

MODUL V-PHP MYSQL

5.1 Tujuan

Setelah mengikuti praktikum ini mahasiswa diharapkan dapat:

1. Menggunakan MySQL untuk mengelola database
2. Mampu membuat database pada MySQL
3. Mampu membuat tabel pada MySQL melalui aplikasi web
4. Mampu memasukkan data ke dalam tabel di database MySQL
5. Mampu menampilkan data dari table
6. Mampu melakukan proses pencarian data pada table
7. Mampu melakukan delete data pada tabel
8. Mampu melakukan update data pada tabe

5.2 Koneksi PHP MYSQL

Untuk dapat menggunakan dan mengakses database MySQL, PHP menyediakan 3 cara koneksi: menggunakan PDO (PHP Data Objects), mysqli extension dan mysql extension.

5.2.1 Perkembangan Cara Koneksi PHP dan MySQL

PHP merupakan bahasa pemrograman yang selalu di update dan berkembang mengikuti teknologi terbaru. Saat ini, **Pemrograman Berorientasi Objek** (*Object Oriented Programming*) merupakan trend pemrograman PHP, dan hal ini juga mempengaruhi cara mengakses database MySQL dari PHP.

PHP memiliki 3 cara pengaksesan MySQL, yakni melalui **PDO (PHP Data Objects)**, **mysqli extension** dan **mysql extension**. **PDO** menggunakan *pemrograman objek*, **mysqli extension** tersedia dalam bentuk *objek* dan *prosedural* (diakses melalui fungsi-fungsi) sedangkan **mysql extension** sepenuhnya menggunakan pemrograman *prosedural*.

5.2.1.1 Koneksi MySQL dengan mysql extension

Saat pertama kali mempelajari PHP MySQL sekitar tahun 2008 (atau jika anda pernah mempelajari PHP-MySQL beberapa tahun yang lalu), untuk mengakses MySQL dari PHP, kita menggunakan fungsi-fungsi seperti *mysql_connect()*, *mysql_query()*, dan *mysql_fetch_array()*. Fungsi-fungsi ini tergabung ke dalam **mysql extension**.

Namun sekarang (tepatnya mulai PHP versi 5.5.0) PHP memutuskan untuk membuat **mysql extension** berstatus *deprecated*. Yang artinya pengaksesan database MySQL menggunakan fungsi *mysql extension* sudah tidak disarankan lagi. Programmer PHP diharapkan pindah ke **mysqli extension** atau **PDO** yang berbasis objek.

Alasannya, MySQL versi terbaru memiliki fitur-fitur yang semakin lengkap dan kompleks, sehingga PHP memutuskan untuk membuat fungsi extension baru agar programmer PHP bisa menggunakan fitur-fitur ini.

5.2.1.2 Koneksi MySQL dengan mysqli extension

Sebagai pengganti **mysql extension**, PHP menyediakan **mysqli extension** (**mysqli** merupakan singkatan dari **MySQL Improved**). Mysqli extension ini pada dasarnya adalah perbaikan dari *mysql extension* dan dikembangkan untuk mendukung fitur-fitur terbaru untuk **MySQL 4.1** keatas.

Hampir semua fungsi yang ada pada *mysql extension* juga tersedia pada *mysqli*. Syntax (aturan penulisan) **mysqli** sangat mirip dengan **mysql extension**. Sehingga jika anda telah lama menggunakan *mysql extension*, akan sangat mudah untuk beralih menggunakan *mysqli extension*.

Selain menggunakan *mysql* maupun *mysqli extension*, cara ketiga untuk pengaksesan database MySQL dari PHP adalah dengan menggunakan PHP Data Objects (atau sering disingkat dengan **PDO**).

5.2.1.3 Koneksi MySQL dengan PDO (PHP Data Objects)

PDO (PHP Data Objects), adalah *extension* atau penambahan fitur dalam PHP yang dirancang sebagai *interface universal* untuk pengaksesan berbagai jenis database (tidak hanya MySQL). Contohnya, jika kita menggunakan **PDO** dalam menulis kode pemograman, lalu suatu saat website kita bertukar database dari **MySQL** ke **Oracle**, maka kita tidak perlu mengubah semua kode program, cukup mengubah cara pemanggilan **PDO** diawal program saja.

