



# Modul Pemrograman Web

Ilmu Komputer

## Pertemuan 12:

CRUD PHP Native

### **Disusun Oleh:**

Asisten Dosen Pemrograman Web

Jurusan Ilmu Komputer Universitas Lampung

## Tim Penyusun

### Pengarah

Rizky Prabowo, M.Kom.

M. Iqbal Parabi, S.Si., M.T.

### Penyusun

- M Arya Dzaky A (2217051007)
- Gilbert Hasiholan Sibuea (2217051118)
- Dafa Farhan Haqiqi (2267061001)
- Luthfi Aditya (2217051062)
- Berli Anta Atrizki (2217051076)
- Cakrawangsa Praba Alkautsar (2217051003)
- M Nurfaiz Satriani Azra (2217051001)
- Jhon Valentino Nababan (2217051087)

## Gambaran Umum

Materi ini membahas tentang penerapan fitur dasar CRUD (Create, Read, Update, Delete) dengan menggunakan PHP native, MySQL, dan Bootstrap sebagai framework CSS. Penggunaan Bootstrap membantu membuat antarmuka aplikasi yang responsif dan menarik tanpa harus menulis banyak kode CSS. Praktikum ini bertujuan memberikan pemahaman dasar mengenai manipulasi data dalam database melalui antarmuka form yang rapi dan profesional.

## Capaian Pembelajaran

Setelah menyelesaikan materi ini, mahasiswa diharapkan mampu:

1. Menjelaskan konsep dasar CRUD dan penerapannya dalam aplikasi web.
2. Membangun antarmuka form dengan Bootstrap.
3. Mengintegrasikan PHP dengan MySQL untuk proses input, tampil, edit, dan hapus data.
4. Menerapkan validasi form dasar.
5. Menggunakan struktur kode yang rapi dan mudah dipahami.

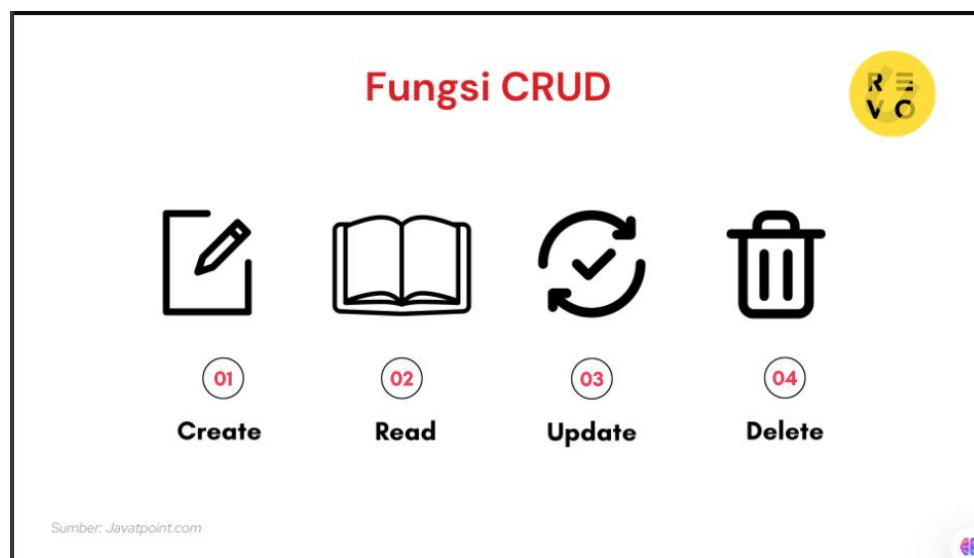
## Materi Praktikum

### Apa Itu CRUD

Sebelum membahas lebih jauh tentang bagaimana CRUD bekerja, mari kita lihat terlebih dahulu **apa itu CRUD** secara mendasar. Seperti disebutkan sebelumnya, CRUD adalah singkatan dari empat operasi utama dalam pengelolaan data:

1. **Create:** Menambahkan data baru.
2. **Read:** Membaca atau menampilkan data yang ada.
3. **Update:** Memperbarui atau mengubah data yang sudah ada.
4. **Delete:** Menghapus data dari sistem.

