

**LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEB DAN MOBILE I**



Nama : Oscar Oktorian Almando
NIM : 193020503037
Kelas : A
Modul : III (PHP MySQL Database)

JURUSAN/PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021

BAB I

LANDASAN TEORI

1.1 Tujuan

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

1.2 Landasan Teori

1.2.1 Cara Membuka Koneksi dengan Database MySQL PHP

PHP memiliki banyak fungsi berbeda yang telah built-in untuk manipulasi database (Nyekrip 2015). Berikut ini adalah fungsi-fungsi built-in yang berhubungan dengan PHP dan MySQL :

```
mysql_connect()  
mysql_select_db()  
mysql_close()
```

a. Membuka Koneksi database MySQL PHP

Sebelum memanipulasi isi tabel dari sebuah database, hal pertama yang harus dilakukan adalah membuat script agar terhubung ke database. Fungsi *mysql_connect()*, sesuai namanya digunakan untuk menghubungkan ke database MySQL.

```
<?php  
  
$user_name = "root";  
$password = "";  
$database = "namadatabase";  
$host_name = "localhost";  
  
mysql_connect($host_name, $user_name, $password);  
  
echo "Koneksi Terbuka";
```

```
?>
```

Perlu diperhatikan empat baris pertama sintaks yang ada di atas, sintaks tersebut merupakan pengaturan variabel. Variabel pertama adalah *\$user_name* sedangkan variabel kedua adalah *\$password*, variabel pertama berisi nilai root sedangkan variabel kedua bernilai "" alias kosong, nilai ini diisi kosong karena nilai ini adalah nilai default untuk database MySQL.

Berikutnya adalah variabel *\$database* yang berisi nama database yang ingin dituju dan variabel *\$host_name* yang berisi nama host tempat database berasal.

```
mysql_connect($host_name, $user_name, $password);
```

Setelah variabel dibuat, barulah fungsi *mysql_connect* dapat dipanggil sehingga fungsi ini dapat diisi parameternya dengan variabel yang telah dibuat.

b. Menentukan Database yang ingin dibuka

Setelah membuat sintaks untuk variabel dan fungsi untuk menghubungkan MySQL, berikut ini merupakan sintaks untuk menentukan atau memilih sebuah database.

```
<?php
```

```
$user_name = "root";
```

```
$password = "";
```

```
$database = "namadatabase";
```

```
$host_name = "localhost";
```

```
mysql_connect($host_name, $user_name, $password);
```

```
mysql_select_db($database);
```

```
echo "Koneksi Terbuka";
```

```
?>
```

Pada sintaks diatas terdapat fungsi `mysql_select_db()`, fungsi ini adalah fungsi built-in PHP yang digunakan untuk memilih database MySQL yang ingin dihubungkan. Selanjutnya fungsi `mysql_select_db()` mengembalikan nilai true jika database yang dimasukkan ditemukan dan mengembalikan nilai false jika database tidak ditemukan, jika nilai tersebut true maka akan menampilkan pesan “Koneksi Terbuka”.

c. Cara Menutup Koneksi Database MySQL

Dalam PHP, `mysql_close()` digunakan untuk menutup koneksi ke server database MySQL. Berikut ini sintaks yang digunakan untuk menutup koneksi.

```
mysql_close($connect_db)
```

d. Menampilkan isi Tabel dari Database MySQL PHP

Setelah mengetahui bagaimana menghubungkan database dan kemudian menutup koneksinya, selanjutnya adalah untuk mengambil data atau membaca data tabel dari database MySQL menggunakan PHP.

Sebelum data ditampilkan, maka harus memastikan bahwa terdapat tabel dalam database yang akan dibuka. Untuk mengambil dan menampilkan isi dari tabel maka perlu menggunakan fungsi `mysql_fetch_assoc()` seperti pada sintaks berikut.

```
<?php
```

```
$user_name = "root";
```

```
$password = "";
```

```
$database = "namadatabase";
```

```

$host_name = "localhost";

$connect_db=mysql_connect($host_name, $user_name,
$password);

$find_db=mysql_select_db($database);

if ($find_db) {

$query = "SELECT * FROM namatabel";
$hasil = mysql_query($query);

while ( $kolom_db = mysql_fetch_assoc($hasil) ) {

echo $kolom_db['kolom_id'] . "<br />";
echo $kolom_db['kolom_ket'] . "<br />";

}

mysql_close($connect_db);

}else {

echo "Database Tidak Ada";

mysql_close($connect_db);

}

?>

```

Jika database tersedia maka query “SELECT * FROM namatabel” akan berjalan menggunakan fungsi *mysql_query()* dan

hasilnya disimpan dalam variabel *\$hasil*. Fungsi *mysql_fetch_assoc()* digunakan untuk mengambil data dari *\$hasil* dan disimpan dalam *\$field_db*. Setelah tersimpan dalam *\$field_db*, maka selanjutnya akan menampilkan data sesuai nama kolom.

1.2.2 Memasukkan Data ke Database

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

- Query SQL harus diberikan kutip dalam PHP.
- Nilai string di dalam 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 (Jurusan Teknik Informatika 2021), query untuk menambahkan data tersebut adalah sebagai berikut :

```
INSERT INTO table_name(column1, column2, column3, ...) VALUES  
(value1, value2, value3, ...)
```

1.2.3 Menghapus Data dari Database

Statement *DELETE* digunakan untuk menghapus baris data atau records dari tabel (Jurusan Teknik Informatika 2021). 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.

