

MODUL 10

PEMROGRAMAN WEB

CREATE READ UPDATE DELETE

A. TUJUAN PERKULIAHAN

- a. Mahasiswa mengenali konsep CRUD
- b. Mahasiswa mampu mengimplementasikan CRUD dengan PHP

B. ALOKASI WAKTU 3 x 50 menit

C. DASAR TEORI

1. DEFINISI

CRUD, istilah tersebut merupakan singkatan dari Create, Read, Update dan Delete. CRUD merupakan istilah untuk kumpulan teknik manipulasi data pada database. Sebutan CRUD biasa digunakan pada sisi frontend sebuah sistem. PHP sering dijodohkan dengan MySQL untuk membangun aplikasi web. MySQL berperan sebagai tempat menyimpan data (basis data) sedangkan PHP adalah program yang mengelola data tersebut.

Sebenarnya PHP tidak hanya bisa dijodohkan dengan MySQL saja. PHP juga bisa dijodohkan dengan SQLite, PostgreSQL, MongoDB, Ms. Access, dsb. PHP dan MySQL dapat saling berhubungan melalui perantara API (Application Programming Interface). Ada tiga jenis API MySQL yang disediakan di PHP:

- mysql API ini sudah tidak didukung/dikembangkan lagi (kadaluarsa).
- mysqli (mysql improvement) peningkatan dari API mysql
- PDO (PHP Data Object)



API yang pertama sudah tidak didukung lagi sejak PHP versi 5.5.x dan digantikan dengan mysqli yang lebih canggih. Karena mysqli dapat digunakan secara OOP dan prosedural.

Sedangkan API PDO digunakan untuk menghubungkan PHP dengan vendor database yang lainnya. Jadi tidak hanya untuk MySQL saja.

Pada modul ini, kita akan belajar cara menghubungkan PHP dan MySQL menggunakan API mysqli secara prosedural. Berikut penjelasan singkat terkait CRUD.

- **C (Create):** Membuat sebuah data baru. Kita membuat dan menyimpan data registrasi ke database. Contoh ketika kita sedang melakukan registrasi di sebuah laman web sudah merupakan Create dari CRUD.
- **R (Read):** Membaca atau menampilkan suatu data yang berada di database MySQL misalnya, kemudian ditampilkan di WEB menggunakan bahasa pemrograman PHP
- **U (Update):** Mengubah sebuah data dari database yang kemudian di edit menggunakan bahasa pemrograman PHP berupa WEB. Contoh edit profil facebook.
- **D (Delete):** Fungsinya hampir sama dengan Update akan tetapi proses ini adalah untuk melakukan penghapusan data di database melalui bahasa PHP. Contoh pada sebuah blog terkadang ada komentar, kemudian kita hapus komentar tersebut,. Langkah itu sudah termasuk proses delete dalam CRUD.

