

**LAPORAN HASIL PRAKTIKUM
PEMPROGRAMAN WEB & MOBILE**



NAMA : FIRDAUS ZAILANI
NIM : 193020503040
KELAS : A
MODUL : III (PHP MySQL Database)

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021

BAB I

TUJUAN DAN LANDASAN TEORI

A. 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.

B. 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. 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";

?>
```

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";

?>
```

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

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

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);
?>
```

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 table  
$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);  
?>
```

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 table 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);
?>

```

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);
?>

```

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	namade pan	namabelak ang	email	reg_date
1	John	Doe	john@exampl e.com	2014-10-22 14:26:15
2	Mary	Moe	mary@examp le.com	2014-10-23 10:22:30
3	Julie	Dooley	Dooleyjulie@ 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);  
?>
```

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 {
```

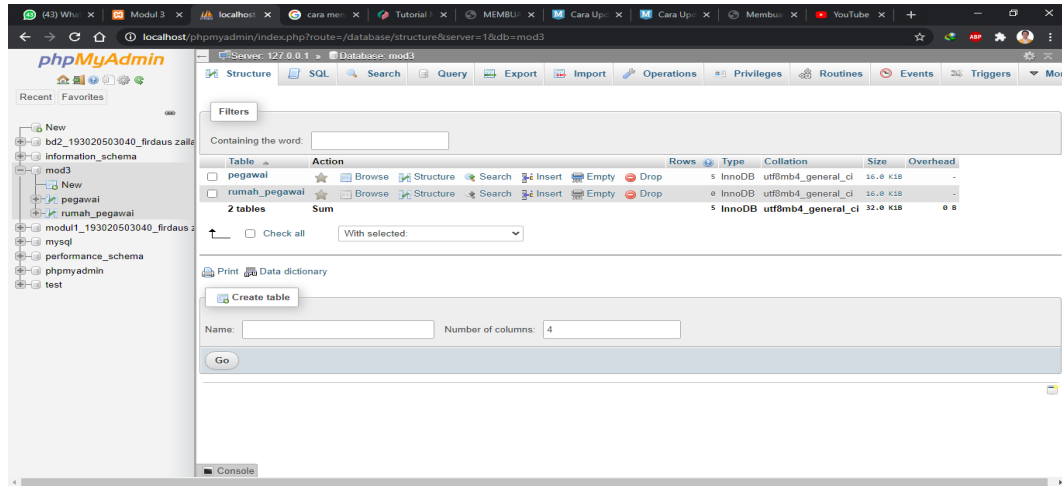
```
        echo "Error updating record: " . mysqli_error($conn);  
    }  
    mysqli_close($conn);  
?>
```

C. TUGAS

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

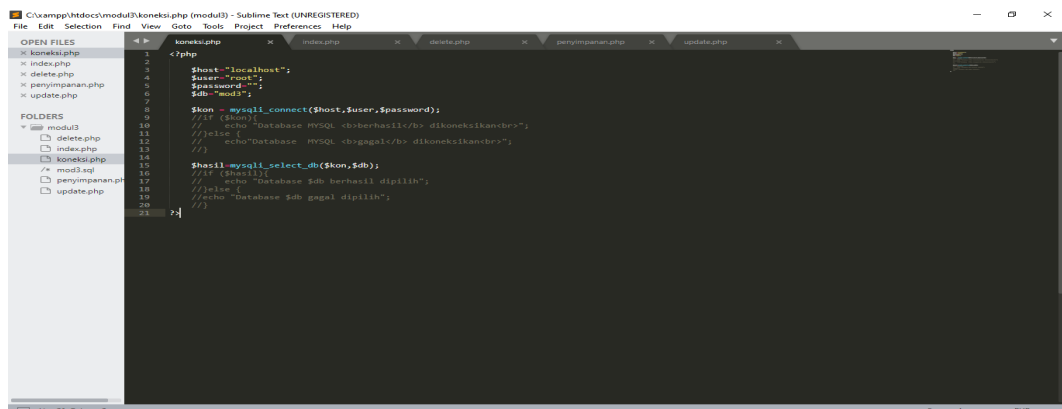
BAB II

PEMBAHASAN



Gambar 1.1. *Databases mod3.*

Gambar diatas dari sebuah Databases, Untuk membuat database, Anda dapat menggunakan mariadb melalui command prompt, php myadmin atau sqlyog untuk menyelesaikannya. Dalam laporan ini, pembuatan database dilakukan dengan menggunakan php myadmin. Ini adalah gambar dan kueri yang diperoleh dari database di php myadmin. Anda membutuhkan database yang nantinya akan ditautkan ke website yang akan dibuat, dan database ini akan digunakan sebagai penyimpan data pada website yang akan terhubung dengan website tersebut.



Gambar 1.2. *Koneksi.php*.

