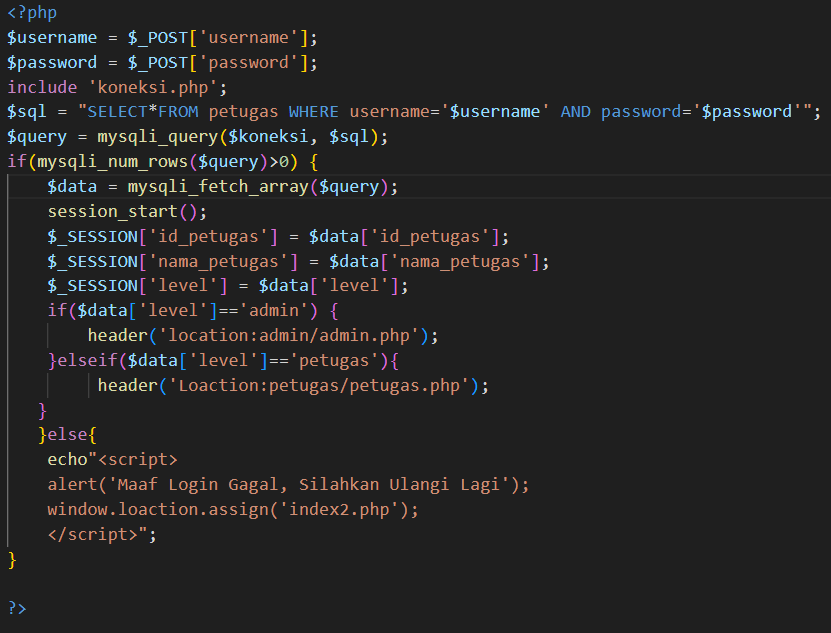
**Laporan Jurusan PPLG pak Sena **

Penjelasan Baris per Baris:

1. `<?php`

Membuka tag PHP untuk memulai script PHP.

1. `username’ = $\_POST[‘username’];`

Mengambil data ‘username’ dari form login yang menggunakan metode POST.

1. `$password = $\_POST[‘password’];`

Mengambil data ‘password’ dari form login yang menggunakan metode POST.

1. ‘include ‘koneksi.php’;`

Menyertakan file `koneksi.php` yang berisi koneksi ke database.

1. ‘$sql = “SELECT \* FROM petugas WHERE username=’$username’ AND password=’$password’ ”; ’

Membuat query SQL untuk mencari user di table ‘petugas’ yang username dan passwordnya sesuai dengan input dari form.

1. `$query = mysqli\_query($koneksi, $sql);`

Menjalankan query SQL yang sudah dibuat dan menyimpan hasilnya ke variable `$query`.

1. `if(mysqli\_num\_rows($query)>0) {

Mengecek Apakah baris data yang ditemukan dari hasil query.

1. `$data = mysqli\_fetch\_array($query);`

Mengambil data pengguna yang ditentukan dalam bentuk array.

1. `session\_start();

Memulai sesi untuk menyimpan data pengguna yang login.

1. `\_SESSION[‘id\_petugas’] = $data[‘id\_petugas’];`

Menyimpan ‘nama\_petugas’ pengguna ke dalam sesi.

1. `$\_SESSION[‘nama\_petugas’] = $data[‘nama\_petugas’];`

Menyimpan ‘nama\_petugas’ pengguna ke dalam sesi.

1. `$\_SESSION[‘level’] = $data[‘level’];`

Menyimpan ‘level’ pengguna ke dalam sesi.

1. `if($data[‘level’]==’admin’) {‘

Mengecek apakah level pengguna adalah’admin’.

1. `header(‘location:admin/admin.php’);

Jika level pengguna adalah ‘admin,’ mengarahkan penggguna ke halaman admin.

1. `} elseif($data[‘level’]==’petugas’){`

Mengecek apakah level pengguna adalah ‘petugas’.

1. `header(‘location:petugas/petugas.php’);`

Jika level pengguna adalah ‘petugas,’ mengarahkan pengguna ke halaman petugas.

1. `} else {`

Menutup kondisi if untuk pengecekan level pengguna.

1. `echo”<script>`

Jika login gagal, menampilkan pesan kesalahan menggunakan JavaScript.

1. `alert(‘Maaf Login Gagal, Silahkan Ulangi Lagi’);`

Menampilkan pesan kesalahan login.

1. `window.location.assign(‘index2.php’);`

Mengarahkan pengguna Kembali ke halaman login ‘index2.php’.

1. `</script>”;`

Menutup tag script.

1. `}`

Menutup kondisi if untuk pengecekan hasil query.

1. `?>`

Menutup tag PHP.



**Penjelasan Detil Kode Program**

**Berikut adalah penjelasan detil dari tiap barus kode program PHP yan menghubungkan ke database MySQL:**

1. **<?php**

**Baris ini adalah tag pembuka untuk kode PHP. Semua kode PHP harus dimulai dengan tag ini.**

1. **$koneksi = mysqli\_connect(‘localhost’,’root’,’’,’db\_spp\_ukk’);**

**`mysqli\_connect` adalah fungsi bawaan PHP yang digunakan untuk menghubungkan aplikasi PHP ke database MySQL.**

1. `’localhost’` adalah nama host dari server database. Dalam kasus ini, server database berada di mesin local (komputer yang sama di mana skrip PHP dijalankan).
2. `’root’` adalah username yang digunakan untuk mengakses database MySQL. Ini adalah username default untuk MySQL.
3. `’’` adalah password yang terkait dengan username. Dalam contoh ini,, tidak ada password yang digunakan.
4. `’db\_spp\_ukk’` adalah nama database yang ingin dihubungkan.

Secara keseluruhan, baris ini menginisialisasi variable `$koneksi` dengan hasil koneksi ke database MySQL yang spesifik.