D. PRAKTIKUM : Pembuatan CRUD

1. Membuat Database

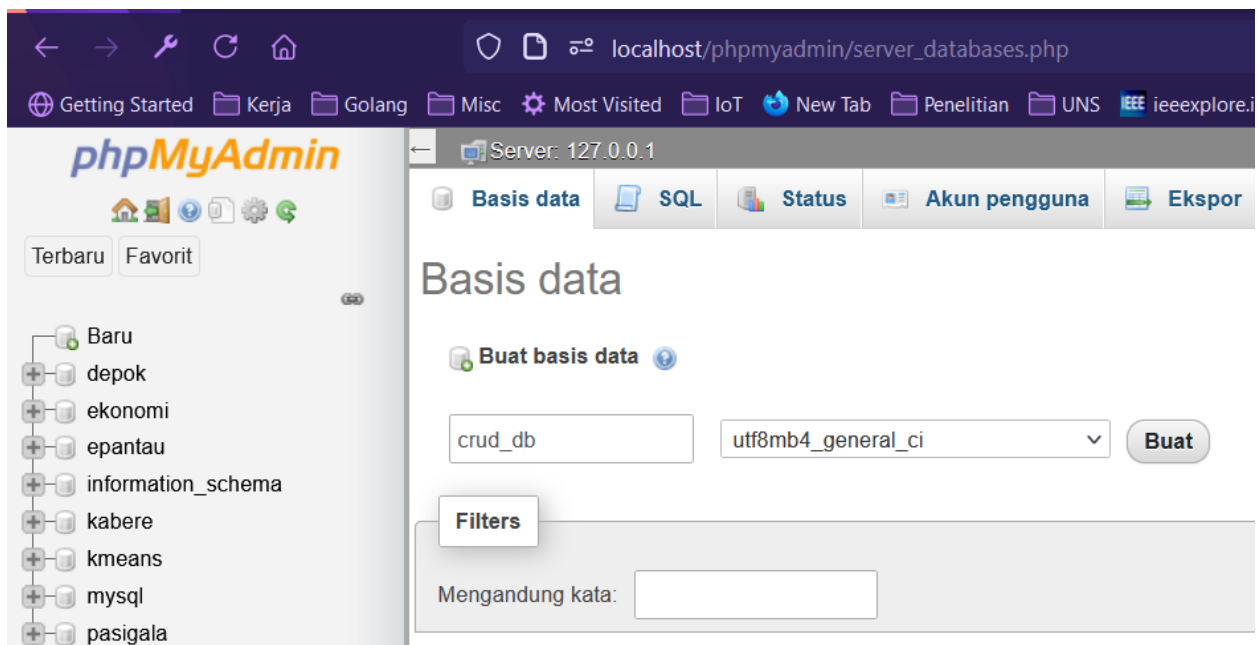
- Pertama kita buat terlebih dahulu file **database.sql** kemudian kita import file database.sql ini ke dalam database (menggunakan PHPMyAdmin).

```
/* Membuat tabel */
create database crud_db;

use crud_db;

CREATE TABLE `mahasiswa` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(100),
  `email` varchar(100),
  `mobile` varchar(15),
  PRIMARY KEY (`id`)
);
```

- Anda juga bisa menggunakan phpMyAdmin untuk membuat database dengan menggunakan interface bawaan dari XAMPP.



2. Membuat file config.php

- File **config.php** menyimpan informasi tentang database host, username dan password. Sebagian besar server lokal bekerja dengan detail yang diberikan. Anda dapat mengubahnya sesuai dengan detail host dan database Anda.

```
<?php
/**
 * menggunakan mysqli
 */

$databaseHost = 'localhost';
$databaseName = 'crud_db';
$databaseUsername = 'root';
$databasePassword = '';

$mysqli = mysqli_connect($databaseHost, $databaseUsername,
$databasePassword, $databaseName);

?>
```

- Untuk menghubungkan PHP dengan MySQL, kita menggunakan fungsi `mysqli_connect()` dengan alamat server sebagai parameter pertama, user database sebagai parameter kedua, password user sebagai parameter ketiga, dan nama database sebagai parameter keempat.
- Silahkan isi password sesuai dengan yang dibuat di servernya. Secara default, username dan password dari database yang dibuat oleh xampp adalah 'root' dan '' atau string kosong.