Gambar diatas adalah codingan *koneksi.php*. Digunakan untuk membuat koneksi dengan database, sehingga perubahan data pada website akan mempengaruhi database, begitu pula sebaliknya. Pada baris 8 digunakan untuk membuat variabel yang akan digunakan pada baris 10, jika tidak berhasil akan menampilkan kata gagal terhubung ke database, fungsi gagal pada baris 19

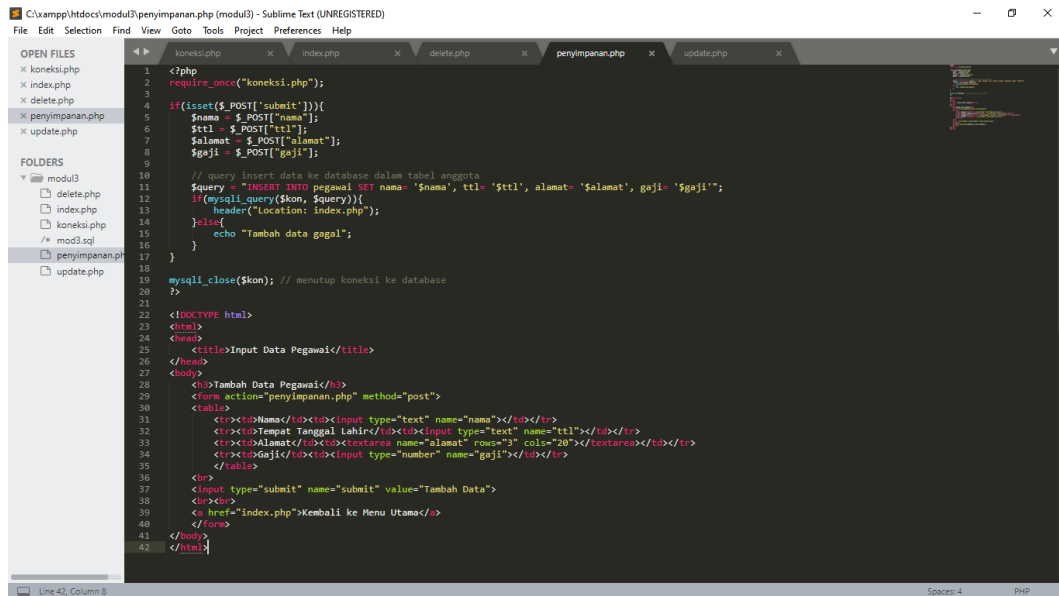
```

1 <?php
2 require("koneksi.php");
3 $data = mysqli_query($kon, "SELECT * FROM pegawai");
4 ?>
5
6 <!DOCTYPE html>
7 <html>
8 <head>
9 <title>Modul 3</title>
10 </head>
11 <body>
12 <h3>Daftar Data Pegawai</h3>
13 <a href="penyimpanan.php">Tambah Data</a>
14
15 <table border="1px">
16 <tr>
17 <th>ID Pegawai</th>
18 <th>Nama</th>
19 <th>Tempat Tanggal Lahir</th>
20 <th>Alamat</th>
21 <th>Gaji</th>
22 </tr>
23
24 <?php while($row = mysqli_fetch_assoc($data)){ ?>
25 <tr>
26 <td><?php echo $row['id'] ?></td>
27 <td><?php echo $row['nama'] ?></td>
28 <td><?php echo $row['ttl'] ?></td>
29 <td><?php echo $row['alamat'] ?></td>
30 <td><?php echo $row['gaji'] ?></td>
31 <td>
32 <a href="update.php?id=<?php echo $row['id']; ?>">Edit</a> |
33 <a href="delete.php?id=<?php echo $row['id']; ?>">Hapus</a>
34 </td>
35 </tr>
36 <?php
37 } // end while
38
39 mysqli_close($kon); // menutup koneksi ke database
40 ?>
41 </table>
42 </body>
43 </html>

```

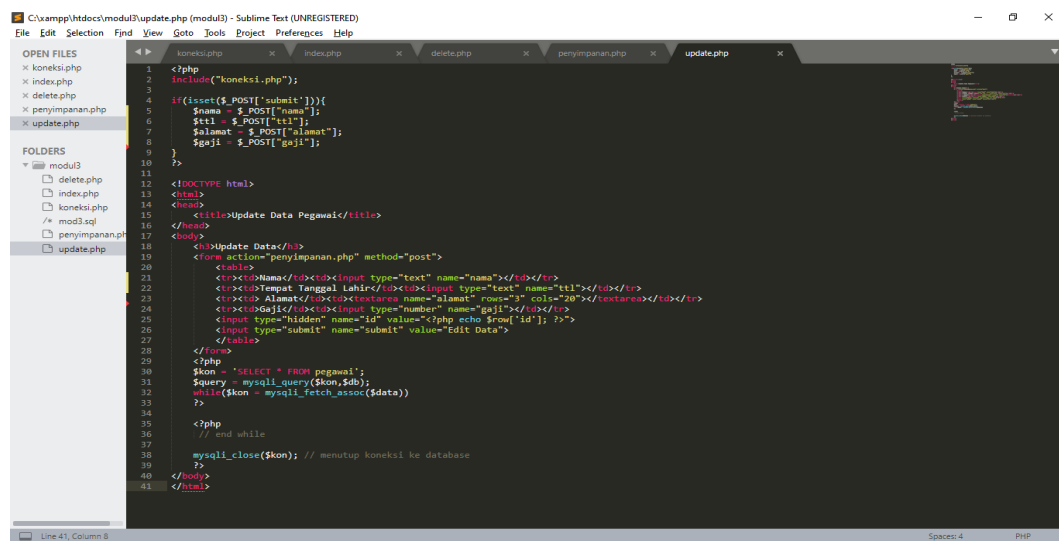
Gambar 1.3. *Index.php*.

