

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN WEB & MOBILE I**



**NAMA : Jorgi Jacko Excel**  
**NIM : 193030503064**  
**MODUL : III (PHP MySQL Database)**

**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PALANGKARAYA**  
**2021**

## **BAB I**

### **TUJUAN DAN LANDASAN TEORI**

#### **1.1 Tujuan**

- Mahasiswa mampu membuat program yang bisa menyimpan data dalam jumlah yang banyak.
- Mahasiswa mampu membuat program yang bisa mengolah data yang tersimpan dalam database.

#### **1.2 Landasan Teori**

Data dalam database MySQL disimpan dalam tabel-tabel. Sebuah tabel adalah koleksi dari data yang berelasi dan mengandung kolom dan baris. Database sangat bermanfaat untuk menyimpan informasi secara kategori. Contoh yang akan diberikan pada modul praktikum ini adalah tabel yang mengandung data Employees (Pekerja), Products (Produk), Customers (Pelanggan) dan Orders (Pesanan).

##### **1.2.1 Membuka Koneksi**

Sebelum mengakses data dalam database MySQL, kita harus terhubung ke server database MySQL. Berikut adalah contoh kode program agar terhubung dengan server mySQL:

```
<?php
    $servername = "localhost";
    $username = "username";
    $password = "password";
    // Membuat hubungan
    $conn = new mysqli($servername, $username, $password);
    // Memeriksa hubungan
    if ($conn->connect_error){
        die("Connection failed: " . $conn->connect_error);
    }
```

```
        echo "Connected successfully";  
    ?>
```

kode diatas tidak berhasil, kemungkinan variabel `$connect_error` sudah tidak tersedia pada versi PHP yang kita gunakan, maka gunakan kode program berikut:

```
<?php  
    $servername = "localhost";  
    $username = "username";  
    $password = "password";  
    // Membuat Hubungan  
    $conn = mysqli_connect($servername, $username, $password);  
    // Memeriksa  
    Hubungan if (!$conn){  
        die("Connection failed: " . mysqli_connect_error());  
    }  
    echo "Connected successfully";  
    ?>
```

Ketika selesai menggunakan data dari database, sebaiknya koneksi atau hubungan ke server ditutup, caranya dengan menggunakan kode program berikut:

```
    mysqli_close($conn);
```

### 1.2.2 Membuat Database

Database pada MySQL bisa juga dibuat menggunakan kode program PHP. Program tersebut akan berisi statement SQL “CREATE DATABASE”. Berikut adalah contoh program yang digunakan untuk membuat database “myDB”:

```
<?php  
    $servername = "localhost";  
    $username = "username";
```

```

$password = "password";
// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password);
// Memeriksa Koneksi
if (!$conn){
    die("Connection failed: " . mysqli_connect_error());
}
// Membuat Database
$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {
    echo "Database created successfully";
} else {
    echo "Error creating database: " . mysqli_error($conn);
}
mysqli_close($conn);
?>

```

### 1.2.3 Membuat Tabel

Pembuatan tabel pada bahasa pemrograman PHP juga menggunakan statement SQL, yaitu statement “CREATE TABLE”. Contoh yang akan diberikan adalah pembuatan tabel MyGuests. Statement pembuatan tabel MyGuests adalah sebagai berikut:

```

CREATE TABLE MyGuests (
    id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY
    KEY, firstname VARCHAR(30) NOT NULL, lastname
    VARCHAR(30) NOT NULL,
    email VARCHAR(50),
    reg_date TIMESTAMP
)

```

Berikut adalah contoh program untuk membuat tabel diatas:

```
<?php
```

```

$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa koneksi
if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
// SQL untuk membuat tabel
$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY
KEY, firstname VARCHAR(30) NOT NULL, lastname
VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP
)";
if (mysqli_query($conn, $sql)) {
echo "Table MyGuests created successfully";
} else {
echo "Error creating table: " . mysqli_error($conn);
}
mysqli_close($conn);
?>

```

#### 1.2.4 Memasukkan Data ke Database

Untuk memasukkan data kedalam tabel di database, terdapat beberapa aturan sintaks yang harus diikuti:

- Query SQL harus diberikan kutip dalam PHP.
- Nilai string didalam query SQL harus diberikan kutip.