1.2.4 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 ..
```

BAB II

PEMBAHASAN

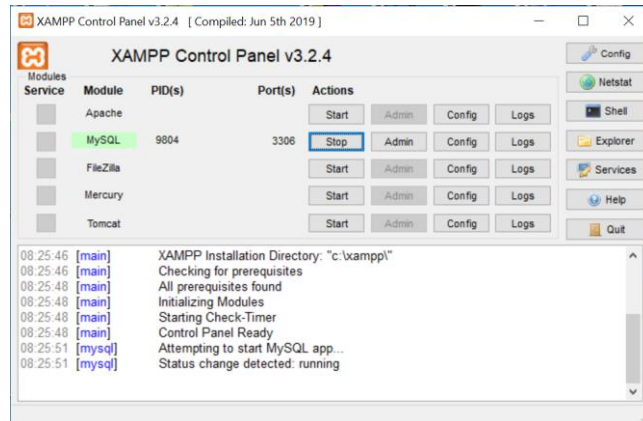
Tugas pada modul 3 adalah untuk membuat sebuah database yang berisi data-data pegawai serta tabel relasi. Kemudian database tersebut akan dihubungkan pada sebuah website atau program yang dapat menyimpan, menghapus dan mengubah data-data pegawai tersebut.

Dalam praktikum modul 3 ini, aplikasi yang digunakan untuk menuliskan sintaks PHP dan HTML adalah aplikasi Komodo Edit, sedangkan untuk membuat database menggunakan aplikasi SQLyog dan Command Prompt (CMD). Sintaks yang telah dibuat akan disimpan ke dalam format *.php dan dieksekusi untuk ditampilkan hasilnya menggunakan aplikasi XAMPP (untuk memproses PHP dan mengaktifkan koneksi database) dan browser untuk menampilkannya. File yang dibuat adalah tentunya satu file berekstensi .php yang merupakan jawaban tugas praktikum modul 3 ini, dan satu file berekstensi .css yang digunakan untuk mengubah tampilan yang ada pada file .php.

2.1 Membuat Database Pegawai

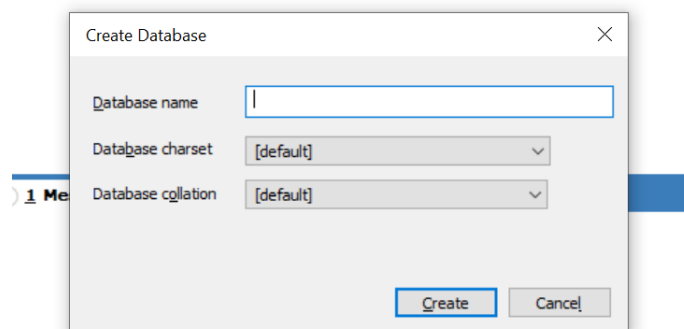
Database yang akan dibuat adalah sebuah database yang berisi tabel data pegawai, dan tabel yang berelasi dengan tabel tersebut. Tabel yang akan dibuat yaitu tabel *karyawan* yang akan menampung data pegawai, dan tabel yang berelasi adalah tabel *divisi*, yang mana merupakan tempat atau bagian dari sebuah kantor tempat pegawai bekerja.

Langkah pertama yang dilakukan adalah membuka aplikasi XAMPP untuk memulai koneksi database. Setelah database terkoneksi, maka selanjutnya membuka aplikasi SQLyog.



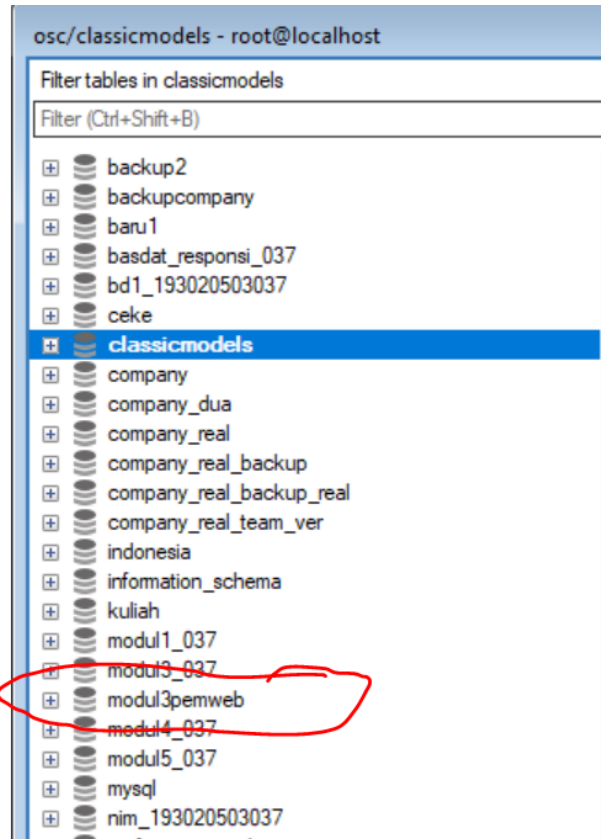
Gambar 2. 1 Tampilan XAMPP ketika sedang menjalankan koneksi database.

Langkah selanjutnya yaitu membuat database baru, dengan cara klik kanan pada bagian mana saja di *layout* yang berisi nama user beserta daftar database pada bagian kiri layar. Cara alternatifnya yaitu dengan cara menekan tombol kombinasi *Ctrl+d* pada keyboar, maka akan muncul jendela untuk pengaturan database baru seperti pemberian nama.



Gambar 2. 2 Tampilan ketika membuat sebuah database baru pada aplikasi SQLyog.

Kemudian database akan dinamai *modul3pemweb* karena database ini dibuat untuk menyelesaikan tugas praktikum yang diberikan. Setelah database berhasil dibuat, dapat dibuktikan dengan adanya nama database *modul3pemweb* pada daftar database di bagian kiri aplikasi.



Gambar 2. 3 Database yang berhasil dibuat.

Setelah database dibuat, maka selanjutnya akan membuat tabel dengan nama yang telah ditentukan sebelumnya. Tabel akan dibuat menggunakan Command Prompt. Berikut ini merupakan sintaks dalam membuat tabel-tabel tersebut.

```
MariaDB [modul3pemweb]> SHOW CREATE TABLE karyawan \G
***** 1. row *****
Table: karyawan
Create Table: CREATE TABLE `karyawan` (
  `id_karyawan` int(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `divisi` int(3) unsigned zerofill DEFAULT NULL,
  `Tgl_Lahir` date DEFAULT NULL,
  PRIMARY KEY (`id_karyawan`),
  KEY `divisi` (`divisi`),
  CONSTRAINT `karyawan_ibfk_1` FOREIGN KEY (`divisi`) REFERENCES `divisi` (`id_divisi`)
) ENGINE=InnoDB AUTO_INCREMENT=21 DEFAULT CHARSET=utf8mb4
1 row in set (0.035 sec)
```

Gambar 2. 4 Sintaks membuat tabel *karyawan*.

```
MariaDB [modul3pemweb]> DESC karyawan;
+-----+-----+-----+-----+-----+-----+
| Field      | Type                | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_karyawan | int(3) unsigned zerofill | NO   | PRI | NULL    | auto_increment |
| nama        | varchar(50)          | YES  |     | NULL    |                |
| divisi      | int(3) unsigned zerofill | YES  | MUL | NULL    |                |
| Tgl_Lahir   | date                 | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.027 sec)

MariaDB [modul3pemweb]>
```

Gambar 2. 5 Tabel *karyawan* yang berhasil dibuat.

```
MariaDB [modul3pemweb]> SHOW CREATE TABLE divisi \G
***** 1. row *****
      Table: divisi
Create Table: CREATE TABLE `divisi` (
  `id_divisi` int(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
  `nama_divisi` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id_divisi`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4
1 row in set (0.051 sec)
```

Gambar 2. 6 Sintaks membuat tabel *divisi*.

```
MariaDB [modul3pemweb]> DESC divisi;
+-----+-----+-----+-----+-----+-----+
| Field      | Type                | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_divisi   | int(3) unsigned zerofill | NO   | PRI | NULL    | auto_increment |
| nama_divisi | varchar(50)          | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.026 sec)

MariaDB [modul3pemweb]>
```

Gambar 2. 7 Tabel *divisi* yang berhasil dibuat.

Setelah tabel dibuat, selanjutnya yaitu mengisi beberapa data pada masing-masing tabel sebagai ‘tampilan’ awal. Pengisian awal beberapa data ini menggunakan aplikasi SQLyog.

2.2 Membuat Program Menyimpan, Menghapus dan Mengubah Data

Program atau website yang dibuat terdiri dari beberapa file .php, dan 1 file .css untuk mengatur tampilan website. Berikut ini merupakan sintaks beserta penjelasannya pada masing-masing file.

2.2.1 Connect.php

File yang bernama *connect.php* merupakan file yang berisi sintaks untuk menghubungkan database SQL dan melakukan perintah-perintah SQL melalui fungsi tertentu. Dapat dikatakan file *connect.php*

merupakan file yang sangat penting dan berperan dalam manipulasi data yang akan dimasukkan dan data yang telah dimasukkan.

Berikut ini merupakan sintaks dari file *connect.php*.

```
<?php
    $db = mysqli_connect("localhost", "root", "",
    "modul3pemweb");

    function query($query){
        global $db;
        $result = mysqli_query($db, $query);
        $rows = [];
        while($row = mysqli_fetch_assoc($result)){
            $rows[]=$row;
        }
        return $rows;
    }

    function tambahKaryawan($data){
        global $db;
        $nama = htmlspecialchars($data["nama"]);
        $divisi =
        htmlspecialchars($data["id_divisi"]);
        $tgl = htmlspecialchars($data["Tgl_lahir"]);
        $query = "INSERT INTO karyawan VALUES
        (', '$nama', '$divisi', '$tgl')";
        mysqli_query($db, $query);
        return mysqli_affected_rows($db);
    }

    function ubahKaryawan($data){
        global $db;
        $id_karyawan = $data["id_karyawan"];
```

