Nama : Wong, Richard Giovanni Ardie

NIM : 1415015033

### Mengkoneksikan database MySQL di PHP

1. **Modul 8.1**

Sebelum merubah isi tabel dari database, pertama yang harus kita mulai adalah membuat skrip agar terhubung ke database. Fungsi **mysqli\_connect()** digunakan untuk menghubungkan ke database MySQL.

1. <?php
2. $username = "root";
3. $password = "";
4. $database = "PWeb";
5. $hostname = "localhost";
6. mysqli\_connect($hostname, $username, $password);
8. echo "Koneksi Terbuka";
9. ?>

|  |  |
| --- | --- |
|  |  |

Empat baris skrip pertama yang ditunjukkan di skrip tersebut merupakan pengaturan variabel. Variabel pertama adalah **$username** sedangkan variabel kedua adalah **$password**, variabel pertama berisi nilai root sedangkan variabel kedua bernilai "" alias kosong, kita menentukan nilai ini karena nilai ini adalah nilai default untuk database MySQL.

Berikutnya adalah variabel **$database** yang berisi nama database yang ingin dituju dan variabel **$hostname** yang berisi nama host tempat database berasal. Jika anda belum mempunyai server lokal yang berisi PHP dan MySQL.

1. **Modul 8.2**
2. <?php
4. $username = "root";
5. $password = "";
6. $database = "PWeb";
7. $hostname = "localhost";
9. mysql\_connect($hostname, $username, $password);
11. mysql\_select\_db($database);
13. echo "Koneksi Terbuka";
15. ?>

Pada skrip diatas terdapat fungsi **mysqli\_select\_db()**, fungsi ini adalah fungsi built-in PHP yang digunakan untuk memilih database MySQL yang ingin kita hubungkan. Selanjutnya fungsi **mysqli\_select\_db()** mengembalikan nilai true jika database yang kita masukkan ditemukan dan mengembalikan nilai false jika database tidak ditemukan, jika nilai tersebut true maka akan menampilkan pesan “Koneksi Terbuka”.

Skrip diatas memiliki kekurangan dalam hal penampilan pesan jika database yang dituju tidak tersedia, maka dari itu kita membuat dan menggunakan variabel untuk menyimpan nilai **mysqli\_select\_db()** dan **mysqli\_connect()**. Dengan variabel tersebut kita dapat menguji dengan if else untuk memeriksa apakah database ditemukan atau tidak.

Contoh :

1. <?php
3. $username = "root";
4. $password = "";
5. $database = "PWeb";
6. $hostname = "localhost";
8. $connect\_db=mysqli\_connect($hostname, $username, $password);
10. $find\_db=mysqli\_select\_db($database);
12. if ($find\_db) {
14. echo "Database  Ada";
16. }else {
18. echo "Database Tidak Ada";
19. }
20. ?>

### Cara Menutup Koneksi Database MySQL PHP

1. **Modul 8.3**

Dalam PHP, **mysqli\_close()** digunakan untuk menutup koneksi ke server database MySQL. Kita dapat menggunakan skrip berikut untuk menutup koneksi :

1. <?php
3. $username = "root";
4. $password = "";
5. $database = "PWeb";
6. $hostname = "localhost";
8. $connect\_db=mysqli\_connect($hostname, $username, $password);
10. $find\_db=mysqli\_select\_db($database);
12. if ($find\_db) {
14. echo "Database Ada";
16. mysqli\_close($connect\_db);
18. }
19. else {
21. echo "Database Tidak Ada";
23. mysqli\_close($connect\_db);
25. }
27. ?>

## Menampilkan isi Tabel dari Database MySQL PHP

## Modul 8.4

Sekarang kita telah mengetahui bagaimana menghubungkan database dan kemudian menutupnya, selanjutnya mari kita mencoba mengambil data atau membaca data tabel dari database MySQL menggunakan PHP.

Sebelum ditampilkan, maka kita harus memastikan bahwa terdapat tabel dalam database yang kita buka, anggaplah nama tabel tersebut adalah “PW” dan telah memiliki 3 baris isi. Untuk mengambil dan menampilkan isi dari tabel “PW” dalam database “PWeb”, maka kita menggunakan fungsi **mysqli\_fetch\_assoc()**, selengkapnya perhatikan skrip berikut :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34 | <?php    $username = "root";  $password = "";  $database = "PWeb";  $hostname = "localhost";    $connect\_db=mysqli\_connect($hostname, $username, $password);    $find\_db=mysqli\_select\_db($database);    if ($find\_db) {    $query = "SELECT \* FROM PW";  $hasil = mysqli\_query($query);    while ( $kolom\_db = mysql\_ifetch\_assoc($hasil) ) {      echo $kolom\_db['Nama'] . "<br />";    echo $kolom\_db['NIM'] . "<br />";    }      mysqli\_close($connect\_db);    }else {      echo "Database Tidak Ada";      mysqli\_close($connect\_db);    }    ?> |

Perhatikan skrip diatas, jika database tersedia maka query “SELECT \* FROM PWeb” akan berjalan menggunakan fungsi **mysqli\_query()** dan hasilnya disimpan dalam variabel **$hasil**. Fungsi **mysqli\_fetch\_assoc()** digunakan untuk mengambil data dari **$hasil** dan disimpan dalam **$field\_db**. Setelah tersimpan dalam **$field\_db**, maka kita tinggal menampilkan data sesuai nama kolom.

