

**LAPORAN PRAKTIKUM  
PEMROGRAMAN WEB & APLIKASI I**



**NAMA : Muhammad Ilyas Abdullah**  
**NIM : 185110500111013**  
**KELAS : C (Permata Merdeka)**  
**MODUL : 3**

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

## BAB 1

### TUJUAN DAN LANDASAN TEORI

#### 1. Tujuan Praktikum

- 1.1. Mahasiswa mampu membuat handling yang mampu mengolah data dari form HTML.
- 1.2 Mahasiswa mampu membuat batasan-batasan untuk menangani inputan dari form HTML.

#### 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”:

```
<?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);
?>

```

**Gambar 1.3** 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 (

```

```

        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 1.4** Membuat Tabel

### Memasukkan Data ke Database

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

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

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

```

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

```

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

```

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

```

```

$dbname = "myDB";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Memeriksa Koneksi
if (!$conn){
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

if (mysqli_query($conn, $sql)) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}

mysqli_close($conn);
?>

```

**Gambar 1.5** Memasukan 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){

```

```

        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 1.6** Memasukan Data

### 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);
?>

```

**Gambar 1.7** Mengambil Data

### 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. 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);
?>

```

**Gambar 1.8 Menghapus**

### 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);
?>
```

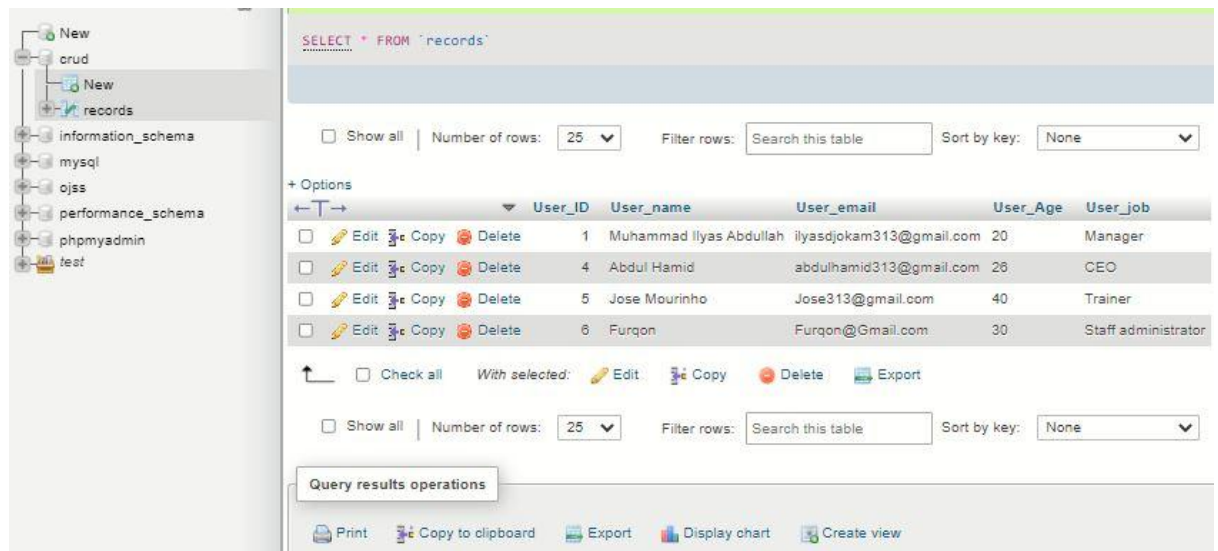
**Gambar 1.9** Update Data

## BAB 2

### PEMBAHASAN

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.

Pembuatan tugas pertama adalah untuk membuat database MySQL untuk menyimpan data pegawai sebuah perusahaan. Untuk membuat database, kita akan membuka <http://localhost/phpmyadmin/>. Setelah itu, kita dapat membuat sebuah database dengan nama yang kita inginkan. Disini penulis menggunakan nama crud dengan structure bernama records.



Gambar 2.1. Database untuk menyimpan data pegawai

Dalam data pegawai yang ada di atas, kita membuat beberapa kolom antara lain User\_ID, User\_name, User\_email, User\_Age dan User\_job. Fungsi dari User\_ID sendiri adalah sebagai kodek unik untuk kemudian dapat digunakan untuk dipanggil nantinya ketika kita akan menghapus sebuah data.

Selanjutnya untuk penugasan kedua, disini penulis diminta untuk membuat sebuah program web yang dapat digunakan untuk menyimpan, menghapus serta mengubah data yang ada di dalam database. Untuk pembuatannya sendiri, penulis akan membuat 8 file antara lain:

1. Connection.php
2. Delete.php
3. Edit.php
4. Index.php
5. Insert.php
6. Update.php
7. View.php
8. bootstrap.css

### **Connection.php**

Fungsi dari connection.php adalah untuk memanggil data yang kita miliki dalam MySQL sehingga dapat kita manipulasi di dalam Index.php serta view.php. Untuk fungsi yang digunakan yakni:

