

LAPORAN TUGAS PEMROGRAMAN WEB

TUGAS KETIGA—*Praktikum Modul Ketiga: PHP dan MYSQL*



Nama : Theo Jesen Naftali Sirait

NRP : 3125600106

Dosen Pengajar : Prof. M. Udin Harun Al Rasyid S. Kom, Ph. D

PROGRAM STUDI S. Tr. TEKNIK INFORMATIKA

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA (PENS)

TAHUN 2026

BAB I

PERCOBAAN

1.1. Percobaan Pertama—Membuat Script Koneksi Database.

1.1.1. *Source Code:*

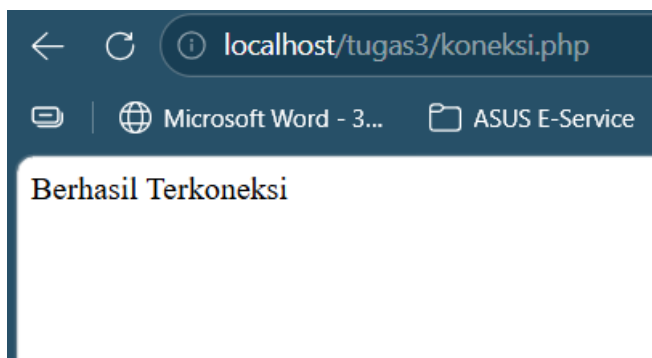
```
<?<?php
$host = "localhost";
$user = "root";
$pass = "128206";
$db = "kurir_db";

$conn = mysqli_connect($host, $user, $pass, $db);

if(!$conn)
    die("Gagal Terkoneksi: ".mysqli_connect_error());

echo "Berhasil Terkoneksi<br>";
?>
```

1.1.2. *Output:*



1.1.3. Penjelasan:

Percobaan ini bertujuan untuk memahami cara menghubungkan aplikasi PHP dengan server database MySQL menggunakan ekstensi mysqli. Pada script ini digunakan fungsi `mysqli_connect()` yang menerima parameter host, username, password, dan nama database untuk membuka koneksi ke server MySQL. Jika koneksi berhasil maka program dapat menjalankan perintah SQL pada database tersebut, sedangkan jika gagal maka fungsi `die()` akan menghentikan

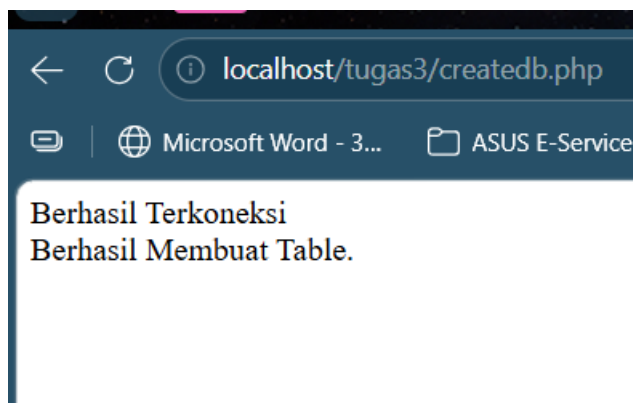
eksekusi program dan menampilkan pesan kesalahan dari `mysqli_connect_error()`.

1.2. Percobaan Kedua—Membuat Tabel pada MySQL via PHP.

1.2.1. *Source Code:*

```
<?php
include 'koneksi.php';
$sql = "CREATE TABLE kurir(
    id INT AUTO_INCREMENT PRIMARY KEY,
    nama VARCHAR(50) NOT NULL,
    sektor VARCHAR(15) NOT NULL,
    jenis_barang VARCHAR(7)
)";
if(mysqli_query($conn, $sql))
    echo "Berhasil Membuat Table.";
else
    echo "Gagal Terkoneksi: ".mysqli_error($conn);
?>
```

