

**LAPORAN PRAKTIKUM
PEMROGRAMAN WEB DAN MOBILE I**



NAMA : ABIGAEL GERRY PRATAMA
NIM : 193020503021
KELAS : A
MODUL : III (PHP MySQL Database)

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021**

BAB I

TUJUAN DAN 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

MySQL adalah sistem manajemen basis data relasional open-source (RDBMS). Ini adalah sistem database paling populer yang digunakan dengan PHP. MySQL dikembangkan, didistribusikan, dan didukung oleh Oracle Corporation.

Data dalam database MySQL disimpan dalam tabel yang terdiri dari kolom dan baris.

- 1. MySQL adalah sistem database yang berjalan di server.
- 2. MySQL ideal untuk aplikasi kecil dan besar.
- 3. MySQL adalah sistem database yang sangat cepat, andal, dan mudah digunakan, menggunakan SQL standar
- 4. MySQL dikompilasi pada sejumlah platform.

Tidak diragukan lagi PHP adalah platform paling menonjol untuk pengembangan web saat ini. Selain situs web, aplikasi perusahaan seperti ERP, CRM open source berbasis web untuk UKM, dan solusi e-Commerce sangat menonjol saat ini. Ada banyak orang yang percaya bahwa mySQL adalah satu-satunya database yang terhubung dengan PHP sementara yang lain mengetahui fakta bahwa PHP juga mendukung database lain tetapi mereka tidak menyadarinya. Apa yang sering terjadi adalah bahwa pelanggan pilih-pilih dalam hal database yang ingin mereka miliki untuk implementasi.

Jadi di sini kami membantu pengembang PHP, perusahaan pengembangan web, dan lainnya membuat mereka memahami database yang didukung PHP.

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.

Membuat Database pada MySQL

Database pada MySQL bisa juga dibuat menggunakan kode program PHP. Program tersebut akan berisi statement SQL “CREATE DATABASE”.

Membuat Tabel pada PHP

Pembuatan tabel pada bahasa pemrograman PHP juga menggunakan statement SQL, yaitu statement “CREATE TABLE”.

Memasukkan Data ke Database

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

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

Statement INSERT INTO digunakan untuk menambahkan baris pada tabel

MySQL, query untuk menambahkan data tersebut adalah sebagai berikut:

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

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.

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.

Update Data dalam Database

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

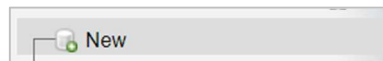
```
UPDATE nama_tabel
SET kolom1=nilai1, kolom2=nilai2, ...
WHERE kolom_penentu=nilai_penentu ...
```

BAB II

PEMBAHASAN

Database pada MySql untuk menyimpan data-data pegawai beserta relasi tabelnya.

Untuk membuat database, menggunakan phpmyadmin. Lalu dengan menu new untuk membuat database baru. Setelah itu buat nama untuk databasenya. Sebelumnya aktifkan xampp untuk mengaktifkan server lokalnya.



Gambar 2.1 Create New Database

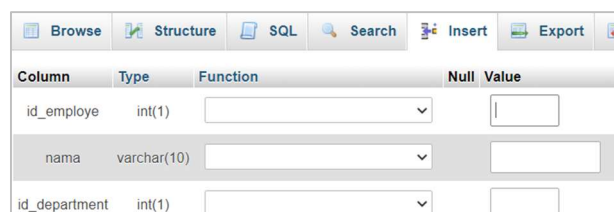
A screenshot of the 'Create database' form in phpMyAdmin. It features a text input field labeled 'Database name', a dropdown menu showing 'utf8mb4_general_ci', and a 'Create' button.

Gambar 2.2 Database Name

Pada tabel employe, terdapat tiga atribut yaitu id_employe int(1), nama varchar(10), dan id_department int(1). Lalu ke menu insert data untuk menambahkan data ke dalam database. Berikut hasil database yang dibuat.

#	Name	Type
<input type="checkbox"/> 1	id_employe	int(1)
<input type="checkbox"/> 2	nama	varchar(10)
<input type="checkbox"/> 3	id_department	int(1)

Gambar 2.3 Table Employe

A screenshot of the 'Insert' menu in phpMyAdmin. It shows a table with columns: Column, Type, Function, Null, and Value. The rows are: id_employe (int(1)), nama (varchar(10)), and id_department (int(1)). Each row has a dropdown menu for the Function and a text input field for the Value.