```
<?php

$con = mysqli_connect("localhost", "root", "", "crud");

if (!$con) {
    die("connection error".mysqli.error);
}

?>
```

### **Delete.php**

Delete.php disini adalah program yang digunakan agar kemudian user dapat menghapus data pegawai yang ada di dalam database yang kita miliki. Dalam hal proses penghapusan, kita menggunakan fungsi:

```
if(isset($_GET['Del']))
{
    $UserID = $_GET['Del'];
    $query = "delete from records where User_ID = ".$UserID."";
    $result = mysqli_query($con,$query);
```

## Edit.php

Edit.php disini adalah program yang dapat membuat membuat user untuk melakukan edit yang ada pada database yang kita miliki. Adapun fungsi yang digunakan yaitu:

```
<?php
```

```
require_once("connection.php");
$UserID = $_GET['GetID'];
$query = " select * from records where User_ID='".$UserID."'";
$result = mysqli_query($con,$query);

while($row=mysqli_fetch_assoc($result))
{
    $UserID = $row['User_ID'];
    $UserName = $row['User_name'];
    $UserEmail = $row['User_email'];
    $UserAge = $row['User_Age'];
    $UserJob = $row['User_job'];
}
```

```
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <link rel="stylesheet" a href="bootstrap.css"/>
    <title>Modul 3 Praktikum</title>
</head>
<body class="bg-dark">

    <div class="container">
        <div class="row">
            <div class="col-lg-6 m-auto">
                <div class="card mt-5">
                    <div class="card-title">
```

```

<h3 class="bg-success text-white
text-center py-3"> Update Database</h3>
</div>
<div class="card-body">

    <form action="update.php?ID=<?php echo
$UserID ?>" method="post">

        <input type="text"
class="form-control mb-2" placeholder=" User Name " name="name"
value="<?php echo $UserName ?>">

        <input type="email"
class="form-control mb-2" placeholder=" User Email " name="email"
value="<?php echo $UserEmail ?>">

        <input type="text"
class="form-control mb-2" placeholder=" User Age " name="age"
value="<?php echo $UserAge ?>">

        <input type="text"
class="form-control mb-2" placeholder=" User occupation "
name="job" value="<?php echo $UserJob ?>">

        <button class="btn btn-primary"
name="update">Update</button>
    </form>

</div>
</div>
</div>
</div>
</div>

</body>
</html>

```

```

<?php
require_once("connection.php");
$UserID = $_GET['GetID'];
$query = " select * from records where User_ID='".$UserID."'";
$result = mysqli_query($con,$query);

while($row=mysqli_fetch_assoc($result))
{
    $UserID = $row['User_ID'];
    $UserName = $row['User_name'];
    $UserEmail = $row['User_email'];
    $UserAge = $row['User_Age'];
    $UserJob = $row['User_job'];
}