- Nilai numeris tidak harus diberikan kutip.
- Nilai NULL tidak harus diberikan kutip.

Statement INSERT INTO digunakan untuk menambahkan baris pada tabel MySQL, query untuk menambahkan data tersebut adalah sebagai berikut:

```
INSERT INTO table_name (column1, column2, column3,...)
```

```
VALUES (value1, value2, value3,...)
```

adapun contoh kode program PHP untuk menambahkan data tersebut adalah sebagai berikut:

```
<?php
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "myDB";
    // Membuat Koneksi
    $conn = mysqli_connect($servername, $username, $password,
    $dbname);
    // Memeriksa Koneksi
    if (!$conn){
        die("Connection failed: " . mysqli_connect_error());
    }
    $sql = "INSERT INTO MyGuests (firstname, lastname, email)
    VALUES ('John', 'Doe', 'john@example.com)";
    if (mysqli_query($conn, $sql)) {
        echo "New record created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
```

```
}  
mysqli_close($conn);  
?>
```

Jika kita melakukan perintah INSERT pada tabel yang menggunakan field dengan opsi AUTO\_INCREMENT, kita bisa mendapatkan ID dari baris yang terakhir diinputkan. Caranya menggunakan fungsi mysqli\_insert\_id(\$conn); nilai kembalian dari fungsi ini akan memberikan id dari record atau baris terakhir yang diinputkan.

Untuk menambahkan record dengan jumlah yang banyak, kita bisa menggunakan fungsi mysqli\_multi\_query(\$conn, \$sql), berikut adalah contoh program untuk menambahkan record dengan jumlah yang banyak:

```
<?php  
  
$servername = "localhost";  
$username = "username";  
$password = "password";  
$dbname = "myDB";  
  
// Membuat Koneksi  
$conn = mysqli_connect($servername, $username, $password,  
$dbname);  
  
// Memeriksa Koneksi  
if (!$conn){  
die("Connection failed: " . mysqli_connect_error());  
}  
  
$sql = "INSERT INTO MyGuests (firstname, lastname, email)  
VALUES ('John', 'Doe', 'john@example.com')";  
$sql .= "INSERT INTO MyGuests (firstname, lastname, email)  
VALUES ('Mary', 'Moe', 'mary@example.com')";  
$sql .= "INSERT INTO MyGuests (firstname, lastname, email)  
VALUES ('Julie', 'Dooley', 'julie@example.com')";  
if (mysqli_multi_query($conn, $sql)) {
```

```

        echo "New records created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
    }
    mysqli_close($conn);
?>

```

### 1.2.5 Mengambil Data dari Database

Untuk mengambil data, statement SQL yang digunakan adalah SELECT nama\_kolom() FROM nama\_tabel, atau kita bisa menggunakan karakter \* untuk memilih semua kolom yang ada pada tabel. Pengambilan data dengan kriteria tertentu bisa dilakukan dengan menggunakan statement WHERE setelah nama\_tabel. Adapun contoh kode program untuk mengambil data dari database adalah sebagai berikut:

```

<?php

$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);

// Memeriksa Koneksi
if (!$conn){
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "SELECT id, firstname, lastname FROM
MyGuests"; $result = mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
    // Menampilkan data pada setiap baris
    while($row = mysqli_fetch_assoc($result)) {

```



```

        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " .
        $row["lastname"]. "<br>";
    }
    } else {
        echo "0 results";
    }
    mysqli_close($conn);
?>

```

### 1.2.6 Menghapus Data dari Database

Statement DELETE digunakan untuk menghapus baris data atau records dari tabel. Clause WHERE digunakan untuk menspesifikasikan baris yang akan dihapus. Jika statement DELETE digunakan tanpa menggunakan clausa WHERE, maka semua record yang ada pada tabel akan dihapus.