Pada bagian gambar *index.php*. berfungsi untuk menampilkan halaman awal, pada baris 9,12 untuk menampilkan header tulisan dan pada baris 24-31 adalah bagian tampilan awal yaitu table dari databases terdiri dari id,nama,tt,alamat,gaji. Sedangkan pada baris ke 22 dan 23 adalah bagian *update.php* dan *delete.php* jika mengklik salah satu nya maka akan langsung tertuju sesuai dengan isi coding dan tujuan nya “Edit” yang artinya mengedit(update) sedangkan “Hapus” yaitu bertujuan untuk menghapus databases.



Gambar 1.4. *Penyimpanan.php*.

Gambar diatas merupakan php untuk menambahkan data yang nantinya jika menambahkan data menggunakan php maka nantinya di database juga bertambah pada baris 11-15 adalah query insert data ke database dalam tabel anggota dan pada baris 31-39 adalah untuk tampilan yang nantinya jika mengklik tambah data maka akan muncul bari atau kolom dan setelah selesai bias mengklik submit yang nantinya langung kembali ke menu utama. Fungsinya hamper sama seperti index.php.



Gambar 1.5. *Delete.php*.

Pada gambar diatas file hapus atau delete diginakan untuk menghapus database dan kita bias menghapus semua database yang tampil di tampilan awal. Pada bagian delete php ini berhubungan dengan koneksi php dimana jika ingin menghapus database yang berada di penyimpanan php maka akan secara otomatis hilang selain itu delete php jika selesai menghapus akan menghilangkan databases.

```

1 <?php
2 include("koneksi.php");
3
4 if(isset($_POST['submit'])){
5     $nama = $_POST['nama'];
6     $ttl = $_POST['ttl'];
7     $alamat = $_POST['alamat'];
8     $gaji = $_POST['gaji'];
9 }
10
11
12 <!DOCTYPE html>
13 <html>
14 <head>
15 <title>Update Data Pegawai</title>
16 </head>
17 <body>
18 <h3>Update Data</h3>
19 <form action="penyimpanan.php" method="post">
20 <table>
21 <tr><td>Nama</td><td><input type="text" name="nama"></td></tr>
22 <tr><td>Tempat Tanggal Lahir</td><td><input type="text" name="ttl"></td></tr>
23 <tr><td>Alamat</td><td><input type="text" name="alamat" value=""></td></tr>
24 <tr><td>Gaji</td><td><input type="number" name="gaji"></td></tr>
25 <input type="hidden" name="id" value=""></td></tr>
26 <input type="submit" name="submit" value="Edit Data">
27 </table>
28 </form>
29 <?php
30 $kon = 'SELECT * FROM pegawai';
31 $query = mysqli_query($kon,$db);
32 while($kon = mysqli_fetch_assoc($data))
33 {
34
35 <?php
36 // end while
37
38 mysqli_close($kon); // menutup koneksi ke database
39 }
40 </body>
41 </html>

```

Gambar 1.6. *Update.php.*

Update.php ini digunakan untuk membuat perubahan pada isi table yang telah didapat sehingga tidak perlu menghapus data dan menambahkan ulang jika ada kesalahan tidak perlu berulang ulang menambah dan hapus data melainkan hanya mengedit atau update saja. Pada baris 4-8 adalah query insert data ke database dalam tabel anggota dan pada baris 15-26 adalah untuk tampilan yang nantinya jika mengklik tambah data maka akan muncul bari atau kolom dan setelah selesai bias mengklik submit yang nantinya langsung kembali ke menu utama.

Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama	Tempat Tanggal Lahir	Alamat	Gaji	
01	Steven Duling	Kuala Kapuas, 20-02-2002	Jl. Merak II	3000000	Edit Hapus
02	Hussain Alam	Amuntai, 10-01-2001	Jl. Merdeka I	2500000	Edit Hapus
03	Elnita	Kuala Lumpur, 21-01-2000	Jl. Karon	3500000	Edit Hapus
05	Mamunah	Barabai, 30-03-2000	Jl. Permai	3000000	Edit Hapus
06	Musram	Kanamt, 22-06-2001	Jl. Panjaitan	2000000	Edit Hapus

Gambar 1.7. *Tampilan Awal.*

Pada gambar diatas menampilkan tampilan awal dari website yang terhubung dengan semua php dan database, pada tampilan menu utama ini menampilkan banyak pilihan yaitu, Tambah Data, Edit dan Hapus. Dan juga table database sendiri.

Tambah Data Pegawai

Nama:

Tempat Tanggal Lahir:

Alamat:

Gaji:

[Kembali ke Menu Utama](#)

Gambar 1.8. *Tampilan Tambah Data.*

Pada Tambah Data ini kita dapat menambahkan data secara manual dimana kita dapat mengisi nama, tt, alamat dan juga gaji. Selain itu kita juga dapat melihat pada gambar diatas bahwa ada pilihan “ Kembali ke Menu Utama “ jika kita tidak ingin menambah data maka dapat mengklik nya yang nanti nya akan kembali ke menu utama yang sudah kita lihat pada gambar 1.7. Dan jika selesai mengisi maka dapat mengklik(submit) Simpaan data dan tampilan nya di menu utama akan seperti gambar dibawah.

