu pilihan. Gambar 5.6 adalah contoh source code check box dan tampilan pada browser check box dapat dilihat pada Gambar 5.7.

Nama File: checkbox.php

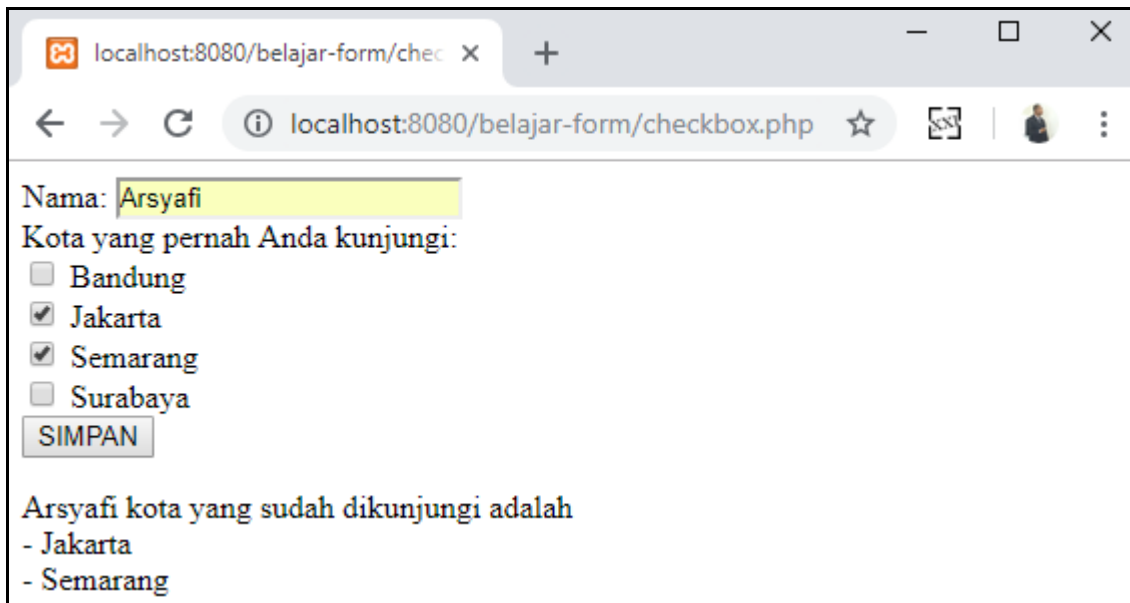
Lokasi: .../htdocs/belajar-form/checkbox.php

```

1 <form method="post" action="<?php echo $_SERVER["PHP_SELF"];?>">
2 <div>Nama: <input type="text" name="txtNama"></div>
3 <div> Kota yang pernah Anda kunjungi: <br>
4 <input type="checkbox" name="kota[]" value="Bandung"> Bandung <br>
5 <input type="checkbox" name="kota[]" value="Jakarta"> Jakarta <br>
6 <input type="checkbox" name="kota[]" value="Semarang"> Semarang <br>
7 <input type="checkbox" name="kota[]" value="Bandung"> Surabaya <br>
8 </div>
9 <div><input type="submit" name="simpan" value="SIMPAN"></div>
10 </form>
11
12 <?php
13 $nama = $_POST['txtNama'];
14 $kota = $_POST['kota'];
15
16 echo "$nama kota yang sudah dikunjungi adalah <br>";
17 foreach ($kota as $data) {
18     echo "- $data <br>";
19 }
20 ?>

```

Gambar 5.6 Source Code checkbox.php



Gambar 5.7 Tampilan Form Input Check Box

#### 5.4 Form Input Radio Button

Form input radio button digunakan untuk membuat inputan radio. Radio biasanya digunakan untuk memilih satu pilihan dari beberapa pilihan yang ada. Gambar 5.8 adalah contoh source code radio button dan tampilan pada browser radio button dapat dilihat pada Gambar 5.9.

Nama File: radiobutton.php

Lokas File: ../htdocs/belajar-form/radiobutton.php

```

1 <form method="post" action="<?php echo $_SERVER["PHP_SELF"];?>">
2 <div>Nama: <input type="text" name="txtNama"></div>
3 <div>Kapan waktu paling nyaman untuk belajar?
4     <input type="radio" name="time" value="pagi"> Pagi &nbsp;
5     <input type="radio" name="time" value="siang"> Siang &nbsp;
6     <input type="radio" name="time" value="malam"> Malam &nbsp;
7 </div>
8 <div><input type="submit" name="simpan" value="SIMPAN"></div>
9 </form>
10
11 <?php
12     $nama = $_POST['txtNama'];
13     $waktu = $_POST['time'];
14
15     echo "Halo... " . $nama. "<br>";
16     echo "Menurut Anda, " . $waktu. " hari paling nyaman untuk belajar";
17 ?>