Dari ketiga cara koneksi PHP dengan MySQL ini, metode yang disarankan adalah menggunakan **mysqli** atau **PDO**.

Cara Penulisan *mysql extension*, *mysqli extension*, dan **PDO** (PHP Data Objects)

Untuk mengetahui secara sekilas perbedaan cara pengaksesan database MySQL menggunakan **mysql extension**, **mysqli extension**, dan **PDO**, berikut contoh kode PHP-nya:

```
<?php
// cara mengakses MySQL menggunakan mysql extension
$link = mysql_connect("localhost", "root", "qwerty");
mysql_select_db("universitas");
$result = mysql_query("SELECT * FROM mahasiswa");
$row = mysql_fetch_assoc($result);

// cara mengakses MySQL menggunakan mysqli extension
$mysqli = new mysqli("localhost", "root", "qwerty", "universitas");
$result = $mysqli->query("SELECT * FROM mahasiswa");
$row = $result->fetch_assoc();
```

// cara mengakses MySQL menggunakan PDO

```
$pdo = new PDO('mysql:host=localhost;dbname=universitas', 'root', 'qwerty');  
$statement = $pdo->query("SELECT * FROM mahasiswa");  
$row = $statement->fetch(PDO::FETCH_ASSOC);  
?>
```

Dalam contoh di atas, dianggap bahwa user MySQL adalah **root**, password '**qwerty**', dan nama database '**universitas**'.

Untuk tahap pertama, kita akan menggunakan metode **mysql extension** yang berbasis fungsi terlebih dahulu. Walaupun metode ini tidak disarankan lagi, namun metode inilah yang paling mudah dipelajari dan telah dikenal luas.

Setelah selesai membahas **mysql extension**, selanjutnya kita akan beralih ke **mysqli extension** yang bisa ditulis dengan *procedural style* (menggunakan fungsi-fungsi) ataupun dengan *object style*. Terakhir, kita akan membahas cara membuat koneksi PHP MySQL menggunakan PDO.

5.2.2 Perbedaan Antara mysql extension Dengan mysqli extension

Terhitung dari PHP versi 5.5, extension **mysql** tidak lagi disarankan penggunaannya dan berstatus *deprecated*, yang berarti mungkin akan dihapus pada PHP versi berikutnya. Kita disarankan untuk mulai beralih menggunakan **mysqli** atau **PDO**. Mari kita bahas tentang **mysqli** terlebih dahulu.

Mysqli merupakan kependekan dari **MySQL Improved Extension**. Seperti yang terlihat dari namanya, extension ini merupakan versi perbaikan dan penambahan dari extension **mysql** sebelumnya yang umum digunakan. Extension **mysqli** dibuat untuk mendukung fitur-fitur terbaru dari MySQL Server versi 4.1 keatas.

Secara garis besar, tidak ada perbedaan mencolok antara *mysql extension* dengan *mysqli extension*. Nama-nama fungsi didalam **mysqli** sebagian besar mirip dengan apa yang telah kita pelajari (*extension mysql*).

Sebagai contoh, untuk membuat koneksi dengan MySQL Server, di dalam **mysql** kita menggunakan fungsi **mysql_connect()**, sedangkan di dalam **mysqli**, kita menggunakan **mysqli_connect()**. Begitu juga dengan fungsi lain seperti **mysql_query()** menjadi **mysqli_query()**.

Selain menambah huruf "i" di dalam nama fungsi, argumen-argumen yang dibutuhkan juga hampir mirip. Perbedaannya, jika di dalam extension **mysql** umumnya kita meletakkan argumen *resources* di akhir fungsi, maka di dalam **mysqli**, argumen ini diletakkan di awal.

Sebagai contoh, di dalam **mysql** kita menulis:

```
mysql_query("SELECT * FROM mahasiswa", $link)
```

Sedangkan di dalam **mysqli** penulisannya menjadi:

```
mysqli_query($link, "SELECT * FROM mahasiswa")
```

Namun perbedaan paling mendasar di dalam mysqli adalah: mysqli mendukung cara penulisan **object oriented programming**.