```

        $nama = htmlspecialchars($data["nama"]);
        $divisi = htmlspecialchars($data["divisi"]);
        $tgl = htmlspecialchars($data["Tgl_lahir"]);
        $query = "UPDATE karyawan SET nama =
'$nama', divisi = '$divisi', Tgl_lahir= '$tgl' WHERE
id_karyawan = $id_karyawan";

        mysqli_query($db, $query);
        return mysqli_affected_rows($db);
    }

```

```

    function hapusKaryawan($id){
        global $db;

        mysqli_query($db, "DELETE FROM karyawan
WHERE id_karyawan = $id");
        return mysqli_affected_rows($db);
    }

```

?>

Di dalam file ini, hanya berisi sintaks PHP, yang mana terdiri dari 1 deklarasi variabel, dan 4 buah fungsi. Satu variabel tersebut adalah variabel *\$db*, yang mana berisi perintah untuk menghubungkan PHP dan SQL dengan fungsi *mysqli* yang memiliki parameter *host*, *username*, *password* dan *nama database*.

Adapun 4 fungsi yang ada pada file ini yaitu *query()*, *tambahKaryawan()*, *ubahKaryawan* dan *hapusKaryawan()*.

Fungsi *query()* digunakan untuk mengambil data yang ada pada database, dan memasukkan setiap baris data tersebut ke dalam sebuah variabel array yang nantinya akan dikembalikan nilainya ke fungsi *query()* itu sendiri. Pada tugas praktikum ini, fungsi *query()* digunakan untuk memanggil data pada sebuah tabel yang ada di database *modul3pemweb*.

Fungsi *tambahKaryawan()* digunakan untuk menangkap data hasil inputan pengguna pada saat berada di website dan diolah agar dapat menyimpannya ke dalam database menggunakan perintah/kueri *INSERT INTO (..)*.

Fungsi *ubahKaryawan()* digunakan untuk menangkap data hasil inputan pengguna pada saat berada di website dan diolah agar dapat mengubah data yang sudah ada pada tabel database, dengan menggunakan perintah/kueri *UPDATE (..) SET*.

Fungsi *hapusKaryawan()* digunakan untuk menghapus data tertentu pada tabel *karyawan* dengan menangkap data *id_karyawan*.

2.2.2 Modul3

File ini merupakan file yang menampilkan halaman utama dari program tugas praktikum yang dibuat. Pada file ini tabel akan ditampilkan untuk melihat data-data yang telah dimasukkan, dan pada bagian sisi tabel akan terdapat perintah untuk mengubah atau menghapus data tersebut. Selain itu, pada halaman ini juga terdapat bagian atau *layout* untuk menambahkan data, yang mana tentunya akan memerlukan sintaks dari file *connect.php* untuk menjalankannya.

Berikut ini merupakan bagian-bagian sintaks yang dibuat.

```
<?php
    require 'connect.php';
    $karyawan = query("SELECT k.id_karyawan AS idka,
k.nama AS nam,
        d.nama_divisi AS divi, k.Tgl_lahir as tglh
        FROM karyawan as k, divisi as d
        WHERE k.divisi=d.id_divisi ORDER BY
k.id_karyawan");
    $divisi = query("SELECT * FROM divisi");

    if(isset($_POST["submit"])) {
        if (tambahKaryawan($_POST)>0){
            echo"<script>
```

```
        alert('data berhasil ditambahkan!');
        document.location.href = 'modul3.php';
    </script>";
} else {
    echo "<script>
    alert('data gagal ditambahkan!');
    document.location.href = 'modul3.php';
    </script>";
}
}
?>
```

Sintaks di atas merupakan sintaks PHP (ditambah Javascript), yang mana pada baris pertama berisi perintah *require 'connect.php'*; yang mana akan memanggil fungsi-fungsi, serta variabel yang telah ditulis pada file *connect.php* sehingga dapat digunakan pada halaman ini.

Selanjutnya terdapat variabel *\$karyawan* untuk menampung data dari gabungan tabel (*INNER JOIN*) dari tabel *karyawan* dan tabel *divisi*. Selain itu terdapat juga variabel *\$divisi* untuk menampung data dari tabel *divisi*.

Sintaks selanjutnya terdapat pengecekan apakah terdapat nilai yang ditampung pada metode *post* (submit) atau tidak. Jika benar, maka akan memeriksa apakah terdapat nilai yang di-*INSERT* menggunakan fungsi *tambahKaryawan* atau tidak. Jika terdapat data, maka akan memunculkan pesan bahwa data berhasil dimasukkan dan tabel akan secara otomatis menambahkan baris data baru. Jika tidak ada data yang dimasukkan, maka halaman akan memunculkan pesan bahwa data gagal dimasukkan. Jika tidak ada nilai yang ditampung pada metode *post*, maka halaman tidak akan melakukan apapun. Penanganan jika tidak ada data yang masuk tidak dituliskan karena berguna ketika pengguna baru mengakses halaman, sehingga tidak memunculkan pesan apapun yang mungkin dapat membuat pengguna keliru.

```

<!DOCTYPE html>
<html>
<head>
    <title> Modul 3 | Oscar Oktorian Almando </title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <a href="#TK" id="awal">
        <div class="rad">
            Tambah Data
        </div>
    </a>

    <div class="TabelKaryawan" id="tbk">
        <div class="tengah"><h1>Tabel Karyawan</h1> </div>
        <div class="tabeltengah">
            <table class="table1">
                <tr>
                    <th>ID Karyawan</th>
                    <th>Nama</th>
                    <th>Divisi</th>
                    <th>Tanggal Lahir</th>
                </tr>

                <?php foreach ($karyawan as $row) : ?>
                <tr>
                    <td><?php echo $row["idka"];?></td>
                    <td><?php echo $row["nam"]; ?></td>
                    <td><?php echo $row["divi"]; ?></td>
                    <td><?php echo $row["tglh"]; ?></td>

```

```

        <td>
            <a href='modul3Update.php?id=<?php
echo $row['idka']; ?>'>ubah</a> |
            <a class='warn'
href='modul3Hapus.php?id=<?php echo $row['idka']; ?>'
onclick='return confirm('Hapus data?');'>hapus</a>
        </td>
    </tr>
<?php endforeach; ?>
</table>
</div>
</div>

<br><br><br><br><br><br><br><br>

<div class="TambahKaryawan" id="TK">
    <div class="tengah"><h1>Tambah Data
Karyawan</h1></div>
    <form action="" method="post">
        <div class="tengah">
            <label for="nama">Nama</label>
        </div>
        <div class="tengah">
            <input type="text" name="nama"
id="nama" required>
        </div>
        <div class="tengah">
            <label for="id_divisi">Divisi</label>
        </div>
        <div class="tengah">
            <select name="id_divisi" class="form-
control" id="id_divisi">

```



```

        <option selected>Pilih Divisi..</option>
        <?php foreach($divisi as $divis) : ?>
            <option value="<?= $divis['id_divisi']
?>"><?= $divis['nama_divisi'] ?></option>
        <?php endforeach; ?>
    </select>
</div>
<div class="tengah">
    <label for="Tgl_lahir">Tanggal
    Lahir</label>
</div>
<div class="tengah">
    <input type="date" name="Tgl_lahir"
id="Tgl_lahir" required>
</div>
<div class="tengah">
    <button class="rad" type="submit"
name="submit">Tambah</button>
</div>
</form>
</div>

<a href="#awal">
<div class="tengah">
    Kembali ke Tabel
</div>
</a>
</body>
</html>

```

Sintaks di atas merupakan bagian yang nantinya akan tampil pada browser atau ketika dijalankan. Terdapat dua bagian atau *layout* utama yang akan ditampilkan, yakni *layout* untuk menampilkan tabel *INNER*

JOIN dari database *modul3pemweb* dan *layout* untuk melakukan penambahan data pada tabel *karyawan*.

Pada *layout* pertama, untuk menampilkan sebuah tabel, maka diperlukan tag `<table>` dan tag turunannya yaitu table row `<tr>` dan table data `<td>`. Tabel baris pertama akan menampilkan judul atau keterangan atribut yang dimiliki tabel database.

Khusus untuk tabel baris kedua, akan menggunakan fungsi PHP yaitu *foreach* agar masing-masing data pada setiap baris dapat ditampilkan sebanyak data yang disimpan. Penggunaan fungsi *foreach* ditulis pada sintaks karena data tabel (dalam hal ini adalah variabel *\$karyawan*) adalah sebuah data yang berupa *array*. Di dalam *foreach* tersebut terdapat satu baris tabel `<tr>` yang menampung masing-masing atribut atau kolom, dan kolom tabel data `<td>` terakhir akan berisi dua perintah, yaitu *ubah* dan *hapus*. Kedua perintah ini akan mengarahkan pengguna ke suatu halaman agar dapat mengeksekusi suatu sintaks yang hanya ada pada halaman itu.

Pada *layout* kedua, untuk dapat mengirimkan data inputan, diperlukan tag `<form>` dengan atribut *action* dan *method* agar dapat menampung hasil input pengguna. Masing-masing kolom memiliki tipe tersendiri agar dapat menampung nilai presisi sesuai pada tabel dan agar tidak menghasilkan sebuah error saat memasukkannya ke dalam database.

Untuk menampung kolom nama, tipe yang digunakan adalah *text*. Kemudian untuk menampung divisi (*id_divisi*) maka menggunakan tag `<select>` agar pengguna dapat memilih divisi mana karyawan ini ditempatkan. Di dalam tag `<select>` juga terdapat tag `<option>` yang akan memunculkan pilihan id divisi (yang direpresentasikan sebagai nama divisi). Untuk menampung *tanggal_lahir*, maka tipe input adalah *date*.

2.2.3 Modul3Update.php

File ini akan dijalankan ketika pengguna melakukan klik pada sisi tabel yang bertuliskan *ubah* di halaman utama. Berikut ini merupakan sintaks yang ada pada file *modul3Update.php*.

