

LAPORAN PRAKTIKUM CRUD DAN LOGIN PHP NATIVE



NAMA ANGGOTA KELOMPOK:

ADNAN NOVALINO L.N (01)
BAGAS RYANDHIKA P (06)

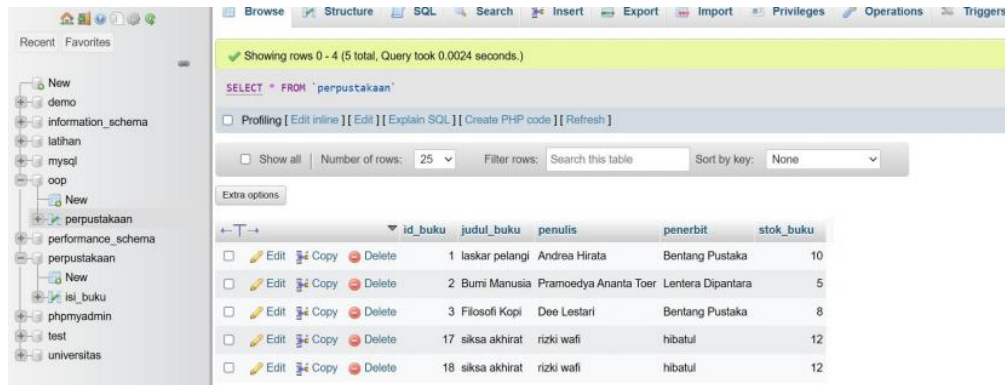
REKAYASA PERANGKAT LUNAK

SMK TEXMACO SEMARANG

2024

I. TAHAPAN

1. MEMBUAT DATABASE PERPUSTAKAAN



2. membuat koneksi dengan nama file config.php

```
<?php
class Database {
    private $host = "localhost";
    private $db_name = "OOP1";
    private $username = "root";
    private $password = "";
    public $conn;

    public function getConnection() {
        $this->conn = null;
        try {
            $this->conn = new PDO("mysql:host=" . $this->host . ";dbname=" . $this->db_name, $this->username,
            $this->password);
            $this->conn->exec("set names utf8");
        } catch(PDOException $exception) {
            echo "Connection error:" . $exception->getMessage();
        }
        return $this->conn;
    }
}
```

- Fungsi: Mengelola koneksi ke database MySQL dengan PDO.
- Metode `getConnection`: Membuat dan mengembalikan koneksi; menangani kesalahan.

Kegunaan

1. Akses Database: Memudahkan interaksi dengan database.
2. Keamanan: Melindungi dari SQL injection.
3. Modularitas: Kode lebih terstruktur dan mudah dipelihara.

3. menampilkan data/output

```
<?php
include 'config/config.php';
include 'classes/perpustakaan.php';
$db = new Database();
$perpustakaan = new perpustakaan($db->getConnection());
?>
<script src="https://cdn.tailwindcss.com"></script>
<section class="antialiased bg-gray-100 text-gray-600 h-screen px-4">
  <div class="flex flex-col justify-center h-full">
    <!-- Table -->
    <div class="w-full max-w-2xl mx-auto bg-white shadow-lg rounded-sm border border-gray-
200">
      <header class="px-5 py-4 border-b border-gray-100 flex justify-between">
        <h2 class="font-semibold text-gray-800">Data perpustakaan</h2>
        <a href="./create.php" class="mr-3 text-sm bg-blue-500 hover:bg-blue-700 text-
white py-1 px-2 rounded focus:outline-none focus:shadow-outline">Add Data</a>
      </header>
      <div class="p-3">
        <div class="overflow-x-auto">
          <table class="table-auto w-full">
            <thead class="text-xs font-semibold uppercase text-gray-400 bg-gray-
50">
              <tr>
                <th class="p-2 whitespace-nowrap">
                  <div class="font-semibold text-left">id_buku</div>
                </th>
                <th class="p-2 whitespace-nowrap">
                  <div class="font-semibold text-left">judul_buku</div>
                </th>
                <th class="p-2 whitespace-nowrap">
                  <div class="font-semibold text-center">penulis</div>
                </th>
                <th class="p-2 whitespace-nowrap">
                  <div class="font-semibold text-left">penerbit</div>
                </th>
                <th class="p-2 whitespace-nowrap">
                  <div class="font-semibold text-left">stok_buku</div>
                </th>
              </tr>
            </thead>
            <tbody class="text-sm divide-y divide-gray-100">
              <?php
              $no = 1;
              foreach($perpustakaan->read() as $x){
                $id_buku = $x["id_buku"];
                $judul_buku = $x["judul_buku"];
                $penulis = $x["penulis"];
                $penerbit = $x["penerbit"];
                $stok_buku = $x["stok_buku"];
                $f_stok_buku = number_format($x["stok_buku"]);
                $dimas = ["IMG-20241004-WA0001.jpg", "IMG-20241004-WA0004.jpg",
"SmartSelect_20241018_080405_Photos.jpg", "SmartSelect_20241018_080442_Photos.jpg",
"SmartSelect_20241018_080620_Photos.jpg"];
                $farrel = $dimas[array_rand($dimas)];

                echo "
              <tr>
                <td class='p-2 whitespace-nowrap'>
                  <div class='text-left'>$id_buku</div>