1. If(!koneksi){

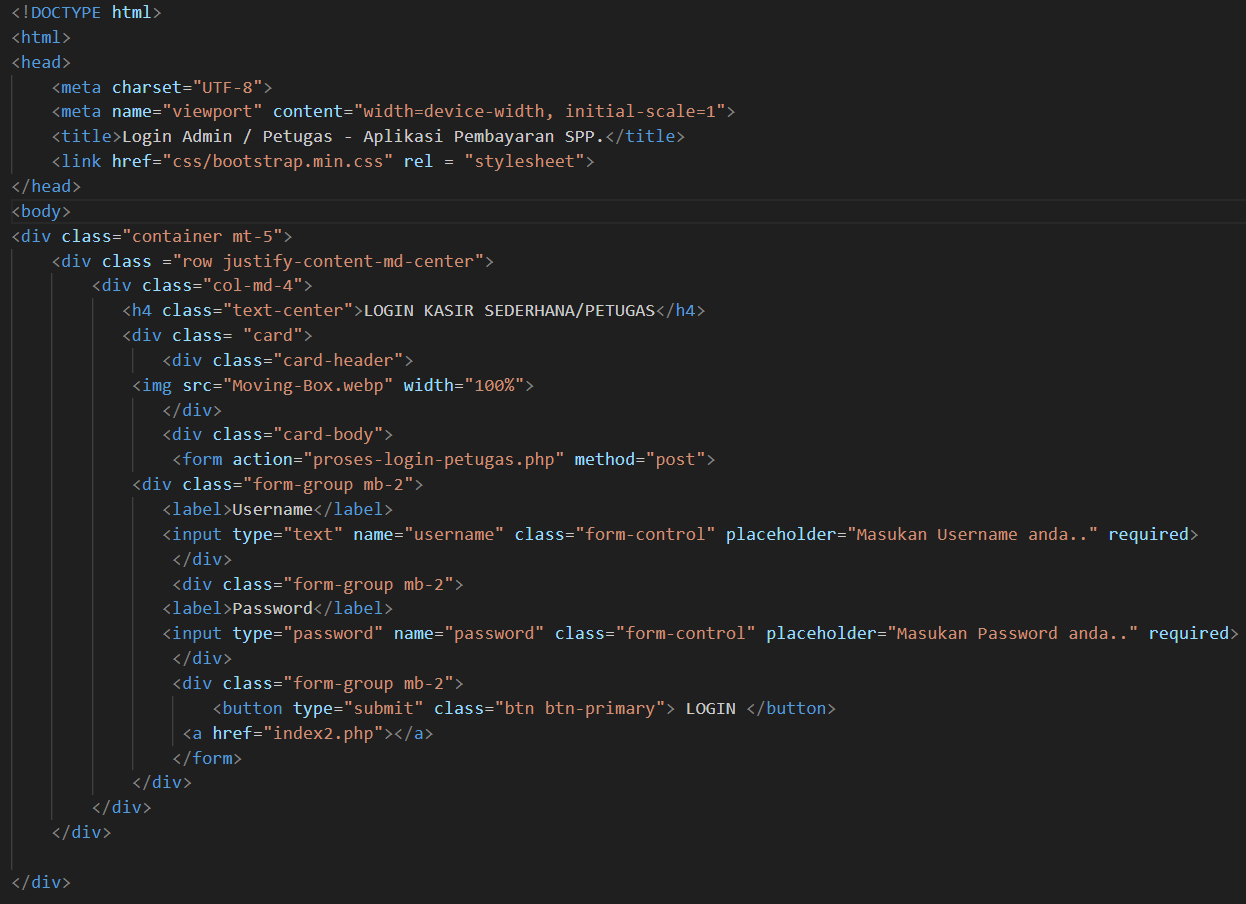
Baris ini memeriksa apakah koneksi ke database berhasil atau tidak.

1. `$koneksi` adalah operator not yang digunakan untuk memeriksa apakah `$koneksi` adalah operator not yang digunakan untuk memeriksa apakah `$koneksi` bernilai false. Jika `mysqli\_connect` gagal membuat koneksi, `$koneksi` akan bernilai false.
2. Echo “Koneksi Anda Gagal”;
3. Jika koneksi ke database gagal, baris ini akan mencetak pesan “Koneksi Anda Gagal”.
4. Secara keseluruhan, blok `if` ini berfungsi sebagai pengecekan untuk memastikan bahwa koneksi ke database berhasil dan memberikan pesan kesalahaan jika koneksi gagal.

Kesimpulan

**Kode PHP ini bertugas untuk menghubungkan aplikasi PHP dengan database MySQL yang ada id server local dengan menggunakan username `root` tanpa password, serta nama database `db\_spp\_ukk`.**

**Jika koneksi gagal, akan ditampilkan pesan kesalahan. Pastikan untuk selalu memeriksa koneksi database dalam aplikasi web untuk memastikan data dapat diakses dan dimanipulasi dengan benar.**

****

### Penjelasan Detil Kode Program

Berikut adalah penjelasan detil dari tiap baris kode dalam file HTML yang digunakan untuk halaman login admin/petugas dalam aplikasi pembayaran SPP:

### Penjelasan Detil Kode:

1. **<!DOCTYPE html>**: Deklarasi tipe dokumen untuk memberitahu browser bahwa dokumen ini menggunakan HTML5.
2. <html>:Elemen root dari dokumen HTML.
3. <head>:Bagian dokumen yang berisi informasi meta tentang dokumen.

<meta chasrset=”UTF-8”>:Menetapkan karakter encoding dokumen ke UTF-8.

<title>:Menetapkan judul dokumen.

<link href=”css/bootstrap.min.css” rel=”stylesheet”>:

Menghubungkan ke file CSS Bootstrap untuk styling.

4. **<body>**: Bagian dokumen yang berisi konten yang akan ditampilkan di browser.

<div class=”container mt-5”>:Container Bootstrap dengan margin top 5.

<div class=”row justify-content-md-center”>:Baris Bootsrap dengan konten yang diposisikan di tengah.

<div class=”col-md-4”>:Kolom Boostrap dengan lebar medium 4.

<h4 class**="text-center">LOGIN KASIR SEDERHANA/PETUGAS</h4>**: Header dengan teks yang diposisikan di tengah.

<div class=”card”>: Elemen kartu Bootstrap.

<div class=”card-header”>: Header kartu.

<**img src="Moving-Box.webp" width="100%">**: Gambar dengan lebar 100%.

<div class=”card-body”>: Body kartu.

<**form action="proses-login-petugas.php" method="post">**: Form dengan metode POST.

**<div class="form-group mb-2">**: Grup form dengan margin bottom 2.

**<label>Username</label>**: Label untuk input username.

**<input type="text" name="username" class="form-control" placeholder="Masukan Username anda.." required>**: Input teks untuk username.

<**div class="form-group mb-2">**: Grup form dengan margin bottom 2.

<label>Password</label>:

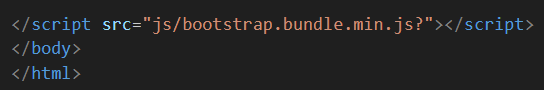
Label untuk input password.

<input type=”password” name=”password” class”form-control” placeholder=”Masukan Password anda..” required>: Input password.

<**div class="form-group mb-2">**: Grup form dengan margin bottom 2.

<button type=”submit” class=”btn btn- primary”>LOGIN</button>:Tombol submit.

<a href=index2.php></a>: Link kosong.



`<script src=”js/bootstrap.bundle.min.js”></script>`:

Menghubungkan file JavaScript Bootstrap untuk fungsionalitas tambahan.