```
<?php
    require 'connect.php';
    $id = $_GET["id"];
    $karyawan = query("SELECT * FROM karyawan
INNER JOIN divisi ON karyawan.divisi=divisi.id_divisi
    WHERE karyawan.id_karyawan = $id")[0];
    $divisi = query("SELECT * FROM divisi");

    if(isset($_POST["submit"])) {
        if (ubahKaryawan($_POST)>0){
            echo "<script>
                alert('data berhasil diubah!');
                document.location.href =
' modul3.php';
            </script>";
        } else {
            echo "<script>
                alert('data gagal diubah!');
                document.location.href =
' modul3.php';
            </script>";
        }
    }

?>

<!DOCTYPE html>
```

```

<html>
  <head>
    <title>Halaman Update Data | Oscar Oktorian
    Almando</title>
    <link rel="stylesheet" type="text/css"
href="style.css">
  </head>
  <body>
    <div class="updt">
      <div class="tengah">
        <h1> Update Data </h1>
      </div>
      <form action="" method="post">
        <div class="tengah">
          <label for="nama">Nama</label>
        </div>
        <div class="tengah">
          <input type="text" name="nama" id="nama"
required value="<?php echo $karyawan["nama"]; ?>">
        </div>
        <div class="tengah">
          <label for="nama">ID : </label>
        </div>
        <div class="tengah">
          <input type="text" name="id_karyawan"
id="nama" required value="<?php echo
$karyawan["id_karyawan"]; ?>">
        </div>
        <div class="tengah">
          <label for="divisi">Pilih
Divisi</label>

```

```

</div>
<div class="tengah">
    <select name="divisi" class="form-control"
id="divisi">
        <option          value="<?php      echo
$aryawan["divisi"]; ?>" selected>
            <?php          echo
$aryawan["nama_divisi"]; ?>
        </option>
        <?php foreach($divisi as $divis) : ?>
            <option          value="<?=
$divis["id_divisi"]      ?>"><?=      $divis["nama_divisi"]
?></option>
        <?php endforeach; ?>
    </select>
</div>
<div class="tengah">
    <label for="Tgl_lahir">Tanggal Lahir :
</label>
</div>
<div class="tengah">
    <input type="date" name="Tgl_lahir"
id="Tgl_lahir" required value="<?php      echo
$aryawan["Tgl_lahir"]; ?>">
</div>
<div class="tengah">
    <button class="rad" type="submit"
name="submit">Update</button>
</div>
</form>
</div>

```

```
<a href="modul3.php">
  <div class="tengah">
    Kembali ke Tabel
  </div>
</a>
</body>
</html>
```

Sintaks di atas sama seperti sintaks pada file *modul3.php* yakni sintaks untuk menambah data *karyawan*. Namun, sintaks pada file ini memiliki perbedaan, file ini akan menangkap *id* dari variabel method *POST* setelah pengguna melakukan klik pada baris tabel tertentu. Kemudian *id* tersebut akan digunakan sebagai acuan atau penentu data mana yang akan diubah.

Kemudian terdapat variabel *\$karyawan* dan *\$divisi* untuk menangkap informasi mengenai data yang ingin diubah. Sehingga, pada kolom ubah/update data tersebut langsung berisi data yang ingin diubah. Hal ini bertujuan agar pengguna tidak keliru ketika ingin mengubah data, yang mungkin menyebabkan data tidak berubah atau menyebabkan error.

Selanjutnya terdapat pengecekan apakah terdapat sebuah data atau tidak pada metode *POST* dalam halaman ini. Jika benar, maka akan melakukan pemeriksaan kembali apakah fungsi *ubahData* sudah berisi nilai atau tidak. Jika benar, maka akan memunculkan pesan bahwa data berhasil diubah dan halaman akan dialihkan ke halaman utama yang menampilkan tabel dengan data terbaru. Jika tidak, maka halaman akan memunculkan pesan bahwa data gagal diubah dan halaman akan dialihkan menuju halaman utama (*modul3.php*).

2.2.4 Modul3Hapus.php

File ini akan dieksekusi sintaksnya ketika pengguna di halaman pertama melakukan klik pada sisi tabel yang bertuliskan ‘hapus’. Berikut ini merupakan sintaks untuk file *modul3Hapus*.

```
<?php
    require 'connect.php';

    $id = $_GET["id"];
    if(hapusKaryawan($id)>0 ){
        echo "<script>

            alert('data berhasil dihapus!');

            document.location.href = 'modul3.php';

        </script>";
    } else {
        echo "<script>

            alert('data gagal dihapus!');

            document.location.href = 'modul3.php';

        </script>";
    }
?>
```

Tentunya pada sintaks di atas memerlukan file *connect.php* agar dapat menjalankan fungsi *hapusKaryawan*, karena itulah diperlukan perintah *require 'connect.php'* untuk memanggilnya.

Halaman ini akan mengambil nilai *id* yang mana akan menampung nilai dari *id_karyawan* yang telah dikirimkan dari halaman utama. Kemudian akan diperiksa, apakah fungsi *hapusKaryawan* berisi sebuah nilai untuk dieksekusi dan berhasil menjalankannya. Jika

ya, maka halaman akan memunculkan pesan bahwa data berhasil dihapus dan halaman dialihkan menuju halaman utama. Jika tidak, maka halaman akan memunculkan pesan bahwa data gagal untuk dihapus dan halaman dialihkan menuju halaman utama.

2.2.5 Style.css

File ini memiliki fungsi untuk menambah atau mengatur tampilan pada file *modul3.php* dan *modul3Update.php*.

Berikut ini merupakan sintaks untuk file ini.

```
body{
    height: 420px;
    background-color: lightblue;
}

div.updt{
    padding : 30px 20px 60px 20px;
    margin: 5% 20% 10% 20%;
    border: 1px solid white;
    border-radius: 20px;
    background-color: white;
    box-shadow: 3px 3px 10px #000000;
}

div.TabelKaryawan{
    padding : 30px 20px 60px 20px;
    margin: 5% 10% 10% 10%;
    border: 1px solid white;
    border-radius: 20px;
    background-color: white;
    box-shadow: 3px 3px 10px #000000;
}
```



```
div.TambahKaryawan{  
    padding : 30px 20px 60px 20px;  
    margin: 5% 20% 10% 20%;  
    border: 1px solid white;  
    border-radius: 20px;  
    background-color: white;  
    box-shadow: 3px 3px 10px #000000;  
}
```

```
div.tengah{  
    margin-top: 6px;  
    margin-bottom: 6px;  
    text-align: center;  
    font-weight: bold;  
}
```

```
div.tabeltengah{  
    margin-top: 6px;  
    margin-bottom: 6px;  
    margin-left: 120px;  
    margin-right: 20px;  
}
```

```
.rad{  
    margin: 0% 45% 0% 45%;  
    background-color: white;  
    padding : 5px 5px 5px 5px;  
    border-radius: 5px;  
    text-align: center;  
    border: 2px solid black;  
}
```

```

.table1 {
    font-family: sans-serif;
    color: #232323;
    border-collapse: collapse;
}