```

```

        </td>
        <td class='p-2 whitespace-nowrap'>
            <div class='flex items-center'>
                <div class='w-10 h-10 flex-shrink-0 mr-2 sm:mr-3'><img class='rounded-full' src='https://raw.githubusercontent.com/Nopalaryan/A/main/$farrel'
width='40' height='40' alt='$judul_buku'></div>
                <div class='font-medium text-gray-800'>$judul_buku</div>
            </div>
        </td>
        <td class='p-2 whitespace-nowrap'>
            <div class='text-left'>$penulis</div>
        </td>
        <td class='p-2 whitespace-nowrap'>
            <div class='text-left'>$penerbit</div>
        </td>
        <td class='p-2 whitespace-nowrap'>
            <div class='text-left font-medium'>$f_stok_buku</div>
        </td>
        <td class='p-2 whitespace-nowrap'>
            <div class='text-lg text-center'>
                </div>
        </td>
        <td class='p-3 px-5 flex justify-end'><a
href=\"./edit.php?id_buku=\" . $id_buku . "&judul_buku=\" . $judul_buku . "&penulis=\" .
$penulis . "&penerbit=\" . $penerbit . "&stok_buku=\" . $stok_buku . "\"" type='button'
class='mr-3 text-sm bg-blue-500 hover:bg-blue-700 text-white py-1 px-2 rounded focus:outline-
none focus:shadow-outline'>Edit</a><button onclick=\"if (confirm('yakin hapus ini?'))
window.location.href = './delete_action.php?id_buku=\" . $id_buku . "\";\" type='button'
class='text-sm bg-red-500 hover:bg-red-700 text-white py-1 px-2 rounded focus:outline-none
focus:shadow-outline'>Delete</button></td>
    </tr>";
    }
    ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</section>

```

Kode ini menciptakan antarmuka web untuk mengelola data buku di perpustakaan. Pertama, kode ini mengimpor file konfigurasi untuk menghubungkan ke database dan kelas `perpustakaan` untuk mengakses data buku. Setelah membuat objek `Database` dan `perpustakaan`, tampilan HTML dibangun menggunakan Tailwind CSS untuk menampilkan tabel dengan informasi buku, seperti ID, judul, penulis, penerbit, dan stok. Data buku diambil dari metode `read()` dalam kelas `perpustakaan`, dan setiap buku ditampilkan dengan gambar acak. Terdapat juga tombol untuk menambah, mengedit, dan menghapus data buku, dengan konfirmasi untuk tindakan penghapusan. Secara keseluruhan, kode ini menyediakan antarmuka pengguna yang fungsional dan mudah digunakan untuk mengelola koleksi buku di perpustakaan.