Gambar 2.4 Insert Data Employe

id_employe	nama	id_department
1	Stefy	2
2	Thea	1

Gambar 2.5 Value Table

Program untuk menyimpan, menghapus dan mengubah data-data pegawai tersebut.

Dibagian ini terdapat 4 file php yang yaitu untuk index, function, update, dan insert. Dan fungsinya untuk menyimpan, menghapus, dan mengubah database. Dan tujuan php ini untuk menghubungkan database. Pada file index.php untuk menampilkan file database yang diinputkan. Terdapat 3 menu yang ada dalam file ini, yaitu insert, untuk menambahkan data, lalu delete untuk menghapus data, dan update untuk memperbaiki atau mengubah data. Jika mengklik menu insert maka akan di hyperlink atau masuk ke file insert.php.

Terdapat 4 atribut dalam database ini yaitu ID employe, Nama, ID Department, dan Menu. Jika memilih delete maka akan tampil pop up dengan kata Data Dihapus, jika ada kesalahan maka akan terjadi Data Gagal Dihapus. Jika mengklik menu update maka akan dialihkan ke file update.php, jika data berhasil diubah maka juga akan tampil pop up. Nama web ini diberi nama Database Pegawai | M3, dan terdapat heading bernama Database Pegawai. Lalu untuk datanya menggunakan table agar terlihat lebih rapi.

```
<?php
require("function.php");
$data = show("SELECT * FROM employee");
if (isset($_GET["delete"])) {
    if (delete($_GET > 0)) {
        echo "<script>
            alert('Data Dihapus');
            document.location.href = 'index.php';
        </script>";
    } else {
        echo "<script>
            alert('Data Gagal Dihapus');
            document.location.href = 'index.php';
        </script>";
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Database Pegawai | M3</title>
```

```

</head>
<body>
  <h1>Database Pegawai</h1><br>
</form>
<table border="1">
<style>
  table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
    padding: 10px;
  }
</style>
  <tr>
    <td>ID Employee</td>
    <td>Nama</td>
    <td>ID Department</td>
    <td>Menu</td>
  </tr>
  <?php
  foreach ($data as $d) {
  ?>
    <tr>
      <td><?php echo $d["id_employe"]; ?></td>
      <td><?php echo $d["nama"]; ?></td>
      <td><?php echo $d["id_department"]; ?></td>
      <td>
        <form action="" method="GET">
          <button type="submit" name="delete" value=<?php echo $d["id_employe"]; ?>>DELETE</button>
        </form><br>
        <form action="update.php" method="GET">
          <button type="submit" name="update" value=<?php echo $d["id_employe"]; ?>>UPDATE</button>
        </form>
      </td>
    </tr>
  <?php
  }
  ?>
</table>
<form action="insert.php">
  <br><button>INSERT</button>
</body>
</html>

```

Gambar 2.6 Table Index

Berikut file function.php, didalamnya terdapat fungsi-fungsi untuk menampilkan data, menghapus, mengubah, dan menambahkan data. Terdapat parameter dalam program ini yaitu servername, database, username, dan password. Dan diisi sesuai dengan data yang dibuat. Mysqli_connect digunakan untuk menghubungkan ke database yang telah dibuat.

```

<?php
$servername = "localhost";
$dbname = "modul3";
$username = "root";
$password = "";

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

function show($query){

```

```

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

function insert($data){
    global $conn;
    $id_employe = $_POST["id_employe"];
    $nama = $_POST["nama"];
    $id_department = $_POST["id_department$id_department"];

    mysqli_query($conn, "INSERT INTO employee VALUES('$id_employe','$nama','$id_department')");
    return mysqli_affected_rows($conn);
}

function delete($data){
    global $conn;
    $id_employe = $_GET["delete"];
    mysqli_query($conn, "DELETE FROM employee WHERE id_employe = $id_employe");
    return mysqli_affected_rows($conn);
}

function update($data){
    global $conn;
    $id_employe = $_POST["id_employe"];
    $nama = $_POST["nama"];
    $id_department = $_POST["id_department$id_department"];
    mysqli_query($conn, "UPDATE employee SET nama = '$nama',
    id_department = '$id_department'
    WHERE id_employe = $id_employe");
    return mysqli_affected_rows($conn);
}

?>