.table1, th, td {
    border: 1px solid #999;
    padding: 8px 20px;
}

.warn{
    color: #ff0000;
    font-style: italic;
    font-weight: lighter;
}

```

2.3 Tampilan Halaman yang Dibuat

Berikut ini merupakan tampilan yang ada pada halaman utama.

[Tambah Data](#)

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah hapus
002	Roll	IT	1998-02-22	ubah hapus
003	Ricita	Public Relation	1998-09-30	ubah hapus
004	Richman	Finance	1992-10-23	ubah hapus
005	Amin	Finance	1999-11-10	ubah hapus
006	Elon	Business Strategy	1997-12-12	ubah hapus
007	Musk	Business Strategy	1997-12-12	ubah hapus
008	Ricardo	Product Development	1992-12-25	ubah hapus
009	Milos	Product Development	1992-12-25	ubah hapus
010	Herp	IT	2001-01-01	ubah hapus
018	Samir	IT	1998-12-02	ubah hapus
019	Abdul	Business Strategy	1987-07-14	ubah hapus
020	Sarukh	Business Strategy	2002-12-11	ubah hapus

Gambar 2. 8 Halaman utama - layout menampilkan tabel data Karyawan.

Tambah Data Karyawan

Nama

Divisi

Tanggal Lahir

[Kembali ke Tabel](#)

Gambar 2. 9 Halaman utama - Layout tambah data karyawan.

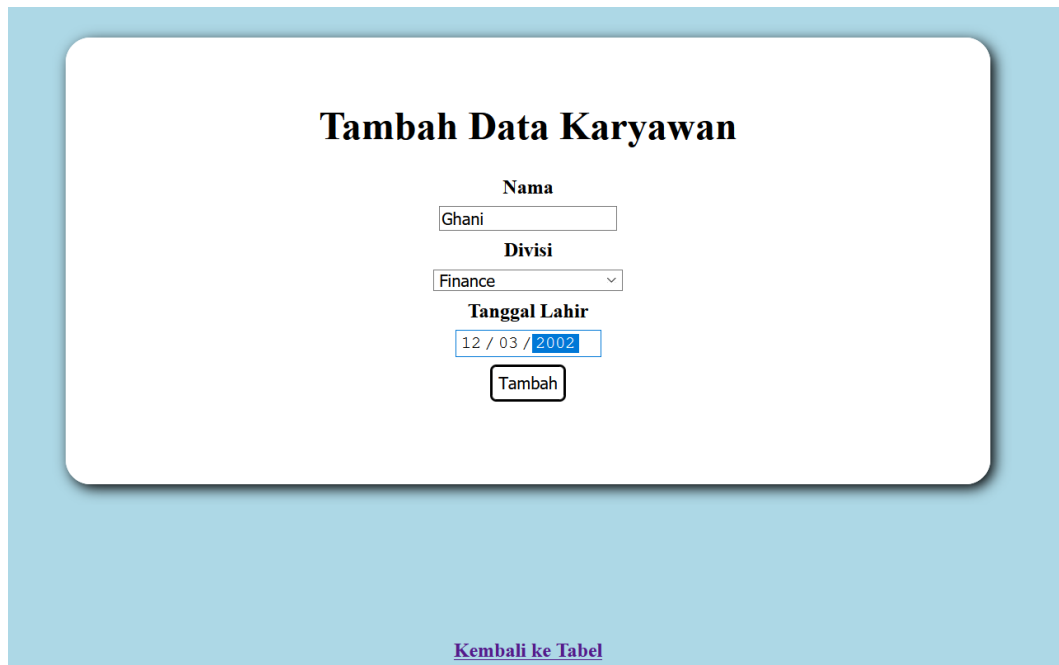
Sebagai perbandingan, dengan menggunakan perintah yang sama pada nilai variabel *\$karyawan* di file *modul3.php*, berikut ini merupakan data yang muncul pada jendela *Command Prompt*.

```

C:\WINDOWS\system32\cmd.exe - mysql -u root
MariaDB [(none)]> use modul3pemweb
Database changed
MariaDB [modul3pemweb]> SELECT k.id_karyawan AS idka, k.nama AS nam,
  -> d.nama_divisi AS divi, k.Tgl_lahir as tglh
  -> FROM karyawan as k, divisi as d
  -> WHERE k.divisi=d.id_divisi ORDER BY k.id_karyawan
  -> ;
+-----+-----+-----+-----+
| idka | nam  | divi                | tglh      |
+-----+-----+-----+-----+
| 001  | Rick | IT                  | 1998-12-10 |
| 002  | Roll | IT                  | 1998-02-22 |
| 003  | Ricitas | Public Relation    | 1998-09-30 |
| 004  | Richman | Finance            | 1992-10-23 |
| 005  | Amin  | Finance            | 1999-11-10 |
| 006  | Elon  | Business Strategy   | 1997-12-12 |
| 007  | Musk  | Business Strategy   | 1997-12-12 |
| 008  | Ricardo | Product Development | 1992-12-25 |
| 009  | Milos  | Product Development | 1992-12-25 |
| 010  | Herp  | IT                  | 2001-01-01 |
| 018  | Samir  | IT                  | 1998-12-02 |
| 019  | Abdul  | Business Strategy   | 1987-07-14 |
| 020  | Sarukh | Business Strategy   | 2002-12-11 |
+-----+-----+-----+-----+
13 rows in set (0.127 sec)