Data Perpustakaan					Tambah Data	
ID BUKU	JUDUL BUKU	PENULIS	PENERBIT	STOK BUKU		
1	 laskar pelangi	Andrea Hirata	Bentang Pustaka	10	Edt	Delete
2	 Bumi Manusia	Pramoedya Ananta Toer	Lentera Dipantara	5	Edt	Delete
3	 Filosofi Kopi	Dee Lestari	Bentang Pustaka	8	Edt	Delete
17	 siksa akhirat		hibatul	12	Edt	Delete
19	 shumala	bagas	adnan	4	Edt	Delete

4. tambah data

- create.php

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Tambahkan Buku</title>
<style>
body {
font-family: Arial, sans-serif;
background-color: #f4f4f4;
margin: 0;
padding: 20px;
}
h1 {
color: #333;
}
form {
background: white;
padding: 20px;
border-radius: 5px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
max-width: 400px;
margin: auto;
}
label {
display: block;
margin-bottom: 5px;
font-weight: bold;
}
input[type="text"],
input[type="number"] {
width: 100%;
padding: 10px;
margin-bottom: 10px;
border: 1px solid #ccc;
border-radius: 3px;
}
```

```

input[type="submit"] {
    background-color: #5cb85c;
    color: white;
    border: none;
    padding: 10px 15px;
    border-radius: 3px;
    cursor: pointer;
    font-size: 16px;
}
input[type="submit"]:hover {
    background-color: #4cae4c;
}
</style>
</head>
<body>
<h1>Mendaftarkan Buku</h1>
<form method="POST" action="./create_action.php">
<label for="judul_buku">Judul Buku:</label>
<input type="text" name="judul_buku" id="judul_buku" required>
<label for="penulis">Penulis:</label>
<input type="text" name="penulis" id="penulis" required>
<label for="penerbit">Penerbit:</label>
<input type="text" name="penerbit" id="penerbit" required>
<label for="stok_buku">Stock Buku:</label>
<input type="number" name="stock_buku" id="stock_buku" required min="1">
<input type="submit" value="Kirim">
</form>
</body>
</html>

```

secara keseluruhan, kode HTML ini berfungsi sebagai formulir pendaftaran buku yang sederhana namun efektif. Dengan penggunaan elemen yang tepat, pengaturan atribut, dan styling CSS yang baik, formulir ini siap digunakan untuk mengumpulkan data buku dari pengguna

- create-action.php

```

<?php
include_once 'config/config.php';
include_once 'classes/perpustakaan.php';
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $judul_buku = $_POST['judul_buku'];
    $penulis = $_POST['penulis'];
    $penerbit = $_POST['penerbit'];
    $stock_buku = (int)$_POST['stock_buku'];
    $db = new Database();
    $perpustakaan = new Perpustakaan($db->getConnection());
    if ($perpustakaan->create($judul_buku, $penulis, $penerbit, $stock_buku)) {
        echo "<h1>Data Berhasil Dikirim</h1> <br>";
    } else {
        echo "<h1>Data Gagal Dikirim</h1> <br>";
    }
    echo "-<a href='/perpustakaan/'>Cek List Data</a> <br>";
    echo "-<a href='/perpustakaan/create.php'>Masukkan Data Lainnya</a>";
} else {
    echo "Tidak ada data yang dikirim.";
}
?>

```

Kode PHP ini berfungsi untuk memproses data dari formulir pendaftaran buku. Pertama, ia mengimpor konfigurasi dan kelas `Perpustakaan`. Kemudian, ia memeriksa apakah data dikirim melalui metode POST. Jika ya, kode ini mengambil

input dari formulir, seperti judul, penulis, penerbit, dan stok buku. Selanjutnya, ia membuat koneksi ke database dan menggunakan metode `create` untuk menyimpan data tersebut. Setelah itu, kode menampilkan pesan sukses jika data berhasil disimpan, atau pesan kesalahan jika gagal. Tautan juga disediakan untuk memeriksa daftar buku atau memasukkan data baru. Jika tidak ada data yang dikirim, pesan yang sesuai ditampilkan. Secara keseluruhan, kode ini efektif untuk menangani pendaftaran buku dan memberikan umpan balik kepada pengguna.

- output tambah data

Mendaftarkan Buku

Judul Buku:







Penulis:

Penerbit:

Stock Buku:

Kirim

- hasil output

Data Perpustakaan					Tambah Data	
ID BUKU	JUDUL BUKU	PENULIS	PENERBIT	STOK BUKU		
1	 laskar pelangi	Andrea Hirata	Bentang Pustaka	10	Edt	Delete
2	 Bumi Manusia	Pramoedya Ananta Toer	Lentera Dipantara	5	Edt	Delete
3	 Filosofi Kopi	Dee Lestari	Bentang Pustaka	8	Edt	Delete
17	 siksa akhirat		hibatul	12	Edt	Delete
19	 shumala	bagas	adnan	4	Edt	Delete
20	 bawang putih	vallen	arya	6	Edt	Delete

5. edit data

- edit.php

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Edit</title>
<style>
body {
font-family: Arial, sans-serif;
background-color: #f4f4f4;
margin: 0;
padding: 20px;
}
h1 {
color: #333;
}
form {
background: #fff;
padding: 20px;
border-radius: 5px;
box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
}
input[type="text"], input[type="number"] {
width: 100%;
padding: 10px;
margin: 10px 0;
border: 1px solid #ccc;
border-radius: 4px;
}
input[type="submit"] {
background-color: #5cb85c;
color: white;
padding: 10px;
border: none;
border-radius: 4px;
cursor: pointer;
width: 100%;
}
input[type="submit"]:hover {
background-color: #4cae4c;
}
</style>
</head>
<body>
<h1>Edit Data Peerpustakaan</h1>
<form action="edit_action.php" method="POST">
Id: <input type="number" name="id_buku" id="id" min="0" <?php $x = $_GET["id_buku"]; if
($x !==
null) echo "value='$x'"; ?>><br>
Judul_buku: <input type="text" name="judul_buku" id="judul_buku" <?php $x =
$_GET["judul_buku"]; if
($x !== null) echo "value='$x'"; ?>><br>
Penulis: <input type="text" name="penulis" id="penulis" <?php $x = $_GET["penulis"]; if
($x !==
null) echo "value='$x'"; ?>><br>
Penerbit: <input type="text" name="penerbit" id="penerbit" <?php $x = $_GET["penerbit"]; if
($x !==
null) echo "value='$x'"; ?>><br>
```



```

Stok Buku: <input type="number" name="stok_buku" id="stok_buku" min="1" <?php $x =
$_GET["stok_buku"]; if ($x !== null) echo "value='$x'"; ?>><br>
<input type="submit" value="Kirim">
</form>
</body>
</html>