?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<link rel="stylesheet" href="CSS/bootstrap.css"/>
<title>Modul 3 Praktikum</title>
</head>
<body class="bg-dark">
    <div class="container">
        <div class="row">
            <div class="col-lg-6 m-auto">
                <div class="card mt-5">
                    <div class="card-title">
                        <h3 class="bg-success text-white text-center py-3"> Update Database</h3>
                    </div>
                    <div class="card-body">
                        <form action="update.php?ID=<?php echo $UserID ?>" method="post">
                            <input type="text" class="form-control mb-2" placeholder=" User Name " name="name" value="<?php echo $UserName ?>">
                            <input type="email" class="form-control mb-2" placeholder=" User Email " name="email" value="<?php echo $UserEmail ?>">
                            <input type="text" class="form-control mb-2" placeholder=" User Age " name="age" value="<?php echo $UserAge ?>">
                            <input type="text" class="form-control mb-2" placeholder=" User occupation " name="job" value="<?php echo $UserJob ?>">
                            <button class="btn btn-primary" name="update">Update</button>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

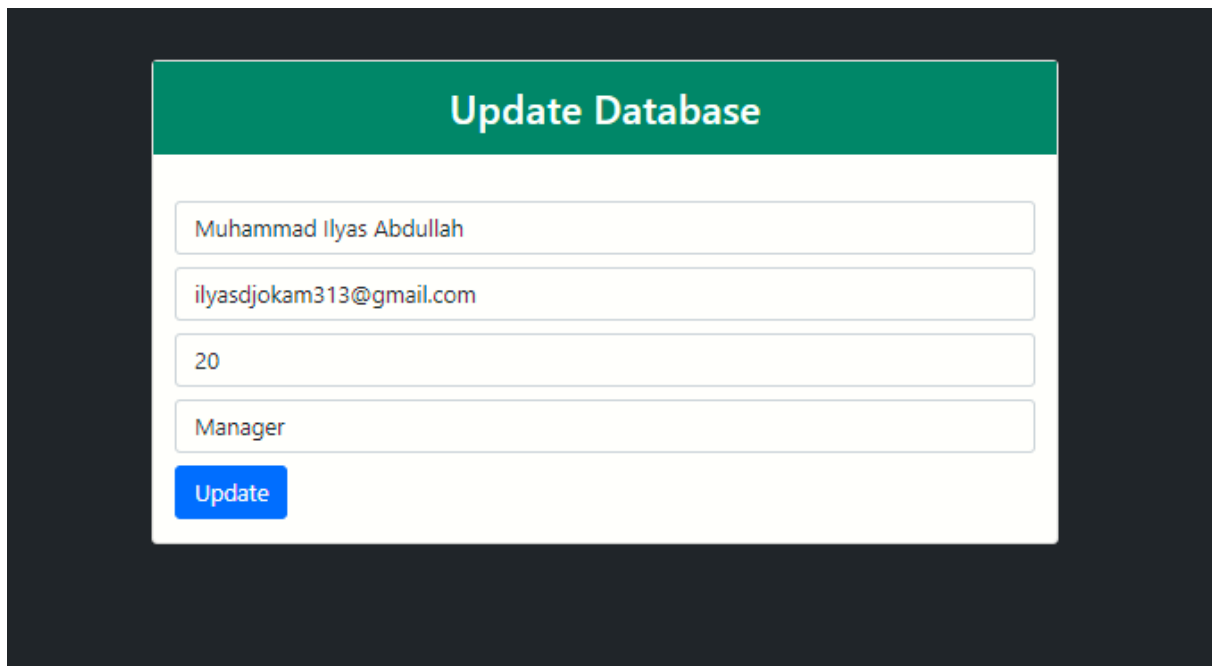
Gambar 2.2. Program yang ada pada *edit.php* yang mana digunakan untuk mengubah data

Hal-hal yang dapat diedit dalam database meliputi nama, email, umur, serta posisi di dalam perusahaan. Hal tersebut dapat dilihat melalui gambar di bawah ini:

User ID	User Name	User Email	User Age	User Job	Edit	Delete
1	Muhammad Ilyas Abdullah	ilyasdjokam313@gmail.com	20	Manager	<a href="#">Edit</a>	<a href="#">Delete</a>
4	Abdul Hamid	abdulhamid313@gmail.com	26	CEO	<a href="#">Edit</a>	<a href="#">Delete</a>
5	Jose Mourinho	Jose313@gmail.com	40	Trainer	<a href="#">Edit</a>	<a href="#">Delete</a>
6	Furqon	Furqon@Gmail.com	30	Staff administrator	<a href="#">Edit</a>	<a href="#">Delete</a>