MariaDB [modul3pemweb]>
  
```

Gambar 2. 10 Tabel *INNER JOIN* pada jendela CMD.



The screenshot shows a web form titled "Tambah Data Karyawan" (Add Employee Data) centered on a light blue background. The form is contained within a white rounded rectangle. It includes three input fields: a text box for "Nama" (Name) containing "Ghani", a dropdown menu for "Divisi" (Department) with "Finance" selected, and a date picker for "Tanggal Lahir" (Date of Birth) showing "12 / 03 / 2002". Below these fields is a "Tambah" (Add) button. At the bottom of the page, there is a link labeled "Kembali ke Tabel" (Return to Table).

Tambah Data Karyawan

Nama
Ghani

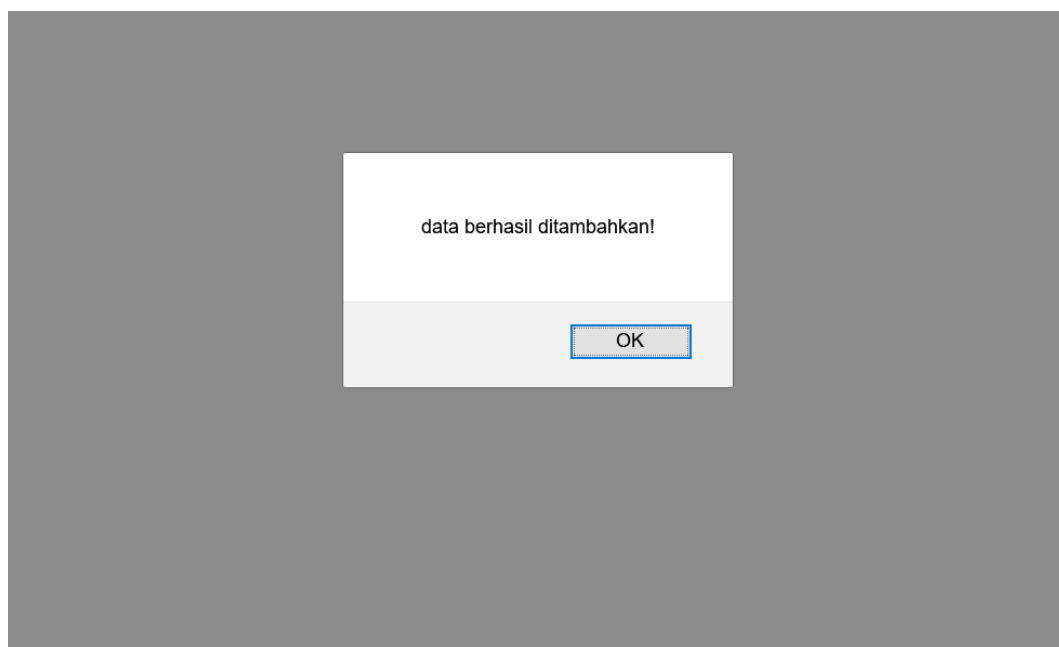
Divisi
Finance

Tanggal Lahir
12 / 03 / 2002

Tambah

[Kembali ke Tabel](#)

Gambar 2. 11 Halaman utama - tampilan saat pengisian data.



Gambar 2. 12 Halaman utama - tampilan saat data berhasil ditambahkan.

Tampilkan Data

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah / hapus
002	Roli	IT	1998-02-22	ubah / hapus
003	Riclas	Public Relation	1998-09-30	ubah / hapus
004	Richman	Finance	1992-10-23	ubah / hapus
005	Amin	Finance	1999-11-10	ubah / hapus
006	Elon	Business Strategy	1997-12-12	ubah / hapus
007	Musk	Business Strategy	1997-12-12	ubah / hapus
008	Ricardo	Product Development	1992-12-25	ubah / hapus
009	Milos	Product Development	1992-12-25	ubah / hapus
010	Harp	IT	2001-01-01	ubah / hapus
018	Samir	IT	1998-12-02	ubah / hapus
019	Abdul	Business Strategy	1987-07-14	ubah / hapus
020	Sanukh	Business Strategy	2002-12-11	ubah / hapus

Tampilkan Data

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah / hapus
002	Roli	IT	1998-02-22	ubah / hapus
003	Riclas	Public Relation	1998-09-30	ubah / hapus
004	Richman	Finance	1992-10-23	ubah / hapus
005	Amin	Finance	1999-11-10	ubah / hapus
006	Elon	Business Strategy	1997-12-12	ubah / hapus
007	Musk	Business Strategy	1997-12-12	ubah / hapus
008	Ricardo	Product Development	1992-12-25	ubah / hapus
009	Milos	Product Development	1992-12-25	ubah / hapus
010	Harp	IT	2001-01-01	ubah / hapus
018	Samir	IT	1998-12-02	ubah / hapus
019	Abdul	Business Strategy	1987-07-14	ubah / hapus
020	Sanukh	Business Strategy	2002-12-11	ubah / hapus
021	Ghani	Finance	2002-12-03	ubah / hapus

Gambar 2. 13 Halaman utama - Tampilan tabel sebelum (kiri) dan sesudah data dimasukan (kanan).

Update Data

Nama

ID :

Pilih Divisi

Finance ▼

Tanggal Lahir :

Gambar 2. 14 Halaman update data.



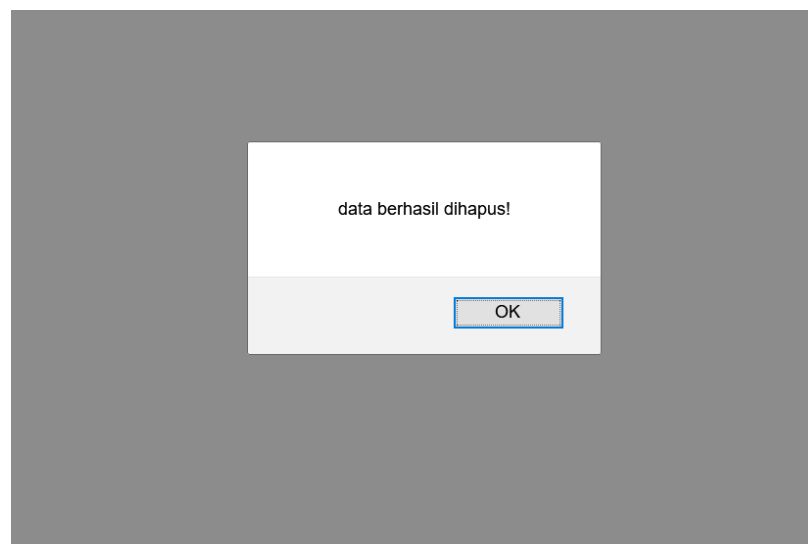
Gambar 2. 15 Halaman update data - ketika data berhasil diubah.

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah hapus
002	Roll	IT	1998-02-22	ubah hapus
003	Ricitas	Public Relation	1998-09-30	ubah hapus
004	Richman		1998-10-23	ubah hapus
005	Amin		1999-11-10	ubah hapus
006	Elon		1999-12-12	ubah hapus
007	Musk		1997-12-12	ubah hapus
008	Ricardo	Product Development	1992-12-25	ubah hapus
009	Milos	Product Development	1992-12-25	ubah hapus
010	Herp	IT	2001-01-01	ubah hapus
018	Samir	IT	1998-12-02	ubah hapus
019	Abdul	Business Strategy	1987-07-14	ubah hapus
020	Sarukh	Business Strategy	2002-12-11	ubah hapus
021	Ghani	Public Relation	2002-12-03	ubah hapus

Hapus data?

OK Cancel

Gambar 2. 16 Halaman utama - tampilan konfirmasi ketika data ingin dihapus.



Gambar 2. 17 Halaman hapus data - tampilan ketika data berhasil dihapus.

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah hapus
002	Roll	IT	1998-02-22	ubah hapus
003	Ricitas	Public Relation	1998-09-30	ubah hapus
004	Richman	Finance	1998-10-23	ubah hapus
005	Amin	Finance	1999-11-10	ubah hapus
006	Elon	Business Strategy	1999-12-12	ubah hapus
007	Musk	Business Strategy	1997-12-12	ubah hapus
008	Ricardo	Product Development	1992-12-25	ubah hapus
009	Milos	Product Development	1992-12-25	ubah hapus
010	Herp	IT	2001-01-01	ubah hapus
018	Samir	IT	1998-12-02	ubah hapus
019	Abdul	Business Strategy	1987-07-14	ubah hapus
020	Sarukh	Business Strategy	2002-12-11	ubah hapus
021	Ghani	Finance	2002-12-03	ubah hapus

Gambar 2. 18 Halaman utama - sebelum (kiri) dan sesudah data dihapus (kanan).

BAB III

KESIMPULAN

Berdasarkan hasil dari tugas praktikum yang dikerjakan, hal yang dapat disimpulkan dari praktikum modul 3 ini yaitu PHP dapat menghubungkan database di MySQL sehingga suatu database itu dapat ditampilkan, ditambah, diubah maupun dihapus baik tabel, data, ataupun database itu sendiri. Agar dapat mengubungkannya, diperlukan beberapa fungsi yang dipanggil dalam PHP. Fungsi pertama yang harus dimasukkan adalah fungsi *mysqli_connect()*, yang mana fungsi ini akan menghubungkan MySQL dengan database tertentu yang telah ditentukan dalam parameteranya.

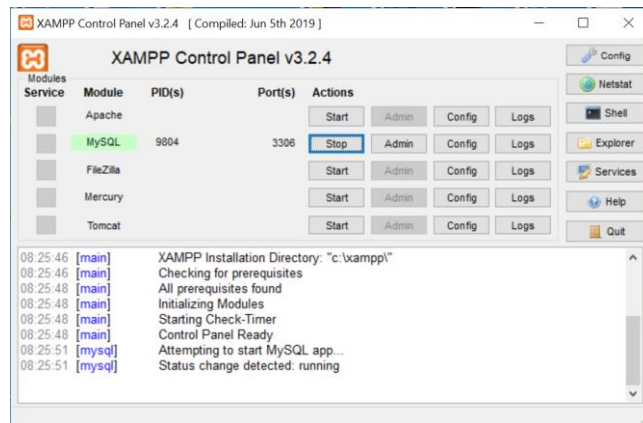
Penggunaan fungsi *mysql* pada PHP tergantung pada kebutuhan website itu sendiri, yang berarti bahwa tidak semua fungsi harus dimasukkan tapi hanya masukkan fungsi yang diperlukan.

DAFTAR PUSTAKA

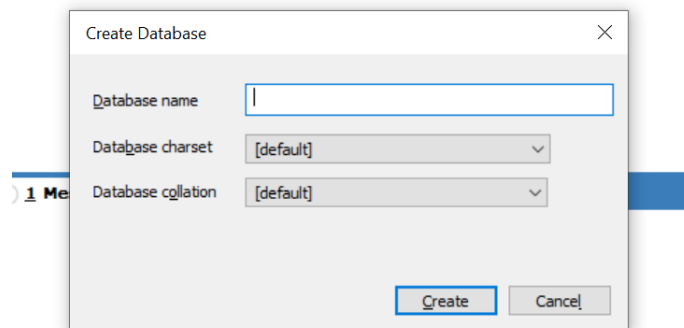
Jurusan Teknik Informatika. 2021. "MODUL PRAKTIKUM PEMROGRAMAN WEB I."

Nyekrip, Admin. 2015. "Cara Koneksi Database MySQL Dengan PHP."
<https://www.nyekrip.com/cara-koneksi-database-mysql-dengan-php/> (April 9, 2021).

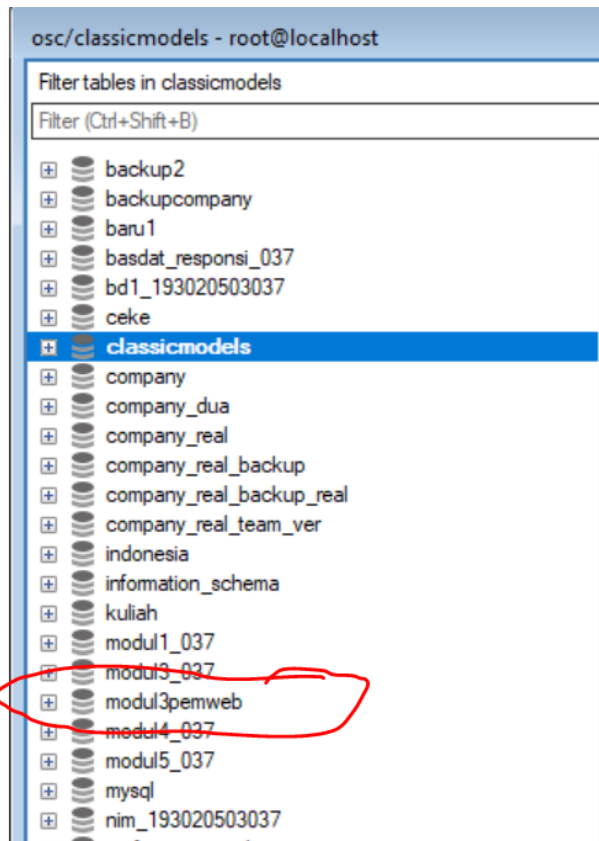
LAMPIRAN



Gambar 2. 1 Tampilan XAMPP ketika sedang menjalankan koneksi database.



Gambar 2. 2 Tampilan ketika membuat sebuah database baru pada aplikasi SQLyog.



Gambar 2. 3 Database yang berhasil dibuat.