```

Gambar 2.7 Function

Pada program insert.php, digunakan untuk menambahkan data untuk dimasukkan ke dalam database. Data yang ditambahkan berupa id employe, nama, dan id department. Jika data yang ditambahkan berhasil, maka akan tampil pop up data ditambahkan. Jika tidak, maka akan tampil pop up data gagal ditambahkan. Terdapat method post, untuk array dari variable yang dikirim dari skrip.

```

<?php
require("function.php");
if (isset($_POST["insert"])) {
    if (insert($_POST) > 0) {
        echo "<script>
            alert('Data Ditambahkan');
            document.location.href = 'index.php';
        </script>";
    } else {
        echo "<script>
            alert('Data Gagal Ditambahkan');
            document.location.href = 'index.php';
        </script>";
    }
}

```



```

}
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <link rel="stylesheet" href="style.css" type="text/css">
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Insert</title>
</head>
<body>
  <a href="index.php">HOME</a>
  <form action="" method="POST">
    <label for="id_employe"><br>ID Employe<br></label>
    <input type="text" name="id_employe" id="id_employe"><br>
    <label for="nama"><br>Nama<br></label>
    <input type="text" name="nama" id="nama"><br>
    <label for="id_department"><br>ID Department<br></label>
    <input type="text" name="id_department" id="id_department"><br><br>
    <button type="submit" name="insert">OK</button>
  </form>
</body>
</html>

```

Gambar 2.8 Insert Data

Program dibawah ini merupakan update.php, yang fungsinya untuk mengupdate atau mengubah file yang telah dibuat. Data yang diubah berupa id employe, nama, dan id department. Jika data yang ditambahkan berhasil, maka akan tampil pop up data diupdate. Jika tidak, maka akan tampil pop up data gagal diupdate. Terdapat method post, untuk array dari variable yang dikirim dari skrip.

```

<?php
require("function.php");
$id_employe = $_GET["update"];
$data = show("SELECT * FROM employee WHERE id_employe = $id_employe");
if (isset($_POST["ubah"])) {
  if (update($_GET > 0)) {
    echo "<script>
      alert('data diupdate');
      document.location.href = 'index.php';
    </script>";
  } else {
    echo "<script>
      alert('data gagal diupdate');
      document.location.href = 'index.php';
    </script>";
  }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Update</title>
</head>
<body>
  <a href="index.php">home</a>

```

```

<form action="" method="POST">
  <?php
  foreach ($data as $d) {
    ?>
    <ul>
      <li>
        <label for="id_employe">id_employe</label>
        <input type="text" name="id_employe" id="id_employe" value=<?php echo
$d["id_employe"]; ?> readonly>
      </li>
      <li>
        <label for="nama">nama</label>
        <input type="text" name="nama" id="nama" value=<?php echo $d["nama"];
?>>
      </li>
      <li>
        <label for="id_department">id_department</label>
        <input type="text" name="id_department" id="id_department" value=<?php
echo $d["id_department"]; ?>>
      </li>
      <li>
        <button type="submit" name="ubah">update</button>
      </li>
    </ul>
    <?php
  }
  ?>
</form>
</body>
</html>

```

Gambar 2.9 Update

Database Pegawai			
ID Employee	Nama	ID Department	Menu
1	Stefy	2	<input type="button" value="DELETE"/> <input type="button" value="UPDATE"/>
2	Thea	1	<input type="button" value="DELETE"/> <input type="button" value="UPDATE"/>
<input type="button" value="INSERT"/>			

Gambar 2.10 Output Database

[HOME](#)

ID Employee

Nama

ID Department

Gambar 2.11 Insert Data

[HOME](#)

ID Employee

Nama

ID Department

Gambar 2.12 Update Data

BAB III

KESIMPULAN

Kesimpulan pada praktikum ini yaitu agar membuat program yang bisa menyimpan data dalam jumlah yang banyak dan dapat membuat program yang bisa mengolah data yang tersimpan dalam database. Database terdiri dari satu atau lebih tabel. Sintak create untuk membuat atau menghapus database MySQL. MySQL adalah sistem manajemen basis data relasional open-source (RDBMS). Ini adalah sistem database paling populer yang digunakan dengan PHP. Data dalam database MySQL disimpan dalam tabel yang terdiri dari kolom dan baris. MySQL adalah sistem database yang berjalan di server. MySQL ideal untuk aplikasi kecil dan besar.