Gambar 2.3. Database yang ada pada *view.php*

Sebagai contoh, disini kita akan mengganti *user job* dari Muhammad Ilyas Abdullah yang awalnya adalah Manager menjadi General Manager. Maka kita akan menekan tombol edit sehingga kemudian akan diarahkan sebuah halaman yang ada pada *edit.php*. Tampilan dari halaman tersebut dapat kita lihat melalui gambar di bawah ini:



Update Database

Muhammad Ilyas Abdullah

ilyasdjokam313@gmail.com

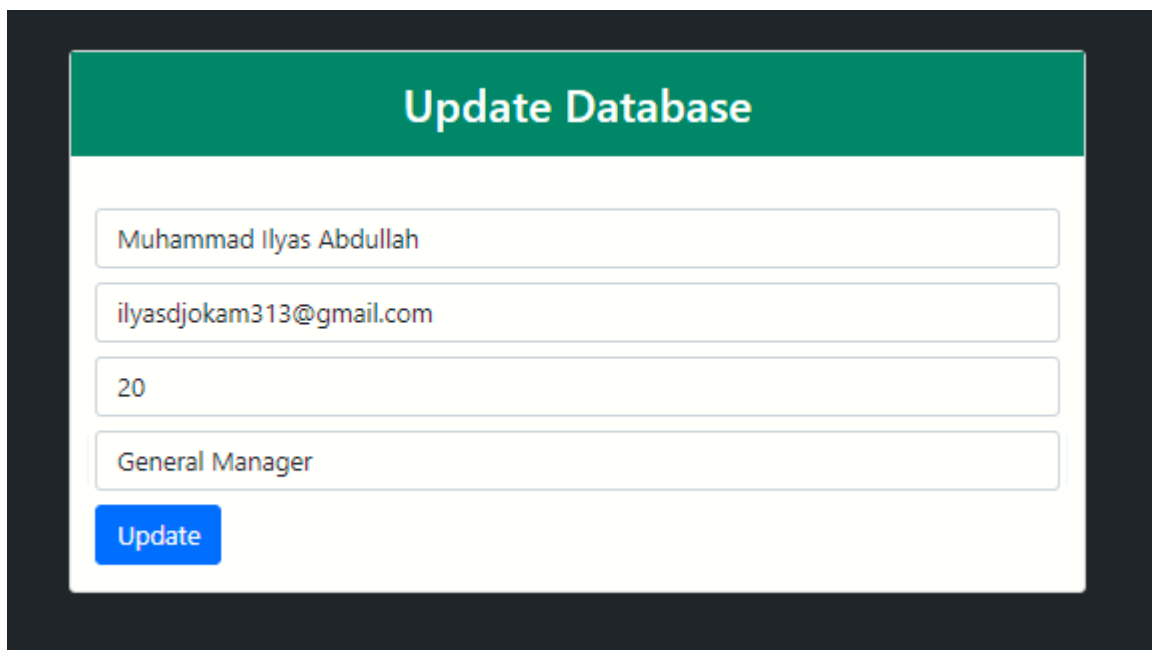
20

Manager

Update

Gambar 2.4. *Sebelum diedit*

Disini kita hanya perlu menghapus kolom yang ada pada baris keempat dan mengubahnya menjadi General Manager.



Update Database

Muhammad Ilyas Abdullah

ilyasdjokam313@gmail.com

20

General Manager

Update

Gambar 2.5. *Sesudah di edit*

Setelah itu kita dapat memencet tombol *update*. Kemudian data yang ada di dalam database akan otomatis berubah.