id	namadepan	namabelakang	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30
3	Julie	Dooley	julie@example.com	2014-10-26 10:48:23

Jika kita memiliki data seperti tabel “MyGuest” seperti terlihat diatas, kemudian kita ingin menghapus data dengan nama depan Julie, maka kode program untuk menghapus data tersebut adalah sebagai berikut:

```

<?php
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "myDB";
    // Membuat Koneksi

```

```

$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa koneksi
if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
// sql untuk menghapus record
$sql = "DELETE FROM MyGuests WHERE id=3";
if (mysqli_query($conn, $sql)) {
echo "Record deleted successfully";
} else {
echo "Error deleting record: " . mysqli_error($conn);
}
mysqli_close($conn);
?>

```

### 1.2.7 Update Data dalam Database

Untuk melakukan perubahan data di dalam database, statement UPDATE digunakan, yaitu sebagai berikut:

```

UPDATE nama_tabel
SET kolom1=nilai1, kolom2=nilai2, ...
WHERE kolom_penentu=nilai_penentu ...

```

Jika statement ini digunakan untuk update data ke dua pada tabel “MyGuest” diatas untuk mengganti nama belakang dari “Moe” menjadi “Doe”, maka kode program untuk update tersebut adalah sebagai berikut

```

<?php
$servername = "localhost";
$username = "username";
$password = "password";

```

```
$dbname = "myDB";  
// Membuat Koneksi  
$conn = mysqli_connect($servername, $username, $password,  
$dbname);  
// Memeriksa Koneksi  
if (!$conn){  
die("Connection failed: " . mysqli_connect_error());  
}  
$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";  
if (mysqli_query($conn, $sql)) {  
echo "Record updated successfully";  
} else {  
echo "Error updating record: " . mysqli_error($conn);  
}  
mysqli_close($conn);
```

?>

## BAB II

### PEMBAHASAN

Database.sql

```
create database Karyawan;

use karyawan;

CREATE TABLE `info_karyawan` (
  `id` int(11) NOT NULL auto_increment,
  `nama` varchar(100),
  `TTL` varchar(100),
  `alamat` varchar(15),
  `jenis_kelamin` varchar(15),
  `no_telpon` varchar(15),
  PRIMARY KEY (`id`)
);
```

Pertama kita buat terlebih dahulu file database.sql kemudian kita impor file database.sql ini ke dalam database (menggunakan PHPMysqlAdmin)

Config.php

```
<?php

$dbHost = 'localhost';
$dbName = 'karyawan';
$dbUser = 'root';
$dbPass = '';

$mysqli = mysqli_connect($dbHost, $dbUser, $dbPass, $dbName);
```

```
Password, $databaseName);
```

```
?>
```

File `**config.php` menyimpan informasi tentang database host, username dan password. Sebagian besar server lokal bekerja dengan detail yang diberikan. Disini dapat diubah sesuai dengan detail host dan database kita sendiri. Untuk menghubungkan PHP dengan MySQL, kita menggunakan fungsi `mysqli_connect()` dengan alamat server sebagai parameter pertama, user database sebagai parameter kedua, password user sebagai parameter ketiga, dan nama database sebagai parameter keempat.

Index.php

```
<?php
// Create database connection using config file
include_once("config.php");

// Fetch all users data from database
$result = mysqli_query($mysqli, "SELECT * FROM info_karyawan ORDER BY id DESC");
?>

<html>
<head>
    <title>Homepage</title>
</head>

<body>
<a href="add.php">Add New User</a><br/><br/>

    <table width='80%' border=1>
```

```

<tr>
    <th>Name</th> <th>TTL</th> <th>Alamat</th> <th>Jenis_Kelamin</th>
<th>No_Telpon</th> <th>Update</th>
</tr>
<?php
while($user_data = mysqli_fetch_array($result)) {
    echo "<tr>";
    echo "<td>".$user_data['nama']. "</td>";
    echo "<td>".$user_data['TTL']. "</td>";
    echo "<td>".$user_data['alamat']. "</td>";
    echo "<td>".$user_data['jenis_kelamin']. "</td>";
    echo "<td>".$user_data['no_telpon']. "</td>";
    echo "<td><a href='edit.php?id=$user_data[id]>Edit</a> | <a href='delete.p
hp?id=$user_data[id]>Delete</a></td></tr>";
}
?>
</table>
</body>
</html>

