

LAPORAN PRAKTIKUM
PEMROGRAMAN WEB & MOBILE



NAMA : AHMAD FALDHI YUDianto
NIM : E1E118064
KELAS : C
MODUL : III

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021

BAB I

LANDASAN TEORI

1.1 Tujuan Praktikum

1. Mahasiswa mampu membuat program yang bisa menyimpan data dalam jumlah yang banyak.
2. 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).

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";
?>
```

Gambar 1.1 Koneksi

Jika 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";

?>
```

Gambar 1.2 Koneksi

Ketika selesai menggunakan data dari database, sebaiknya koneksi atau hubungan ke server ditutup, caranya dengan menggunakan kode program berikut: **mysqli_close(\$conn);**

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”:

```
<html>
  <body>

    <form action="welcome_get.php" method="get">
      Name: <input type="text" name="name"><br>
      E-mail: <input type="text" name="email"><br>
      <input type="submit">
    </form>
  </body>
</html>
```

Gambar 1.3 HTML dan PHP

```

        echo "Database created successfully";
    } else {
        echo "Error creating database: " . mysqli_error($conn);
    }

    mysqli_close($conn);
?>

```

Gambar 1.4 Membuat Database

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 (

```

Gambar 1.5 Membuat Tabel

```

        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);
?>

```

Gambar 1.6 Membuat Tabel

Memasukkan Data ke Database

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

1. Query SQL harus diberikan kutip dalam PHP.
2. Nilai string didalam query SQL harus diberikan kutip.
3. Nilai numeris tidak harus diberikan kutip.
4. Nilai NULL tidak harus diberikan kutip.

Statement INSERT INTO digunakan untuk menambahkan baris pada tabelMySQL, 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";

```

Gambar 1.7 Menambahkan Data

```

$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);
?>

```

Gambar 1.8 Menambahkan Data

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){

```

Gambar 1.9 Menambahkan Data

```

        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);
?>

```

Gambar 1.10 Menambahkan Data

Text Field

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 table. 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());
    }

```

Gambar 1.11 Mengambil Data

```

$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);
?>

```

Gambar 1.12 Mengambil Data

Menghapus Data

Dari Database statement DELETE digunakan untuk menghapusbaris 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.

Jika kita memiliki data seperti table “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);

```

Gambar 1.13 Menghapus Data


```

// 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);
?>

```

Gambar 1.14 Menghapus Data

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){

```

Gambar 1.15 Update Data

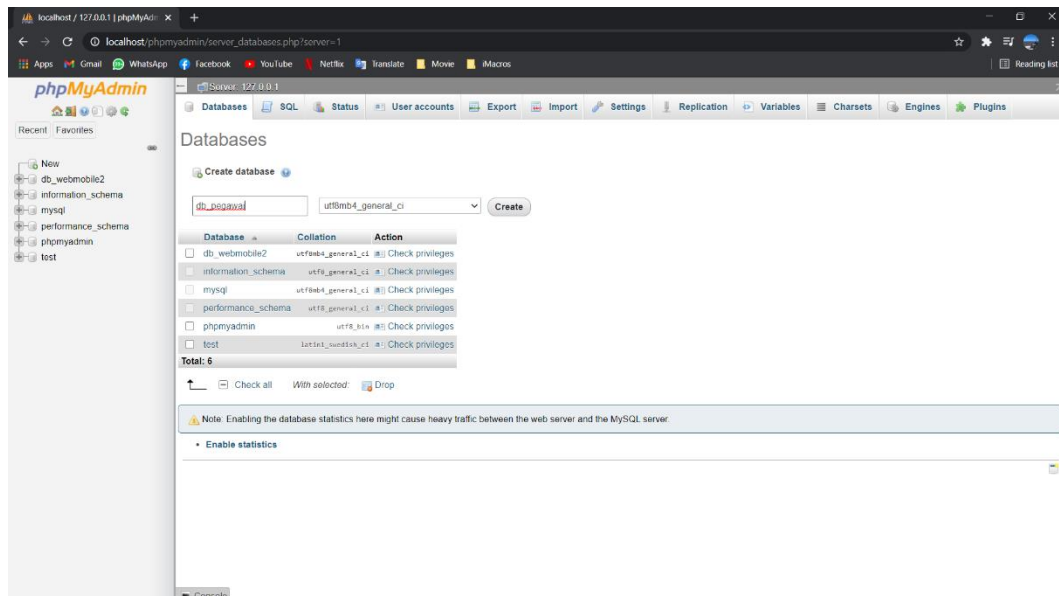
BAB II

PEMBAHASAN

Buatlah program web untuk mengolah nama-nama anggota keluarga anda dengan kriteria sebagai berikut :

1. Buatlah database pada MySQL untuk menyimpan data-data pegawai beserta tabelnya.
2. Buatlah program untuk menyimpan, menghapus dan mengubah data-data pegawai tersebut.