```

Kode HTML ini menciptakan halaman untuk mengedit data buku di perpustakaan. Halaman ini memiliki struktur dasar HTML5 dengan meta tags untuk pengaturan karakter dan responsivitas. Judul halaman ditetapkan sebagai "Edit" dan dilengkapi dengan CSS untuk tampilan yang bersih. Formulir berisi input untuk ID buku, judul, penulis, penerbit, dan stok buku, dengan nilai yang diambil dari parameter URL jika tersedia, sehingga pengguna dapat melihat dan mengedit data yang ada. Terdapat tombol kirim yang mengarahkan data yang diedit ke `edit_action.php`. Secara keseluruhan, halaman ini memungkinkan pengguna untuk memperbarui informasi buku dengan mudah.

- edit_action.php

```

<?php
include_once 'config/config.php';
include_once 'classes/perpustakaan.php';

$databse = new Database();
$db = $databse->getConnection();
$perpustakaan = new perpustakaan ($db);
$perpustakaan->id_buku = $_POST['id_buku'];
$perpustakaan->judul_buku = $_POST['judul_buku'];
$perpustakaan->penulis = $_POST['penulis'];
$perpustakaan->penerbit = (int)$_POST['penerbit'];
$perpustakaan->stok_buku = (int)$_POST['stok_buku'];
$hasil = $perpustakaan->edit();
echo "<h1 align='center'>";
if($hasil) {
    echo "Data berhasil di edit.";
} else {
    echo "Data tidak bisa di edit.";
}
echo "<br>";
echo "<a href='dashboard.php'>Kembali</a>";
echo "</h1>";
?>

```

Kode PHP ini berfungsi untuk mengedit data buku di perpustakaan. Pertama, ia mengimpor konfigurasi database dan kelas `Perpustakaan`, kemudian membuat koneksi ke database. Selanjutnya, objek `Perpustakaan` diinstansiasi, dan data buku diambil dari permintaan POST untuk diisi ke dalam properti objek. Kode ini kemudian memanggil metode `edit()` untuk memperbarui informasi di database. Setelah itu, hasil pengeditan ditampilkan, baik sebagai pesan sukses atau gagal, serta tautan untuk kembali ke dashboard. Secara keseluruhan, kode ini memungkinkan pengguna untuk memperbarui informasi buku dan memberikan umpan balik tentang proses tersebut.