```

File index.php merupakan file utama yang menyertakan file konfigurasi untuk koneksi database. Kemudian menampilkan semua daftar pengguna menggunakan MySQL Select Query. Pengguna yang akan ditampilkan di dalam daftar perlu menambahkan terlebih dahulu menggunakan tautan ‘Tambahkan Pengguna Baru’.

Add.php

```

<html>
<head>
    <title>Add Info Karyawan</title>
</head>

```

```
<body>
  <a href="index.php">Go to Home</a>
  <br/><br/>

  <form action="add.php" method="post" name="form1">
    <table width="25%" border="0">
      <tr>
        <td>nama</td>
        <td><input type="text" name="nama"></td>
      </tr>
      <tr>
        <td>TTL</td>
        <td><input type="text" name="TTL"></td>
      </tr>
      <tr>
        <td>alamat</td>
        <td><input type="text" name="alamat"></td>
      </tr>
      <tr>
        <td>jenis_kelamin</td>
        <td><input type="text" name="jenis_kelamin"></td>
      </tr>
      <tr>
        <td>no_telpon</td>
        <td><input type="text" name="no_telpon"></td>
      </tr>
      <tr>
        <td></td>
        <td><input type="submit" name="Submit" value="Add"></td>
      </tr>
    </table>
  </form>

```

```

        </table>
    </form>

    <?php

    // Check If form submitted, insert form data into users table.
    if(isset($_POST['Submit'])) {
        $nama = $_POST['nama'];
        $TTL = $_POST['TTL'];
        $alamat = $_POST['alamat'];
        $jenis_kelamin = $_POST['jenis_kelamin'];
        $no_telpon = $_POST['no_telpon'];

        // include database connection file
        include_once("config.php");

        // Insert user data into table
        $result = mysqli_query($mysqli, "INSERT INTO info_karyawan(nama,TTL,
alamat,jenis_kelamin,no_telpon) VALUES('$nama','$TTL','$alamat', '$jenis_kela
min', '$no_telpon')");

        // Show message when user added
        echo "User added successfully. <a href='index.php'>View Users</a>";
    }
    ?>
</body>
</html>

```

File add.php berfungsi untuk menambahkan pengguna baru. Formulir HTML digunakan untuk menerima masukan data pengguna. Setelah data pengguna diserahkan, MySQL INSERT Query digunakan untuk memasukkan data pengguna ke dalam database.



## Edit.php

```
<?php
// include database connection file
include_once("config.php");

// Check if form is submitted for user update, then redirect to homepage after update
if(isset($_POST['update']))
{
    $id = $_POST['id'];

    $nama = $_POST['nama'];
    $TTL = $_POST['TTL'];
    $alamat = $_POST['alamat'];
    $jenis_kelamin = $_POST['jenis_kelamin'];
    $no_telpon = $_POST['no_telpon'];

    // update user data
    $result = mysqli_query($mysqli, "UPDATE info_karyawan SET nama='$nama',TTL='$TTL',alamat='$alamat', jenis_kelamin='$jenis_kelamin',no_telpon='$no_telpon' WHERE id=$id");

    // Redirect to homepage to display updated user in list
    header("Location: index.php");
}
?>
<?php
// Display selected user data based on id
// Getting id from url
$id = $_GET['id'];
```

```

// Fetech user data based on id
$result = mysqli_query($mysqli, "SELECT * FROM info_karyawan WHERE id=
$id");

while($user_data = mysqli_fetch_array($result))
{
    $nama = $user_data['nama'];
    $TTL = $user_data['TTL'];
    $alamat = $user_data['alamat'];
    $jenis_kelamin = $user_data['jenis_kelamin'];
    $no_telpon = $user_data['no_telpon'];
}
?>
<html>
<head>
    <title>Edit User Data</title>
</head>

<body>
    <a href="index.php">Home</a>
    <br/><br/>

    <form name="update_user" method="post" action="edit.php">
        <table border="0">
            <tr>
                <td>nama</td>
                <td><input type="text" name="nama" value=<?php echo $nama;?></t
d>
            </tr>
            <tr>