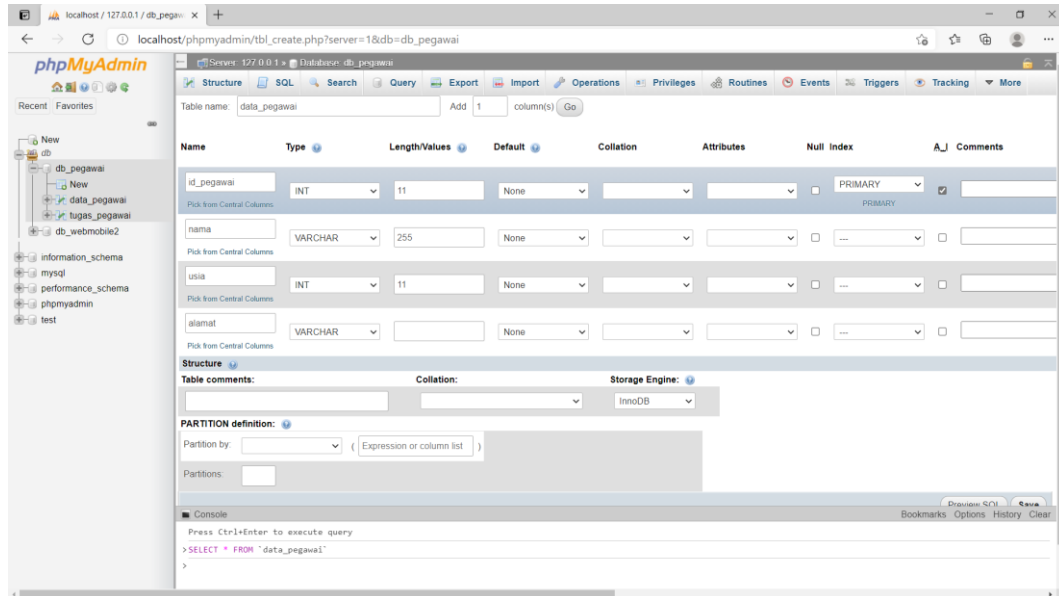
Pada bagian pertama dimana perintah tersebut meminta pembuatan database pada MySQL untuk menyimpan data-data pegawai beserta relasinya. Pembuatan database tersebut melalui localhost/phpmyadmin, setelah itu isi nama database dengan nama “db_pegawai” kemudian tekan tombol create, seperti berikut:



Gambar 2.1 Membuat Database

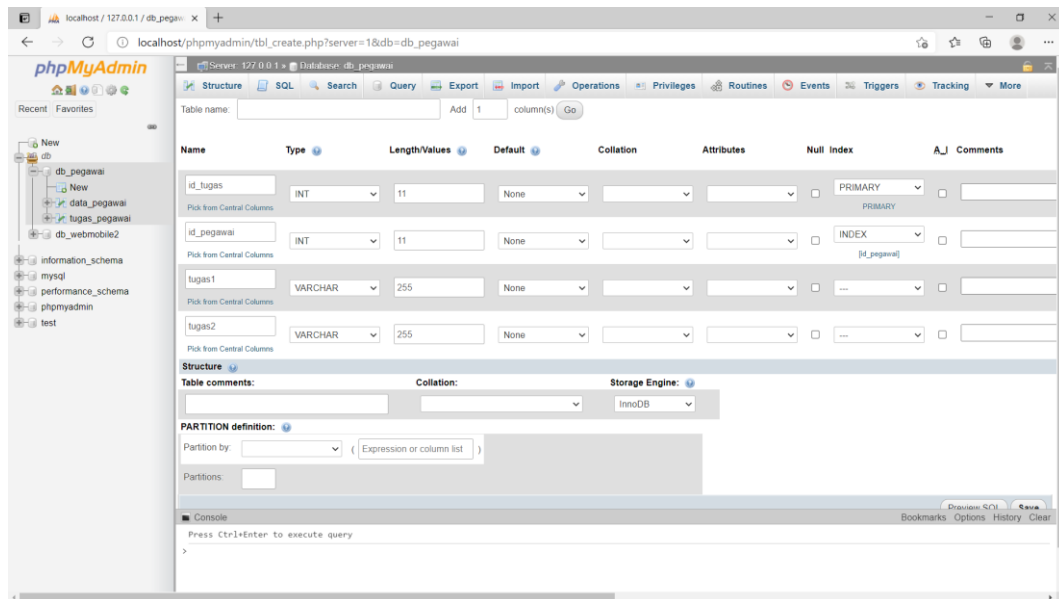
Langkah selanjutnya buatlah tabel beserta namanya “data_pegawai” kemudian tentukan berapa kolom yang akan dibuat contohnya membuat 4 kolom yang berisikan id_pegawai sebagai primary key, nama, usia dan alamat. Keempat

kolom tersebut mempunyai karakteristik masing-masing seperti id_pegawai int(11), nama varchar(255), usia int(11) dan alamat varchar(11) kemudian tekan tombol save.