-tampil data edit

Edit Data Perpustakaan

Id:	<input type="text"/>
Judul_buku:	<input type="text"/>
Penulis:	<input type="text"/>
Penerbit:	<input type="text"/>
Stok Buku:	<input type="text"/>
<input type="button" value="Kirim"/>	

6. delete data

- delete.php

```
<?php
include_once 'config/config.php';
include_once 'classes/perpustakaan.php';

$database = new Database();
$db = $database->getConnection();
```

```
$sepatu = new perpustakaan ($db);
$sepatu->id_buku = $_GET["ID"];
```

```
$hasil = $sepatu->delete();
```

```
if ($hasil) header("Location: ./");
else echo "<h1>Data Gagal Dihapus<h1><br> - <a href='./\"/>
```

Kode PHP ini digunakan untuk menghapus data buku dari perpustakaan. Pertama, ia mengimpor konfigurasi database dan kelas 'Perpustakaan', kemudian membuat koneksi ke database. Setelah itu, objek 'Perpustakaan' diinstansiasi, dan ID buku diambil dari parameter URL. Kode ini kemudian memanggil metode 'delete()' untuk menghapus buku tersebut. Jika penghapusan berhasil, pengguna dialihkan ke halaman utama; jika gagal, pesan kesalahan ditampilkan dengan tautan untuk kembali ke beranda. Secara keseluruhan, kode ini mengelola proses penghapusan data buku dan memberikan umpan balik kepada pengguna.

- delete action.php

```
<?php
include_once 'config/config.php';
include_once 'classes/perpustakaan.php';

$database = new Database();
$db = $database->getConnection();
```

```
$sepatu = new perpustakaan ($db);
$sepatu->id_buku = $_GET['ID'];
```

```
$hasil = $sepatu->delete();
```

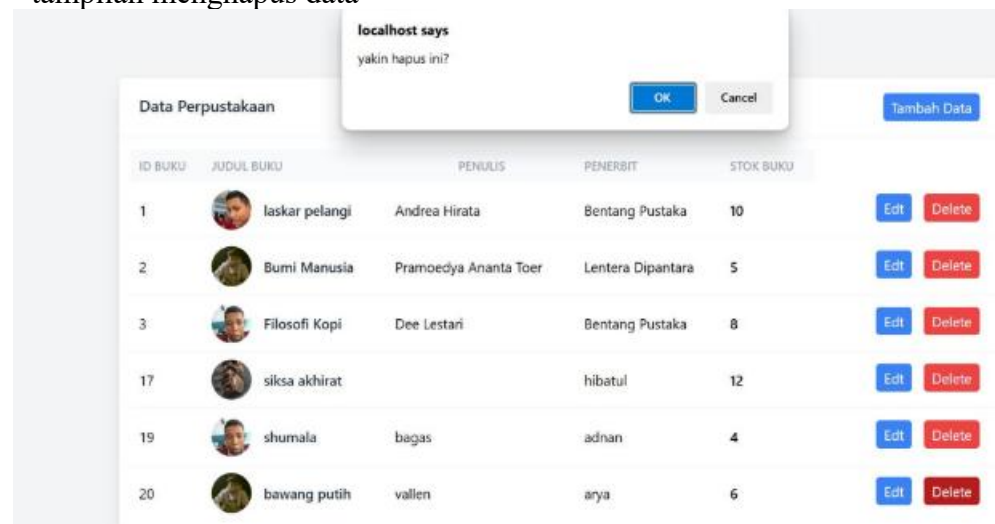
```

if($hasil) {
    header('location: ./');
} else {
    echo "<h1 align='center'>";
    echo "Data tidak bisa di hapus.";
    echo "</h1>";
    echo "<a href='dashboard.php'>Back Home</a>";
}
?>