Keempat operasi ini merupakan elemen penting dalam hampir semua aplikasi berbasis data, seperti sistem manajemen inventaris, aplikasi keuangan, hingga media sosial. Sebagai contoh, dalam aplikasi media sosial, saat kita membuat akun (Create), melihat profil (Read), memperbarui status (Update), atau menghapus postingan (Delete), kita sudah melakukan proses CRUD.



### Mengapa CRUD Sangat Penting ?

CRUD tidak hanya memudahkan pengelolaan data tetapi juga memastikan integritas data dalam aplikasi. Dalam dunia pengembangan, konsep CRUD membantu pengembang untuk membangun aplikasi yang efisien dan mudah dikelola.

### Fungsi masing-masing CRUD

#### Create

Dengan fungsi Create, pengguna dapat menambahkan catatan atau entri baru ke *database*. Fungsi ini meliputi memasukkan data ke dalam *database*, membuat baris baru dalam tabel, dan menentukan nilai untuk atribut (kolom) dari baris itu. Sebagai contoh, saat pengguna membuat akun pengguna baru di *website*, informasi yang relevan seperti nama, alamat *e-mail*, dan *password* dimasukkan ke dalam *database*.

## Read

Fungsi Read membantu pengguna untuk mengambil atau mengakses data yang ada dari *database*, seperti saat mencari catatan tertentu atau memperoleh semua catatan yang memenuhi kriteria tertentu. Misalnya, saat pengguna masuk ke *website*, aplikasi akan melakukan operasi Read untuk mengakses informasi akun mereka dari *database* dan menampilkannya di layar.

## Update

Update memungkinkan pengguna untuk mengubah atau mengedit data yang ada di *database*. Misalnya, jika pengguna ingin mengubah alamat *e-mail*-nya atau memperbarui informasi tertentu di profilnya, aplikasi akan menjalankan operasi Update untuk mengubah data yang relevan di *database*.

## Delete

Fungsi Delete memberi akses ke pengguna untuk menghapus data yang tidak diinginkan dari *database*. Dengan demikian, saat pengguna ingin menghapus akunnya di suatu *website*, aplikasi akan melakukan *Delete* untuk menghapus data tersebut dari *database*.

## Bagaimana cara CRUD bekerja ?

Setelah mengetahui **apa itu CRUD**, langkah berikutnya adalah memahami cara kerjanya. Setiap aplikasi yang menggunakan CRUD akan memiliki fungsi atau metode yang mendukung keempat operasi ini. Cara kerja CRUD sering kali ditentukan oleh bahasa pemrograman atau framework yang digunakan.

Misalnya, dalam **PHP**, CRUD dapat diimplementasikan melalui koneksi database MySQL, menggunakan query SQL dasar. **Fungsi CRUD** dalam PHP akan mencakup proses menambahkan data ke database, membaca data, memperbarui data yang ada, dan menghapus data ketika tidak lagi diperlukan. Contoh CRUD ini dapat diaplikasikan pada aplikasi seperti sistem inventaris, katalog produk, atau daftar pengguna.

Berikut adalah gambaran sederhana cara kerja CRUD:

1. **Create:** Pengguna memasukkan data baru ke database.
2. **Read:** Data yang tersimpan di database ditampilkan kepada pengguna.
3. **Update:** Pengguna dapat mengubah data yang sudah ada.
4. **Delete:** Data yang tidak lagi diperlukan dapat dihapus.