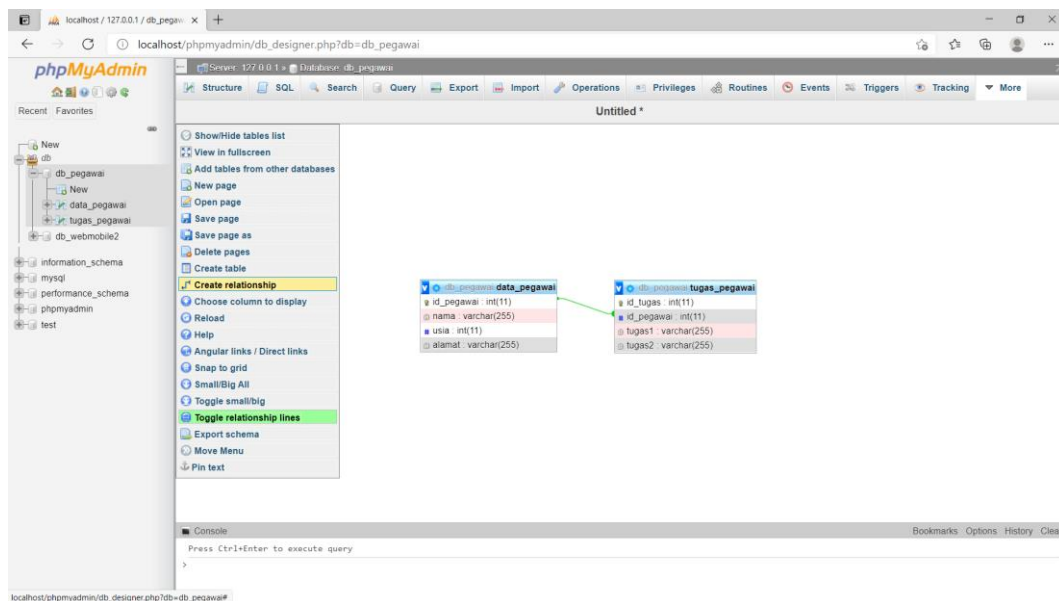
Gambar 2.2 Membuat Tabel 1

Berikutnya buatlah relasi tabel, dimana buatlah satu tabel yang menghubungkan dengan tabel “data_pegawai”. Buatlah tabel dengan nama tabel “tugas_pegawai” dengan membuat 4 kolom yang berisikan id_tugas sebagai primary key, id_pegawai sebagai foreign key (membuat index), tugas1 dan tugas2. Keempat kolom tersebut mempunyai karakteristik masing-masing id_tugas int(11), id_pegawai int(11), tugas1 varchar(255), tugas2 varchar(11) kemudian tekan tombol save.



Gambar 2.3 Membuat Tabel 2

Kemudian buatlah suatu relasi antara tabel “data_pegawai” dengan tabel _pegawai” dengan cara menggabungkan melalui menu designer lalu hubungkan “id_pegawai” antara kedua tabel tersebut menggunakan create relationship agar kedua tabel saling berelasi melalui “id_pegawai”.



Gambar 2.4 Membuat Relasi Tabel

Pada bagian kedua, yang dimana perintah tersebut yaitu buatlah program untuk menyimpan, menghapus dan mengubah data-data pegawai tersebut. Pertama buatlah sebuah file yang bernama “index.php” dalam file tuliskanlah syntax yang berisikan menampilkan data pegawai, insert, delete dan update serta menghubungkan dengan database MySQL.

```
1  <?php
2  $connect = mysqli_connect("localhost", "root", "", "db_pegawai");
3  if (!$connect) {
4      die("Gagal Terhubung" . mysqli_connect_error());
5  }
6  ?>
7
8  <!DOCTYPE html>
9  <html lang="en">
10 <head>
11     <meta charset="UTF-8">
12     <meta http-equiv="X-UA-Compatible" content="IE=edge">
13     <meta name="viewport" content="width=device-width, initial-scale=1.0">
14     <title>Tugas Modul 3</title>
15 </head>
16 <body>
17     <h1>Data Pegawai</h1>
18     <table border="1">
19         <tr>
20             <td>ID Pegawai</td>
21             <td>Nama Pegawai</td>
22             <td>Usia</td>
23             <td>Alamat</td>
24             <td colspan="2" align="center">Aksi</td>
25         </tr>
26     <?php
27     ini_set("display_errors", 0);
28     if ($_POST['btn'] == 'Input') {
29         $id_pegawai = $_POST['id_pegawai'];
30         $nama = $_POST['nama'];
31         $usia = $_POST['usia'];
32         $alamat = $_POST['alamat'];
33         $sql = "INSERT INTO data_pegawai (id_pegawai, nama, usia, alamat) VALUES
34             ('$id_pegawai', '$nama', '$usia', '$alamat')";
35         mysqli_query($connect, $sql);
```

Gambar 2.5 Source Code Halaman Index Bagian 1

```

40     if ($_GET['btn'] == 'Delete') {
41         $sql = "DELETE FROM data_pegawai WHERE id_pegawai = '$_GET[id_pegawai]' ";
42         mysqli_query($connect, $sql);
43     }
44     $sql = "SELECT * from data_pegawai";
45     $result = mysqli_query($connect, $sql);
46     if (mysqli_num_rows($result)) {
47         while ($row = mysqli_fetch_assoc($result)) {
48             echo
49             "<tr>
50                 <td>".$row["id_pegawai"]."</td>
51                 <td>".$row["nama"]."</td>
52                 <td>".$row["usia"]."</td>
53                 <td>".$row["alamat"]."</td>
54                 <td><a href='\"update_data.php?id_pegawai=\".$row[\"id_pegawai\"].\"&btn=Update\">
55                     Update</a></td>
56                 <td><a href='\"index.php?id_pegawai=\".$row[\"id_pegawai\"].\"&btn=Delete\">
57                     Delete</a> </td>
58             </tr>";
59         }
60     }
61 }>
62 </table>
63 <br><a href="input_data.php">Input Data</a>
64 </body>
65 </html>

```

Gambar 2.6 Source Code Halaman Index Bagian 2

Pada awal baris dimana kegunaan syntax tersebut bertujuan untuk menghubungkan program dengan database “db_pegawai” dan dimana jika gagal akan muncul tulisan “Gagal Terhubung”.

```

<?php
$connect = mysqli_connect("localhost", "root", "", "db_pegawai");
if (!$connect) {
    die("Gagal Terhubung" . mysqli_connect_error());
}
?>

```

Syntax untuk menginput, menghapus dan mengupdate ke dalam tabel “data_pegawai” pada database “db_pegawai”.