```

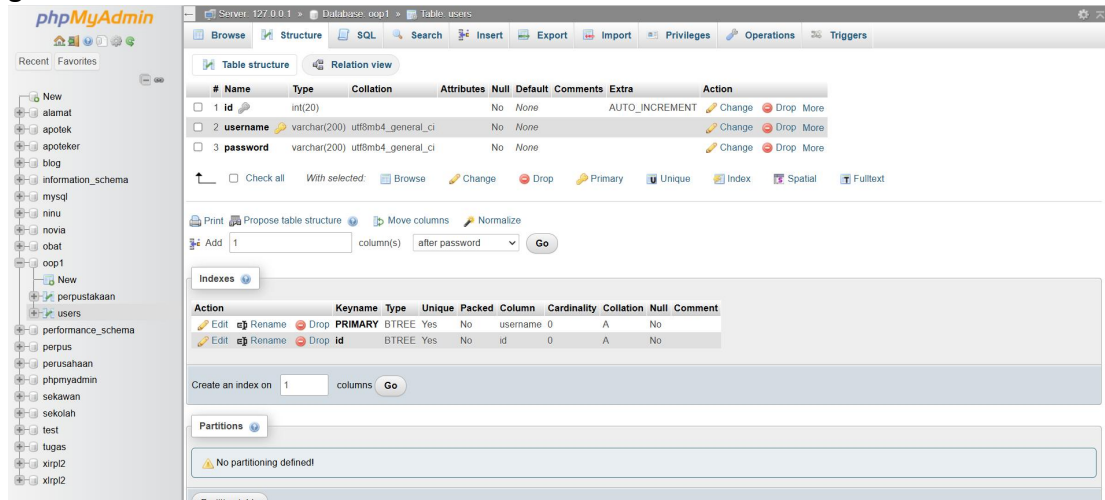
Kode PHP ini digunakan untuk menghapus data buku dari perpustakaan. Pertama, ia mengimpor konfigurasi database dan kelas `Perpustakaan`, lalu membuat koneksi ke database. Selanjutnya, ID buku diambil dari parameter URL, dan metode `delete()` dipanggil untuk menghapus buku tersebut. Jika penghapusan berhasil, pengguna akan dialihkan ke halaman utama; jika gagal, pesan kesalahan ditampilkan dengan tautan untuk kembali ke dashboard. Kode ini mengelola proses penghapusan buku dan memberi umpan balik kepada pengguna.

- tampilan menghapus data



II. Login

1. pertama tama buatlah sebuah database diphpMyAdmin/Laragon dengan nama OOP lalu buatlah sebuah table bernama kan perpustakaan dan users seperti di gambar dibawah ini



2. membuat koneksi php ke localhost

```
<?php
class Database {
    private $host = "localhost";
    private $db_name = "OOP1";
    private $username = "root";
    private $password = "";
    public $conn;

    public function getConnection() {
        $this->conn = null;
        try {
            $this->conn = new PDO("mysql:host=" . $this->host . ";dbname=" . $this->db_name, $this->username, $this->password);
            $this->conn->exec("set names utf8");
        } catch(PDOException $exception) {
            echo "Connection error:" . $exception->getMessage();
        }
        return $this->conn;
    }
}
```

Kode PHP ini mendefinisikan kelas `Database` yang digunakan untuk mengelola koneksi ke database MySQL. Kelas ini memiliki properti untuk host, nama database, username, password, dan objek koneksi (`\$conn`). Metode `getConnection()` mencoba membuat koneksi menggunakan PDO, mengatur encoding ke UTF-8, dan menangani kesalahan jika koneksi gagal. Metode ini mengembalikan objek koneksi yang dapat digunakan untuk menjalankan kueri database. Secara keseluruhan, kelas ini menyederhanakan dan mengelola proses koneksi ke database.