## Langkah Praktikum

Berikut adalah Langkah-langkah pembuatan CRUD dengan PHP native :

1. Persiapkan database

Gunakan database yang kita gunakan pada pertemuan praktikum sebelumnya, buatlah table mahasiswa dengan script sql sebagai berikut :

```
CREATE TABLE mahasiswa (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nama VARCHAR(100),  
    nim VARCHAR(20)  
);
```

2. Koneksi Database (koneksi/db.php)

```
<?php  
$host = "localhost";  
$user = "root";  
$pass = "";  
$db = "Latihan_login_db ";  
  
$conn = mysqli_connect($host, $user, $pass, $db);  
if (!$conn) {  
    die("Koneksi gagal: " . mysqli_connect_error());  
}  
?>
```

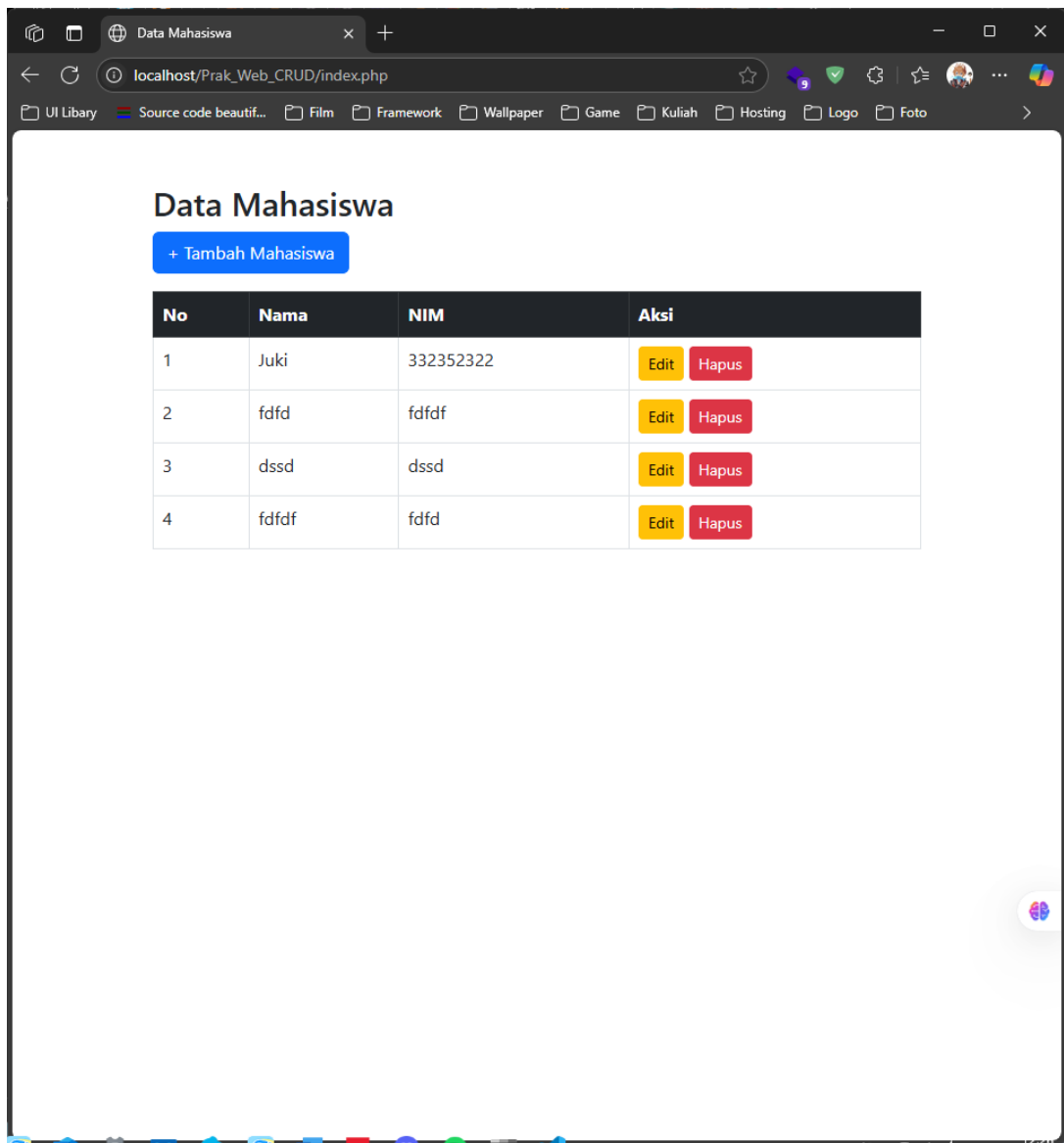
3. Menampilkan Data (index.php)

```

<?php include "koneksi/db.php"; ?>
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Data Mahasiswa</title>
    <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap
.min.css" rel="stylesheet">
</head>
<body class="container mt-5">
    <h2>Data Mahasiswa</h2>
    <a href="tambah.php" class="btn btn-primary mb-3">+ Tambah
Mahasiswa</a>
    <table class="table table-bordered">
        <thead class="table-dark">
            <tr><th>No</th><th>Nama</th><th>NIM</th><th>Aksi</th></tr>
        </thead>
        <tbody>
            <?php
            $no = 1;
            $result = mysqli_query($conn, "SELECT * FROM mahasiswa");
            while ($row = mysqli_fetch_assoc($result)) {
                echo "<tr>
                    <td>$no</td>
                    <td>{$row['nama']}</td>
                    <td>{$row['nim']}</td>
                    <td>
                        <a href='edit.php?id={$row['id']}' class='btn btn-
warning btn-sm'>Edit</a>
                        <a href='hapus.php?id={$row['id']}' class='btn btn-
danger btn-sm' onclick='return confirm(\"Hapus data
ini?\")'>Hapus</a>
                    </td>
                </tr>";
                $no++;
            }
            ?>
        </tbody>
    </table>
</body>
</html>