Syntax di bawah ini berguna untuk menginput data dengan memakai perintah \$_POST dikarenakan untuk memasukan data berada pada halaman lain :

```

if ($_POST['btn'] == 'Input') {
    $id_pegawai = $_POST['id_pegawai'];
    $nama = $_POST['nama'];
    $usia = $_POST['usia'];
    $alamat = $_POST['alamat'];
    $sql = "INSERT INTO data_pegawai (id_pegawai, nama, usia,
alamat) VALUES ('$id_pegawai', '$nama', '$usia', '$alamat')";
    mysqli_query($connect, $sql);
}

```

```
}
```

Syntax di bawah ini berguna untuk menghapus data dengan memakai perintah \$_GET dikarenakan untuk menghapus data mengambil id pada halaman ini :

```
if ($_GET['btn'] == 'Delete') {  
    $sql = "DELETE FROM data_pegawai WHERE id_pegawai = '$_GET[id_pegawai]' ";  
    mysqli_query($connect, $sql);  
}
```

Syntax di bawah ini berguna untuk memperbaiki data dengan memakai perintah \$_POST dikarenakan untuk memperbaiki data berada pada halaman lain :

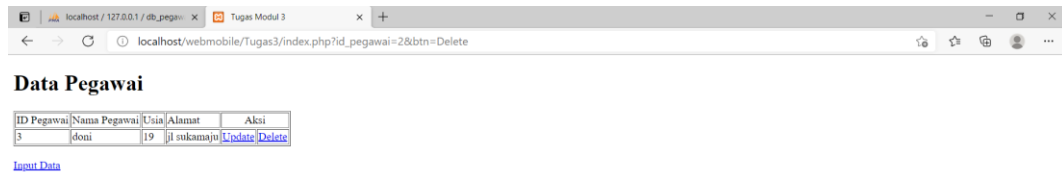
```
if ($_POST['btn'] == 'Update') {  
    $sql = "UPDATE data_pegawai SET nama = '$_POST[nama]', usia = '$_POST[usia]', alamat = '$_POST[alamat]' WHERE id_pegawai = '$_POST[id_pegawai]'";  
    mysqli_query($connect, $sql);  
}
```

Untuk memunculkan hasil data input, delete dan update dalam halaman index yang mempunyai tabel serta perpindahan halaman ke halaman input dan halaman update menggunakan syntax sebagai berikut :

```
$sql = "SELECT * from data_pegawai";  
$result = mysqli_query($connect, $sql);  
if (mysqli_num_rows($result)) {  
    while ($row = mysqli_fetch_assoc($result)) {  
        echo  
        "<tr>  
            <td>".$row["id_pegawai"]."</td>  
            <td>".$row["nama"]."</td>  
            <td>".$row["usia"]."</td>  
            <td>".$row["alamat"]."</td>  
            <td><a href=\"update_data.php?id_pegawai=".$row["id_pegawai"]."&btn=Update\">  
                Update</a></td>  
            <td><a href=\"index.php?id_pegawai=".$row["id_pegawai"]."&btn=Delete\">  
                Delete</a> </td>  
        </tr>";  
    }  
} else {  
    echo "0 result in database";  
}
```

}

Adapun hasil keluaran dari keseluruhan source code halaman index yang telah dituliskan adalah sebagai berikut :



Gambar 2.7 Keluaran Source Code Halaman Input

Langkah berikutnya buatlah file yang bernama “input_data” yang berisikan source code untuk memasukan data dari halaman tersebut lalu membawanya kembali ke halaman index menggunakan attribute name yang diberikan masing-masing serta menggunakan method post untuk membawanya ke halaman index, source codenya sebagai berikut :


```

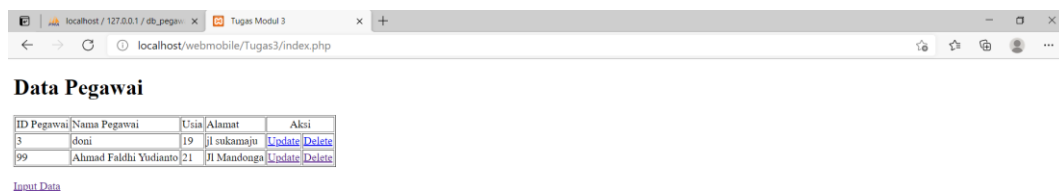
1  <h3>Input Data Pegawai</h3>
2  <form method="post" action="index.php">
3      <table>
4          <tr>
5              <td>ID Pegawai</td>
6              <td>:</td>
7              <td><input type="text" name="id_pegawai"></td>
8          </tr>
9          <tr>
10             <td>Nama Pegawai</td>
11             <td>:</td>
12             <td><input type="text" name="nama"></td>
13         </tr>
14         <tr>
15             <td>Usia</td>
16             <td>:</td>
17             <td><input type="text" name="usia"></td>
18         </tr>
19         <tr>
20             <td>Alamat</td>
21             <td>:</td>
22             <td><input type="text" name="alamat"></td>
23         </tr>
24     </table>
25     <br><input type="submit" name="btn" value="Input">
26 </form>
27 <p>Keterangan :<br>
28 Primary Key : ID Pegawai</p>