`</body>`: Penutup elemen body. `</html>`:Penutup elemen HTML.



### Penjelasan Detil Kode Program

Kode berikut merupakan bagian dari aplikasi yang menangani tampilan form untuk menambah barang ke dalam tabel barang. Saya akan menjelaskan setiap baris kode secara rinci:

Penjelasan Kode

1. Header dan Tombol Kembali

`<h5> Tabel Barang</h5>: Menampilkan judul bagian halaman sebagai "Tabel Barang".

`<a href="?url=barang" class="btn btn-primary"> KEMBALI </a>`:Membuat tombol berlabel "KEMBALI" yang mengarahkan pengguna kembali ke halaman tabel barang. URL tujuan ditentukan oleh parameter `url` dengan nilai `barang`.

`<hr>`:Menambahkan garis horizontal untuk memisahkan header dari form dibawahnya.

1. Formulir Tambah Barang

`<form method="post" action="?url=proses-tambah-barang">`:

Membuka tag form yang menggunakan metode POST untuk mengirim data ke URL yang ditentukan oleh parameter `url` dengan nilai `proses-tambah-barang`.

1. Input Nama Barang

`<div class="form-group mb-2">`:Membuat div dengan kelas `form-group mb-2` untuk styling elemen form.

`<div class=”form-group mb-2”>`: Membuat div dengan kelas `form-group mb-2` untuk sty;ing elemen form.

`<label>Nama Barang</label>`: Label untuk input nama barang.

`<input type="text" name="nama\_barang" class="form-control" required>`: Input field untuk nama barang dengan tipe teks, nama `nama\_barang`, kelas`form\_control` untuk styling, dan atribut `required` agar tidak kosong.

1. Input Harga

`<label>Harga</label>`: Label untuk input harga.

`<input type=”text” name=”harga” class=”form-control” required>`: Input field untuk harga dengan tipe teks, nama `harga`, kelas `form-control` untuk styling, dan atribut `required` agar tidak kosong.

1. Input Jumlah

`<label>Jumlah</label>`: Label untuk input jumlah.

`<input type="number" name="jumlah" class="form-control" required>`:Input field untuk jumlah dengan tipe angka, nama `jumlah`, kelas `form-control` untuk styling, dan atribut `required` agar tidak kosong.

1. Tombol Submit dan Reset

`<button>`:type="reset" class="btn btn-warning"> KOSONGKAN </button>`: Tombol reset dengan kelas `btn -btn-warning` untuk styling dan label “KOSONGKAN”,

Kesimpulan

Kode di atas menyediakan antarmuka form yang memungkinkan pengguna untuk menambahkan barang ke tabel barang dengan memasukkan nama barang, harga, dan jumlah. Setiap input field memiliki atribut `required` untuk memastikan bahwa pengguna tidak dapat mengirim form dengan data yang kosong. Tombol submit mengirimkan data ke URL yang ditentukan untuk diproses, sementara tombol reset mengosongkan semua input field.

#### Penjelasan Detil Tiap Baris Kode Program

**Berikut adalah penjelasan detil dari tiap baris kode program pada bagian halaman barang:**

1. `<h5>Halaman Barang</h5>`:Menampilkan judul halaman “Halaman Barang” dengan format heading 5.
2. `<a href=”?url=tambah-barang” class=”btn btn-primary”> Tambah Barang </a>`:Membuat link untuk menambah barang baru. Saat diklik, pengguna akan diarahkan ke halaman `tambah-barang`.Link ini diberi kelas CSS `btn btn-primary` untuk gaya tombol bootstrap.
3. `<hr>`:Menambahkan garis horizontal untuk memisahkan elemen halaman.
4. `<table class=”table table-striped table-bordered”>`:Memulai table dengan kelas bootstrap untuk gaya table bergaris dan berbingkai.
5. `<tr class=”fw-bold”>:`Memulai baris table untuk header dengan kelas `fw-bold` untuk teks tebal.
6. `<td>No</td>`:Header kolom pertama untuk nomor urut.
7. `<td>Nama Barang</td>`:Header kolom kedua untuk nama barang.
8. `<td>Harga</td>`:Header kolom ketiga untuk harga barang.
9. `<td>Pembelian</td>`:Header kolom keempat untuk aksi pembelian.
10. `<td>Hapus</td>`:Header kolom kelima untuk aksi penghapusan.
11. `<?php>`:Membuka blok kode PHP.
12. `include`../koneksi.php’;`:Menyertakan file`koneksi.php` yang berisi detail koneksi ke database.
13. `$no =1;`: Menginisialisasi variabel `$no` untuk nomor urut baris.
14. `$sql = “SELECT \* FROM barang ORDER BY id\_barang DESC”;`: Membuat query SQL untuk mengambil semua data dari tabel `barang` diurutkan berdasarkan `id\_barang` secara menurun.
15. `$query = mysqli\_query($koneksi, $sql);`:Mengeksekusi query SQL dan menyimpan hasilnya dalam variable `$query`.
16. `foreach($query as $data){ ?>`:Memulai loop untuk literasi melalui hasil query. Setiap baris hasil query akan disimpan dalam variable `$data`.
17. `<tr>`: Memulai baris table baru.
18. `<td><?= $no++; ?></td>`:Menampilkan nomor urut dan menambah nilainya satu setiap kali baris baru ditampilkan.
19. `<td><?= $data[‘nama\_barang’] ?></td>`:Menampilkan nama barang dari hasil query.
20. `<td>`:Memulai kolom aksi pembelian.
21. `<a href=”url=beli-barang&id\_barang=<?=$data[‘id\_barang’] ?

>” class=’btn btn-warning’>BELI</a>`:Membuat link untuk membeli barang. Saat diklik, pengguna akan diarahkan ke halaman `beli-barang` dengan parameter `id\_barang` yang sesuai. Link ini diberi kelas CSS `btn btn-warning` untuk gaya tombol bootstrap.

23.`</td>`:Menutup kolom aksi pembelian.

24.`<td>: Memulai kolom aksi penghapusan.