5.2.3 Cara Membuat Koneksi MySQL Dengan Fungsi `mysqli_connect()`

Untuk membuat koneksi antara PHP dengan MySQL Server, kita menggunakan fungsi **`mysqli_connect()`**. Fungsi `mysqli_connect()` ini membutuhkan argumen yang sama dengan fungsi `mysql_connect()`, yakni: **alamat host**, **nama user**, dan **password user**.

Sebagai contoh, untuk masuk kedalam MySQL di **localhost** menggunakan user **root** dan dengan password “**qwerty**”, kita menggunakan kode program sebagai berikut:

```
1 | <?php
2 |     $link = mysqli_connect("localhost", "root", "qwerty");
3 | ?>
```

Sebagai fitur tambahan, fungsi `mysqli_connect()` memiliki argumen ke-4 yang bersifat opsional. Kita bisa menambahkan **nama database** yang ingin digunakan. Misalkan kita ingin sewaktu proses koneksi dengan MySQL, langsung memilih database “*universitas*”, maka contoh penulisan nya adalah sebagai berikut:

```
1 | <?php
2 |     $link = mysqli_connect("localhost", "root", "qwerty", "universitas");
3 | ?>
```

5.2.4 Cara Menutup Koneksi Dengan Fungsi `mysqli_close()`

Walaupun PHP akan menutup koneksi dengan MySQL pada saat halaman selesai diproses, kita bisa menutupnya secara manual menggunakan fungsi **`mysqli_close()`**. Fungsi ini membutuhkan 1 argumen, yakni variabel *resources* hasil pemanggilan fungsi `mysqli_connect()`.

Berikut contoh penggunaan fungsi `mysqli_close()`:

```
1 | <?php
2 |     // buat koneksi dengan MySQL
3 |     $link = mysqli_connect("localhost", "root", "qwerty", "universitas");
4 |
5 |     //.... proses PHP
6 |     //.... proses PHP
7 |     //.... proses PHP
8 |
9 |     // tutup koneksi
10 |    mysqli_close($link);
11 | ?>
```

5.2.5 Cara Menjalankan Query MySQL Dengan Fungsi `mysqli_query()`

Cara menjalankan query MySQL dengan *mysqli extension* juga mirip dengan *mysql extension*. Di dalam `mysqli`, kita menggunakan fungsi **`mysqli_query()`**. Namun berbeda dengan fungsi `mysql_query()` yang meletakkan variabel *resources* hasil **`mysql_connect()`** sebagai

argumen kedua dan bersifat opsional, di dalam fungsi *mysqli_query()*, variabel ini diletakkan sebagai argumen pertama dan harus ditulis.

Sebagai contoh, dengan menggunakan *mysql_extension*, kita biasa menulis kode PHP sebagai berikut:

Sedangkan dengan menggunakan *mysqli extension*, berikut adalah perubahan cara penggunaannya:

```
1  <?php
2      //buat koneksi dengan MySQL
3      $link=mysql_connect('localhost','root','');
4
5      //gunakan database universitas
6      $result=mysql_query('USE universitas',$link);
7
8      //tampilkan tabel mahasiswa_ilkom
9      $result=mysql_query('SELECT * FROM mahasiswa_ilkom',$link);
10 ?>
```

Perhatikan bahwa kita menggunakan format: **mysqli_query(\$link, “query_MySQL”)** untuk menjalankan query MySQL.

```
1  <?php
2      // buat koneksi dengan MySQL, gunakan database: universitas
3      $link = mysqli_connect("localhost", "root", "", "universitas");
4
5      // jalankan query
6      $result = mysqli_query($link, "SELECT * FROM mahasiswa_ilkom");
7  ?>
```

Perhatikan bahwa kita menggunakan format: **mysqli_query(\$link, “query_MySQL”)** untuk menjalankan query MySQL.

5.2.6 Cara Menampilkan Data MySQL Dengan mysqli

Untuk menampilkan hasil query MySQL, *mysqli extension* memiliki banyak fungsi. Kali ini kita akan membahas 3 fungsi yang paling sering digunakan: **mysqli_fetch_row()**, **mysqli_fetch_array()**, dan **mysqli_fetch_object()**.

Ketiga fungsi ini hampir sama cara penggunaannya dengan versi *mysql extension* yang telah kita bahas, yakni: *mysql_fetch_row()*, *mysql_fetch_array()*, dan *mysql_fetch_object()*. Jika anda perhatikan, perbedaan namanya hanya dengan menambah huruf “i”.