Membuat Tabel di Database MySQL dengan PHP

1. **Modul 8.5**

Untuk membuat tabel dalam database, yang perlu kita lakukan sama seperti kita membuat database. Pertama kita membuat query SQL untuk membuat tabel dan kemudian jalankan query tersebut menggunakan fungsi **mysqli\_query()**.

Perhatikan contoh skrip membuat tabel di database MySQL dengan PHP berikut ini :

1. <?php
2. $dbhost = 'localhost';
3. $dbuser = 'root';
4. $dbpass = 'password';
5. $conn = mysqli\_connect($dbhost, $dbuser, $dbpass);
6. if(! $koneksi )
7. {
8. die('Gagal Koneksi: ' . mysqli\_error());
9. }
10. echo 'Koneksi Berhasil';
11. $sql = 'CREATE TABLE karyawan( '.
12. 'id\_karyawan INT NOT NULL AUTO\_INCREMENT, '.
13. 'nama\_karyawan VARCHAR(20) NOT NULL, '.
14. 'alm\_karyawan  VARCHAR(20) NOT NULL, '.
15. 'gaji\_karyawan   INT NOT NULL, '.
16. 'tgl\_gabung    timestamp(14) NOT NULL, '.
17. 'primary key ( id\_karyawan ))';
19. mysqli\_select\_db('test\_db');
20. $buattabel = mysqli\_query( $sql, $koneksi );
21. if(! $buattabel )
22. {
23. die('Gagal Membuat Tabel: ' . mysql\_error());
24. }
25. echo "Tabel Karyawan sukses dibuat\n";
26. mysqli\_close($koneksi);
27. ?>

Perhatikan isi dari **$sql**, pada variabel tersebut berisi query SQL untuk membuat tabel. Sedikit saran jika anda ingin membuat banyak tabel, maka akan sangat merepotkan jika harus mengetik-kan banyak query, maka dari itu manfaatkan-lah software untuk membuat tabel di MySQL seperti [HeidiSQL](http://www.heidisql.com/).

Cara Menghapus Tabel MySQL di PHP

1. **Modul 8.6**

Sekali lagi kita akan mneggunakan perintah SQL melalui fungsi **mysqli\_query**, kali ini untuk menghapus tabel dalam database. Namun saya ingatkan untuk berhati-hati saat menggunakan perintah ini karena dengan demikian, kita dapat menghapus beberapa informasi penting yang ada dalam tabel.

Perhatikan contoh skrip menghapus tabel MySQL di PHP berikut ini:

1. <?php
2. $dbhost = 'localhost';
3. $dbuser = 'root';
4. $dbpass = 'password';
5. $koneksi = mysqli\_connect($dbhost, $dbuser, $dbpass);
6. if(! $koneksi )
7. {
8. die('Gagal Koneksi: ' . mysqli\_error());
9. }
10. $sql = 'DROP TABLE karyawan';
11. $hapustabel = mysqli\_query( $sql, $koneksi );
12. if(! $hapustabel )
13. {
14. die('Gagal Hapus tabel karyawan: ' . mysqli\_error());
15. }
16. echo "Berhasil hapus tabel karywan\n";
17. mysqli\_close($koneksi );
18. ?>

Setelah memperhatikan kedua skrip diatas, sebenarnya dalam menghapus tabel dan database kita hanya memanfaatkan perintah SQL yang disematkan dalam melalui fungsi **mysqli\_query**, dan pastinya kita bisa mengubah dengan perintah sql lainnya.

Cara Menghapus Database MySQL di PHP

1. **Modul 8.7**

Jika database tidak lagi diperlukan maka dapat dihapus selamanya. Kita dapat menggunakan perintah SQL dalam **mysqli\_query** untuk menghapus database.