25. `<a onclick=”return confirm(‘Apakah Anda Yakin Menghapus Data’)” href=”?url=hapus-barang\*id\_barang=<?=

$data[‘id\_barang’] ?>” class=’ btn btn-danger’>HAPUS</a>`:

**Membuat link untuk menghapus barang. Saat diklik, akan muncul konfirmasi penghapusan. Jika dikonfirmasi, pengguna akan diarahkan ke halaman `hapus-barang` dengan parameter**

**`id\_barang` yang sesuai. Link ini diberi kelas CSS `btn btn-danger` untuk gaya tombol bootstrap.**

**26. `</td>`:Menutup kolom aksi penghapusan.**

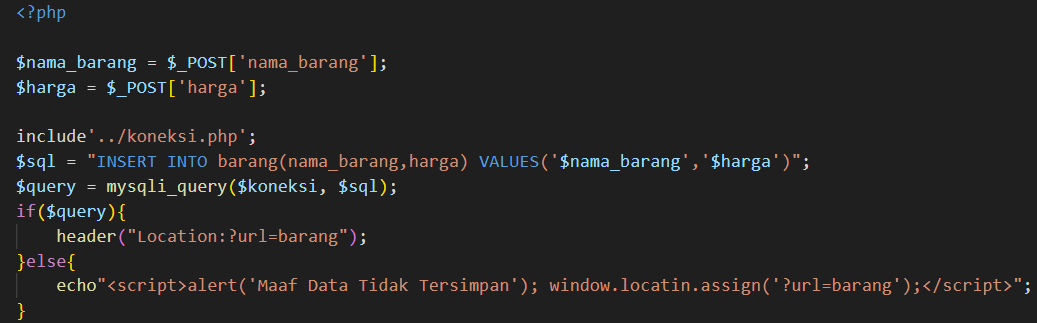
**27. `</tr>:Menutup baris tabel.**

**28. `<?php } ?>`:Menutup blok loop `foreach`.**

**29. `</table>`:Menutup tabel HTML.**

### **Kesimpulan**

**Kode di atas adalah implementasi halaman barang yang menampilkan daftar barang dalam bentuk tabel, dengan opsi untuk menambah, membeli, dan menghapus barang. Kode ini menggunakan PHP untuk menghubungkan dan mengambil data dari database MySQL, serta Bootstrap untuk gaya antarmuka pengguna.**



**Penjelasan Detail dari Tiap Baris Kode Program**

**Ini adalah tag pembuka untuk kode PHP. Setiap skrip PHP dimulai dengan**

**`<?php` dan diakhiri dengan `?>`(meskipun dalam kode ini tidak ditampilkan penutupnya karena tidak ada HTML atau skrip lainnya setelahnya).**

**`$\_POST` adalah sebuah superglobal array yang digunakan untuk menggunakan metode POST.**

**`$nama\_barang` menyimpan nilai dari input form dengan nama ‘nama\_barang’.**

**`$harga` menyimpan nilai dari input form dengan nama 'harga'.**

**`include` digunakan untuk menyertakan file `koneksi.php` yang berisi konfigurasi koneksi ke database. Dengan menyertakan file ini, koneksi ke** **database dapat digunakan di dalam skrip ini.**

**`$sql` menyimpan perintah SQL yang digunakan untuk memasukkan data ke dalam tabel `barang`.**

**`INSERT INTO barang(nama\_barang, harga)` adalah perintah SQL untuk memasukkan data ke dalam kolom `nama\_barang` dan `harga` di tabel `barang`.**

**`VALUES(‘$nama\_barang’,’$harga’)` adalah nilai yang akan dimasukkan ke dalam kolom `nama\_barang dan `harga`. Variabel PHP disisipkan ke dalam perintah SQL** **untuk memasukkan data dari form.**

**`mysqli\_query` adalah fungsi PHP yang digunakan untuk menjalankan perintah SQL terhadap database.**

**`$koneksi` adalah variabel yang menyimpan koneksi ke database (disediakan oleh `koneksi.php`).**

**`$sql` adalah perintah SQL yang akan dijalankan.**

**`$query` menyimpan hasil dari eksekusi perintah SQL tersebut, apakah berhasil atau tidak.**

**`id($query)` memeriksa apakah perintah SQL berhasil dijalankan (`$query` bernilai true jika berhasil).**

**`header(“Location:?url=barang`) jika perintah SQL berhasil.**

**`else` mengeksekusi blok kode jika perintah SQL gagal.**

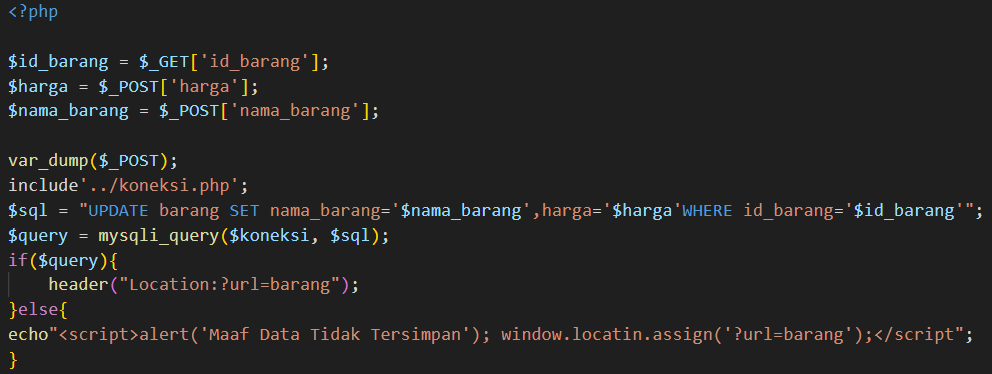
**`echo “<script>alert(‘Maaf Data Tidak Tersimpan’);**

**Window.location.assign(‘?url=barang’);</script>”;`**

**menampilkan pesan alert JavaScript bahwa data tidak tersimpan dan mengarahkan pengguna kembali ke halaman `?url=barang`.**

### **Kesimpulan**

**Kode ini mengambil data dari form HTML melalui metode POST, menghubungkan ke database, dan memasukkan data tersebut ke dalam tabel barang. Jika operasi berhasil, pengguna akan diarahkan ke halaman lain. Jika gagal, pesan kesalahan akan ditampilkan dan pengguna akan tetap berada di halaman yang sama.**



## **Penjelasan Kode Program**

### **Kode Program**

**Penjelasan Detil**

**`<?php`**

**Tag pembuka untuk kode PHP.**

**~$id\_barang = $\_GET[‘id\_barang’];`**

**Mengambil nilai `id\_barang` dari parameter URL menggunakan metode GET dan menyimpannya ke dalam variabel `$id\_barang`.**

**`$harga = $\_POST[‘harga’];`**

**Mengambil nilai `harga` dari form yang dikirim menggunakan metode POST dan menyimpannya ke dalam variabel `$harga`.**

**`$nama\_barang = $\_POST['nama\_barang'];`**

**Mengambil nilai `nama\_barang` dari form yang dikirim menggunakan metode POST dan menyimpannya ke dalam variabel `$nama\_barang`.**

**`var\_dump($\_POST)`**

**Mencetak seluruh data yang dikirim melalui metide POST untuk tujuan debugging. Fungsi `var\_dump` menampilkan struktur dan isi dari variabel.**