Langsung saja kita masuk ke contoh program, untuk memahami cara penggunaan ketiga fungsi ini. Di dalam contoh-contoh berikut kita menggunakan tabel **mahasiswa** di dalam database **universitas**. Tabel ini kita buat sewaktu membahas tentang fungsi mysqli_fetch_row.

Pertama, mari kita lihat cara menampilkan data MySQL dengan fungsi **mysqli_fetch_row()**:

```

1 <?php
2 // buat koneksi dengan MySQL, gunakan database: universitas
3 $link = mysqli_connect("localhost", "root", "", "universitas");
4
5 // jalankan query
6 $result = mysqli_query($link, "SELECT * FROM mahasiswa_ilkom");
7
8 // tampilkan query
9 while ($row=mysqli_fetch_row($result))
10 {
11     echo "$row[0] $row[1] $row[2] $row[3] $row[4]";
12     echo "<br />";
13 }
14 ?>

```

Berikutnya, kita bahas cara menampilkan data MySQL dengan fungsi **mysqli_fetch_array()**:

```

1 <?php
2 // buat koneksi dengan MySQL, gunakan database: universitas
3 $link = mysqli_connect("localhost", "root", "", "universitas");
4
5 // jalankan query
6 $result = mysqli_query($link, "SELECT * FROM mahasiswa_ilkom");
7
8 // tampilkan query
9 while ($row=mysqli_fetch_array($result,MYSQLI_ASSOC))
10 {
11     echo $row['nim']." ".$row['nama']." ".$row['umur']." ";
12     echo $row['tempat_lahir']." ".$row['IPK'];
13     echo "<br />";
14 }
15 ?>

```

Cara ketiga yang akan kita bahas adalah dengan menggunakan fungsi **mysqli_fetch_object()**.

```

1 <?php
2 // buat koneksi dengan MySQL, gunakan database: universitas
3 $link = mysqli_connect("localhost", "root", "", "universitas");
4
5 // jalankan query
6 $result = mysqli_query($link, "SELECT * FROM mahasiswa_ilkom");
7
8 // tampilkan query
9 while ($row=mysqli_fetch_object($result))
10 {
11     echo $row->nim." ".$row->nama." ".$row->umur." ";
12     echo $row->tempat_lahir." ".$row->IPK;
13     echo "<br />";
14 }
15 ?>

