

**LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEB & MOBILE**



NAMA : Bryant Aprilian Bahan
NIM : DBC 118 100
KELAS : C
MODUL : III (PHP MySQL Database)

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021

BAB I

TUJUAN DAN LANDASAN TEORI

1.1. Tujuan

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

1.2. Landasan Teori

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

1.2.1 Membuka Koneksi

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

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

// Membuat hubungan
$conn = new mysqli($servername, $username, $password);

// Memeriksa hubungan
if ($conn->connect_error){
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

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);
```

1.2.2 Membuat Database

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

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

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password);
// Memeriksa Koneksi if (!$conn){

    die("Connection failed: " . mysqli_connect_error());
}
// Membuat Database

$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {

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

1.2.3 Membuat Tabel

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

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

Berikut adalah contoh program untuk membuat tabel diatas:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

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

// SQL untuk membuat tabel
$sql = "CREATE TABLE MyGuests (
  id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  firstname VARCHAR(30) NOT NULL, lastname VARCHAR(30) NOT
  NULL,
  email VARCHAR(50),
  reg_date TIMESTAMP
)";
if (mysqli_query($conn, $sql)) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

1.2.4 Memasukkan Data ke Database

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

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

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

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

adapun contoh kode program PHP untuk menambahkan data

```
<?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);
?>
```

tersebut adalah sebagai berikut:

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

Untuk menambahkan record dengan jumlah yang banyak, kita bisa menggunakan fungsi `mysqli_multi_query($conn, $sql)`, berikut adalah contoh program untuk menambahkan record dengan jumlah yang banyak:

```

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

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

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

```

1.2.5 Mengambil Data dari Database

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

```

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

// Membuat koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){

die("Connection failed: " . mysqli_connect_error());
}

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

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

```

1.2.6 Menghapus Data dari Database

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

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

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

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

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

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

1.2.7 Update Data dalam Database

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

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

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

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
```



```
die("Connection failed: " . mysqli_connect_error());
}
$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";
if (mysqli_query($conn, $sql)) {
    echo "Record updated successfully";
} else {
    echo "Error updating record: " . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

BAB II PEMBAHASAN

Berikut ini adalah penjelasan mengenai source code program web untuk menginputkan username dan password menggunakan form dan penanganan input data dengan kriteria sebagai berikut.

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.

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

Blok code di atas merupakan source code dari file PHP pertama yang diberi nama connect berikut ini merupakan penjelasan tentang blok code di atas. Fungsi `mysql_query` dalam PHP adalah untuk menjalankan perintah atau instruksi query ke database MySQL dimana `mysql_query()` merupakan sarana dasar yang akan menampung jenis query apapun di dalamnya.

Function `tambahKaryawan` berfungsi untuk memasukan data karyawan, lalu

Htmlspecialchars adalah sebuah fungsi atau sebuah perintah atau syntax yang dimiliki oleh PHP yang berguna untuk menontaktifkan seluruh perintah – perintah html mengubah special pada sebuah halaman web. Yang artinya mempertahankan penulisan HTML pada sebuah tampilan HTML.

Function hapuskaryawan berfungsi untuk menghapus data karyawan yang ingin kita hapus berdasarkan id_karyawan.

Modul3.php

```
<?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>";
    }
}
?>

<!DOCTYPE html>
<html>
<head>
    <title> Modul 3</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <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>
        </form>
    </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>

<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 $i=1; ?>
            <?php foreach ($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 class="warn" href="modul3Hapus.php?id=<?php echo $row["idka"]; ?>" onclick="r
return confirm('Hapus data?');">hapus</a>
                    </td>
                </tr>
            <?php $i++; ?>
            <?php endforeach; ?>
        </table>
    </div>
</div>
</body>
</html>

```

Program ke dua y aitu program modul3.php, program ini berfungsi sebagai program utama dari web ini untuk menampilkan desain dari web yang dibuat. require() merupakan fungsi yang digunakan untuk menyertakan file php lain ke dalam suatu program PHP. Hal sangat membantu proses pemrograman karena

tidak perlu menulis program PHP secara berulang-ulang, cukup dalam satu file saja.

DOCTYPE adalah suatu deklarasi yang digunakan untuk mengidentifikasi jenis dokumen HTML yang digunakan sehingga browser dapat menentukan bagaimana memperlakukan kode tersebut.

Class=TambahKaryawan berfungsi untuk user menambah data karyawan yang akan di panggil ke connect.php

Class=tabelkaryawan bertujuan untuk membuat sebuah tabel agar user lebih mudah untuk melihat data yang sudah diinput dan akan di panggil ke connect.php

```
<?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>";
}
?>
```

Program ini adalah program if dan else yang bertujuan untuk mengkonfirmasi apakah data yang kita hapu berhasil dihapus atau tidak. Jika data terhapus web akan menampilkan tulisan “data berhasil dihapus” dan jika tidak terhapus web akan menampilkan tulisan “data gagal dihapus”

BAB III

KESIMPULAN

Kesimpulan yang dapat diambil dari modul 3 ini adalah suatu aplikasi banyak menerima masukan/input data dari pengguna, misalkan ketika registrasi kesuatu website atau ketika login pada halaman dan sebagainya. Untuk memungkinkan hal tersebut diperlukan suatu antarmuka pada klien dalam bentuk form HTML dan suatu mekanisme untuk penanganan data yang dikirim oleh pengguna pada sisi server.

mysql dapat di gabungkan dengan function php melalui syntax `Mysqli_connect()`, kemudian dengan membuat database ataupun memakai database yang telah ada, maka function pada php dapat dimuat kedalam database tersebut

DAFTAR PUSTAKA

Tanpa Tahun. *MODUL PRAKTIKUM PEMROGRAMAN WEB I*. Palangka Raya:

Jurusan Teknik Informatika, Universitas Palangka Raya.

Belajarbootstrapframework. (2013). *fungsi index*. 2013.

<http://belajarbootstrapframework.blogspot.com/2015/12/fungsi-index-dalamwebsite.html>