**`include’../koneksi.php’:`**

**Menyertakan file `koneksi.php` yang berisi kode untuk menghubungkan ke database. File ini biasanya berisi informasi koneksi seperti host, username, password, dan nama database.**

**`$sql = “UPDATE barang SET**

**nama\_barang=’$nama\_barang’,harga=’$harga’ WHERE id\_barang=’$id\_barang’”;`**

**Membuat query SQL untuk memperbarui tabel `barang`. Query**

**Ini akan mengubah nilai kolom `nama\_barang` dan `harga` berdasarkan `id\_barang` yang diterima dari URL.**

**`$query = mysqli\_query($koneksi, $sql);`**

**Menjalankan query SQL yang telah dibuat menggunakan koneksi database yang diambil dari `koneksi.php`. Hasil dari operasi ini disimpan dalam variabel `$query`.**

**`if($query){`**

**Memeriksa apakah query berhasil dijalankan. Jika berhasil, kode di dalam blok if akan dieksekusi.**

**`headwe(“Location:?url=barang”);`**

**Mengarahkan browser ke halaman lain menggunakan header HTTP. Dalam kasus ini, diarahkan ke halaman dengan parameter URL `url=barang`.**

**`}else{`**

**Blok else yang akan dieksekusi jika query gagal dijalankan.**

**`echo”<script>alert(‘Maaf Data Tidak Tersimpan’);**

**Window.locatin.assign(‘?url=barang’);</script”;`**

**Menggunakan JavaScript untuk menampilkan pesan peringatan bahwa data tidak tersimpan, kemudian mengarahkan pengguna kembali ke halaman dengan parameter URL `url=barang`. (Catatan: Ada kesalahan penulisan pada `window.locatin.assign`, yang seharusnya `window.location.assign`).**

**`}`**

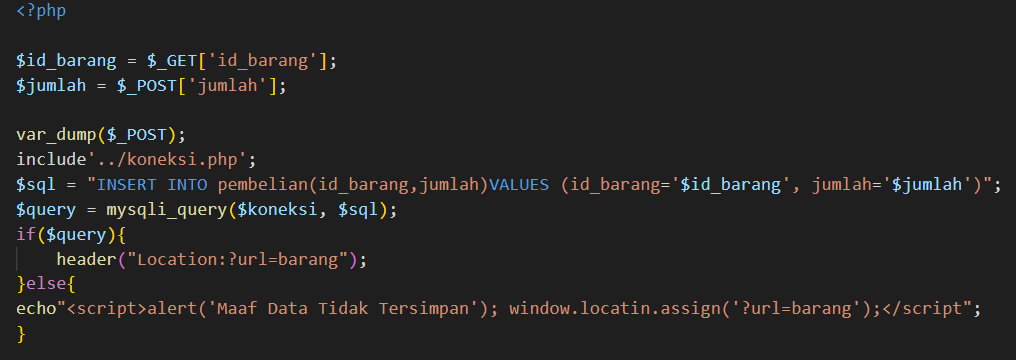
**Penutup blok if-else.**

**`?>`**

**Tag penutup untuk kode PHP.**

### **Kesimpulan**

**Kode PHP di atas digunakan untuk memperbarui data barang dalam database berdasarkan `id\_barang` yang diterima dari URL. Data baru untuk `nama\_barang` dan `harga` diambil dari form yang dikirim menggunakan metode POST. Jika proses pembaruan berhasil, pengguna akan diarahkan ke halaman tertentu, jika gagal, pesan peringatan akan ditampilkan dan pengguna akan diarahkan kembali.**

****

**Penjelasan Detil Kode Program**

**Berikut adalah penjelasan detil dari setiap baris kode pada bagian program PHP yang disediakan:**

**<?php**

**Baris ini menandakan awal dari kode PHP.**

$id\_barang = $\_GET[‘id\_barang’];

Baris ini mengambil nilai dari parameter `id\_barang` yang dikirim melalui metode GET dari URL dan menyimpannya dalam variabel `$id\_barang`.

$jumlah = $\_POST[‘jumlah’];

Var\_dump($\_POST);

Baris ini digunakan untuk mencetak semua data yang dikirim melaui metode POST. `var\_dump` sangat berguna untuk debugging

Karena menampilkan tipe data dan nilai dari variabel.

Include’../koneksi.php’;

Baris ini memasukkan file `koneksi.php` yang biasanya berisi skrip

Untuk menghubungkan ke database MySQL. Dengan menggunakan `include`, file ini dimasukkan sehingga bisa digunakan dalam skrip ini.

$sql = “INSERT INTO pembelian(id\_barang,jumlah)VALUES (id\_barang=’$id\_barang’, jumlah=’$jumlah’)”;

Baris ini membuat query SQL untuk memasukkan data ke tabel `pembelian`. `id\_barang` dan `$jumlah`. Namun, query ini salah karena penulisan sintaks SQL yang tidak benar.

Query yang benar seharusnya:

$sql = “INSERT INTO pembelian (id\_barang, jumlah) VALUES (‘$id\_barang’, ‘$jumlah’)”;

$query = mysqli\_query($koneksi, $sql);

Baris ini menajalankan query SQL yang telah diibuat sebelumnya menggunakan koneksi database `$koneksi`. Hasil dair operasi ini disimpan dalam variabel `$query’.

Id($query){

Baris ini mengecek apakah query berhasil dijalankan. Jika `mysqli\_query` berhasil, maka variable `$query` akan bernilai `true`.

Header(“Location:?url=barang”);

Jika query berhasil, maka pengguna akan dialihkan ke halaman yang ditentukan oleh parameter URL `barang`.

}else{

Bagian ini menangani kondisi jika query gagal dijalankan.

Echo=<script>alert(‘Maaf Data Tidak Tersimpan’); window.location.assign.(‘?url=barang’);</script>”;

Jika query gagal, pengguna anak melihat pesan alert “Maaf Data Tidak Tersimpan” dan kemudian dialihkan Kembali ke halaman yang ditentukan oleh parameter URL `barang`. Namun, terdapat kesalahan penulisan pada `window.locationnassign’, yang seharusnya adalah`window.locatin.assign’.

}

Akhir dari blok kondisi `if-else’.

?>

Baris ini menandakan akhir dari kode PHP.