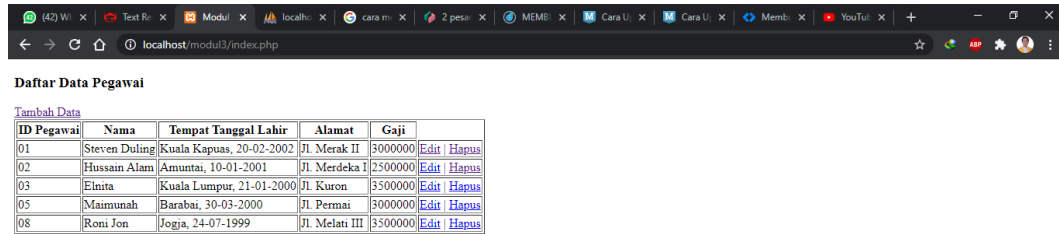
Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama	Tempat Tanggal Lahir	Alamat	Gaji
01	Steven Duling	Kuala Kapuas, 20-02-2002	Jl. Merak II	3000000 Edit Hapus
02	Hussain Alam	Amuntai, 10-01-2001	Jl. Merdeka I	2500000 Edit Hapus
03	Elnita	Kuala Lumpur, 21-01-2000	Jl. Kuron	3500000 Edit Hapus
05	Maimunah	Barabai, 30-03-2000	Jl. Permai	3000000 Edit Hapus
06	Musram	Kanamit, 22-06-2001	Jl. Panjaitan	2000000 Edit Hapus
08	Roni Jon	Jogja, 24-07-1999	Jl. Melati III	3500000 Edit Hapus

Gambar 1.9. *Tampilan setelah Tambah Data.*

Gambar diatas adalah hasil dari gambar 1.8. yaitu menambahkan data pada database.

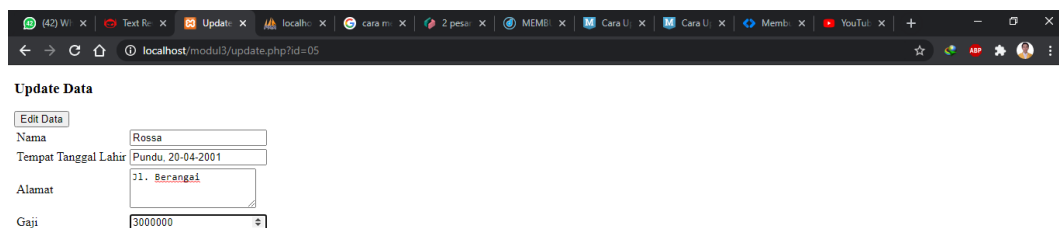


The screenshot shows a web browser window with the address bar displaying 'localhost/modul3/index.php'. The page title is 'Daftar Data Pegawai'. Below the title is a link 'Tambah Data'. The main content is a table with 6 columns: ID Pegawai, Nama, Tempat Tanggal Lahir, Alamat, Gaji, and two empty columns. The table contains 6 rows of data. Each row has 'Edit' and 'Hapus' links in the last two empty columns.

ID Pegawai	Nama	Tempat Tanggal Lahir	Alamat	Gaji		
01	Steven Duling	Kuala Kapuas, 20-02-2002	Jl. Merak II	5000000	Edit	Hapus
02	Hussain Alam	Amuntai, 10-01-2001	Jl. Merdeka I	2500000	Edit	Hapus
03	Elnita	Kuala Lumpur, 21-01-2000	Jl. Kuron	3500000	Edit	Hapus
05	Maimunah	Barabai, 30-03-2000	Jl. Permai	3000000	Edit	Hapus
08	Roni Jon	Jogja, 24-07-1999	Jl. Melati III	3500000	Edit	Hapus

Gambar 1.10. *Tampilan Hapus Data.*

Pada gambar diatas menunjukkan bahwa database dapat dihapus jika mengklik pada bagian Hapus secara otomatis semua biodata nya akan hilang dan jika ingin menambahkan data maka id pegawai nya akan terus berlanjut.



The screenshot shows a web browser window with the address bar displaying 'localhost/modul3/update.php?id=05'. The page title is 'Update Data'. Below the title is a link 'Edit Data'. The main content is a form with four fields: Nama (text input), Tempat Tanggal Lahir (text input), Alamat (text input), and Gaji (text input). The form contains the following data: Nama: Rossa, Tempat Tanggal Lahir: Pundu, 20-04-2001, Alamat: Jl. Berangai, Gaji: 3000000.

Gambar 1.11. *Tampilan Edit Data.*

Pada bagian edit atau update ini bertujuan untuk mengedit data setiap pegawai, tanpa harus hapus lalu menambahkan data berulang ulang.