1.2.2. *Output:*



1.2.3. Penjelasan:

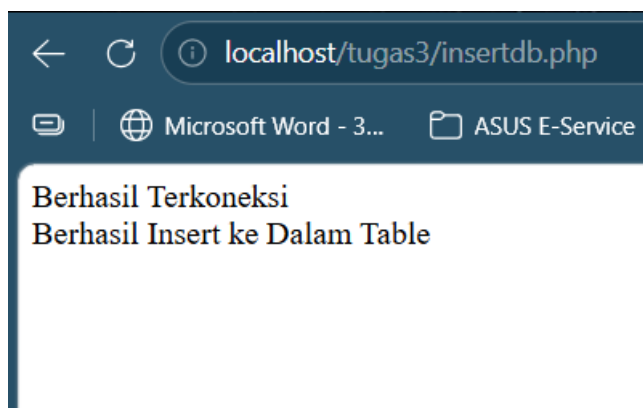
Percobaan ini menunjukkan bahwa struktur database seperti tabel tidak hanya dapat dibuat melalui phpMyAdmin, tetapi juga dapat dibuat langsung melalui script PHP menggunakan perintah SQL DDL (Data Definition Language). Pada contoh ini, perintah CREATE TABLE digunakan untuk membuat tabel dengan beberapa kolom seperti id, nim, nama, dan jurusan. Script PHP mengirimkan perintah SQL tersebut ke database menggunakan fungsi `mysqli_query()`. Jika query berhasil dijalankan maka tabel akan dibuat di dalam database, sedangkan jika terjadi kesalahan maka pesan error dari MySQL akan ditampilkan.

1.3. Percobaan Ketiga—Memasukkan Data ke Tabel (Insert).

1.3.1. *Source Code:*

```
<?php
include 'koneksi.php';
$sql = "INSERT INTO kurir(nama, sektor, jenis_barang) VALUE ('Guinevere','Jabodetabek', 'Besar)";
if(mysqli_query($conn, $sql))
    echo "Berhasil Insert ke Dalam Table";
else
    echo "Gagal Terkoneksi: ".$sql."<br>".mysqli_error($conn);
?>
```

1.3.2. *Output:*



1.3.3. Penjelasan:

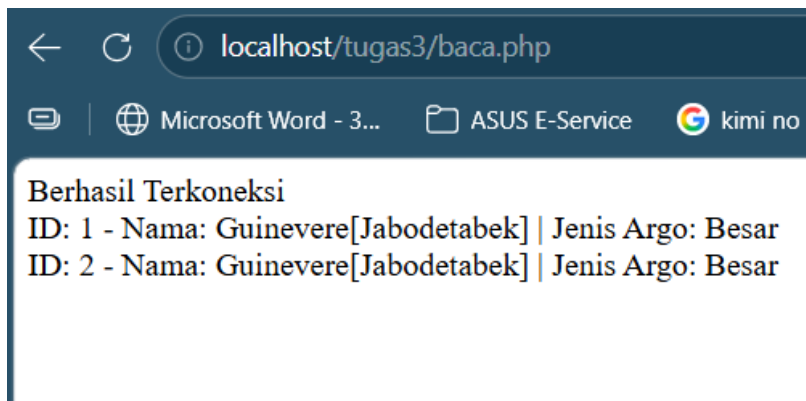
Pada percobaan ini dipelajari bagaimana menambahkan data baru ke dalam tabel database menggunakan perintah SQL INSERT INTO. Data yang dimasukkan berupa nilai untuk kolom nim, nama, dan jurusan pada tabel. Script PHP mengirimkan query tersebut menggunakan fungsi `mysqli_query()`, sehingga data yang sebelumnya berada di dalam variabel program dapat disimpan sebagai baris baru pada tabel database. Jika proses penyimpanan berhasil maka program menampilkan pesan bahwa data telah ditambahkan, sedangkan jika gagal maka ditampilkan pesan kesalahan dari database.

1.4. Percobaan Keempat—Menampilkan Data Tabel (Select).

1.4.1. *Source Code:*

```
<table border=6>
  <tr>
    <th>ID</th>
    <th>Nama</th>
    <th>Sektor</th>
    <th>Jenis</th>
  </tr>
<?php
include 'koneksi.php';
$result = mysqli_query($conn, "SELECT * FROM kurir");
while($data = mysqli_fetch_array($result))
  echo
  "<tr><td>$data[id]</td><td>$data[nama]</td><td>$data[sektor]</td><td>$data[jenis_barang]</td></tr>";
  ?>
</table<?php
include 'koneksi.php';
$sql = 'SELECT * FROM kurir';
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result) >0)
  while($row = mysqli_fetch_assoc($result))
    echo "ID: ". $row["id"]." - Nama: ". $row["nama"].".".$row["sektor"]."." | Jenis Argo: ".$row["jenis_barang"]."<br>";
else
  echo "Tidak ada data";
?>
```