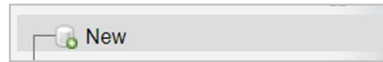
BAB IV
DAFTAR PUSTAKA

Likins, S. (2017). What is the relationship between SQL and PHP? - Quora. In *Quora* (p. 1). <https://www.quora.com/What-is-the-relationship-between-SQL-and-PHP>

PHP _ MySQL Database Introduction - GeeksforGeeks. (n.d.). <https://www.geeksforgeeks.org/php-mysql-database-introduction/>

Top Database Technologies for PHP Web Applications - The Crazy Programmer. (n.d.). <https://www.thecrazyprogrammer.com/2016/06/top-database-technologies-php-web-applications.html>

LAMPIRAN



Gambar 2.1 Create New Database

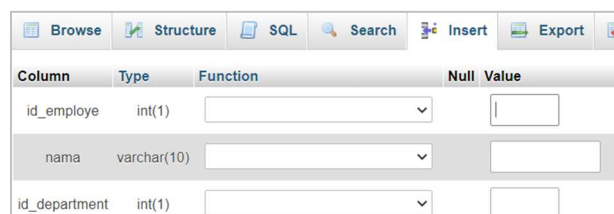
A screenshot of a 'Create database' dialog box. It has a title bar with a green icon and the text 'Create database'. Below the title bar, there is a text input field labeled 'Database name', a dropdown menu showing 'utf8mb4_general_ci', and a 'Create' button.

Gambar 2.2 Database Name

Pada tabel employee, terdapat tiga atribut yaitu id_employee int(1), nama varchar(10), dan id_department int(1). Lalu ke menu insert data untuk menambahkan data ke dalam database. Berikut hasil database yang dibuat.

#	Name	Type
<input type="checkbox"/> 1	id_employee	int(1)
<input type="checkbox"/> 2	nama	varchar(10)
<input type="checkbox"/> 3	id_department	int(1)

Gambar 2.3 Table Employee

A screenshot of a database management interface showing the 'Insert' menu. The menu bar includes 'Browse', 'Structure', 'SQL', 'Search', 'Insert', and 'Export'. Below the menu bar, there is a table with columns 'Column', 'Type', 'Function', 'Null', and 'Value'. The table has three rows: 'id_employee' with type 'int(1)', 'nama' with type 'varchar(10)', and 'id_department' with type 'int(1)'. Each row has a dropdown menu in the 'Function' column and a text input field in the 'Value' column.

Gambar 2.4 Insert Data Employee

id_employee	nama	id_department
1	Stefy	2
2	Thea	1

Gambar 2.5 Value Table

```
require("function.php");
$data = show("SELECT * FROM employee");
if (isset($_GET["delete"])) {
    if (delete($_GET > 0)) {
```

```

        echo "<script>
            alert('Data Dihapus');
            document.location.href = 'index.php';
        </script>";
    } else {
        echo "<script>
            alert('Data Gagal Dihapus');
            document.location.href = 'index.php';
        </script>";
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Database Pegawai | M3</title>
</head>
<body>
    <h1>Database Pegawai</h1><br>
    </form>
    <table border="1">
    <style>
        table, th, td {
            border: 1px solid black;
            border-collapse: collapse;
            padding: 10px;
        }
    </style>
    <tr>
        <td>ID Employee</td>
        <td>Nama</td>
        <td>ID Department</td>
        <td>Menu</td>
    </tr>
    <?php
    foreach ($data as $d) {
    ?>
        <tr>
            <td><?php echo $d["id_employee"]; ?></td>
            <td><?php echo $d["nama"]; ?></td>
            <td><?php echo $d["id_department"]; ?></td>
            <td>
                <form action="" method="GET">
                    <button type="submit" name="delete" value=<?php echo $d["id_employ
e"]; ?>>DELETE</button>
                </form><br>
                <form action="update.php" method="GET">
                    <button type="submit" name="update" value=<?php echo $d["id_employ
e"]; ?>>UPDATE</button>
                </form>
            </td>
        </tr>
    <?php
    }
    ?>
    </table>
    <form action="insert.php">
        <br><button>INSERT</button>
    </body>
</html>