User ID	User Name	User Email	User Age	User Job	Edit	Delete
1	Muhammad Ilyas Abdullah	ilyasdjokam313@gmail.com	20	General Manager	<a href="#">Edit</a>	<a href="#">Delete</a>

Gambar 2.6. Database setelah di edit

### Index.php & Bootstrap.css

bootstrap.css berfungsi sebagai front-end dari program mengubah, menghapus dan menyimpan. Penggunaan bootstrap disini untuk membuat tampilan modern dan dinamis. Index.php disini memiliki fungsi sebagai halaman utama ketika user akan memasukan data baru ke dalam database pegawai. Tampilan dari Index.php dapat dilihat melalui gambar di bawah ini:

Gambar 2.7. Tampilan awal ketika user ingin menambahkan data baru ke dalam database

### Insert.php

Insert.php disini berfungsi sebagai program yang digunakan untuk memasukan data pegawai baru ke dalam database. Insert.php menggunakan fungsi if else sebagai fungsi yang digunakan di dalam program. Berikut ini program yang digunakan di dalamnya:

```
<?php
    require_once("connection.php");

    if (isset($_POST['submit'])) {
        if (empty($_POST['name']) ||
empty($_POST['email']) || empty($_POST['age']) ||
empty($_POST['job']) )
        {
            echo 'please fill data in the blank';
        }

        else
        {
            $UserName = $_POST['name'];
            $UserEmail = $_POST['email'];
            $UserAge = $_POST['age'];
            $UserJob = $_POST['job'];

            $query = "insert into records (User_name,
User_Email, User_Age, User_job) value('$UserName',
'$UserEmail', '$UserAge', '$UserJob')";

            $result = mysqli_query($con, $query);

            if ($result) {
                header("location: view.php");
            }

            else {
                echo 'please check your query';
            }
        }
    }

    else {
        header("location: index.php");
    }
?>
```

## Update.php

Update.php disini adalah program yang digunakan agar dapat melakukan update pada database yang terhubung melalui edit.php. Di dalam update.php, penulis menggunakan fungsi if else untuk menjalankan program yang ada di dalamnya. Berikut ini adalah *source code* yang ada di dalam update.php :

```
<?php

require_once("connection.php");

if(isset($_POST['update']))
{
    $UserID = $_GET['ID'];
    $UserName = $_POST['name'];
    $UserEmail = $_POST['email'];
    $UserAge = $_POST['age'];
    $UserJob = $_POST['job'];

    $query = "update records set User_name =
'".$UserName."', User_email='".$UserEmail."',
User_Age='".$UserAge."', User_job='".$UserJob.'" where
User_ID='".$UserID.'"";
    $result = mysqli_query($con,$query);

    if($result)
    {
        header("location:view.php");
    }
    else
    {
        echo 'please check your query';
    }

}
else {
    header("location:view.php");
}

?>
```

## View.php

View.php disini memiliki fungsi sebagai tampilan dari database yang ada di dalam MySql. Dari view.php kita dapat melihat seluruh data pegawai yang ada di dalam database. Berikut adalah kode nya:

```

                                while
($row=mysqli_fetch_assoc($result))
                                {
                                    $UserID =
                                    $UserName =
                                    $UserEmail =
                                    $UserAge =
                                    $UserJob =

                                ?>
                                <tr>
                                    <td><?php echo
$UserID ?></td>
                                    <td><?php echo
$UserName ?></td>
                                    <td><?php echo
$UserEmail ?></td>
                                    <td><?php echo
$UserAge ?></td>
                                    <td><?php echo
$UserJob ?></td>
                                    <td><a
href="edit.php?GetID=<?php echo $UserID ?>">Edit</td>
                                    <td><a
href="delete.php?Del=<?php echo $UserID ?>">Delete</a></td>
                                </tr>

                                <?php
                                }

                                ?>
                            </table>

                        </div>

                    </div>

```

```

        </div>
</div>
</body>
</html>

```

Dalam kode yang digunakan di atas, penulis menggunakan `mysqli_fetch_assoc()`. Fungsi `mysqli_fetch_assoc()` untuk memanggil data yang ada pada database yang diletakan pada `mysql`. `mysqli_fetch_assoc()` akan mengembalikan nilai berupa associative array. Disini key array bukan lagi angka, tapi nama kolom tabel. Selain mengembalikan nilai, fungsi `mysqli_fetch_assoc()` secara otomatis menaikkan posisi pointer dari variabel `$query` pada tiap pemanggilan.

### Menambahkan, mengubah dan menghapus data pada program

Seperti yang diminta dalam penugasan, program yang dibuat mencakup 8 file, 8 file tersebut kemudian digunakan sehingga dapat menghapus dan menambahkan data pada program.