```

Gambar 2.8 Source Code Halaman Input



Gambar 2.9 Keluaran Source Code Halaman Input



Gambar 2.10 Hasil Input

Langkah selanjutnya buatlah file yang bernama “update_data” yang berisikan source code untuk memperbarui data dari halaman tersebut, pada awal baris terdapat syntax menghubungkan dengan database MySQL serta syntax untuk menampilkan data yang telah diinputkan pada id yang akan diupdate.

```
<?php
$connect = mysqli_connect("localhost", "root", "", "db_pegawai");
```

```

if (!$connect) {
    die("Gagal Terhubung" . mysqli_connect_error());
}
$sql = "SELECT * FROM data_pegawai WHERE id_pegawai= '$_GET[id_peg
awai]'";
$result = mysqli_query($connect, $sql);
$row = mysqli_fetch_assoc($result);
?>

```

Data yang akan diupdate tersebut kembali ke halaman index menggunakan attribute name yang diberikan masing-masing serta menggunakan method post untuk membawanya ke halaman index, source codenya sebagai berikut :

```

1  <?php
2  $connect = mysqli_connect("localhost", "root", "", "db_pegawai");
3  if (!$connect) {
4      die("Gagal Terhubung" . mysqli_connect_error());
5  }
6  $sql = "SELECT * FROM data_pegawai WHERE id_pegawai= '$_GET[id_pegawai]'";
7  $result = mysqli_query($connect, $sql);
8  $row = mysqli_fetch_assoc($result);
9  ?>
10
11 <h3>Update Data Pegawai</h3>
12 <form method="post" action="index.php">
13     <table>
14         <tr>
15             <td>ID Pegawai</td>
16             <td>:</td>
17             <td><input type="text" name="id_pegawai" value="<?php echo $row["id_pegawai"]; ?
18                 >"></td>
19         </tr>
20         <tr>
21             <td>Nama Pegawai</td>
22             <td>:</td>
23             <td><input type="text" name="nama" value="<?php echo $row["nama"]; ?>"></td>
24         </tr>
25         <tr>
26             <td>Usia</td>
27             <td>:</td>
28             <td><input type="text" name="usia" value="<?php echo $row["usia"]; ?>"></td>
29         </tr>
30         <tr>
31             <td>Alamat</td>
32             <td>:</td>
33             <td><input type="text" name="alamat" value="<?php echo $row["alamat"]; ?>"></td>
34         </tr>
35     </table>
36     <br><input type="submit" name="btn" value="Update">
37     <br><button onclick="index.php">Kembali</button>
38 </form>
39 <p>Keterangan :<br>
    Primary Key : ID Pegawai</p>

```

Gambar 2.11 Source Code Halaman Update

Update Data Pegawai

ID Pegawai : 99

Nama Pegawai : Ahmad Faldhi

Usia : 25

Alamat : Jl. Mandonga 27

Update

Kembali

Keterangan :
Primary Key : ID Pegawai

Gambar 2.12 Keluaran Source Code Halaman Update

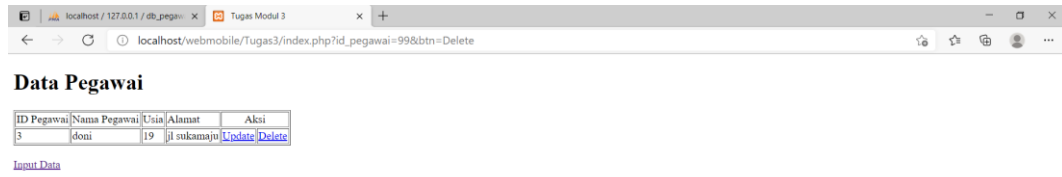
Data Pegawai

ID Pegawai	Nama Pegawai	Usia	Alamat	Aksi
3	doni	19	Jl. Sukamaju	Update Delete
99	Ahmad Faldhi	25	Jl. Mandonga 27	Update Delete

[Input Data](#)

Gambar 2.13 Hasil Update

Fungsi delete tidak memerlukan sebuah halaman baru dikarenakan id yang diambil berasal dari halaman index sendiri dengan memakai fungsi \$_GET.