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

3. Membuat file index.php

- File **index.php** merupakan file utama yang menyertakan file konfigurasi untuk koneksi database. Kemudian menampilkan semua daftar pengguna menggunakan MySQL Select Query. Pengguna yang akan ditampilkan di dalam daftar perlu menambahkan terlebih dahulu menggunakan tautan 'Tambahkan Pengguna Baru'.

```
<?php
// Membuat koneksi dengan config.php
include_once("config.php");
```

```
// Fetch semua mahasiswa dari DB
$result = mysqli_query($mysqli, "SELECT * FROM mahasiswa ORDER BY id
DESC");
?>

<html>
<head>
    <title>Homepage</title>
</head>

<body>
<a href="add.php">Add Mahasiswa</a><br/><br/>

    <table width='80%' border=1>

        <tr>
            <th>Name</th> <th>Mobile</th> <th>Email</th> <th>Update</th>
        </tr>
        <?php
        while($user_data = mysqli_fetch_array($result)) {
            echo "<tr>";
            echo "<td>".$user_data['name']. "</td>";
            echo "<td>".$user_data['mobile']. "</td>";
            echo "<td>".$user_data['email']. "</td>";
            echo "<td><a href='edit.php?id=$user_data[id]'>Edit</a> | <a
href='delete.php?id=$user_data[id]'>Delete</a></td></tr>";
        }
        ?>
    </table>
</body>
</html>
```

4. Membuat file add.php

- File **add.php** berfungsi untuk menambahkan pengguna baru. Formulir HTML digunakan untuk menerima masukan data pengguna. Setelah data pengguna diserahkan, MySQL INSERT Query digunakan untuk memasukkan data pengguna ke dalam database.

```
<html>
<head>
    <title>Add Mahasiswa</title>
</head>

<body>
    <a href="index.php">Kembali</a>
    <br/><br/>

    <form action="add.php" method="post" name="form1">
        <table width="25%" border="0">
            <tr>
                <td>Name</td>
```

```

        <td><input type="text" name="name"></td>
    </tr>
    <tr>
        <td>Email</td>
        <td><input type="email" name="email"></td>
    </tr>
    <tr>
        <td>Mobile</td>
        <td><input type="number" name="mobile"></td>
    </tr>
    <tr>
        <td></td>
        <td><input type="submit" name="Submit"
value="Add"></td>
    </tr>
</table>
</form>

<?php
    // Cek apakah ada parameter $_POST yang terisi, jika iya
    masukkan dalam DB.
    if(isset($_POST['Submit'])) {
        $name = $_POST['name'];
        $email = $_POST['email'];
        $mobile = $_POST['mobile'];

        // include database
        include_once("config.php");

        // Insert data
        $result = mysqli_query($mysqli, "INSERT INTO
mahasiswa(name,email,mobile) VALUES('$name','$email','$mobile')");

        // Menampilkan data telah berhasil
        echo "Mahasiswa berhasil ditambahkan! <a href='index.php'>View
Mahasiswa</a>";
    }
    ?>
</body>
</html>

```

5. Membuat file edit.php

- **Edit.php** digunakan untuk mengedit / update data pengguna. Anda dapat mengubah data pengguna dan memperbaharui. File ini akan mengarahkan pengguna kembali ke homepage, setelah update sukses.

```

<?php
// include database
include_once("config.php");

// Cek apakah parameter $_POST terisi, update data jika iya
if(isset($_POST['update']))

```

```

{
    $id = $_POST['id'];

    $name=$_POST['name'];
    $mobile=$_POST['mobile'];
    $email=$_POST['email'];

    // update data
    $result = mysqli_query($mysqli, "UPDATE mahasiswa SET
name='$name',email='$email',mobile='$mobile' WHERE id=$id");

    // Redirect kembali ke halaman utama
    header("Location: index.php");
}
?>
<?php
// Mengambil ID dan menampilkan data berdasarkan ID yang didapat
$id = $_GET['id'];

// Fetch user data
$result = mysqli_query($mysqli, "SELECT * FROM mahasiswa WHERE
id=$id");

while($user_data = mysqli_fetch_array($result))
{
    $name = $user_data['name'];
    $email = $user_data['email'];
    $mobile = $user_data['mobile'];
}
?>
<html>
<head>
    <title>Edit Data Mahasiswa</title>
</head>

<body>
    <a href="index.php">Home</a>
    <br/><br/>

    <form name="update_user" method="post" action="edit.php">
        <table border="0">
            <tr>
                <td>Name</td>
                <td><input type="text" name="name" value=<?php echo
$name;?>></td>
            </tr>
            <tr>
                <td>Email</td>
                <td><input type="text" name="email" value=<?php echo
$email;?>></td>
            </tr>
            <tr>
                <td>Mobile</td>
                <td><input type="text" name="mobile" value=<?php echo
$mobile;?>></td>
            </tr>
            <tr>