BAB III

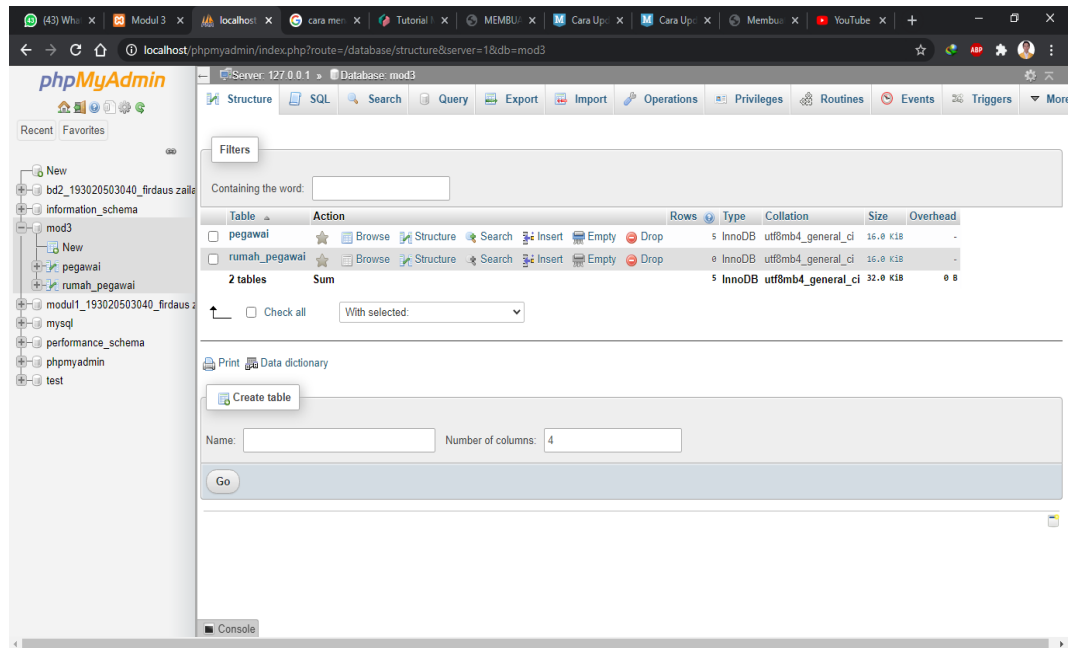
KESIMPULAN

Data dalam database MySQL disimpan dalam tabel. Tabel adalah kumpulan data terkait yang berisi kolom dan baris. Database sangat berguna untuk menyimpan informasi dalam kategori. PHP (PHP: Hypertext Preprocessor) adalah bahasa pemrograman skrip sisi server open source. Sebagai bahasa skrip, PHP menjalankan instruksi pemrograman pada saat runtime. Hasil instruksi pasti akan berbeda-beda sesuai dengan data yang diolah. HTML adalah singkatan dari Hypertext Markup Language, yaitu bahasa pemrograman standar yang digunakan untuk membuat halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di Internet Web browser (browser). HTML juga dapat digunakan sebagai link antar situs atau file menggunakan localhost di komputer, atau untuk menghubungkan link antar situs di Internet.