3. membuat file dashboard

```
<?php
include_once 'config/config.php';
include_once 'classes/perpustakaan.php';
$db = new Database();
$perpustakaan = new Perpustakaan($db->getConnection());
?>
<script src="https://cdn.tailwindcss.com"></script>
<section class="antialiased bg-gray-100 text-gray-600 h-screen px-4">
<div class="flex flex-col justify-center h-full">
<!-- Table -->
<div class="w-full max-w-4xl mx-auto bg-white shadow-lg rounded-sm border border-gray-200">
<header class="px-5 py-4 border-b border-gray-100 flex justify-between">
<h2 class="font-semibold text-gray-800">Data Perpustakaan</h2>
<a href="/create.php" class="mr-3 text-sm bg-blue-500 hover:bg-blue-700 text-white py-1
px-2 rounded focus:outline-none focus:shadow-outline">Tambah Data</a>
</header>
<div class="p-3">
<div class="overflow-x-auto">
<table class="table-auto w-full">
<thead class="text-xs font-semibold uppercase text-gray-400 bg-gray-50">
<tr>
<th class="p-2 whitespace-nowrap">
<div class="font-semibold text-left">Id buku</div>
</th>
<th class="p-2 whitespace-nowrap">
<div class="font-semibold text-left">Judul buku</div>
</th>
<th class="p-2 whitespace-nowrap">
<div class="font-semibold text-center">Penulis</div>
</th>
<th class="p-2 whitespace-nowrap">
<div class="font-semibold text-left">Penerbit</div>
</th>
```

```
<th class="p-2 whitespace-nowrap">
<div class="font-semibold text-left">Stok buku</div>
</th>
</tr>
</thead>
<tbody class="text-sm divide-y divide-gray-100">
<?php
$no = 1;
foreach($perpustakaan->read() as $x){
$id_buku = $x["id_buku"];
$judul_buku = $x["judul_buku"];
$penulis = $x["penulis"];
$penerbit= $x["penerbit"];
$stok_buku = $x["stok_buku"];
$f_stok_buku = number_format($x["stok_buku"]);
```

```
$dimas = ["https://raw.githubusercontent.com/Nopalaryan/A/main/IMG-
20241004-WA0001.jpg", "https://raw.githubusercontent.com/Nopalaryan/A/main/IMG-20241004-
WA0004.jpg",
```

```
"https://raw.githubusercontent.com/Nopalaryan/A/main/SmartSelect_20241018_080405_Photos.jpg",
"https://raw.githubusercontent.com/Nopalaryan/A/main/SmartSelect_20241018_080442_Photos.jpg",
"https://raw.githubusercontent.com/Nopalaryan/A/main/SmartSelect_20241018_080620_Photos.jpg"];
```

```

$farrel = $dimas[array_rand($dimas)];
echo "
<tr>
<td class='p-2 whitespace-nowrap'>
<div class='text-left'>$id_buku</div>
</td>
<td class='p-2 whitespace-nowrap'>
<div class='flex items-center'>
<div class='w-10 h-10 flex-shrink-0 mr-2 sm:mr-3'><img

class='rounded-full' src='$farrel' width='40' height='40' alt='$judul_buku'></div>

<div class='font-medium text-gray-800'>$judul_buku</div>
</div>
</td>
<td class='p-2 whitespace-nowrap'>
<div class='text-left'>$penulis</div>
</td>
<td class='p-2 whitespace-nowrap'>
<div class='text-left'>$penerbit</div>
</td>
<td class='p-2 whitespace-nowrap'>
<div class='text-left font-medium text-grey-500'>$f_stok_buku</div>
</td>
<td class='p-2 whitespace-nowrap'>
<div class='text-lg text-center'>
</div>
</td>
<td class='p-3 px-5 flex justify-end'><a href=\"./edit.php?id_buku=\" .

$id_buku . "&judul_buku=" . $judul_buku . "&penerbit=" . $penerbit . "&penulis=" . $penulis .
"&stok_buku=" . $stok_buku . "\"" type='button' class='mr-3 text-sm bg-blue-500 hover:bg-blue-
700 text-white
py-1 px-2 rounded focus:outline-none focus:shadow-outline'>Edt</a><button onclick=\"if
(confirm('yakin
hapus ini?')) window.location.href = './delete_action.php?id_buku=" . $id_buku . "';\"
type='button'

class='text-sm bg-red-500 hover:bg-red-700 text-white py-1 px-2 rounded focus:outline-none
focus:shadow-
outline'>Delete</button></td>

</tr>";
}
?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</section>

```

Kode PHP ini digunakan untuk menampilkan data perpustakaan dalam bentuk tabel dengan styling menggunakan Tailwind CSS. Pertama, kode ini mengimpor konfigurasi dan kelas `Perpustakaan` untuk mengelola koneksi database. Setelah membuat objek untuk mendapatkan data buku, halaman menampilkan judul dan tombol untuk menambah data. Menggunakan loop, kode ini menampilkan ID, judul, penulis, penerbit, dan stok buku, serta menampilkan gambar acak dari array yang telah disediakan. Selain itu, terdapat tombol untuk mengedit dan menghapus buku, yang dilengkapi dengan konfirmasi sebelum penghapusan. Secara keseluruhan, kode ini menyediakan antarmuka interaktif yang memungkinkan pengguna untuk mengelola data buku di perpustakaan.