Perbaikan

Setelah memeprbaiki beberapa kesalahan penulisan, berikut adalah kode yang sudah diperbaiki:

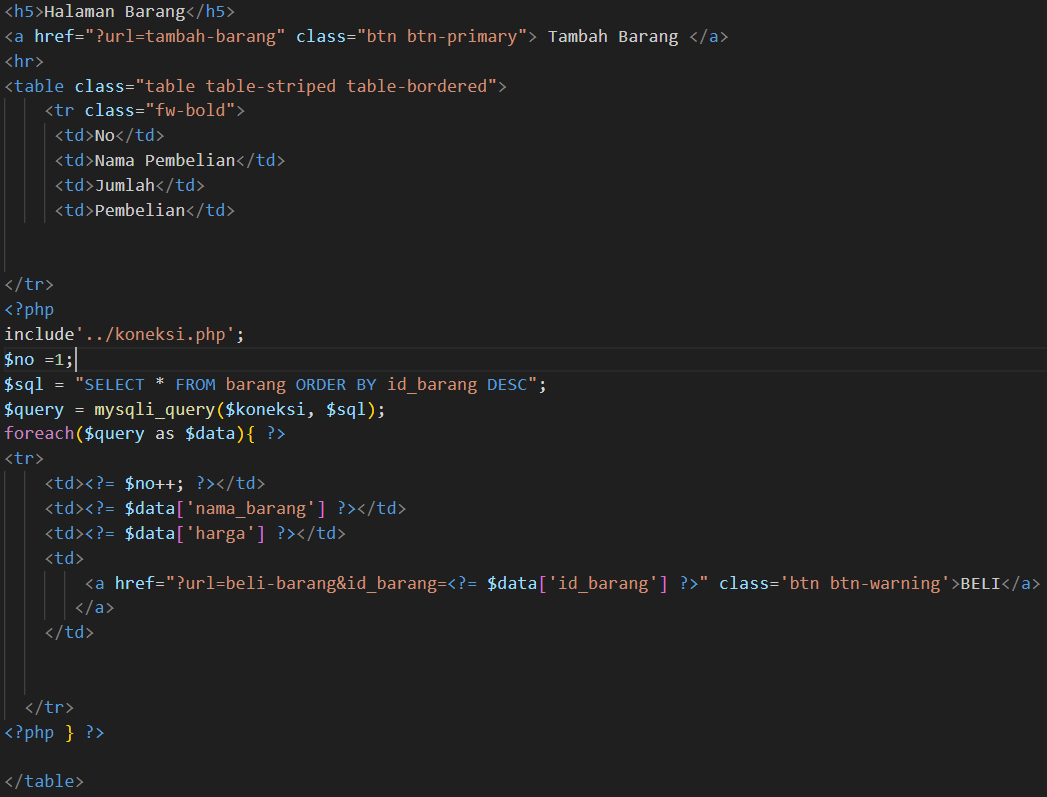
<?php

**$id\_barang = $\_GET['id\_barang']; $jumlah = $\_POST['jumlah']; var\_dump($\_POST); include '../koneksi.php'; $sql = "INSERT INTO pembelian (id\_barang, jumlah) VALUES ('$id\_barang', '$jumlah')"; $query = mysqli\_query($koneksi, $sql); if ($query) { header("Location:?url=barang"); } else { echo "<script>alert('Maaf Data Tidak Tersimpan'); window.location.assign('?url=barang');</script>"; }**

**?>**

**Penutup**

**Demikian penjelasan detil dari tiap baris kode PHP pada bagian yang disediakan.**

`<h5>Halaman Barang</h5>`:Judul halaman yang ditampilkan pada bagian atas halaman.

`<a href=”?url=tambah-barang” class=”btn btn-primary”> Tambah Barang </a>`: Tautan yang mengarah ke halaman/form untuk menambah barang baru. Dengan kelas `btn btn-primary`, tautan ini diformat sebagai tombol berwarna biru.

`<hr>`:Garis horizontal untuk memisahkan bagian – bagian dalam halaman.

<`table class=”table table-striped table-bordered”>`:Memulai tabel dengan kelas Bootstrap untuk menambahkan garis batas dan garis-garis belang.

`<tr class=”fw-bold”>`:Baris header tabel dengan teks tebal.

`<td>`:Kolom dalam header tabel untuk No, Nama Pembelian, Jumlah, dan Pembelian.

2. Menyertakan File Koneksi dan Query Database

<?php include’../koneksi.php’;

$no =1;

$sql = “SELECT \* FROM barang OREDER BY id\_barang DESC”;

$query = mysqli\_query($koneksi, $sql);

Foreach($query as $data){ ?>

`<?php include’../koneksi.php’; ?>`:Menyertakan file `koneksi.php` untuk menghubungkan ke database.

`$no =1`;Menginisialisasi variabel `$no` dengan nilai 1 untuk penomoran baris dalam tabel.

`$sql = “SELECT \* FROM barang ORDER BY id\_barang DESC”;`:

Mendefinisikan query SQL untuk mengambil semua data dari tabel `barang` dan mengurutkannya berdasarkan `id\_barang` secara menurun.

`$query = mysqli-query($koneksi, $sql);`: Menjalankan query

SQL di atas menggunakan koneksi database yang disimpan dalam variabel `$koneksi`.

`foreach($query as $data)`: Memulai loop `foreach` untuk literasi setiap baris data yang dikembalian oleh query.

Menampilkan Data Dalam Tabel

`<tr>`: Memulai bari tabel untuk setiap item barang.

`<td><?= $no++; ?></td>`:Menampilkan nomor urut. Variabel `$no` ditingkatkan nilainya setiap kali baris baru ditambahkan.

`<td><?= $data[‘nama\_barang’] ?></td>`:Menampilkan nama barang dari hasil query.

`<td><?= $data[‘harga’] ?></td>`:Menampilkan harga barang dari hasil query.

`<td>`:Memulai kolom untuk tombol beli.

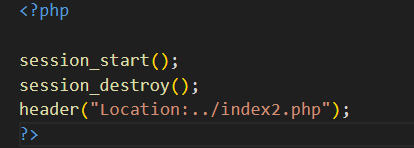
`<a href=”?url=beli-barang&id\_barang=<?= $data[‘id\_barang’] ?>” class=’btn btn-warning’>BELI</a>`:Tautan yang mengarahkkan ke halaman pembelian barang dengan mengirimkan parameter `id\_barang`. Tombol ini memiliki kelas`btn btn-warning` untuk tampilan berwarna kuning.

`<?php } ?>`:Menutup loop `foreach`.

`</table>`:Menutup tabel.

Kesimpulan

Kode di atas adalah bagian dari aplikasi web yang menampilkan daftar barang yang tersedia untuk dbeli. Dengan menggunakan kombinasi HTML, CSS dari Bootstrap, dan PHP untuk berinteraksi dengan database, halaman ini memungkinkan pengguna untuk melihat barang yang tersedian dan memilih untuk membelinya. Setiap bagian dari kode ini memiliki fungsi spesifik untuk memastikan data ditampilkan dengan benar dan interaksi pengguuna ditangani dengan baik.