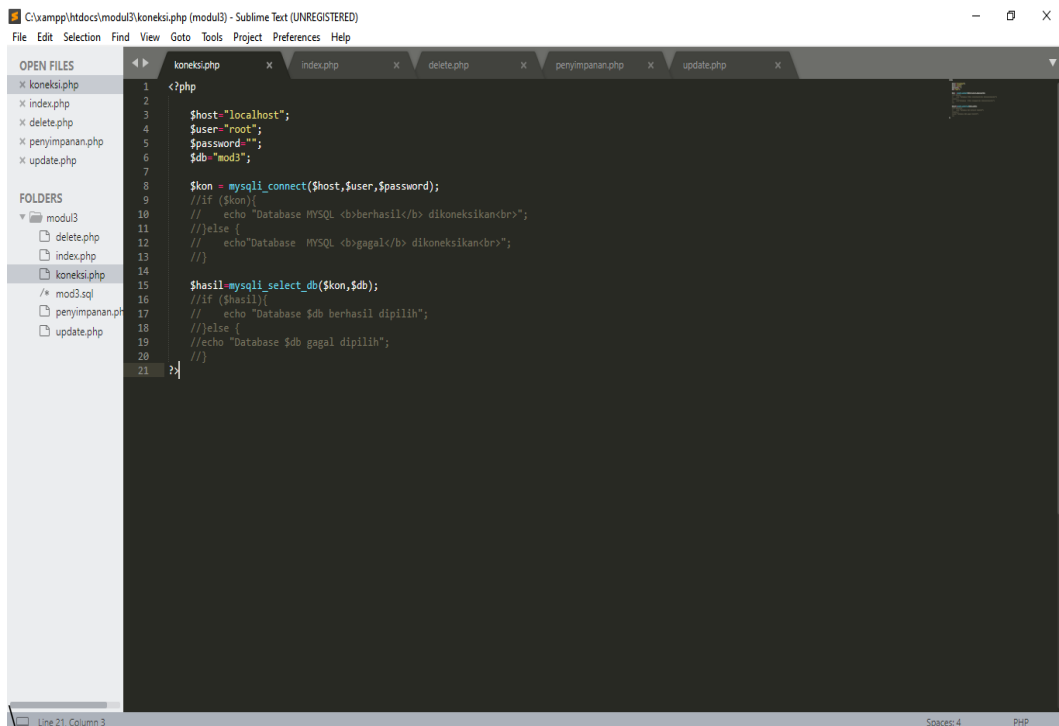
DAFTAR PUSTAKA

- Alfarabi. Diki. 24 Agustus 2020. *Membuat CRUD Dengan PHP Dan MySQL Edit Data*. <https://www.malasngoding.com/membuat-crud-dengan-php-dan-mysql-edit-data/>. (Diakses Pada 12 April 2021 Pukul 02:30 WIB).
- Akbar. Nuris. 10 Agustus 2016. *Cara Update dan Delete Data MySQL Dengan PHP*. <https://belajarphp.net/cara-update-dan-delete-data-mysql-dengan-php/>. (Diakses Pada 12 April 2021 Pukul 00:03 WIB).
- Modul Praktikum Pemrograman Web & Mobile I. Jurusan Teknik Informatika. Fakultas Teknik. Universitas Palangka Raya. 2021.
- Suryadi. Ade. 26 Maret 2018. *Membuat FORM Edit dan Hapus Data Menggunakan PHP*. <https://www.develindo.com/artikel/>. (Diakses Pada 11 April 2021 Pukul 22:20 WIB).
- Yuni. 21 Januari 2019. *Tutorial Membuat CRUD PHP Dengan MySQL*. <https://www.codepolitan.com/tutorial-membuat-crud-php-dengan-mysql>. (Diakses Pada 11 April 2021 Pukul 22:02 WIB).