```

Langkah selanjutnya adalah membuat halaman utama (index.php) yang berfungsi untuk menampilkan seluruh data mahasiswa dari database. Di halaman ini, kita menggunakan query `SELECT * FROM mahasiswa` untuk mengambil semua data. Hasil query kemudian ditampilkan dalam tabel HTML yang sudah didesain dengan Bootstrap agar tampilannya rapi dan responsif. Setiap baris data memiliki tombol "Edit" dan "Hapus" yang mengarah ke halaman `edit.php` dan `hapus.php`, membawa id sebagai parameter. Halaman ini merepresentasikan operasi **Read** dalam konsep CRUD.



#### 4. Menambah Data (tambah.php)

```
<?php include "koneksi/db.php"; ?>
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Tambah Mahasiswa</title>
  <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap
.min.css" rel="stylesheet">
</head>
<body class="container mt-5">
  <h2>Tambah Data Mahasiswa</h2>
  <form method="POST">
    <div class="mb-3">
      <label>Nama</label>
      <input type="text" name="nama" class="form-control" required>
    </div>
```



```

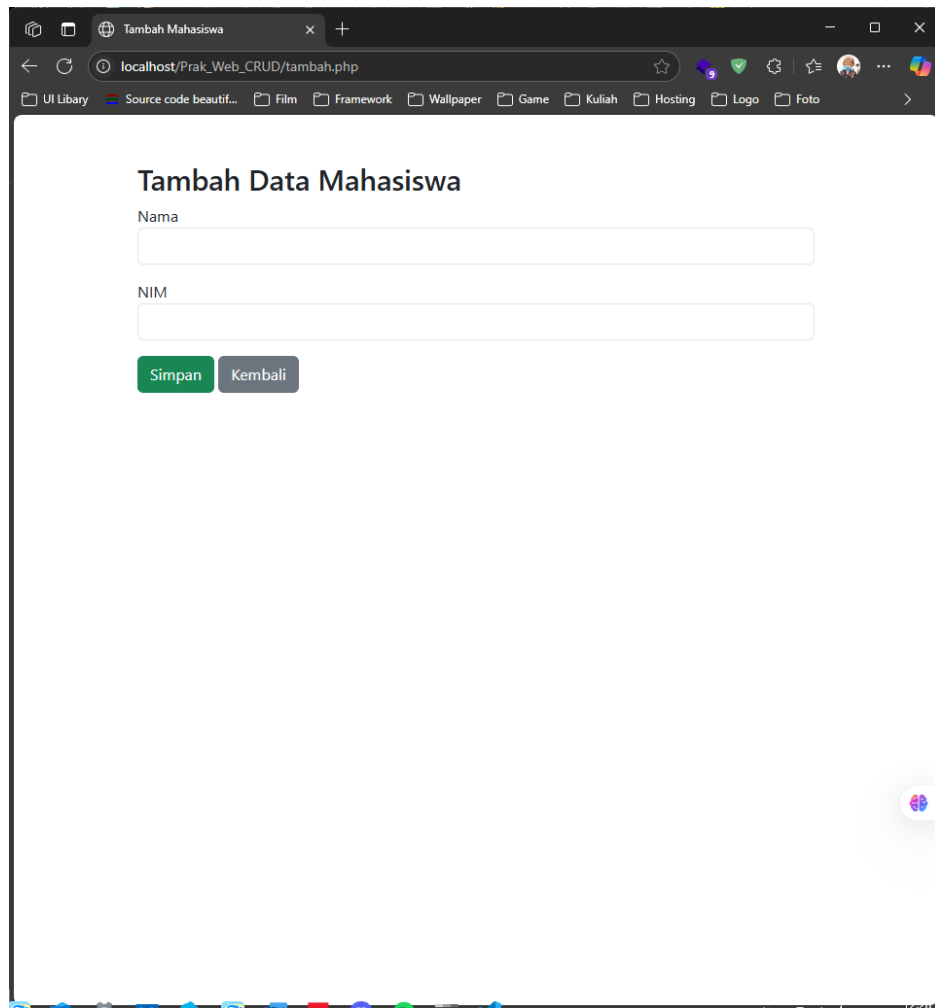
<div class="mb-3">
  <label>NIM</label>
  <input type="text" name="nim" class="form-control" required>
</div>
  <button type="submit" name="simpan" class="btn btn-
success">Simpan</button>
  <a href="index.php" class="btn btn-secondary">Kembali</a>
</form>

<?php