Gambar 2.14 Hasil Delete

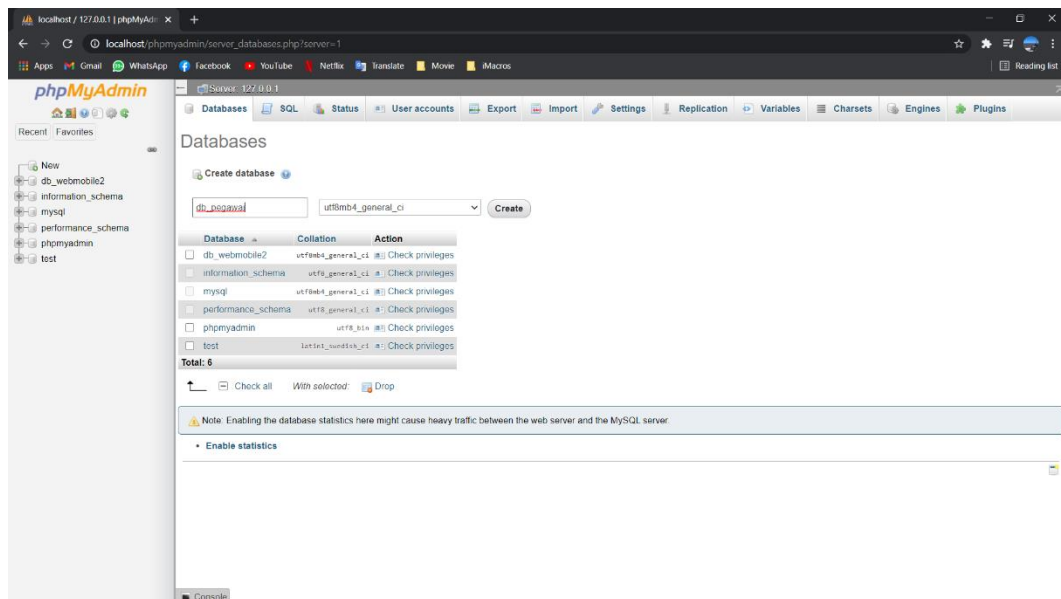
KESIMPULAN

Membuat suatu database dengan mudah serta relasi antar tabel dapat melalui localhost/phpmadmin, serta untuk menginput suatu data yang diinginkan memakai query “INSERT INTO VALUE”, untuk mengupdate suatu data dapat memakai query “UPDATE SET WHERE” dan untuk menghapus suatu dapat memakai query “DELETE FROM WHERE”

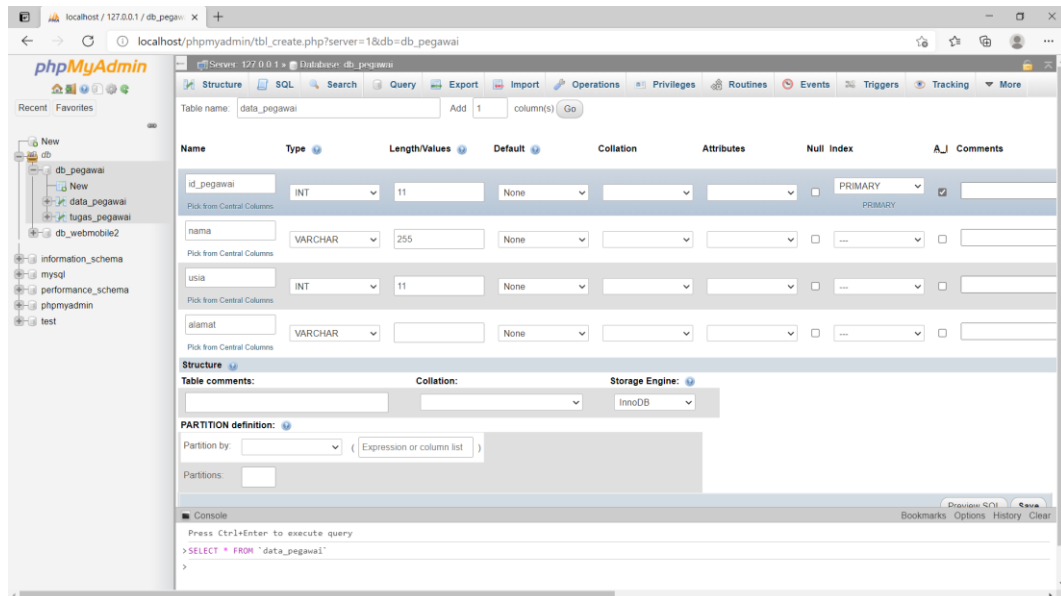
DAFTAR PUSTAKA

- Ariata. “Cara Membuat Koneksi PHP ke Database MySQL”. 18 Agustus 2020.
<https://jagongoding.com/web/php/menengah/regular-expression/> (Di akses
18 April 2020).
- Ariata. “Ingin Tahu Cara Input Data ke Database dengan PHP? Simak Tutorial
Ini!”. 18 Agustus 2020.
<https://jagongoding.com/web/php/menengah/regular-expression/> (Di akses
18 April 2020).
- Teknik Informatika. (2021). “MODUL PRAKTIKUM PEMROGRAMAN WEB I
Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya”.