1.4.2. Output:



1.4.3. Penjelasan:

Percobaan ini bertujuan memahami cara mengambil data dari database menggunakan perintah SQL SELECT. Query `SELECT * FROM kurir` digunakan untuk mengambil seluruh data yang ada di dalam tabel kurir. Hasil query disimpan dalam variabel `$result`, kemudian diperiksa jumlah barisnya menggunakan fungsi `mysqli_num_rows()`. Jika terdapat data, maka data tersebut diambil satu per satu menggunakan `mysqli_fetch_assoc()` yang mengubah setiap baris menjadi array associative sehingga kolom dapat diakses menggunakan nama kolomnya. Data kemudian ditampilkan ke layar menggunakan `echo`. Percobaan ini menunjukkan bagaimana data dari database dapat diolah dan ditampilkan oleh program PHP.

1.5. Percobaan Kelima—Menampilkan Data dalam Tabel HTML.

1.5.1. Source Code:

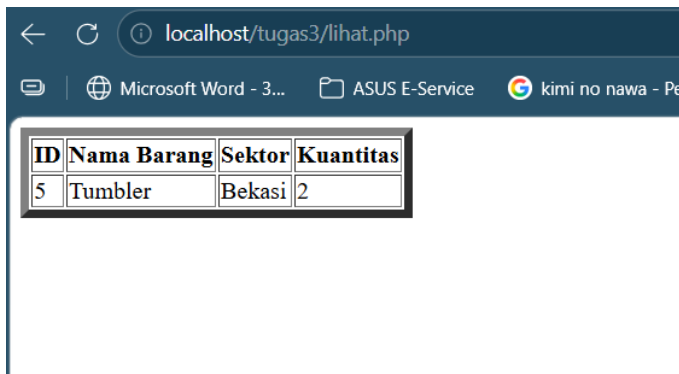
```
<table border=6>
  <tr>
    <th>ID</th>
    <th>Nama</th>
    <th>Sektor</th>
    <th>Jenis</th>
  </tr>
<?php
```

```

include 'koneksi.php';
$result = mysqli_query($conn, "SELECT * FROM kurir");
while($data = mysqli_fetch_array($result))
    echo
    "<tr><td>$data[id]</td><td>$data[nama]</td><td>$data[sektor]</td><td>$data[jenis_barang]</td></tr>";
    ?>
</table>

```

1.5.2. Output:



1.5.3. Penjelasan:

Pada percobaan ini data yang diambil dari database tidak lagi ditampilkan dalam bentuk teks sederhana, tetapi disusun dalam bentuk tabel HTML agar lebih rapi dan mudah dibaca oleh pengguna. Script PHP melakukan query SELECT untuk mengambil data dari tabel kurir, kemudian menggunakan perulangan while untuk menampilkan setiap baris data sebagai baris tabel HTML (<tr>). Nilai dari setiap kolom dimasukkan ke dalam elemen <td> sehingga membentuk struktur tabel yang terorganisir.

1.6. Percobaan Keenam—Input Data via Form HTML.

1.6.1. Source Code:

tambah.php:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>FORM KIRIM</title>

```

```

</head>
<body>
  <form action="simpan.php" method="POST">
    <label for="item_Name">Nama Barang:</label><br>
    <input id="item_Name" type="text" name="item_Name" required/><br>
    <label for="sector">Sektor:</label><br>
    <input id="sector" type="text" name="sector" required/><br>
    <label for="quantity">Kuantitas:</label><br>
    <input id="quantity" type="number" name="quantity" required/><br>
    <button type="submit" name="proses">Simpan Barang</button>
  </form>
</body>
</html>

```

simpan.php:

```

<?php
include 'koneksi_order.php';
if(isset($_POST['proses'])){
    $nama_barang = $_POST['item_Name'];
    $sektor = $_POST['sector'];
    $kuantitas = $_POST['quantity'];
    $sql = "INSERT INTO order_barang (nama_barang, sektor, kuantitas) VALUES ('$nama_barang', '$sektor', '$kuantitas')";
    if(mysqli_query($conn, $sql))
        echo "Berhasil Menambahkan!<br> <a href='tambah.php'>Kembali ke Form</a> || <a href='lihat.php'>Lihat Order Barang</a>";
    else
        echo "Error db: ".mysqli_error($conn);
}
?>

```

1.6.2. Output:

localhost/tugas

Microsoft Word - 3...

Nama Barang:
Tumbler

Sektor:
Surabaya

Kuantitas:
7

Simpan Barang

1.6.3. Penjelasan:

Percobaan ini menjelaskan bagaimana data dari pengguna dapat dikirim dari halaman web ke server menggunakan form HTML. Halaman `form_tambah.php` berfungsi sebagai antarmuka bagi pengguna untuk mengisi data seperti nama barang, stok, dan harga. Ketika tombol simpan ditekan, data tersebut dikirim ke file `simpan_barang.php` menggunakan metode POST, sehingga data tidak terlihat pada URL. Script PHP kemudian menangkap data tersebut menggunakan variabel superglobal `$_POST`, lalu memasukkannya ke dalam database menggunakan perintah SQL INSERT.

1.7. Percobaan Ketujuh—Menambahkan Fitur Hapus Data.

1.7.1. *Source Code:*

lihat.php:

```
<table border=6>
  <tr>
    <th>ID</th>
    <th>Nama Barang</th>
    <th>Sektor</th>
    <th>Kuantitas</th>
  </tr>
<?php
include 'koneksi_order.php';
$result = mysqli_query($conn, "SELECT * FROM order_barang");
while($data = mysqli_fetch_array($result))
  echo
  "<tr><td>$data[id]</td><td>$data[nama_barang]</td><td>$data[sektor]</td><td>$data[kuantitas]</td><td>
    <a href='hapus.php?id=$data[id]'
    onclick=\"return confirm('Yakin ingin menghapus data ini?')\">
    Hapus
    </a>
  </td></tr>";
?>
</table>
```

hapus.php:

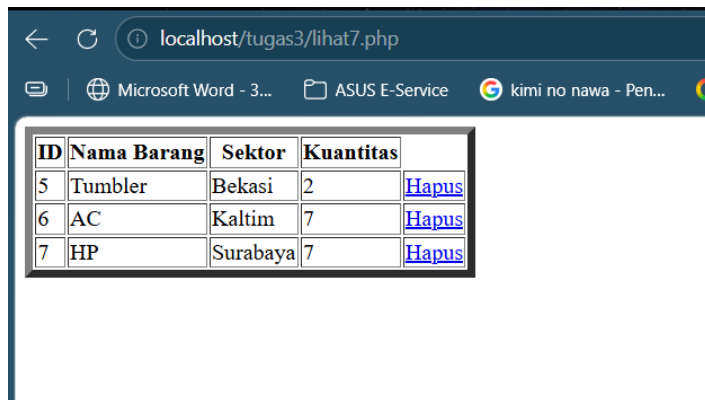
```
<?php
include 'koneksi_order.php';
if(isset($_GET['id'])){
  $id = $_GET['id'];
  $sql = "DELETE FROM order_barang WHERE id='$id'";
```

```

if(mysqli_query($conn, $sql)){
    header("Location: lihat.php");
}
else{
    echo "Gagal menghapus data: ".mysqli_error($conn);
}
}
else{
    echo "Akses tidak valid!";
}
?>

```

1.7.2. Output:



ID	Nama Barang	Sektor	Kuantitas	
5	Tumbler	Bekasi	2	Hapus
6	AC	Kaltim	7	Hapus
7	HP	Surabaya	7	Hapus

1.7.3. Penjelasan:

Percobaan ini menambahkan kemampuan untuk menghapus data dari database. Pada halaman `lihat_barang.php` ditambahkan kolom aksi yang berisi tautan menuju file `hapus_barang.php` dengan membawa parameter `id` melalui URL menggunakan metode GET. File `hapus_barang.php` kemudian mengambil nilai `id` tersebut menggunakan `$_GET` dan menjalankan query `DELETE FROM barang WHERE id = ...` untuk menghapus data yang sesuai. Klausula `WHERE` sangat penting karena tanpa klausula tersebut seluruh data dalam tabel dapat terhapus. Setelah proses penghapusan berhasil, fungsi `header()` digunakan untuk mengarahkan kembali pengguna ke halaman daftar data.

1.8. Percobaan Kedelapan—Fitur Update (Edit) Data

1.8.1. *Source Code:*

lihat.php:

```
<table border=6>
  <tr>
    <th>ID</th>
    <th>Nama Barang</th>
    <th>Sektor</th>
    <th>Kuantitas</th>
    <th>Aksi</th>
  </tr>

<?php
include 'koneksi_order.php';
$result = mysqli_query($conn, "SELECT * FROM order_barang");
while($data = mysqli_fetch_array($result)){
  echo "<tr>
    <td>$data[id]</td>
    <td>$data[nama_barang]</td>
    <td>$data[sektor]</td>
    <td>$data[kuantitas]</td>
    <td>
      <a href='edit.php?id=$data[id]'>Edit</a> |
      <a href='hapus.php?id=$data[id]'
      onclick=\"return confirm('Yakin ingin menghapus data ini?')\">
      Hapus
    </a>
    </td>
  </tr>";
}
?>
</table>
```

edit.php:

```
<?php
include 'koneksi_order.php';
$id = $_GET['id'];
$query = mysqli_query($conn, "SELECT * FROM order_barang WHERE id='$id'");
$data = mysqli_fetch_array($query);
if(isset($_POST['update'])){
  $nama_barang = $_POST['nama_barang'];
  $sektor = $_POST['sektor'];
  $kuantitas = $_POST['kuantitas'];
  $sql = "UPDATE order_barang
    SET nama_barang='$nama_barang',
        sektor='$sektor',
        kuantitas='$kuantitas'
```

```

        WHERE id='$id';
    if(mysqli_query($conn, $sql)){
        header("Location: lihat.php");
    }
    else{
        echo "Gagal update: ".mysqli_error($conn);
    }
}
?>
<h2>Edit Order Barang</h2>
<form method="POST">
Nama Barang : <br>
<input type="text" name="nama_barang" value="<?php echo $data['nama_barang']; ?>">
<br><br>
Sektor : <br>
<input type="text" name="sektor" value="<?php echo $data['sektor']; ?>">
<br><br>
Kuantitas : <br>
<input type="number" name="kuantitas" value="<?php echo $data['kuantitas']; ?>">
<br><br>
<button type="submit" name="update">Simpan Perubahan</button>
<a href="lihat.php">Batal</a>
</form>

```

1.8.2. Output:

ID	Nama Barang	Sektor	Kuantitas	Aksi
5	Tumbler	Bekasi	2	Edit Hapus
6	AC	Kaltim	7	Edit Hapus
7	HP	Surabaya	7	Edit Hapus

1.8.3. Penjelasan:

Percobaan ini memperkenalkan cara memperbarui data yang sudah ada di dalam database. Pada halaman daftar data ditambahkan tombol Edit yang mengirimkan id data ke halaman edit_barang.php. Script tersebut mengambil data berdasarkan id menggunakan query SELECT, kemudian menampilkan data tersebut di dalam form HTML dengan mengisi nilai awal pada atribut value. Pengguna dapat mengubah data tersebut lalu menekan tombol update, sehingga data yang telah diperbarui dikirim kembali ke server menggunakan metode POST. PHP kemudian menjalankan query UPDATE barang SET ... WHERE id=... untuk mengganti data lama dengan data yang baru.

BAB II

TUGAS PRAKTIKUM

2.1. Aplikasi Perpustakaan dengan DB

2.1.1. *Source Code:*

koneksi.php:

```
<?php
$host = 'localhost';
$user = 'root';
$pass = '128206';
$db = 'db_perpus';
$conn = mysqli_connect($host, $user, $pass, $db);
if (!$conn) {
    die("Koneksi gagal: " . mysqli_connect_error());
}
?>
```

tambah.php:

```
<?php
?>
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>Tambah Buku - Perpustakaan</title>
</head>
<body>
    <h2>Tambah Buku</h2>
    <form action="simpan.php" method="post">
        <label>Judul:</label><br>
        <input type="text" name="judul" required maxlength="255"><br><br>
        <label>Pengarang:</label><br>
        <input type="text" name="pengarang" required maxlength="255"><br><br>
        <label>Tahun Terbit (YYYY):</label><br>
        <input type="number" name="tahun_terbit" required min="1000" max="9999"><br><br>
        <button type="submit" name="simpan">Simpan</button>
    </form>
    <p><a href="tampil.php">Lihat Data Buku</a></p>
</body>
</html>
```

simpan.php:

```
<?php
include 'koneksi.php';
if (isset($_POST['simpan'])) {
    $judul = trim($_POST['judul']);
```

```

$pengarang = trim($_POST['pengarang']);
$tahun = intval($_POST['tahun_terbit']);
if ($judul === "" || $pengarang === "" || $tahun < 1000 || $tahun > 9999) {
    die("Input tidak valid. <a href='tambah.php'>Kembali</a>");
}
$stmt = mysqli_prepare($conn, "INSERT INTO buku (judul, pengarang, tahun_terbit) VALUES (?, ?, ?)");
mysqli_stmt_bind_param($stmt, "ssi", $judul, $pengarang, $tahun);
if (mysqli_stmt_execute($stmt)) {
    mysqli_stmt_close($stmt);
    header("Location: tampil.php?pesan=tertambah");
    exit;
} else {
    echo "Gagal menyimpan: " . mysqli_error($conn);
}
} else {
    header("Location: tambah.php");
    exit;
}
?>

```

hapus.php:

```

<?php
include 'koneksi.php';
if (isset($_GET['id'])) {
    $id = intval($_GET['id']);
    $stmt = mysqli_prepare($conn, "DELETE FROM buku WHERE id = ?");
    mysqli_stmt_bind_param($stmt, "i", $id);
    if (mysqli_stmt_execute($stmt)) {
        mysqli_stmt_close($stmt);
        header("Location: tampil.php?pesan=terhapus");
        exit;
    } else {
        echo "Gagal menghapus: " . mysqli_error($conn);
    }
} else {
    die("Akses dilarang.");
}
?>

```

edit.php:

```

<?php
include 'koneksi.php';
if (!isset($_GET['id'])) {
    die("ID tidak ditemukan. <a href='tampil.php'>Kembali</a>");
}
$id = intval($_GET['id']);
$stmt = mysqli_prepare($conn, "SELECT judul, pengarang, tahun_terbit FROM buku WHERE id = ?");
mysqli_stmt_bind_param($stmt, "i", $id);
mysqli_stmt_execute($stmt);
mysqli_stmt_bind_result($stmt, $judul, $pengarang, $tahun);

```

```

if (!mysqli_stmt_fetch($stmt)) {
    mysqli_stmt_close($stmt);
    die("Data tidak ditemukan. <a href='tampil.php'>Kembali</a>");
}
mysqli_stmt_close($stmt);
?>
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>Edit Buku</title>
</head>
<body>
    <h2>Edit Buku</h2>
    <form action="update.php" method="post">
        <input type="hidden" name="id" value="<?php echo htmlspecialchars($id); ?>">
        <label>Judul:</label><br>
        <input type="text" name="judul" required maxlength="255" value="<?php echo
htmlspecialchars($judul); ?>"><br><br>
        <label>Pengarang:</label><br>
        <input type="text" name="pengarang" required maxlength="255" value="<?php echo
htmlspecialchars($pengarang); ?>"><br><br>
        <label>Tahun Terbit (YYYY):</label><br>
        <input type="number" name="tahun_terbit" required min="1000" max="9999" value="<?php echo
htmlspecialchars($tahun); ?>"><br><br>
        <button type="submit" name="update">Simpan Perubahan</button>
        <a href="tampil.php">Batal</a>
    </form>
</body>
</html>

```

tampil.php:

```

<?php
include 'koneksi.php';
$keyword = "";
if (isset($_GET['q'])) {
    $keyword = trim($_GET['q']);
}
?>
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>Daftar Buku - Perpustakaan</title>
</head>
<body>
    <h2>Daftar Buku</h2>
    <?php if (isset($_GET['pesan']) && $_GET['pesan'] == 'terhapus'): ?>
        <p style="color:green;">Data berhasil dihapus.</p>
    <?php elseif (isset($_GET['pesan']) && $_GET['pesan'] == 'tertambah'): ?>

```