Untuk menambahkan data ke dalam database, kita dapat membuka halaman utama dari program yang berada pada `index.php`. Seperti yang ada pada gambar 2.7. Lalu menambahkan data pegawai dengan memasukan ke dalam kolom yang sudah disediakan antara lain nama, email, umur serta jabatan di dalam perusahaan. Kemudian jika kita ingin melihat isi dari database yang berada di `mysql`. Cukup klik hyperlink database admin yang mana nantinya akan mengarahkan user menuju `view.php`.

User ID	User Name	User Email	User Age	User Job	Edit	Delete
1	Muhammad Ilyas Abdullah	ilyasdjokam313@gmail.com	20	General Manager	<a href="#">Edit</a>	<a href="#">Delete</a>
4	Abdul Hamid	abdulhamid313@gmail.com	26	CEO	<a href="#">Edit</a>	<a href="#">Delete</a>
5	Jose Mourinho	Jose313@gmail.com	40	Trainer	<a href="#">Edit</a>	<a href="#">Delete</a>
6	Furqon	Furqon@Gmail.com	30	Staff administrator	<a href="#">Edit</a>	<a href="#">Delete</a>

*Gambar 2.8. Database mysql yang dipanggil ke dalam view.php*

jika kita ingin mengubah data yang ada di dalam database, kita cukup mengklik tombol edit yang mana nantinya akan membuka halaman seperti yang ada pada gambar 2.4 dan 2.5. Jika dirasa data yang dimasukan sudah

benar, maka kita dapat langsung mengklik tombol update, maka data akan secara otomatis sudah berubah, kemudian jika kita ingin menghapus data, maka kita cukup mengklik tombol delete, maka data akan secara otomatis terhapus.

### **BAB III**

### **KESIMPULAN**

Salah satu dari keunggulan PHP adalah dapat digunakan dalam program web untuk membuat database sebuah perusahaan atau institusi. Dengan memanfaatkan mysql sebagai database berbasis *cloud native application* ditambah dengan penggunaan php sebagai bahasa *back-end* proses Pembuatan database menjadi lebih mudah dan ringkas.

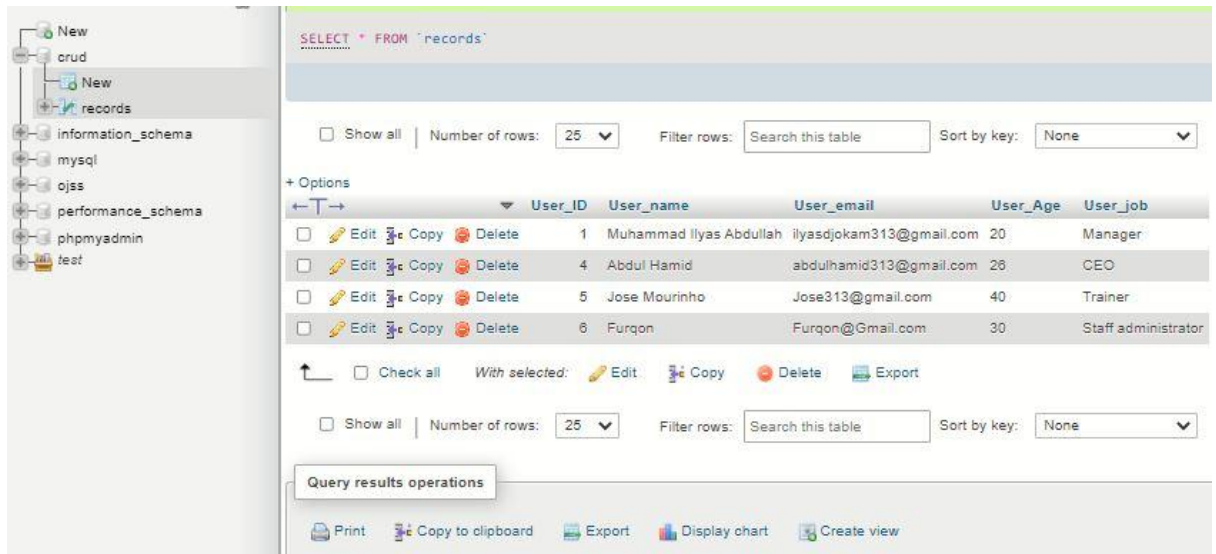
Dengan adanya tugas yang ada pada modul 3 ini. Kita dapat membuat sebuah program web database sederhana yang berfungsi sebagai fundamental dasar sebelum melanjutkan ke dalam pembelajaran yang lebih dalam.