if (isset($_POST['simpan'])) {
  $nama = $_POST['nama'];
  $nim = $_POST['nim'];
  mysqli_query($conn, "INSERT INTO mahasiswa (nama, nim) VALUES
('$nama', '$nim')");
  echo "<div class='alert alert-success mt-3'>Data berhasil
disimpan.</div>
  <script>
    alert('Data Berhasil Ditambah')
    window.location.href = 'index.php'

  </script>
  ";
}
?>
</body>
</html>

```

Untuk menambahkan data baru, dibuat halaman tambah.php yang menampilkan form input berisi dua isian yaitu nama dan NIM mahasiswa. Saat form dikirim (submit), PHP akan memproses input tersebut menggunakan metode POST. Nilai dari input kemudian dimasukkan ke dalam database menggunakan query INSERT INTO mahasiswa (nama, nim) VALUES (...). Setelah berhasil, sistem akan menampilkan pesan konfirmasi bahwa data telah disimpan. Ini merupakan implementasi dari fungsi **Create** dalam CRUD.



## 5. Mengedit Data (edit.php)

```
<?php
include "koneksi/db.php";
$id = $_GET['id'];
$data = mysqli_fetch_assoc(mysqli_query($conn, "SELECT * FROM
mahasiswa WHERE id=$id"));
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Edit Mahasiswa</title>
  <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap
.min.css" rel="stylesheet">
</head>
<body class="container mt-5">
  <h2>Edit Data Mahasiswa</h2>
  <form method="POST">
    <div class="mb-3">
      <label>Nama</label>
```