```

Gambar 5.8 Source Code radiobutton.php

localhost:8080/belajar-form/radiobutton.php

Nama:

Kapan waktu paling nyaman untuk belajar? ☐ Pagi ☐ Siang ☒ Malam

Halo... Rutin Fisekati  
Menurut Anda, malam hari paling nyaman untuk belajar

Gambar 5.9 Tampilan Form Input Radio Button

## 5.5 Form Input Textarea

Kegunaan input type textarea mirip dengan input type text, namun textarea mempunyai kelebihan yaitu bisa menampung input yang panjang dan lebih dari dua baris. Gambar 5.10 adalah contoh source code textarea dan tampilan pada browser textarea dapat dilihat pada Gambar 5.11.

Nama File: textarea.php

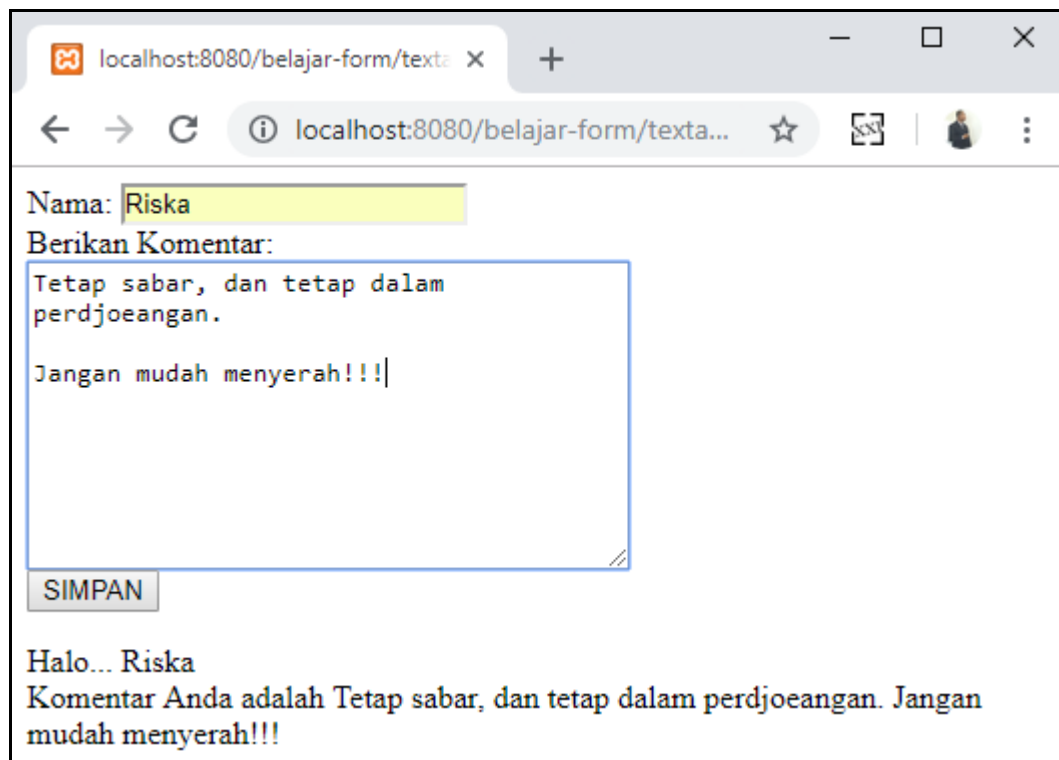
Lokasi File: .../htdocs/belajar-form/textarea.php

```

1 <form method="post" action="<?php echo $_SERVER["PHP_SELF"];?>">
2   <div>Nama: <input type="text" name="txtNama"></div>
3   <div>Berikan Komentar:<br>
4       <textarea name="komentar" cols="40" rows="10"></textarea>
5   </div>
6   <div><input type="submit" name="simpan" value="SIMPAN"></div>
7 </form>
8
9 <?php
10     $nama = $_POST['txtNama'];
11     $pesan = $_POST['komentar'];
12
13     echo "Halo... " . $nama. "<br>";
14     echo "Komentar Anda adalah " . $pesan. "<br>";
15 ?>

```

Gambar 5.10 Source Code textarea.php



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/belajar-form/texta...'. The page content includes a form with the following elements:

- A text input field labeled 'Nama:' containing the text 'Riska'.
- A label 'Berikan Komentar:' followed by a text area containing the text 'Tetap sabar, dan tetap dalam perdjoeangan. Jangan mudah menyerah!!!'.
- A button labeled 'SIMPAN'.
- A feedback message below the form: 'Halo... Riska' and 'Komentar Anda adalah Tetap sabar, dan tetap dalam perdjoeangan. Jangan mudah menyerah!!!'.