## DAFTAR PUSTAKA

Praktikum, K. (n.d.). *MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya*.

## LAMPIRAN



```
<?php
require_once("connection.php");
$UserID = $_GET['GetID'];
$query = " select * from records where User_ID='".$UserID."'";
$result = mysqli_query($con,$query);

while($row=mysqli_fetch_assoc($result))
{
    $UserID = $row['User_ID'];
    $UserName = $row['User_name'];
    $UserEmail = $row['User_email'];
    $UserAge = $row['User_Age'];
    $UserJob = $row['User_job'];
}
}

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<link rel="stylesheet" a href="CSS/bootstrap.css"/>
<title>Modul 3 Praktikum</title>
</head>
<body class="bg-dark">

    <div class="container">
        <div class="row">
            <div class="col-lg-6 m-auto">
                <div class="card mt-5">
                    <div class="card-title">
                        <h3 class="bg-success text-white text-center py-3"> Update Database</h3>
                    </div>
                    <div class="card-body">

                        <form action="update.php?ID=<?php echo $UserID ?>" method="post">
                            <input type="text" class="form-control mb-2" placeholder=" User Name " name="name" value="<?php echo $UserName ?>">
                            <input type="email" class="form-control mb-2" placeholder=" User Email " name="email" value="<?php echo $UserEmail ?>">
                            <input type="text" class="form-control mb-2" placeholder=" User Age " name="age" value="<?php echo $UserAge ?>">
                            <input type="text" class="form-control mb-2" placeholder=" User occupation " name="job" value="<?php echo $UserJob ?>">
                            <button class="btn btn-primary" name="update">Update</button>
                        </form>

                    </div>
                </div>
            </div>
        </div>
    </div>

```

User ID	User Name	User Email	User Age	User Job	Edit	Delete
1	Muhammad Ilyas Abdullah	ilyasdjokam313@gmail.com	20	Manager	<a href="#">Edit</a>	<a href="#">Delete</a>
4	Abdul Hamid	abdulhamid313@gmail.com	26	CEO	<a href="#">Edit</a>	<a href="#">Delete</a>
5	Jose Mourinho	Jose313@gmail.com	40	Trainer	<a href="#">Edit</a>	<a href="#">Delete</a>
6	Furqon	Furqon@Gmail.com	30	Staff administrator	<a href="#">Edit</a>	<a href="#">Delete</a>

Update Database

Muhammad Ilyas Abdullah

ilyasdjokam313@gmail.com

20

Manager

Update

Update Database

Muhammad Ilyas Abdullah

ilyasdjokam313@gmail.com

20

General Manager

Update

User ID	User Name	User Email	User Age	User Job	Edit	Delete
1	Muhammad Ilyas Abdullah	ilyasdjokam313@gmail.com	20	General Manager	<a href="#">Edit</a>	<a href="#">Delete</a>

## Employees Registration Database of PT.Mencari Cinta Sejati

 [Database Admin](#)

User ID	User Name	User Email	User Age	User Job	Edit	Delete
1	Muhammad Ilyas Abdullah	ilyasdjokam313@gmail.com	20	General Manager	<a href="#">Edit</a>	<a href="#">Delete</a>
4	Abdul Hamid	abdulhamid313@gmail.com	26	CEO	<a href="#">Edit</a>	<a href="#">Delete</a>
5	Jose Mourinho	Jose313@gmail.com	40	Trainer	<a href="#">Edit</a>	<a href="#">Delete</a>
6	Furqon	Furqon@Gmail.com	30	Staff administrator	<a href="#">Edit</a>	<a href="#">Delete</a>