```

```
 <input type="hidden" name="id" value=<?php echo $_GET['id'];?>></td>  <input type="submit" name="update" value="Update"></td> </tr> </table> </form> </body> </html> | |
```

6. Membuat file delete.php

- File **delete.php** hanya dipanggil saat kita klik *link* 'Delete' untuk item manapun yang dipilih.

```

<?php
// include database
include_once("config.php");

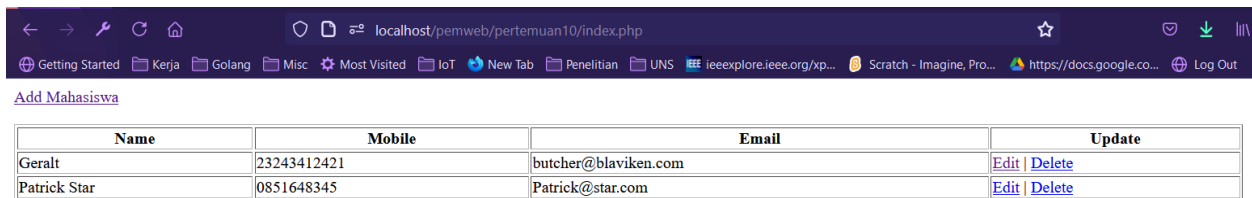
// Mengambil ID
$id = $_GET['id'];

// Menghapus baris data dengan ID yang berkesesuaian
$result = mysqli_query($mysqli, "DELETE FROM mahasiswa WHERE id=$id");

// Redirect kembali ke halaman utama
header("Location:index.php");
?>

```

Hasil dari tampilan tersebut adalah sebagai berikut :




The screenshot shows a web browser window with the address bar displaying 'localhost/pemweb/pertemuan10/index.php'. Below the browser window, there is a link 'Add Mahasiswa'. Below that is a table with four columns: 'Name', 'Mobile', 'Email', and 'Update'. The table contains two rows of data. The first row has 'Geralt', '23243412421', 'butcher@blaviken.com', and a link 'Edit | Delete'. The second row has 'Patrick Star', '0851648345', 'Patrick@star.com', and a link 'Edit | Delete'.

Name	Mobile	Email	Update
Geralt	23243412421	butcher@blaviken.com	Edit Delete
Patrick Star	0851648345	Patrick@star.com	Edit Delete

E. TUGAS

- Buatlah sebuah CRUD sederhana yang memuat tabel karyawan.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_karyawan 	int(10)			No	None		AUTO_INCREMENT
2	nama	varchar(255)	utf8mb4_general_ci		No	None		
3	email	varchar(255)	utf8mb4_general_ci		No	None		
4	telepon	varchar(15)	utf8mb4_general_ci		No	None		
5	alamat	text	utf8mb4_general_ci		No	None		
6	jenis_kelamin	enum('pria', 'wanita')	utf8mb4_general_ci		No	None		
7	tempat_lahir	varchar(100)	utf8mb4_general_ci		No	None		
8	tanggal_lahir	date			No	None		

- Semua field bersifat wajib diisi (required), kecuali telepon.
- Berikan style sederhana (bebas) pada kode yang Anda buat.

F. REFERENSI

https://www.w3schools.com/php/php_forms.asp

<https://www.malasngoding.com/membuat-crud-dengan-php-dan-mysqli-menampilkan-data/>

<https://www.petanikode.com/tutorial-php-mysql/>

<https://www.codepolitan.com/tutorial-membuat-crud-php-dengan-mysql-59897c72d8470>