Perhatikan contoh skrip menghapus database MySQL di PHP berikut ini:

1. <?php
2. $dbhost = 'localhost';
3. $dbuser = 'root';
4. $dbpass = 'password';
5. $koneksi = mysqli\_connect($dbhost, $dbuser, $dbpass);
6. if(! $koneksi )
7. {
8. die('Gagal Koneksi: ' . mysqli\_error());
9. }
10. $sql = 'DROP DATABASE test\_db';
11. $hapusdb = mysqli\_query( $sql, $koneksi );
12. if(! $hapusdb )
13. {
14. die('Gagal hapus database test\_db: ' . mysqli\_error());
15. }
16. echo "Database berhasil dihapus\n";
17. mysqli\_close($koneksi);
18. ?>

Insert Data ke Tabel MySQL tanpa form HTML

1. **Modul 8.8**

Data dapat dimasukkan ke dalam tabel MySQL dengan mengeksekusi pernyataan INSERT SQL melalui fungsi PHP **mysqli\_query**. Di bawah contoh sederhana untuk menyisipkan catatan ke dalam tabel karyawan.

Perhatikan contoh skrip insert data ke tabel di database MySQL dengan PHP berikut ini:

1. <?php
2. $dbhost = 'localhost';
3. $dbuser = 'root';
4. $dbpass = 'password';
5. $koneksi = mysqli\_connect($dbhost, $dbuser, $dbpass);
6. if(! $koneksi )
7. {
8. die('Gagal Koneksi: ' . mysqli\_error());
9. }
10. $sql = 'INSERT INTO karyawan '.
11. '(nama\_karyawan,alm\_karyawan, gaji\_karyawan, tgl\_gabung) '.
12. 'VALUES ( "nyekrip", "alamat nyekrip", 34000, NOW() )';
13. mysqli\_select\_db('test\_db');
14. $tambahdata = mysqli\_query( $sql, $koneksi );
15. if(! $tambahdata )
16. {
17. die('Gagal tambah data: ' . mysqli\_error());
18. }
19. echo "Berhasil tambah data\n";
20. mysqli\_close($koneksi);
21. ?>

Insert Data ke Tabel MySQL dengan form HTML

1. **Modul 8.9**

Dalam aplikasi nyata, kebanyakan proses insert data melalui form dalam HTML, semua nilai akan diambil menggunakan formulir HTML dan kemudian nilai-nilai tersebut akan ditangkap dengan menggunakan skrip PHP (GET atau POST) dan akhirnya nilai-nilai tersebut dimasukkan ke dalam tabel MySQL.

Saat melakukan insert atau tambah data, praktek terbaik yaitu menambahkan fungsi **get\_magic\_quotes\_gpc()** untuk memeriksa apakah konfigurasi saat ini untuk magic quote sudah diatur atau tidak. Fungsi ini sangat bermanfaat, kebanyakan fungsi ini dipakai untuk mencegah error ketika string query yang dijalankan (dieksekusi) mengandung karakter khusus.

DeprecatedUpdate: Fungsi get\_magic\_quotes\_gpc() telah usang pada PHP 5.3.0 dan DIHAPUS di PHP 5.4.0.

Perhatikan contoh skrip insert data ke tabel di database MySQL dengan PHP yang telah dilengkapi dengan form HTML berikut ini:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75 | <html>  <head>  <title>Tambah DATA ke DATABASE - Nyekrip</title>  </head>  <body>  <?php  if(isset($\_POST['tambah']))  {  $dbhost = 'localhost';  $dbuser = 'root';  $dbpass = '';  $koneksi = mysqli\_connect($dbhost, $dbuser, $dbpass);  if(! $koneksi )  {    die('Gagal Koneksi: ' . mysqil\_error());  }    if(! get\_magic\_quotes\_gpc() )  {     $nama\_karyawan = addslashes ($\_POST['nama\_karyawan']);     $alm\_karyawan = addslashes ($\_POST['alm\_karyawan']);  }  else  {     $nama\_karyawan = $\_POST['nama\_karyawan'];     $alm\_karyawan = $\_POST['alm\_karyawan'];  }  $gaji\_karyawan = $\_POST['gaji\_karyawan'];    $sql = "INSERT INTO karyawan ".         "(nama\_karyawan,alm\_karyawan, gaji\_karyawan, tgl\_gabung) ".         "VALUES('$nama\_karyawan','$alm\_karyawan',$gaji\_karyawan, NOW())";  mysqli\_select\_db('test\_db');  $tambahdata = mysqli\_query( $sql, $koneksi );  if(! $tambahdata )  {    die('Gagal Tambah Data: ' . mysqli\_error());  }  echo "Berhasil tambah data\n";  mysqli\_close($koneksi);  }  else  {  ?>  <form method="post" action="index.php">  <table width="500" border="0" cellspacing="1" cellpadding="2">  <tr>  <td width="100">Nama Karyawan</td>  <td><input name="nama\_karyawan" type="text" id="nama\_karyawan"></td>  </tr>  <tr>  <td width="110">Alamat Karyawan</td>  <td><input name="alm\_karyawan" type="text" id="alm\_karyawan"></td>  </tr>  <tr>  <td width="110">Gaji Karyawan</td>  <td><input name="gaji\_karyawan" type="text" id="gaji\_karyawan"></td>  </tr>  <tr>  <td width="110"> </td>  <td> </td>  </tr>  <tr>  <td width="110"> </td>  <td>  <input name="tambah" type="submit" id="tambah" value="Tambah Karyawan">  </td>  </tr>  </table>  </form>  <?php  }  ?>  </body>  </html> |