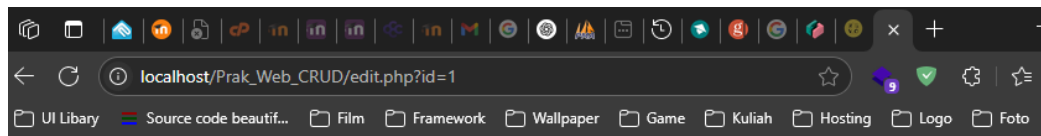
```

        <input type="text" name="nama" class="form-control" value="<?=$data['nama'] ?>" required>
    </div>
    <div class="mb-3">
        <label>NIM</label>
        <input type="text" name="nim" class="form-control" value="<?=$data['nim'] ?>" required>
    </div>
    <button type="submit" name="update" class="btn btn-warning">Update</button>
    <a href="index.php" class="btn btn-secondary">Kembali</a>
</form>

<?php
if (isset($_POST['update'])) {
    $nama = $_POST['nama'];
    $nim = $_POST['nim'];
    mysqli_query($conn, "UPDATE mahasiswa SET nama='$nama',
nim='$nim' WHERE id=$id");
    echo "<div class='alert alert-success mt-3'>Data berhasil
diupdate.</div>";
}
?>
</body>
</html>

```

Halaman edit.php memungkinkan pengguna untuk memperbarui data mahasiswa yang telah dimasukkan sebelumnya. Ketika tombol "Edit" pada halaman index.php diklik, halaman ini akan menerima id mahasiswa melalui URL. Berdasarkan ID tersebut, PHP melakukan query ke database untuk mengambil data lama yang akan ditampilkan di form. Pengguna kemudian dapat mengubah nilai dan mengklik tombol "Update" untuk menyimpan perubahan. Query UPDATE digunakan untuk mengganti nilai nama dan NIM berdasarkan ID. Proses ini merupakan implementasi dari fungsi **Update** dalam CRUD.



## Edit Data Mahasiswa

Nama

NIM

### 6. Menghapus data mahasiswa (hapus.php)

```
<?php
include "koneksi/db.php";
$id = $_GET['id'];
mysqli_query($conn, "DELETE FROM mahasiswa WHERE id=$id");
header("Location: index.php");
?>
```

Langkah terakhir dalam alur CRUD adalah proses penghapusan data. Ketika pengguna menekan tombol “Hapus” pada halaman utama, browser akan diarahkan ke hapus.php dengan membawa ID mahasiswa.

Di dalam halaman ini, PHP mengambil ID tersebut dan menjalankan query DELETE FROM mahasiswa WHERE id=... untuk menghapus data dari tabel. Setelah penghapusan berhasil, sistem akan langsung mengembalikan pengguna ke halaman index.php. Ini adalah implementasi dari fungsi **Delete** dalam CRUD, dan menjadi cara efektif untuk mengelola data yang sudah tidak relevan atau salah.

localhost says

Hapus data ini?

OK

Cancel

## Data Mahasiswa

+ Tambah Mahasiswa

No	Nama	NIM	Aksi
1	Juki Sudrajat	332352322	<a href="#">Edit</a> <a href="#">Hapus</a>
2	fdfd	fdfdf	<a href="#">Edit</a> <a href="#">Hapus</a>
3	dssd	dssd	<a href="#">Edit</a> <a href="#">Hapus</a>

## Data Mahasiswa

+ Tambah Mahasiswa

No	Nama	NIM	Aksi
1	Juki Sudrajat	332352322	<a href="#">Edit</a> <a href="#">Hapus</a>
2	fdfd	fdfdf	<a href="#">Edit</a> <a href="#">Hapus</a>

## Referensi

[Tutorial PHP dan MySQL: Membuat Aplikasi CRUD \[Studi Kasus Pendaftaran Siswa Baru\]](#)

[Apa Itu CRUD? Pengertian dan Fungsi CRUD dalam Aplikasi Web – CODEPOLITAN](#)

[Apa itu CRUD? Arti, Fungsi, Contoh, FAQs 2025 | Revou](#)