4. membuat file user

```
<?php
// classes/User.php
require_once 'config/config.php';

class user {
    private $conn;
    private $table_name = "users";

    public function __construct ($db) {
        $this->conn = $db;
    }

    /**
     * mengecek kredensial login
     * @param string $username
     * @param string $password
     * @return boolean
     */

    public function login ($username, $password) {
        //Query untuk mengambil data user yang sesuai dengan username yang diinput
        $query = "SELECT * FROM " . $this->table_name . " WHERE username = :username LIMIT 1";
        //mempersiapkan statement untuk query diatas
        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(":username", $username);
        $stmt->execute();

        //jika ada data user yang sesuai dengan data username yang diinput
        if($stmt->rowCount() > 0) {
            //ambil data user yang sesuai dengan username yang diinput
            $row = $stmt->fetch(PDO::FETCH_ASSOC);
            //jika password yang diinput sesuai dengan password yang tersimpan di database
            if(password_verify($password, $row['password'])) {
                //kembalikan true
                return true;
            }
        }

        //kembalikan false jika tidak ada data user yang sesuai dengan username yang diinput
        //atau password yang diinput tidak sesuai dengan password yang tersimpan di database
        return false;
    }
}
```

Kode PHP ini mendefinisikan kelas ``User`` yang digunakan untuk mengelola autentikasi pengguna. Kelas ini memiliki properti untuk menyimpan koneksi database dan nama tabel pengguna ("users"). Konstruktor kelas menerima objek koneksi database. Metode ``login()`` mengambil username dan password, menyiapkan query untuk mencari pengguna berdasarkan username, dan memverifikasi password menggunakan ``password_verify()``. Jika autentikasi berhasil, metode ini mengembalikan ``true``; jika tidak, mengembalikan ``false``. Secara keseluruhan, kelas ini menyediakan fungsi untuk memverifikasi kredensial login pengguna dengan aman.

```
<?php
//register.php
require_once 'config/config.php';
require_once 'classes/user.php';

$database = new Database();
$db = $database->getConnection();
$user = new User($db);
```

```
if($user->register($username, $password)) {
    $success_message = "Registrasi berhasil silahkan login";
} else {
    $error_message = "Registrasi gagal. Username Mungkin sudah di gunakan.";
}
}
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTtWfspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">
  <title>Register</title>
</head>
<body>
  <div class="container mt-5">
    <div class="row justify-content-center">
      <div class="col-md-4">
        <h3 class="text-center">register</h3>
        <?php if (isset($error_message)): ?>
          <div class="alert alert-danger">
            <?php echo htmlspecialchars($error_message); ?>
          </div>
        <?php endif; ?>
        <form action="" method="POST">
          <div class="form-group mb-3">
            <label>Username</label>
```



```

        <input type="text" name="username" class="form-control" required>
    </div>
    <div class="form-group mb-3">
        <label>Password</label>
        <input type="password" name="password" class="form-control" required>
    </div>
    <button type="submit" class="btn btn-primary w-100">Daftar</button>
    <p class="text-center mt-3">sudah punya akun?
    <a href="index.php">login di sini</a>
    </form>
</div>
</div>
</div>
</body>
</html>

```

Kode PHP ini mengatur halaman registrasi pengguna dengan mengimpor konfigurasi dan kelas `User` untuk manajemen pendaftaran. Kode ini membuat koneksi ke database dan memeriksa permintaan POST untuk mengambil `username` dan `password`. Setelah itu, metode `register()` dipanggil untuk mencoba mendaftarkan pengguna, dan pesan sukses atau kesalahan ditampilkan berdasarkan hasilnya. Halaman ini menggunakan HTML dan Bootstrap untuk styling, menyediakan formulir untuk input, menampilkan pesan kesalahan jika registrasi gagal, serta menyertakan tautan untuk login bagi pengguna yang sudah memiliki akun. Secara keseluruhan, kode ini menyediakan antarmuka yang jelas untuk mendaftar pengguna baru dengan umpan balik yang sesuai.