```
MariaDB [modul3pemweb]> SHOW CREATE TABLE karyawan \G
***** 1. row *****
Table: karyawan
Create Table: CREATE TABLE `karyawan` (
  `id_karyawan` int(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `divisi` int(3) unsigned zerofill DEFAULT NULL,
  `Tgl_Lahir` date DEFAULT NULL,
  PRIMARY KEY (`id_karyawan`),
  KEY `divisi` (`divisi`),
  CONSTRAINT `karyawan_ibfk_1` FOREIGN KEY (`divisi`) REFERENCES `divisi` (`id_divisi`)
) ENGINE=InnoDB AUTO_INCREMENT=21 DEFAULT CHARSET=utf8mb4
1 row in set (0.035 sec)
```

Gambar 2. 4 Sintaks membuat tabel karyawan.

```
MariaDB [modul3pemweb]> DESC karyawan;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_karyawan | int(3) unsigned zerofill | NO | PRI | NULL | auto_increment |
| nama | varchar(50) | YES | | NULL | |
| divisi | int(3) unsigned zerofill | YES | MUL | NULL | |
| Tgl_Lahir | date | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.027 sec)

MariaDB [modul3pemweb]>
```

Gambar 2. 5 Tabel karyawan yang berhasil dibuat.

```

MariaDB [modul3pemweb]> SHOW CREATE TABLE divisi \G
***** 1. row *****
      Table: divisi
Create Table: CREATE TABLE `divisi` (
  `id_divisi` int(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
  `nama_divisi` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id_divisi`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4
1 row in set (0.051 sec)

```

Gambar 2. 6 Sintaks membuat tabel divisi.

```

MariaDB [modul3pemweb]> DESC divisi;
+-----+-----+-----+-----+-----+-----+
| Field      | Type                               | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_divisi  | int(3) unsigned zerofill          | NO   | PRI | NULL    | auto_increment |
| nama_divisi | varchar(50)                        | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.026 sec)

```

Gambar 2. 7 Tabel divisi yang berhasil dibuat.

[Tambah Data](#)

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah hapus
002	Roll	IT	1998-02-22	ubah hapus
003	Ricitas	Public Relation	1998-09-30	ubah hapus
004	Richman	Finance	1992-10-23	ubah hapus
005	Amin	Finance	1999-11-10	ubah hapus
006	Elon	Business Strategy	1997-12-12	ubah hapus
007	Musk	Business Strategy	1997-12-12	ubah hapus
008	Ricardo	Product Development	1992-12-25	ubah hapus
009	Milos	Product Development	1992-12-25	ubah hapus
010	Herp	IT	2001-01-01	ubah hapus
018	Samir	IT	1998-12-02	ubah hapus
019	Abdul	Business Strategy	1987-07-14	ubah hapus
020	Sarukh	Business Strategy	2002-12-11	ubah hapus

Gambar 2. 8 Halaman utama - layout menampilkan tabel data Karyawan.

Tambah Data Karyawan

Nama

Divisi

Pilih Divisi:

Tanggal Lahir

[Kembali ke Tabel](#)

Gambar 2. 9 Halaman utama - Layout tambah data karyawan.

```

C:\WINDOWS\system32\cmd.exe - mysql -u root
MariaDB [(none)]> use modul3pemweb
Database changed
MariaDB [modul3pemweb]> SELECT k.id_karyawan AS idka, k.nama AS nam,
-> d.nama_divisi AS divi, k.Tgl_lahir as tglh
-> FROM karyawan as k, divisi as d
-> WHERE k.divisi=d.id_divisi ORDER BY k.id_karyawan
-> ;

```

idka	nam	divi	tglh
001	Rick	IT	1998-12-10
002	Roll	IT	1998-02-22
003	Ricitas	Public Relation	1998-09-30
004	Richman	Finance	1992-10-23
005	Amin	Finance	1999-11-10
006	Elon	Business Strategy	1997-12-12
007	Musk	Business Strategy	1997-12-12
008	Ricardo	Product Development	1992-12-25
009	Milos	Product Development	1992-12-25
010	Herp	IT	2001-01-01
018	Samir	IT	1998-12-02
019	Abdul	Business Strategy	1987-07-14
020	Sarukh	Business Strategy	2002-12-11

```

13 rows in set (0.127 sec)