Gambar 5.11 Tampilan Form Textarea

## Bab 6

### APLIKASI CRUD SEDERHANA (TEKNIK PROSEDURAL)

#### 6.1 Pembuatan Database dan Tabel

Database yang digunakan pada praktikum ini MySQL, dibuat dengan program bantu PHPMyAdmin. Berikut adalah langkah-langkah pembuatan database dan tabel:

1. Aktifkan xampp control panel.
2. Buka browser, kemudian ketikkan url <http://localhost/phpmyadmin/> (apabila menggunakan port 8080, diubah menjadi <http://localhost:8080/phpmyadmin/>).
3. Buat database dengan nama **crud**.
4. Buat tabel dengan nama **mahasiswa** (lihat Tabel 6.1).
5. Insert beberapa data pada tabel siswa (lihat Gambar 6.1).

Tabel 6.1 Deskripsi Tabel Mahasiswa

Nama Field	Type Data	Length	Extra
id	int	8	primary key, auto increment
nim	varchar	30	
nama	varchar	50	



id	nim	nama
1	G01	Arsyavie Pramudhita
2	G02	Rutin Fisekati
3	G03	Joko Suntoro

Gambar 6.1 Tampilan Browse Mahasiswa

#### 6.2 Pembuatan Koneksi PHP-MySQL

Agar bisa terhubung (saling berkomunikasi) antara aplikasi web (PHP) dan database (MySQL), kita harus menghubungkan antara aplikasi web dan database dengan cara membuat file koneksi. Berikut adalah langkah-langkah pembuatan file koneksi:

1. Buat folder baru dengan nama **belajar-crud** pada folder htdocs (.../xampp/htdocs/belajar-crud).
2. Buat file dengan nama **koneksi.php**, kemudian simpan di dalam folder belajar-crud yang sudah kita buat pada langkah no. 1
3. Ketikkan source code koneksi.php, seperti yang terlihat pada Gambar 6.2.

```
1 <?php
2     $host = "localhost";
3     $user = "root";
4     $pass = "";
5     $db = "crud";
6
7     $conn = mysqli_connect($host, $user, $pass, $db);
8
9     if (!$conn) {
10         die("Gagal terhubung dengan database: ") .mysqli_connect_error();
11     }
12 ?>
```

Gambar 6.2 Source Code koneksi.php

### 6.3 Tambah Data Mahasiswa (Create)

Untuk membuat modul tambah mahasiswa dibutuhkan 2 (dua) file, yaitu **form-tambah.php** dan **proses-tambah.php**. File form-tambah.php digunakan sebagai view/tampilan form. Sedangkan proses-tambah.php digunakan sebagai controller, agar data yang diinputkan pada form dapat tersimpan ke dalam database.

Berikut adalah langkah-langkah pembuatan file form-tambah.php:

1. Buat file dengan nama **form-tambah.php**
2. Simpan file tersebut ke dalam folder belajar-crud yang telah kita buat (.../htdocs/belajar-crud/form-tambah.php).
3. Ketikkan source code form-tambah.php, lihat Gambar 6.3.

Sedangkan untuk pembuatan file proses-tambah.php adalah sebagai berikut:

1. Buat file dengan nama **proses-tambah.php**
2. Simpan file tersebut ke dalam folder belajar-crud yang telah kita buat (.../htdocs/belajar-crud/proses-tambah.php).
3. Ketikkan source code proses-tambah.php, lihat Gambar 6.4.

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>:: Tambah Mahasiswa ::</title>
5 </head>
6 <body>
7 <h3>Formulir Tambah Data Mahasiswa</h3>
8 <form action="proses-tambah.php" method="POST">
9 <fieldset>
10 <table>
11 <tr>
12 <td>NIM:</td>
13 <td><input type="text" name="nim" placeholder="Nomor Induk Mahasiswa"></td>
14 </tr>
15 <tr>
16 <td>Nama:</td>
17 <td><input type="text" name="nama" placeholder="Nama Mahasiswa"></td>
18 </tr>
19 <tr>
20 <td></td>
21 <td>
22 <input type="submit" name="submit" value="Tambah Data">
23 <input type="reset" name="reset" value="Reset Data">
24 </td>
25 </tr>
26 </table>
27 </fieldset>
28 </form>
29 <a href="index.php">Kembali ke Halaman Index</a>
30 </body>
31 </html>

```

Gambar 6.3 Source Code form-tambah.php