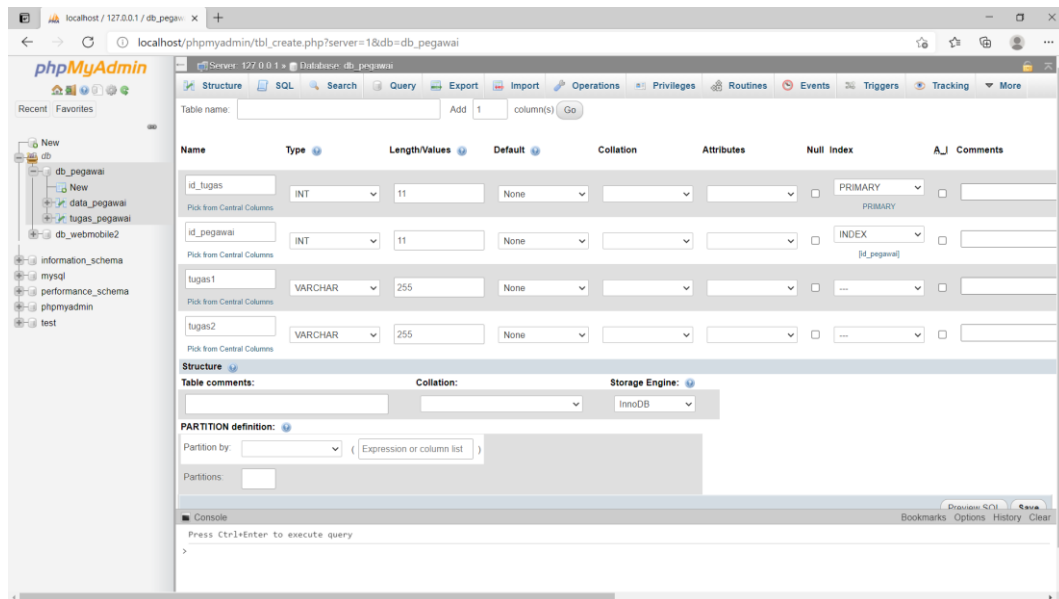
LAMPIRAN



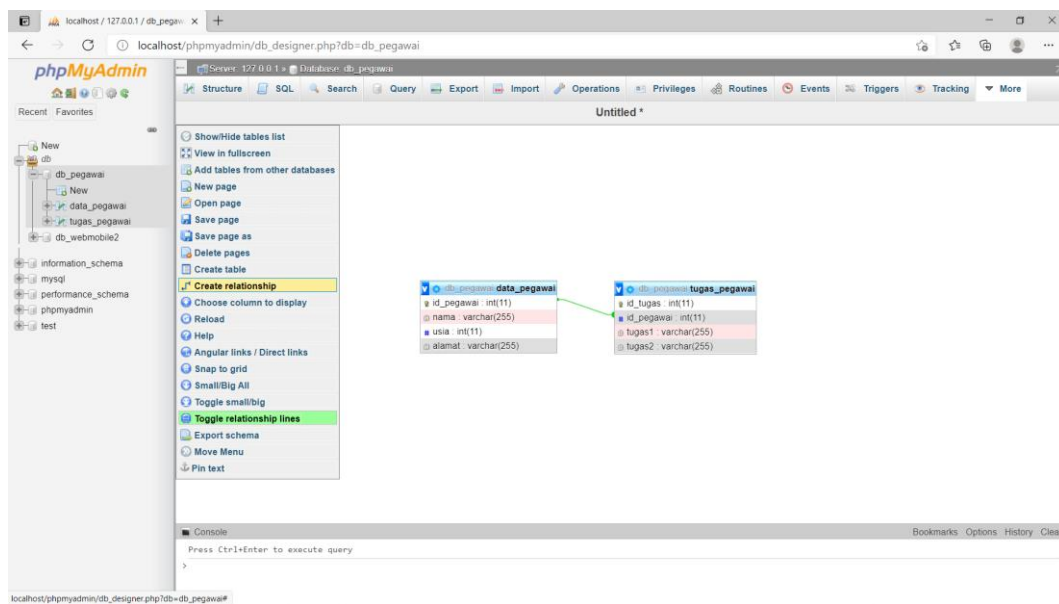
Lampiran 1



Lampiran 2



Lampiran 3



Lampiran 4

```

1  <?php
2  $connect = mysqli_connect("localhost", "root", "", "db_pegawai");
3  if (!$connect) {
4      die("Gagal Terhubung" . mysqli_connect_error());
5  }
6  ?>
7
8  <!DOCTYPE html>
9  <html lang="en">
10 <head>
11     <meta charset="UTF-8">
12     <meta http-equiv="X-UA-Compatible" content="IE=edge">
13     <meta name="viewport" content="width=device-width, initial-scale=1.0">
14     <title>Tugas Modul 3</title>
15 </head>
16 <body>
17     <h1>Data Pegawai</h1>
18     <table border="1">
19         <tr>
20             <td>ID Pegawai</td>
21             <td>Nama Pegawai</td>
22             <td>Usia</td>
23             <td>Alamat</td>
24             <td colspan="2" align="center">Aksi</td>
25         </tr>
26     <?php
27     ini_set("display_errors", 0);
28     if ($_POST['btn'] == 'Input') {
29         $id_pegawai = $_POST['id_pegawai'];
30         $nama = $_POST['nama'];
31         $usia = $_POST['usia'];
32         $alamat = $_POST['alamat'];
33         $sql = "INSERT INTO data_pegawai (id_pegawai, nama, usia, alamat) VALUES
34             ('$id_pegawai', '$nama', '$usia', '$alamat')";
35         mysqli_query($connect, $sql);

```

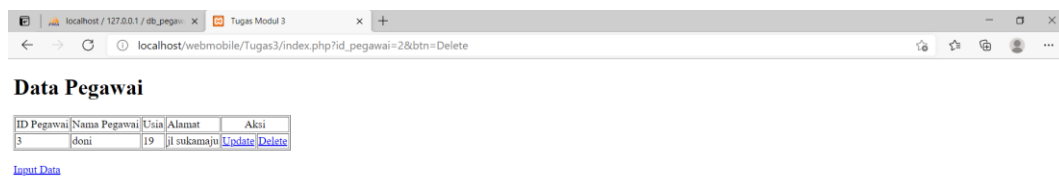
Lampiran 5

```

40     if ($_GET['btn'] == 'Delete') {
41         $sql = "DELETE FROM data_pegawai WHERE id_pegawai = '$_GET[id_pegawai]' ";
42         mysqli_query($connect, $sql);
43     }
44     $sql = "SELECT * from data_pegawai";
45     $result = mysqli_query($connect, $sql);
46     if (mysqli_num_rows($result)) {
47         while ($row = mysqli_fetch_assoc($result)) {
48             echo
49             "<tr>
50                 <td>".$row["id_pegawai"]."</td>
51                 <td>".$row["nama"]."</td>
52                 <td>".$row["usia"]."</td>
53                 <td>".$row["alamat"]."</td>
54                 <td><a href='\"update_data.php?id_pegawai=".$row[\"id_pegawai\"].\"&btn=Update\">
55                     Update</a></td>
56                 <td><a href='\"index.php?id_pegawai=".$row[\"id_pegawai\"].\"&btn=Delete\">
57                     Delete</a> </td>
58             </tr>";
59         }
60     }
61     ?>
62     </table>
63     <br><a href="input_data.php">Input Data</a>
64 </body>
65 </html>