**Penjelasan Detil Kode Program**

**Berikut adalah penjelasan detil dari tiap baris kode pada bagian yang diberikan:**

Baris 1:`<?php`

**Penjelasan: Tag pembuka PHP. Ini menandakan bahwa semua kode setelah tag ini akan diperlakukan sebagai kode PHP sampai tag penutup**

**`?>` ditemukan.**

**Baris 2:`session\_start();`**

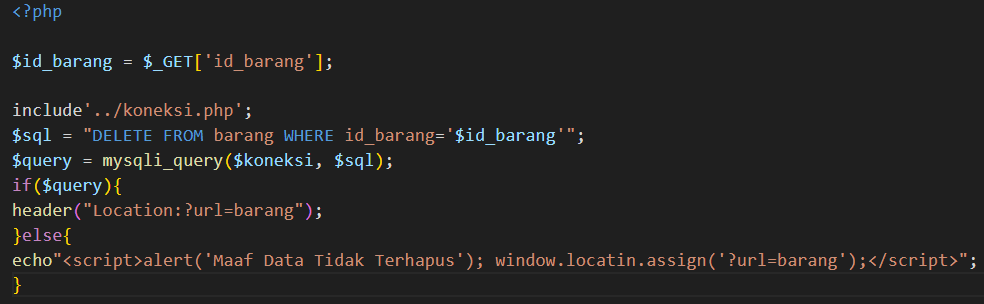
**Penjelasan: Fungsi `session\_start()` digunakan untuk memulai sesi pada PHP. Ketika fungsi ini dipanggil, PHP akan mencoba memulai sesi yang ada atau membuat sesi baru jika belum ada sesi yang aktif. Sesi digunakan untuk menimpan informasi pengguna di antara berbagai halaman (seperti login status, user preferences, dll).**

**Baris3: `session\_destroy();`**

**Penjelasan: Fungsi `session\_destroy()` digunakan untuk mengakhiri sesi yang sedang berjalan dan menghapus semua data yang terkait dengan sesi tersebut. Setelah fungsi ini dipanggil, semua variabel sesi akan dihapus dan sesi akan diakhiri. Fungsi ini berguna ketika pengguna logout atau ketika kita ingin menghapus semua informasi sesi pengguna.**

**Baris 4: `header(“Location:../index2.php”);`**

**Penjelasan: Fungsi`header()` digunakan untuk mengirimakn header HTTP mentah ke browser. Dalam hal ini, header yang dikirim adalah `Location`, yang mengistruksikan browser untuk mengalihkan pengguna ke URL yang ditentukan. Pada baris ini, pengguna akan dialihkan ke halaman`index2.php` yang berada satu direktori di atas (indicated by`../`). Redirection ini biasanya dilakukan setelah proses logout untuk membawa pengguna Kembali ke halaman utama atau halaman login.**

****

**Penjelasan Detil Kode Program**

**Berikur adalah penjelasan detil dari tiap baris kode program PHP untuk menghapus data barang dari database:**

**<?php**

**Baris ini membuka tag PHP, yang menunjukkan bahwa kode di dalamnya adalah kode PHP.**

**$id\_barang = $\_GET['id\_barang'];**

**Baris ini mengambil parameter`id\_barang` dari URL menggunakan metode`GET`.Parameter ini mewakili ID barang yang ingin dihapus dari database. Misalnya, jika URL adalah `delete.php?id\_barang=123`, maka nilai`$id\_barang` akan menjadi `123`.**

**include'../koneksi.php';**

**Baris ini menghubungkan file `koneksi.php` yang berisi skrip untuk menghubungkan ke database. Dengan menggunakan `include`, skrip `koneksi.php`, dijalankan dan koneksi ke database diinisialisasi.**

**$sql = "DELETE FROM barang WHERE id\_barang='$id\_barang'";**

**Baris ini mendefinikasikan query SQL untuk menghapus data dari tabel `barang` di mana kolom `id\_barang` sama dengan nilai`$id\_barang`**

**Yang didapat dari URL. Query ini berbentuk string.**

**$query = mysqli\_query($koneksi, $sql);**

**Baris ini menjalankan query SQL yang telah didefinikasikan menggunakan fungsi`mysqli\_query()`. Fungsi ini membutuhkan dua parameter:**

**Koneksi database(`$koneksi`) dan query SQL (`$sql`). Hasil eksekusi query disimpan dalam variabel`$query`.**

**if($query){**

**Baris ini memulai sebuah blok kondisional`if`.Jika query berhasil dijalankan(`$query`mengembalikan nilai`true`),maka blok kode di dalam `if` akan dieksekusi.**

**header("Location:?url=barang");**

**Jika query berhasil, baris ini akan mengarahkan pengguna ke halaman dengan URL parameter `url=barang`. Ini biasanya digunakan untuk Kembali ke halaman daftar barang setelah penghapusan berhasil.**

**}else{**

**Baris ini memulai blok `else` yang akan dieksekusi jika query gagal (misalnya, jika`$query` mengembalikan nilai `false`).**

**echo"<script>alert('Maaf Data Tidak Terhapus'); window.locatin.assign('?url=barang');</script>";**

**Jika query gagal, barus ini akan menampilkan pesan alert menggunakan JavaScript dan mengarahkan Kembali pengguna ke halaman dengan URL parameter `url=barang`. Terdapat kesalahan pengetikan pada `window.locatin.assign`. seharusnya`window.location.assign`.**

**}**

**Baris ini menutup blok `if-else`.**

**?>**

**Baris ini menutup tag PHP, menunjukkan akhir dari skrip PHP.**

**Keseluruhan Kode**

**<?php $id\_barang = $\_GET['id\_barang']; include'../koneksi.php'; $sql = "DELETE FROM barang WHERE id\_barang='$id\_barang'"; $query = mysqli\_query($koneksi, $sql); if($query){ header("Location:?url=barang"); }else{ echo"<script>alert('Maaf Data Tidak Terhapus'); window.locatin.assign('?url=barang');</script>"; } ?>**

**Kode ini digunakan untuk menghapus data barang dari database berdasarkan `id\_barang` yang diterima dari URL. Jika penghapusan berhasil, pengguna akan diarahkan Kembali ke halaman daftar barang. Jika gagal, pengguna akan mendapatkan pesan error.**

**Kesimpulan**

**Skrip PHP ini secara efektif menghapus data dari tabel barang di database berdasarkan ID yang diterima melalui metode GET. Pastikan untuk memeriksa kesalahan pengetikan dan memastikan bahwa koneksi ke database berhasil dilakukan melalui file`koneksi.php`.**

**Penjelasan Deti dari Kode Program** **Berikut adalah penjelasan detil dari tiap baris kode program PHP yang telah Anda buat:**