```

1 <?php
2 include("koneksi.php");
3
4 //cek apakah tombol submit sudah diklik atau blum?
5 if (isset($_POST['submit'])) {
6     // ambil data dari formulir
7     $nim = $_POST['nim'];
8     $nama = $_POST['nama'];
9
10    // buat query
11    $sql = "INSERT INTO mahasiswa (nim, nama) VALUES ('$nim', '$nama')";
12    $query = mysqli_query($conn, $sql);
13
14    // apakah query simpan berhasil?
15    if ($query) {
16        header('Location: index.php?status=sukses');
17    } else {
18        header('Location: index.php?status=gagal');
19    }
20
21 } else {
22     die("Akses dilarang...");
23 }
24 ?>

```

Gambar 6.4 Source Code proses-tambah.php

## 6.4 Tampil Data Mahasiswa (Read)

Kita akan menampilkan data yang berada di dalam database MySQL ke dalam aplikasi berbasis web yang kita buat. Berikut adalah langkah-langkahnya:

1. Buat file dengan nama **index.php**.
2. Simpan file tersebut ke dalam folder belajar-crud yang telah kita buat.
3. Ketikkan source code index.php, seperti yang terlihat pada Gambar 6.5.

```
1  <?php
2      include("koneksi.php");
3  ?>
4  <!DOCTYPE html>
5  <html>
6  <head>
7      <title>:: Data Mahasiswa ::</title>
8  </head>
9  <body>
10     <h3>Daftar Nama Mahasiswa</h3>
11     <a href="form-tambah.php">[+] Tambah Mahasiswa</a>
12
13     <?php if (isset($_GET['status'])): ?>
14     <p>
15         <?php
16         if ($_GET['status'] == 'sukses') {
17             echo "Tambah mahasiswa berhasil!";
18         } else {
19             echo "Tambah mahasiswa gagal!";
20         }
21         ?>
22     </p>
23     <?php endif; ?>
24     <br>
25     <table border="1">
26     <tr>
27         <th>No</th>
28         <th>NIM</th>
29         <th>Nama</th>
30         <th>Tindakan</th>
31     </tr>
32
33     <?php
34     $sql = "SELECT * FROM mahasiswa";
35     $query = mysqli_query($conn, $sql);
36
37     while ($x = mysqli_fetch_array($query)) {
38         echo "<tr>";
39         echo "<td>" . $x['id'] . "</td>";
40         echo "<td>" . $x['nim'] . "</td>";
41         echo "<td>" . $x['nama'] . "</td>";
42         echo "<td>";
43         echo "<a href='form-edit.php?id=".$x['id']."'>Edit</a> | ";
44         echo "<a href='hapus.php?id=".$x['id']."'>Hapus</a>";
45         echo "</td>";
46         echo "</tr>";
47     }
48     ?>
49 </table>
50 <p>Total Mahasiswa: <?php echo mysqli_num_rows($query) ?></p>
51 </body>
52 </html>
```

Gambar 6.5 Source Code index.php



## 6.5 Ubah Data Mahasiswa (Update)

Pada pembuatan modul ubah data mahasiswa dibutuhkan 2 (dua) file, yaitu **form-edit.php** dan **proses-edit.php**. File form-edit.php digunakan sebagai view/tampilan form untuk ubah data. Sedangkan proses-edit.php digunakan sebagai controller, agar data yang akan diubah pada form dapat tersimpan ke dalam database.

Berikut adalah langkah-langkah pembuatan file form-edit.php:

1. Buat file dengan nama **form-edit.php**
2. Simpan file tersebut ke dalam folder belajar-crud yang telah kita buat (.../htdocs/belajar-crud/form-edit.php).
3. Ketikkan source code form-edit.php, lihat Gambar 6.6.

Sedangkan untuk pembuatan file proses-edit.php adalah sebagai berikut:

1. Buat file dengan nama **proses-edit.php**
2. Simpan file tersebut ke dalam folder belajar-crud yang telah kita buat (.../htdocs/belajar-crud/proses-edit.php).

Ketikkan source code proses-edit.php, lihat Gambar 6.7.

## 6.6 Hapus Data Mahasiswa (Delete)

Berikut adalah langkah-langkah pembuatan hapus data mahasiswa:

1. Buat file dengan nama **hapus.php**
2. Simpan file tersebut ke dalam folder belajar-crud yang telah kita buat (.../htdocs/belajar-crud/hapus.php).
3. Ketikkan source code hapus.php, lihat Gambar 6.8.