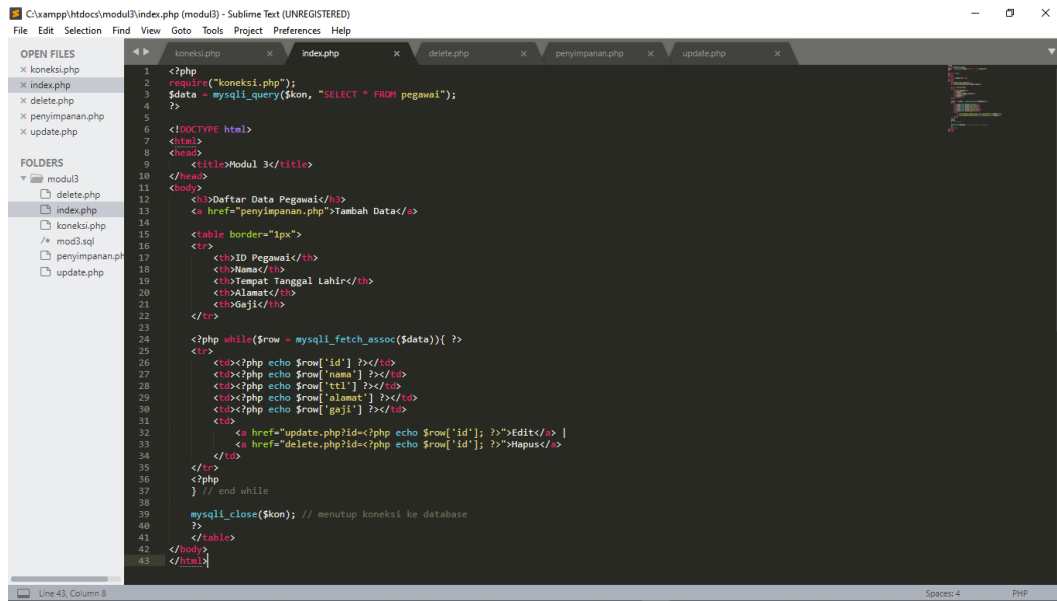
LAMPIRAN



Gambar 1.1. *Databases mod3.*

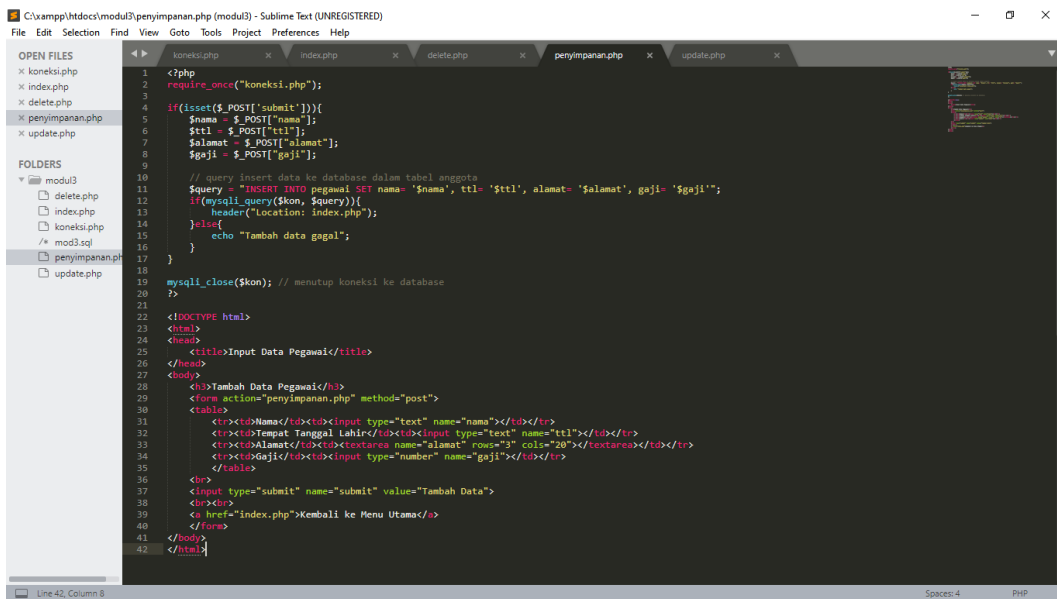


Gambar 1.2. *Koneksi.php.*



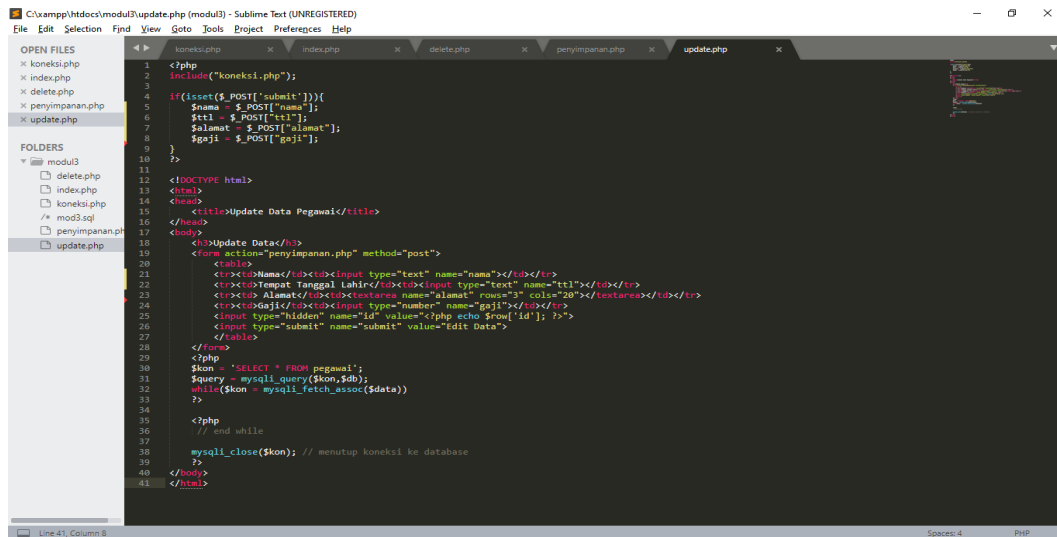
```
1 <?php
2 require("koneksi.php");
3 $data = mysqli_query($kon, "SELECT * FROM pegawai");
4 ?>
5
6 <!DOCTYPE html>
7 <html>
8 <head>
9 <title>Modul 3</title>
10 </head>
11 <body>
12 <h3>Daftar Data Pegawai</h3>
13 <a href="penyimpanan.php">Tambah Data</a>
14
15 <table border="1px">
16 <tr>
17 <th>ID Pegawai</th>
18 <th>Nama</th>
19 <th>Tempat Tanggal Lahir</th>
20 <th>Alamat</th>
21 <th>Gaji</th>
22 </tr>
23
24 <?php while($row = mysqli_fetch_assoc($data)){ ?>
25 <tr>
26 <td><?php echo $row['id'] ?></td>
27 <td><?php echo $row['nama'] ?></td>
28 <td><?php echo $row['ttl'] ?></td>
29 <td><?php echo $row['alamat'] ?></td>
30 <td><?php echo $row['gaji'] ?></td>
31 <td>
32 <a href="update.php?id=<?php echo $row['id'] ?>">Edit</a> |
33 <a href="delete.php?id=<?php echo $row['id'] ?>">Hapus</a>
34 </td>
35 </tr>
36 <?php
37 } // end while
38
39 mysqli_close($kon); // menutup koneksi ke database
40 ?>
41 </table>
42 </body>
43 </html>
```

Gambar 1.3. *Index.php*.