```

Gambar 2.6 Table Index

```

<?php
$servername = "localhost";
$dbname = "modul3";
$username = "root";
$password = "";

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

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

function insert($data){
    global $conn;
    $id_employe = $_POST["id_employe"];
    $nama = $_POST["nama"];
    $id_department = $_POST["id_department$id_department"];

    mysqli_query($conn, "INSERT INTO employee VALUES('$id_employe','$nama','$id_department'
    ')");
    return mysqli_affected_rows($conn);
}

function delete($data){
    global $conn;
    $id_employe = $_GET["delete"];
    mysqli_query($conn, "DELETE FROM employee WHERE id_employe = $id_employe");
    return mysqli_affected_rows($conn);
}

function update($data){
    global $conn;
    $id_employe = $_POST["id_employe"];
    $nama = $_POST["nama"];
    $id_department = $_POST["id_department$id_department"];
    mysqli_query($conn, "UPDATE employee SET nama = '$nama',
    id_department = '$id_department'
    WHERE id_employe = $id_employe");
    return mysqli_affected_rows($conn);
}

?>

```

Gambar 2.7 Function

```

<?php
require("function.php");
if (isset($_POST["insert"])) {
    if (insert($_POST) > 0) {
        echo "<script>
            alert('Data Ditambahkan');
            document.location.href = 'index.php';
        </script>";
    } else {
        echo "<script>
            alert('Data Gagal Ditambahkan');
            document.location.href = 'index.php';
        </script>";
    }
}

```



```

}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <link rel="stylesheet" href="style.css" type="text/css">
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Insert</title>
</head>
<body>
    <a href="index.php">HOME</a>
    <form action="" method="POST">
        <label for="id_employe"><br>ID Employe<br></label>
        <input type="text" name="id_employe" id="id_employe"><br>
        <label for="nama"><br>Nama<br></label>
        <input type="text" name="nama" id="nama"><br>
        <label for="id_department"><br>ID Department<br></label>
        <input type="text" name="id_department" id="id_department"><br><br>
        <button type="submit" name="insert">OK</button>
    </form>
</body>
</html>

```

Gambar 2.8 Insert Data

```

<?php
require("function.php");
$id_employe = $_GET["update"];
$data = show("SELECT * FROM employee WHERE id_employe = $id_employe");
if (isset($_POST["ubah"])) {
    if (update($_GET > 0)) {
        echo "<script>
            alert('data berhasil diupdate');
            document.location.href = 'index.php';
        </script>";
    } else {
        echo "<script>
            alert('data gagal diupdate');
            document.location.href = 'index.php';
        </script>";
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Update</title>
</head>
<body>
    <a href="index.php">home</a>
    <form action="" method="POST">
        <?php
        foreach ($data as $d) {
            ?>
            <ul>
                <li>
                    <label for="id_employe">id_employe</label>
                    <input type="text" name="id_employe" id="id_employe" value=<?php echo
$id["id_employe"]; ?> readonly>
                </li>
                <li>

```

```

        <label for="nama">nama</label>
        <input type="text" name="nama" id="nama" value=<?php echo $d["nama"];
?>>
    </li>
    <li>
        <label for="id_department">id_department</label>
        <input type="text" name="id_department" id="id_department" value=<?php
echo $d["id_department"]; ?>>
    </li>
    <li>
        <button type="submit" name="ubah">update</button>
    </li>
</ul>
<?php
}
?>
</form>
</body>
</html>

```

Gambar 2.9 Update

Database Pegawai			
ID Employee	Nama	ID Department	Menu
1	Stefy	2	<input type="button" value="DELETE"/> <input type="button" value="UPDATE"/>
2	Thea	1	<input type="button" value="DELETE"/> <input type="button" value="UPDATE"/>
<input type="button" value="INSERT"/>			

Gambar 2.10 Output Database

[HOME](#)

ID Employee

Nama

ID Department

Gambar 2.11 Insert Data

[HOME](#)

ID Employe

Nama

ID Department

Gambar 2.12 Update Data