```

1 <?php
2 include("koneksi.php");
3 // kalau tidak ada id di query string
4 if( !isset($_GET['id']) ) {
5     header('Location: index.php');
6 }
7
8 //ambil id dari query string
9 $id = $_GET['id'];
10 // buat query untuk ambil data dari database
11 $sql = "SELECT * FROM mahasiswa WHERE id=$id";
12 $query = mysqli_query($conn, $sql);
13 $mhs = mysqli_fetch_assoc($query);
14 // jika data yang di-edit tidak ditemukan
15 if (mysqli_num_rows($query) < 1 ) {
16     die("data tidak ditemukan...");
17 }
18 ?>
19
20 <!DOCTYPE html>
21 <html>
22 <head>
23     <title>::Edit Data Mahasiswa::</title>
24 </head>
25 <body>
26     <h3>Edit Data Mahasiswa</h3>
27     <form action="proses-edit.php" method="POST">
28     <fieldset>
29     <table>
30         <input type="hidden" name="id" value="<?php echo $mhs['id'] ?>">
31         <input type="hidden" name="nim" value="<?php echo $mhs['nim'] ?>">
32     <tr>
33         <td>NIM:</td>
34         <td><input type="text" name="nim" value="<?php echo $mhs['nim'] ?>" disabled</td>
35     </tr>
36     <tr>
37         <td>Nama:</td>
38         <td><input type="text" name="nama" value="<?php echo $mhs['nama'] ?>"></td>
39     </tr>
40     <tr>
41         <td></td>
42         <td>
43             <input type="submit" name="update" value="Update Data">
44         </td>
45     </tr>
46 </table>
47 </fieldset>
48 </form>
49 <a href="index.php">Kembali ke Halaman Index</a>
50 </body>
51 </html>

```

Gambar 6.6 Source Code form-edit.php

```

1  <?php
2  include("koneksi.php");
3  // cek apakah tombol update sudah diklik atau belum?
4  if (isset($_POST['update'])) {
5      // ambil data dari formulir
6      $id = $_POST['id'];
7      $nim = $_POST['nim'];
8      $nama = $_POST['nama'];
9
10     // buat query update
11     $sql = "UPDATE mahasiswa SET nim='$nim', nama='$nama' WHERE id=$id";
12     $query = mysqli_query($conn, $sql);
13
14     // apakah query update berhasil?
15     if ($query) {
16         // kalau berhasil alihkan ke halaman index.php
17         header('Location: index.php');
18     } else {
19         // kalau gagal tampilkan pesan
20         die("Gagal menyimpan perubahan...");
21     }
22 } else {
23     die("Akses dilarang...");
24 }
25 ?>

```

Gambar 6.7 Source Code proses-edit.php

```

1  <?php
2  include("koneksi.php");
3  if (isset($_GET['id'])) {
4      // ambil id dari query string
5      $id = $_GET['id'];
6      // buat query hapus
7      $sql = "DELETE FROM mahasiswa WHERE id=$id";
8      $query = mysqli_query($conn, $sql);
9
10     // apakah query hapus berhasil?
11     if ($query) {
12         header('Location: index.php');
13     } else {
14         die("gagal menghapus...");
15     }
16 } else {
17     die("akses dilarang...");
18 }
19 ?>

```

Gambar 6.8 Source Code hapus.php

## DAFTAR PUSTAKA

- Delisle, M. (2012). *Mastering phpMyAdmin 2.8for Effective MySQL Management*. Birmingham: Packt Publishing Ltd.
- Duckett, J. (2010). *Begining HTML, XHTML, CSS and JavaScript*. (C. Ullman, Ed.). Canada: Willey Publishing, Inc.
- Gilmore, W. J. (2010). *Beginning PHP and MySQL From Novice to Professional* (4th ed.). United States of America: Apress.
- Hogan, B. P. (2010). *HTML5 and CSS3 Develop with Tomorrow's Standards Today*. United States of America: Pragmatic Programmer, LLC.
- Janet Valade. (2007). *PHP and MySQL For Dummies 3rd Edition*. Canada: Willey Publishing, Inc.
- Marc Rochkind. (2013). *Expert PHP and MySQL*. United States of America: Apress.
- PHP Group. (2018). [php.net/manual/en](http://php.net/manual/en/). Retrieved October 3, 2018, from <http://php.net/manual/en/>
- Timoty Boronczyk. (2015). *Jump Start MySQL*. Cambridge: Sitepoint. <https://doi.org/10.1016/B978-0-08-036833-7.50001-3>
- W3school.com. (2018). W3School. Retrieved October 3, 2018, from <https://www.w3schools.com/php/>