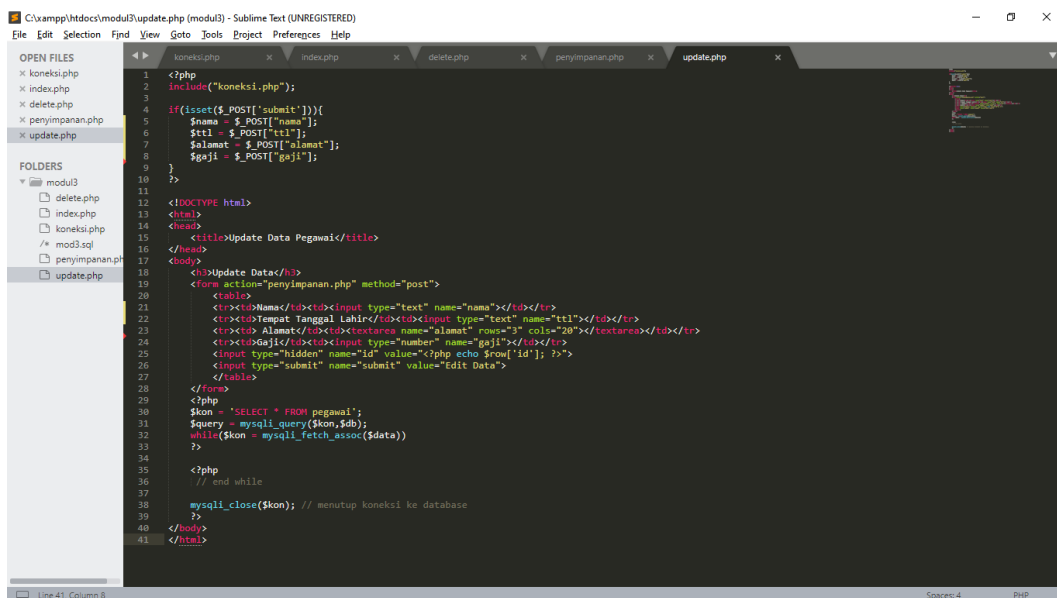


```
1 <?php
2 require_once("koneksi.php");
3
4 if(isset($_POST['submit'])){
5     $nama = $_POST['nama'];
6     $ttl = $_POST['ttl'];
7     $alamat = $_POST['alamat'];
8     $gaji = $_POST['gaji'];
9
10     // query insert data ke database dalam tabel anggota
11     $query = "INSERT INTO pegawai SET nama= '$nama', ttl= '$ttl', alamat= '$alamat', gaji= '$gaji'";
12     if(mysqli_query($kon, $query)){
13         header("Location: index.php");
14     }else{
15         echo "Tambah data gagal";
16     }
17 }
18
19 mysqli_close($kon); // menutup koneksi ke database
20 ?>
21
22 <!DOCTYPE html>
23 <html>
24 <head>
25 <title>Input Data Pegawai</title>
26 </head>
27 <body>
28 <h3>Tambah Data Pegawai</h3>
29 <form action="penyimpanan.php" method="post">
30 <table>
31 <tr><td>Nama</td><td><input type="text" name="nama"></td></tr>
32 <tr><td>Tempat Tanggal Lahir</td><td><input type="text" name="ttl"></td></tr>
33 <tr><td>Alamat</td><td><textarea name="alamat" rows="3" cols="20"></td></tr>
34 <tr><td>Gaji</td><td><input type="number" name="gaji"></td></tr>
35 </table>
36 <br>
37 <input type="submit" name="submit" value="Tambah Data">
38 <br><br>
39 <a href="index.php">Kembali ke Menu Utama</a>
40 </form>
41 </body>
42 </html>
```

Gambar 1.4. *Penyimpanan.php*.



Gambar 1.5. *Delete.php.*



Gambar 1.6. *Update.php.*

Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama	Tempat Tanggal Lahir	Alamat	Gaji	
01	Steven Duling	Kuala Kapuas, 20-02-2002	Jl. Merak II	3000000	Edit Hapus
02	Hussain Alam	Amuntai, 10-01-2001	Jl. Merdeka I	2500000	Edit Hapus
03	Elnita	Kuala Lumpur, 21-01-2000	Jl. Kuron	3500000	Edit Hapus
05	Maimunah	Barabai, 30-03-2000	Jl. Permai	3000000	Edit Hapus
06	Musram	Kanamit, 22-06-2001	Jl. Panjaitan	2000000	Edit Hapus

Gambar 1.7. *Tampilan Awal.*

Tambah Data Pegawai

Nama:

Tempat Tanggal Lahir:

Alamat:

Gaji:

[Kembali ke Menu Utama](#)

Gambar 1.8. *Tampilan Tambah Data.*

Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama	Tempat Tanggal Lahir	Alamat	Gaji	
01	Steven Duling	Kuala Kapuas, 20-02-2002	Jl. Merak II	3000000	Edit Hapus
02	Hussain Alam	Amuntai, 10-01-2001	Jl. Merdeka I	2500000	Edit Hapus
03	Elnita	Kuala Lumpur, 21-01-2000	Jl. Kuron	3500000	Edit Hapus
05	Maimunah	Barabai, 30-03-2000	Jl. Permai	3000000	Edit Hapus
06	Musram	Kanamit, 22-06-2001	Jl. Panjaitan	3000000	Edit Hapus
08	Roni Jon	Jogja, 24-07-1999	Jl. Melati III	3500000	Edit Hapus

Gambar 1.9. *Tampilan setelah Tambah Data.*

Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama	Tempat Tanggal Lahir	Alamat	Gaji	
01	Steven Duling	Kuala Kapuas, 20-02-2002	Jl. Merak II	3000000	Edit Hapus
02	Hussain Alam	Amuntai, 10-01-2001	Jl. Merdeka I	2500000	Edit Hapus
03	Elnita	Kuala Lumpur, 21-01-2000	Jl. Kuron	3500000	Edit Hapus
05	Maimunah	Barabai, 30-03-2000	Jl. Permai	3000000	Edit Hapus
08	Roni Jon	Jogja, 24-07-1999	Jl. Melati III	3500000	Edit Hapus

Gambar 1.10. *Tampilan Hapus Data.*

Update Data

Nama:

Tempat Tanggal Lahir:

Alamat:

Gaji:

Gambar 1.11. *Tampilan Edit Data.*