Dalam skrip diatas kita tidak memasukkan id\_karyawan pada saat proses input, karena kolom ini sudah di-setting menjadi AUTO INCREMENT alias akan membuat isi secara otomatis.

Cara Menampilkan Data di Tabel Database

1. **Modul 8.10**

Data dapat diambil dari tabel MySQL di PHP dengan mengeksekusi pernyataan SQL SELECT melalui fungsi **mysqli\_query**.

Untuk menarik baris data dalam tabel, pilihan yang paling sering digunakan adalah dengan menggunakan fungsi **mysqli\_fetch\_array ()**. Fungsi ini mengembalikan baris data sebagai array asosiatif, array numerik, atau keduanya. Fungsi ini mengembalikan FALSE jika tidak ada lagi baris yang ditemukan.

Di bawah ini adalah contoh skrip sederhana untuk mengambil data dari tabel karyawan.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | <?php  $dbhost = 'localhost';  $dbuser = 'root';  $dbpass = '';  $koneksi = mysqli\_connect($dbhost, $dbuser, $dbpass);  if(! $koneksi )  {    die('Gagal Koneksi: ' . mysqli\_error());  }  $sql = 'SELECT id\_karyawan, nama\_karyawan, gaji\_karyawan FROM karyawan';    mysqli\_select\_db('test\_db');  $ambildata = mysqli\_query( $sql, $koneksi);  if(! $ambildata )  {    die('Gagal ambil data: ' . mysqli\_error());  }  while($row = mysqli\_fetch\_array($ambildata, MYSQL\_ASSOC))  {      echo "ID KARYAWAN :{$row['id\_karyawan']}  <br> ".           "NAMA KARYAWAN : {$row['nama\_karyawan']} <br> ".           "GAJI KARYAWAN : {$row['gaji\_karyawan']} <br> ".           "--------------------------------<br>";  }  echo "Berhasil ambil data\n";  mysqli\_close($koneksi);  ?> |

Cara update data MySQL melalui PHP

Cara update data MySQL melalui PHP bisa dibilang cukup mudah karena kita hanya butuh mengeksekusi pernyataan UPDATE SQL melalui fungsi PHP **mysqli\_query**.

Pada contoh di bawah, data kolom id\_karyawan yang dijadikan klausa bersyarat telah di format menjadi primary key untuk menjaga keunikan tiap row. Silahkan anda coba contoh skrip berikut untuk memahami operasi update. Kita menggunakan ID karyawan untuk memperbarui gaji karyawan.

Contoh :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64 | <html>  <head>    <title>Update data dalam database MySQL</title>  </head>  <body>    <?php  if(isset($\_POST['update']))  {  $dbhost = 'localhost';  $dbuser = 'root';  $dbpass = '';  $conn = mysqli\_connect($dbhost, $dbuser, $dbpass);  if(! $conn )  {    die('Tidak bisa Koneksi: ' . mysqli\_error());  }    $id = $\_POST['id\_karyawan'];  $gaji = $\_POST['gaji\_karyawan'];    $sql = "UPDATE karyawan ".     "SET gaji\_karyawan = $gaji ".     "WHERE id\_karyawan = $id" ;    mysqli\_select\_db('db\_karyawan');  $retval = mysqli\_query( $sql, $conn );  if(! $retval )  {    die('Tidak bisa update data: ' . mysqli\_error());  }  echo "Updated data sukses\n";  mysqli\_close($conn);  }  else  {  ?>  <form method="post" action="index.php">  <table width="500" border="0" cellspacing="1" cellpadding="2">  <tr>  <td width="100">ID Karyawan</td>  <td><input name="id\_karyawan" type="text" id="id\_karyawan"></td>  </tr>  <tr>  <td width="100">Gaji Karyawan</td>  <td><input name="gaji\_karyawan" type="text" id="gaji\_karyawan"></td>  </tr>  <tr>  <td width="100"> </td>  <td> </td>  </tr>  <tr>  <td width="100"> </td>  <td>  <input name="update" type="submit" id="update" value="Update">  </td>  </tr>  </table>  </form>  <?php  }  ?>  </body>  </html> |