```

```

        <td>TTL</td>
        <td><input type="text" name="TTL" value=<?php echo $TTL;?>></td>
    >
</tr>
<tr>
    <td>alamat</td>
    <td><input type="text" name="alamat" value=<?php echo $alamat;?>>
</td>
</tr>
<tr>
    <td>jenis_kelamin</td>
    <td><input type="text" name="jenis_kelamin" value=<?php echo $jenis
_kelamin;?>></td>
</tr>
<tr>
    <td>no_telpon</td>
    <td><input type="text" name="no_telpon" value=<?php echo $no_telpo
n;?>></td>
</tr>
<tr>
    <td><input type="hidden" name="id" value=<?php echo $_GET['id'];?
>></td>
    <td><input type="submit" name="update" value="Update"></td>
</tr>
</table>
</form>
</body>
</html>

```

Edit.php digunakan untuk mengedit / update data pengguna. Anda dapat mengubah data pengguna dan memperbaruinya. File ini akan mengarahkan pengguna kembali ke homepage, setelah update sukses.

## Delete.php

```
<?php
// include database connection file
include_once("config.php");

// Get id from URL to delete that user
$id = $_GET['id'];

// Delete user row from table based on given id
$result = mysqli_query($mysqli, "DELETE FROM info_karyawan WHERE id=$id");

// After delete redirect to Home, so that latest user list will be displayed.
header("Location:index.php");
?>
```

File **\*\* delete.php\*\*** hanya dipanggil saat kita klik link 'Delete' untuk pengguna manapun yang dipilih.

## **BAB III**

### **KESIMPULAN**

PHP adalah bahasa pemrograman script yang paling banyak dipakai saat ini. PHP banyak dipakai untuk memprogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain.

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multithread, multi-user.

PHP sendiri sebenarnya merupakan singkatan dari Hypertext Preprocessor, yang merupakan sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML. MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (DBMS) yang multithread, dan multi-user. MySQL adalah implementasi dari system manajemen basisdata relasional (RDBMS). Kata kunci: PHP, MySQL, HTML, DBMS

Pendahuluan PHP sendiri sebenarnya merupakan singkatan dari Hypertext Preprocessor, yang merupakan sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML. Sebagian besar sintaks dalam PHP mirip dengan bahasa C, Java dan Perl, namun pada PHP ada beberapa fungsi yang lebih spesifik. Sedangkan tujuan utama dari penggunaan bahasa ini adalah untuk memungkinkan perancang web yang dinamis dan dapat bekerja secara otomatis. MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (DBMS) yang multithread, dan multi-user. MySQL adalah implementasi dari system manajemen basisdata relasional (RDBMS).

## DAFTAR PUSTAKA

*Cara Membuat Koneksi PHP ke Database MySQL.* (n.d.). Diambil 12 April 2021, dari <https://www.hostinger.co.id/tutorial/koneksi-database-php>

*How to Use MySQL Database in PHP - Tutorial Republic.* (n.d.). Diambil 12 April 2021, dari <https://www.tutorialrepublic.com/php-tutorial/php-mysql-introduction.php>

*PHP: MySQL Database.* (n.d.). Diambil 12 April 2021, dari [https://www.w3schools.com/php/php\\_mysql\\_intro.asp](https://www.w3schools.com/php/php_mysql_intro.asp)

*Tutorial PHP dan MySQL: Membuat Aplikasi CRUD [Studi Kasus Pendaftaran Siswa Baru].* (n.d.). Diambil 12 April 2021, dari <https://www.petanikode.com/tutorial-php-mysql/>