```

5.2.7 CONTOH KASUS BUKU TAMU

```
MariaDB [(none)]> create database dbBukuTamu;  
Query OK, 1 row affected (0.01 sec)
```

```
MariaDB [(none)]> use dbBukuTamu;  
Database changed  
MariaDB [dbBukuTamu]> create table bukutamu (  
  -> nama varchar(50) not null  
  -> , email varchar(75) not null,  
  -> komentar varchar(100) not null  
  -> );  
Query OK, 0 rows affected (0.91 sec)
```

```
MariaDB [dbBukuTamu]> desc bukutamu;  
+-----+-----+-----+-----+-----+  
| Field | Type      | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+  
| nama  | varchar(50) | NO   |     | NULL    |      |  
| email | varchar(75) | NO   |     | NULL    |      |  
| komentar | varchar(100) | NO   |     | NULL    |      |  
+-----+-----+-----+-----+-----+  
3 rows in set (0.02 sec)
```

```
MariaDB [dbBukuTamu]> insert into bukutamu values  
  -> ('Gunawan','gunawan.telkompdc@gmail.com','Testing....');  
Query OK, 1 row affected (0.12 sec)
```

```
MariaDB [dbBukuTamu]> insert into bukutamu values  
  -> ('Naufal','naufalg@gmail.com','Testing....Hello World...');  
Query OK, 1 row affected (0.15 sec)  
MariaDB [dbBukuTamu]> select * from bukutamu;
```

```
+-----+-----+-----+  
| nama  | email                  | komentar                  |  
+-----+-----+-----+  
| Gunawan | gunawan.telkompdc@gmail.com | Testing....              |  
| Naufal  | naufalg@gmail.com        | Testing....Hello World... |  
+-----+-----+-----+  
2 rows in set (0.00 sec)
```

```
1 <HTML>
2 <HEAD>
3   <title>Buku Tamu</title>
4 </HEAD>
5 <BODY>
6   <h1>Buku Tamu untuk database MySQL</h1>
7   <form action="prosesTambah.php" method="post">
8     Nama : <input type="text" name="nama" size="35"
9     maxlength="50"> <br>
10    Email : <input type="text" name="email" size="35"
11    maxlength="50"> <br>
12    Komentar : <textarea name="komentar" rows="5"
13    cols="30"></textarea> <br>
14    <input type="submit" value="Simpan">
15    <input type="reset" value="Reset">
16  </form>
17 </BODY>
18 </HTML>
```

```
1 <HTML>
2 <HEAD><title>Simpan Buku Tamu</title></HEAD>
3 <BODY>
4   <h1>Simpan Buku Tamu MySQL</h1>
5   <?php
6     $nama = $_POST["nama"];
7     $email = $_POST["email"];
8     $komentar = $_POST["komentar"];
9     $link = mysqli_connect("localhost","root","","dbBukuTamu") or die ("koneksi gagal");
10    echo "Nama : $nama <br>";
11    echo "Email : $email <br>";
12    echo "Komentar : $komentar <br>";
13    $sqlstr="insert into bukutamu (nama,email,komentar)
14    values ('$nama','$email','$komentar)";
15    $hasil = mysqli_query($link,$sqlstr);
16    echo "Simpan bukutamu berhasil dilakukan";
17  ?>
18 </BODY>
19 </HTML>
```



```
1  <?php
2      $link = mysqli_connect("localhost","root","","dbBukuTamu");
3      $hasil = mysqli_query($link, "select * from bukutamu");
4      $jumlah = mysqli_num_rows($hasil);
5      echo "<center>Daftar Pengunjung</center>";
6      echo "Jumlah pengunjung : $jumlah";
7      $a=1;
8      while($baris = mysqli_fetch_array($hasil))
9      {
10         echo "<br>";
11         echo $a;
12         echo "<br>";
13         echo "Nama : ";
14         echo $baris[0];
15         echo "<br>";
16         echo "Email : ";
17         echo $baris[1];
18         echo "<br>";
19         echo "Komentar : ";
20         echo $baris[2];
21         $a++;
22     }
23  ?>
```

Search.html

```
1 <HTML>
2 <HEAD>
3   <title>Cari Database</title>
4 </HEAD>
5 <BODY>
6   <h1>Searching Buku Tamu untuk database MySQL</h1>
7   <form action="hasilsearch.php" method="post">
8     <select name="kolom">
9       <option value="nama">nama</option>
10      <option value="email">email</option>
11    </select>
12    Masukkan kata yang anda cari
13    <input type="text" type="text" name="cari">
14    <input type="submit" value="cari" >
15  </form>
16 </BODY>
17 </HTML>
```

Hasilsearch.php

```
1 <?php
2   $kolom=$_POST['kolom'];
3   $cari=$_POST['cari'];
4   $link = mysqli_connect("localhost","root","","dbBukuTamu");
5   $hasil=mysqli_query($link,"select * from bukutamu where $kolom like '%$cari%'");
6   $jumlah=mysqli_num_rows($hasil);
7   echo "<br>";
8   echo "Ditemukan: $jumlah";
9   echo "<br>";
10  while($baris=mysqli_fetch_array($hasil))
11  {
12      echo "Nama : ";
13      echo $baris[0];
14      echo "<br>";
15      echo "Email : ";
16      echo $baris[1];
17      echo "<br>";
18      echo "Komentar : ";
19      echo $baris[2];
20  }
21 ?>
```

5.2.8 LATIHAN

1. Buat tabel mahasiswa yang berisi 4 field : NIM, Nama, Alamat, ID_Jur, Foto
2. Buat table jurusan yang berisi 2 field : ID_Jur, Nama
3. Kedua table diatas saling berelasi
4. Buat tampilan web yang berisi :
 - a. Proses untuk tambah data
 - b. Proses untuk search data, dengan masukan berupa Nama. Data yang ditampilkan adalah NIM, Nama, Foto dan nama jurusan.
 - c. Proses untuk delete data, dengan masukan berupa NIM.
 - d. Buat form untuk proses update data berdasarkan NIM.