MariaDB [modul3pemweb]>

```

Gambar 2. 10 Tabel INNER JOIN pada jendela CMD.

Tambah Data Karyawan

Nama

Divisi

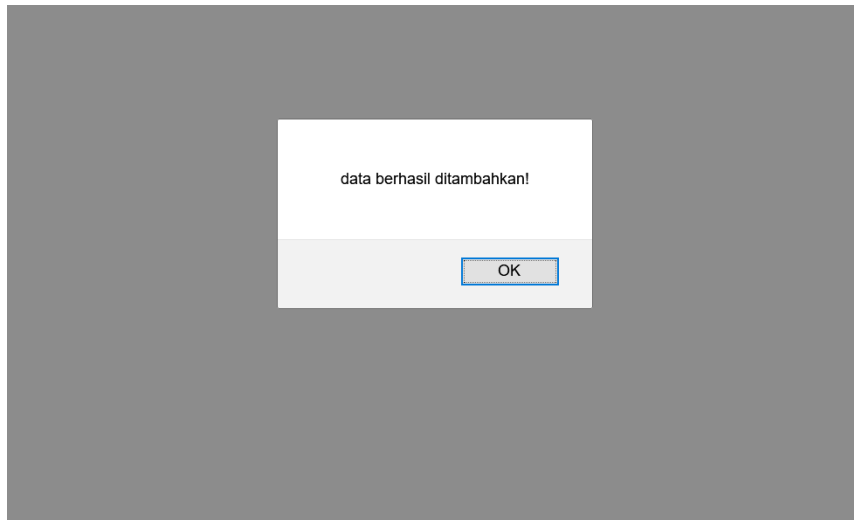
Finance

Tanggal Lahir

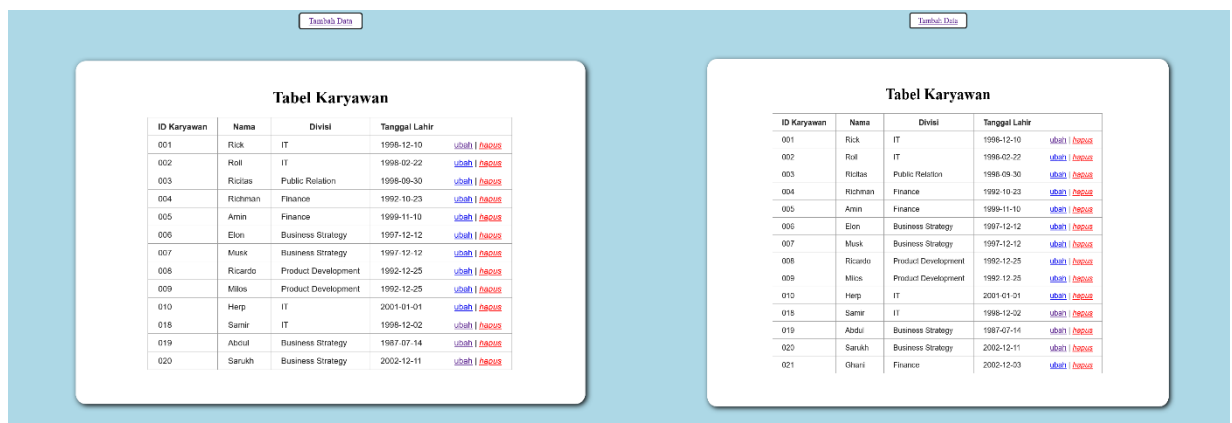
12 / 03 / 2002

[Kembali ke Tabel](#)

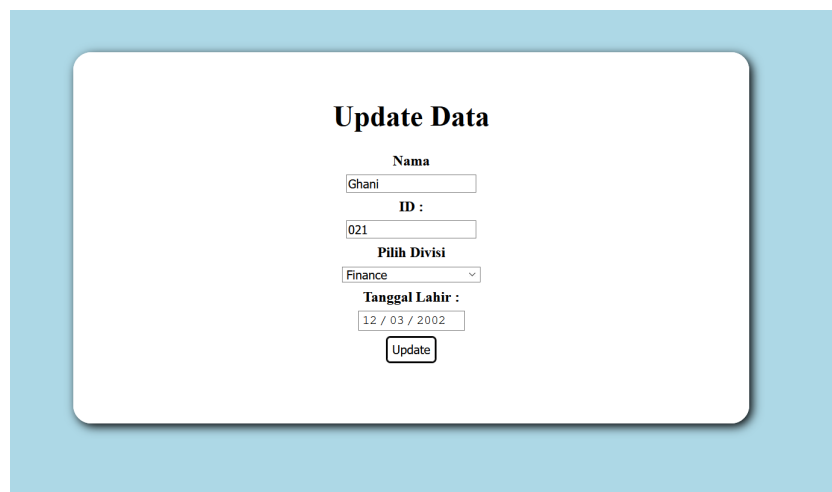
Gambar 2. 11 Halaman utama - tampilan saat pengisian data.



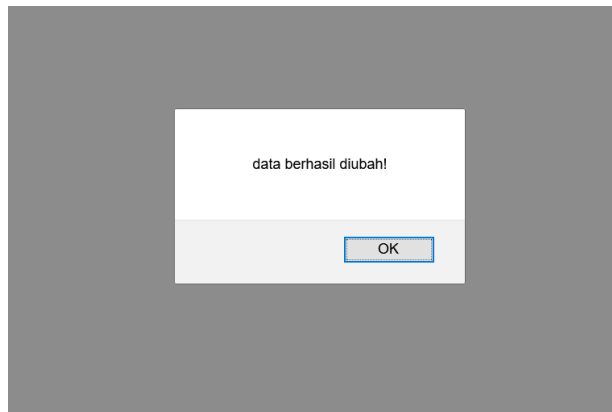
Gambar 2. 12 Halaman utama - tampilan saat data berhasil ditambahkan.



Gambar 2. 13 Halaman utama - Tampilan tabel sebelum (kiri) dan sesudah data dimasukan (kanan).



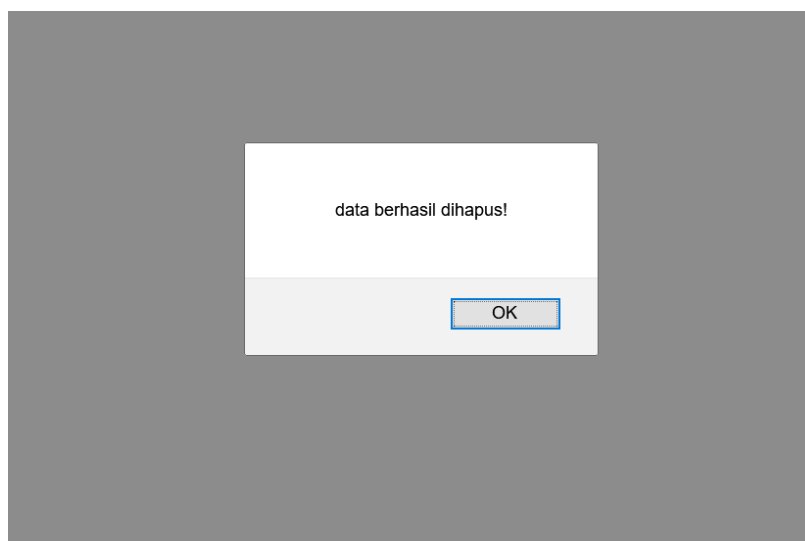
Gambar 2. 14 Halaman update data.



Gambar 2. 15 Halaman update data - ketika data berhasil diubah.

ID Karyawan	Nama	Divisi	Tanggal Lahir	
001	Rick	IT	1998-12-10	ubah hapus
002	Roll	IT	1998-02-22	ubah hapus
003	Ricita	Public Relation	1998-09-30	ubah hapus
004	Richman		2010-10-23	ubah hapus
005	Amin		2011-11-10	ubah hapus
006	Elon		2012-12-12	ubah hapus
007	Musk		2012-12-12	ubah hapus
008	Ricardo	Product Development	1992-12-25	ubah hapus
009	Milos	Product Development	1992-12-25	ubah hapus
010	Herp	IT	2001-01-01	ubah hapus
018	Samir	IT	1998-12-02	ubah hapus
019	Abdul	Business Strategy	1987-07-14	ubah hapus
020	Sarukh	Business Strategy	2002-12-11	ubah hapus
021	Ghani	Public Relation	2002-12-03	ubah hapus

Gambar 2. 16 Halaman utama - tampilan konfirmasi ketika data ingin dihapus.



Gambar 2. 17 Halaman hapus data - tampilan ketika data berhasil dihapus.

Tampilkan Data

Tabel Karyawan

ID Karyawan	Nama	Divisi	Tanggal Lahir
001	Rick	IT	1990-12-12 ubah hapus
002	Rita	IT	1998-02-29 ubah hapus
003	Ricma	Public Relation	1998-05-30 ubah hapus
004	Richard	Finance	1992-10-23 ubah hapus
005	Arlis	Finance	1999-11-10 ubah hapus
006	Clara	Business Strategy	1997-12-12 ubah hapus
007	Muak	Business Strategy	1997-12-12 ubah hapus
008	Kicardo	Product Development	1992-12-25 ubah hapus
009	Milos	Product Development	1992-12-25 ubah hapus
010	Harp	IT	2001-01-01 ubah hapus
018	Serli	IT	1999-12-02 ubah hapus
019	Abdul	Business Strategy	1987-07-14 ubah hapus
020	Surah	Business Strategy	2002-12-11 ubah hapus
021	Gibara	Finance	2002-12-03 ubah hapus

Tampilkan Data

Tabel Karyawan

ID Karyawan	Nama	Divisi	Tanggal Lahir
201	Rek	IT	1990-12-10 ubah hapus
202	Roi	IT	1998-02-22 ubah hapus
203	Riclas	Public Relation	1998-09-30 ubah hapus
204	Richman	Finance	1992-10-23 ubah hapus
205	Amin	Finance	1999-11-10 ubah hapus
206	Eren	Business Strategy	1997-12-12 ubah hapus
207	Muak	Business Strategy	1997-12-12 ubah hapus
208	Kiwanda	Product Development	1992-12-25 ubah hapus
209	Milos	Product Development	1992-12-25 ubah hapus
210	Harp	IT	2001-01-01 ubah hapus
218	Serir	IT	1999-12-02 ubah hapus
219	Abrial	Business Strategy	1987-07-14 ubah hapus
220	Surah	Business Strategy	2002-12-11 ubah hapus

Gambar 2. 18 Halaman utama - sebelum (kiri) dan sesudah data dihapus (kanan).