```

<p style="color:green;">Data berhasil ditambahkan.</p>
<?php elseif (isset($_GET['pesan']) && $_GET['pesan'] == 'terupdate'): ?>
    <p style="color:green;">Data berhasil diperbarui.</p>
<?php endif; ?>
<form method="get" action="tampil.php">
    <input type="text" name="q" placeholder="Cari judul buku..." value="<?php echo
htmlspecialchars($keyword); ?>">
    <button type="submit">Cari</button>
    <a href="tambah.php">+ Tambah Buku</a>
</form>
<br>
<table border="1" cellpadding="8">
    <tr>
        <th>ID</th>
        <th>Judul</th>
        <th>Pengarang</th>
        <th>Tahun Terbit</th>
        <th>Aksi</th>
    </tr>
<?php
if ($keyword != "") {
    $like = "%{$keyword}%";
    $stmt = mysqli_prepare($conn, "SELECT id, judul, pengarang, tahun_terbit FROM buku WHERE judul LIKE ? ORDER BY
id DESC");
    mysqli_stmt_bind_param($stmt, "s", $like);
    mysqli_stmt_execute($stmt);
    mysqli_stmt_bind_result($stmt, $id, $judul, $pengarang, $tahun);
    while (mysqli_stmt_fetch($stmt)) {
        echo "<tr>
            <td>" . htmlspecialchars($id) . "</td>
            <td>" . htmlspecialchars($judul) . "</td>
            <td>" . htmlspecialchars($pengarang) . "</td>
            <td>" . htmlspecialchars($tahun) . "</td>
            <td>
                <a href='edit.php?id=" . urlencode($id) . "'>Edit</a> |
                <a href='hapus.php?id=" . urlencode($id) . "' onlick=\\"return confirm(\"Yakin ingin
menghapus?\"\\")>Hapus</a>
            </td>
        </tr>";
    }
    mysqli_stmt_close($stmt);
} else {
    $query = "SELECT id, judul, pengarang, tahun_terbit FROM buku ORDER BY id DESC";
    if ($res = mysqli_query($conn, $query)) {
        while ($row = mysqli_fetch_assoc($res)) {
            echo "<tr>
                <td>" . htmlspecialchars($row['id']) . "</td>
                <td>" . htmlspecialchars($row['judul']) . "</td>
                <td>" . htmlspecialchars($row['pengarang']) . "</td>

```

```

        <td>" . htmlspecialchars($row['tahun_terbit']) . "</td>
        <td>
            <a href='edit.php?id=" . urlencode($row['id']) . "'>Edit</a> |
            <a href='hapus.php?id=" . urlencode($row['id']) . "' onclick=\"return confirm('Yakin ingin
menghapus?')\">Hapus</a>
        </td>
    </tr>";
}
mysqli_free_result($res);
} else {
    echo "<tr><td colspan='5'>Error: " . mysqli_error($conn) . "</td></tr>";
}
}
?>
</table>
</body>
</html>

```

update.php:

```

<?php
include 'koneksi.php';
if (isset($_POST['update'])) {
    $id = intval($_POST['id']);
    $judul = trim($_POST['judul']);
    $pengarang = trim($_POST['pengarang']);
    $tahun = intval($_POST['tahun_terbit']);
    if ($id <= 0 || $judul === "" || $pengarang === "" || $tahun < 1000 || $tahun > 9999) {
        die("Input tidak valid. <a href='tampil.php'>Kembali</a>");
    }
    $stmt = mysqli_prepare($conn, "UPDATE buku SET judul = ?, pengarang = ?, tahun_terbit = ? WHERE id = ?");
    mysqli_stmt_bind_param($stmt, "ssii", $judul, $pengarang, $tahun, $id);
    if (mysqli_stmt_execute($stmt)) {
        mysqli_stmt_close($stmt);
        header("Location: tampil.php?pesan=terupdate");
        exit;
    } else {
        echo "Gagal update: " . mysqli_error($conn);
    }
} else {
    header("Location: tampil.php");
    exit;
}
?>

```

Output:

localhost/tugas3/prak/tam

Tambah Buku

Judul:

Pengarang:

Tahun Terbit (YYYY):

[Lihat Data Buku](#)

Figure 2 Form Tambah

localhost/tugas3/prak/edit.php?id=1

Edit Buku

Judul:

Pengarang:

Tahun Terbit (YYYY):

Figure 1 Form Edit

localhost/tugas3/prak/tampil.php?pesan=tertambah

Daftar Buku

Data berhasil ditambahkan.

[+ Tambah Buku](#)

ID	Judul	Pengarang	Tahun Terbit	Aksi
3	Madilog	Tan Malaka	1999	Edit Hapus
2	Madilog	Tan Malaka	1999	Edit Hapus
1	Laut Bercerita	Chidor	2021	Edit Hapus

Figure 3 Form Daftar

localhost/tugas3/prak/tampil.php?q=Laut

Daftar Buku

[+ Tambah Buku](#)

ID	Judul	Pengarang	Tahun Terbit	Aksi
1	Laut Bercerita	Chidor	2021	Edit Hapus

Figure 4 Form Search/Like

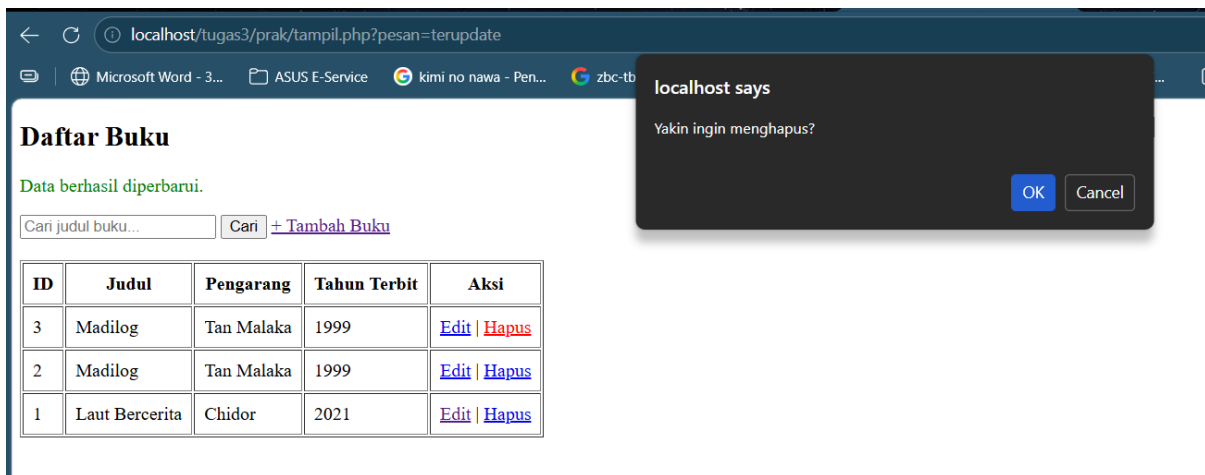


Figure 5 Form Hapus

2.1.2. Penjelasan

Pada tugas praktikum ini diterapkan seluruh konsep yang telah dipelajari pada percobaan sebelumnya untuk membangun sebuah aplikasi sederhana bernama Sistem Manajemen Perpustakaan. Proses dimulai dengan pembuatan database db_perpus yang memiliki tabel buku dengan beberapa atribut utama, yaitu id, judul, pengarang, dan tahun_terbit. Setelah struktur database tersedia, kemudian dibuat beberapa file PHP yang masing-masing memiliki fungsi berbeda, seperti melakukan koneksi ke database, menambahkan data buku baru, menampilkan daftar buku, mengedit data yang sudah ada, serta menghapus data dari tabel. Selain itu, pada halaman tampilan data juga ditambahkan fitur pencarian sederhana berdasarkan judul buku menggunakan perintah SQL LIKE, sehingga data buku dapat ditemukan dengan memasukkan kata kunci tertentu. Melalui implementasi ini, seluruh operasi dasar pengolahan data dalam database, yaitu CRUD (Create, Read, Update, Delete), dapat dijalankan dalam satu sistem aplikasi web sederhana yang terhubung dengan database MySQL.

DOKUMENTASI

naftali.it.student.pens.ac.id