```

Lampiran 6



The screenshot shows a web browser window with the address bar displaying 'localhost/webmobile/Tugas3/index.php?id_pegawai=2&btn=Delete'. The page title is 'Data Pegawai'. Below the title is a table with the following data:

ID Pegawai	Nama Pegawai	Usia	Alamat	Aksi
3	doni	19	jl sukamaju	Update Delete

Below the table, there is a link labeled 'Input Data'.

Lampiran 7

```

1  <h3>Input Data Pegawai</h3>
2  <form method="post" action="index.php">
3      <table>
4          <tr>
5              <td>ID Pegawai</td>
6              <td>:</td>
7              <td><input type="text" name="id_pegawai"></td>
8          </tr>
9          <tr>
10             <td>Nama Pegawai</td>
11             <td>:</td>
12             <td><input type="text" name="nama"></td>
13         </tr>
14         <tr>
15             <td>Usia</td>
16             <td>:</td>
17             <td><input type="text" name="usia"></td>
18         </tr>
19         <tr>
20             <td>Alamat</td>
21             <td>:</td>
22             <td><input type="text" name="alamat"></td>
23         </tr>
24     </table>
25     <br><input type="submit" name="btn" value="Input">
26 </form>
27 <p>Keterangan :<br>
28 Primary Key : ID Pegawai</p>

```

Lampiran 8



Input Data Pegawai

ID Pegawai :

Nama Pegawai :

Usia :

Alamat :

Keterangan :
Primary Key : ID Pegawai

Lampiran 9



Data Pegawai

ID Pegawai	Nama Pegawai	Usia	Alamat	Aksi
3	doni	19	jl sukamaju	Update Delete
99	Ahmad Faldhi Yudianto	21	Jl Mandonga	Update Delete

[Input Data](#)

Lampiran 10

```

1  <?php
2  $connect = mysqli_connect("localhost", "root", "", "db_pegawai");
3  if (!$connect) {
4      die("Gagal Terhubung" . mysqli_connect_error());
5  }
6  $sql = "SELECT * FROM data_pegawai WHERE id_pegawai= '$_GET[id_pegawai]'";
7  $result = mysqli_query($connect, $sql);
8  $row = mysqli_fetch_assoc($result);
9  ?>
10
11 <h3>Update Data Pegawai</h3>
12 <form method="post" action="index.php">
13     <table>
14         <tr>
15             <td>ID Pegawai</td>
16             <td>:</td>
17             <td><input type="text" name="id_pegawai" value="<?php echo $row["id_pegawai"]; ?
18             >"></td>
19         </tr>
20         <tr>
21             <td>Nama Pegawai</td>
22             <td>:</td>
23             <td><input type="text" name="nama" value="<?php echo $row["nama"]; ?>"></td>
24         </tr>
25         <tr>
26             <td>Usia</td>
27             <td>:</td>
28             <td><input type="text" name="usia" value="<?php echo $row["usia"]; ?>"></td>
29         </tr>
30         <tr>
31             <td>Alamat</td>
32             <td>:</td>
33             <td><input type="text" name="alamat" value="<?php echo $row["alamat"]; ?>"></td>
34         </tr>
35     </table>
36     <br><input type="submit" name="btn" value="Update">
37     <br><button onclick="index.php">Kembali</button>
38 </form>
39 <p>Keterangan :<br>
    Primary Key : ID Pegawai</p>

```

Lampiran 11

localhost / 127.0.0.1 / db_pegaw... x localhost/webmobile/Tugas3/... x +

localhost/webmobile/Tugas3/update_data.php?id_pegawai=99&btn=Update

Update Data Pegawai

ID Pegawai : 99
Nama Pegawai : Ahmad Faldhi
Usia : 25
Alamat : Jl. Mandonga 27

Update
Kembali

Keterangan :
Primary Key : ID Pegawai

Lampiran 12

localhost / 127.0.0.1 / db_pegaw... x Tugas Modul 3 x +

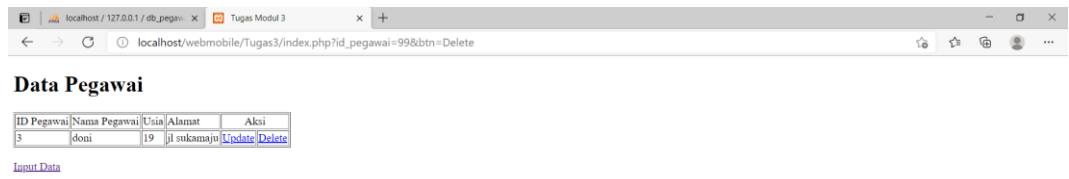
localhost/webmobile/Tugas3/index.php

Data Pegawai

ID Pegawai	Nama Pegawai	Usia	Alamat	Aksi
3	doni	19	Jl. Sukamaju	Update Delete
99	Ahmad Faldhi	25	Jl. Mandonga 27	Update Delete

[Input Data](#)

Lampiran 13



Lampiran 14