**<?php**

**// Mengambil parameter 'id\_barang' dari URL**

**$id\_barang = $\_GET['id\_barang'];**

**// Menyertakan file koneksi ke database**

**include'../koneksi.php';**

**// Query untuk mengambil data barang berdasarkan id\_barang**

**$sql = "SELECT \* FROM barang WHERE id\_barang='$id\_barang'";**

**// Menjalankan query dan menyimpan hasilnya dalam variabel $query**

**$query = mysqli\_query($koneksi, $sql);**

**// Mengambil data hasil query dalam bentuk array**

**$data = mysqli\_fetch\_array($query);**

**?>**

**<!-- Judul halaman -->**

**<h5>Halaman Beli Barang.</h5>**

**<!-- Tombol untuk kembali ke halaman sebelumnya -->**

**<a href="?url=barang" class="btn btn-primary">KEMBALI</a>**

**<hr>**

**<!-- Form untuk mengedit data barang -->**

**<form method="post" action="?url=proses-edit-barang&id\_barang=<?= $id\_barang; ?>">**

**<!-- Input untuk nama barang -->**

**<div class="form-group mb-2">**

**<label>Nama Barang</label>**

**<input value="<?= $data['nama\_barang']; ?>" type="text" name="nama\_barang" class="form-control" required>**

**</div>**

**<!-- Input untuk harga barang -->**

**<div class="form-group mb-2">**

**<label>Harga</label>**

**<input value="<?= $data['harga'] ?>" type="text" name="harga" class="form-control" required>**

**</div>**

**<!-- Tombol untuk menyimpan perubahan dan mengosongkan form -->**

**<div class="form-group mb-2">**

**<button type="submit" class="btn btn-success">SIMPAN</button>**

**<button type="reset" class="btn btn-warning">KOSONGKAN</button>**

**</div>**

**</form>**

**Penjelasan Baris Kode**

**Mengambil parameter ‘if\_barang’ dari URL:**

**$id\_barang = $\_GET['id\_barang'];**

**Baris ini mengambil nilai parameter`id\_barang` dari URL dan menyimpannya dalam variabel`$id\_barang`.**

**Menyerteakan file koneksi ke database:**

**include'../koneksi.php';**

**Baris in menyertakan file`koneksi.php` yang berisi konfigurasi koneksi ke database.**

**Query untuk mengambil data barang berdasarkan id\_barang**:   
**$sql = "SELECT \* FROM barang WHERE id\_barang='$id\_barang'";**

**Baris ini mendefinisikan query SQL untuk mengambil semua data dari tabel `barang` Dimana `id\_barang` sesuai dengan nilai yang diambil dari URL.**

**Menjalankan query dan menyimpan hasilnya dalam variabel $query:**

**$query = mysqli\_query($koneksi, $sql);**

**Baris ini menjalankan query SQL yang telah didefinisikan dan menyimpan hasilnya dalam variabel $query.**

**Mengambil data hasil query dalam bentuk array**:

**$data = mysqli\_fetch\_array($query);**

**Baris ini mengambil baris data hasil query dalam bentuk array dan menyimpannya dalam vairabel`$data`.**

**Judul halaman:**

**<h5>Halaman Belu Barang.</h5>**

**Baris ini menampilkan judul halaman.**

**Tombol untuk Kembali ke halaman sebelumnya:**

**<a href="?url=barang" class="btn btn-primary">KEMBALI</a>**

**<hr>**

**Baris ini menampilkan tombol untuk Kembali ke halaman `barang`**

**Dengan URL parameter `?url=barang:**

**<form method="post" action="?url=proses-edit-barang&id\_barang=<?= $id\_barang; ?>">**

**Baris ini mendefinisikan awa form dengan metode`POST` yang akan mengirim data ke URL`?url=proses-edit-barang&id\_barang=<?= $id\_barang; ?>`.**

**Input untuk nama barang:**

**<div class="form-group mb-2">**

**<label>Nama Barang</label>**

**<input value="<?= $data['nama\_barang']; ?>" type="text" name="nama\_barang" class="form-control" required>**

**</div>**

**Baris ini mendefinisikan input fiels untuk nama barang dengan nilain awal diambil dari array`$data`.**

**Input untuk harga barang:**

**<div class="form-group mb-2">**

**<label>Harga</label>**

**<input value="<?= $data['harga'] ?>" type="text" name="harga" class="form-control" required>**

**</div>**

**Baris ini mendefinisikan input field utnuk harga barang dengan nilai awal diambil dari array `$data`.**

**Tombol untuk menyimpan perubahan dan mengosongkan form:**

**<div class="form-group mb-2">**

**<button type="submit" class="btn btn-success">SIMPAN</button>**

**<button type="reset" class="btn btn-warning">KOSONGKAN</button>**

**</div>**

**Baris ini mendefinisikan tombol untuk menyimpan perubahan(`submit`) dan mengosongkan form(`reset`).**

**Dengan penjelasan ini, diharapkan setiap baris kode dapat dipahami**

**Dengan jelas dan lengkap.**

#### Penjelasan Detil Kode Program

**Berikut adalah penjelasan detil untuk setiap baris kode dari skrip PHP yang telah diberikan:**

**<?php**

**$id\_barang = $\_GET['id\_barang'];**

**include'../koneksi.php';**

**$sql = "SELECT \* FROM barang WHERE id\_barang='$id\_barang'";**

**$query = mysqli\_query($koneksi, $sql);**

**$data = mysqli\_fetch\_array($query);**

**// var\_dump($data);**

**?>**

**<h5>Halaman Beli Barang.<h5>**

**<a href="?url=barang" class="btn btn-primary"> KEMBALI </a>**

**<hr>**

**<form method="post" action="?url=proses-beli-barang&id\_barang=<?= $id\_barang; ?>">**

**<div class="form-group mb-2">**

**<label>Nama Barang</label>**

**<input value="<?= $data['nama\_barang']; ?>" type="text" name="nama\_barang" class="form-control" required>**

**</div>**

**<div class="form-group mb-2">**

**<label>Harga</label>**

**<input value="<?= $data['harga'] ?>" type="text" name="harga" class="form-control" required>**

**</div>**

**<div class="form-group mb-2">**

**<label>Jumlah</label>**

**<input type="number" name="jumlah" class="form-control" required>**

**</div>**

**<div class="form-group mb-2">**

**<button type="submit" class="btn btn-success"> SIMPAN </button>**

**<button type="reset" class="btn btn-warning"> KOSONGKAN </button>**

**</div>**

**</form>**

**Penjelasan Kode**

**`<?php`:Membuka blok kode PHP.**

**~$id\_barang =$\_GET[‘id\_barang’];`:Mengambil nilai`id\_barang` dari URL menggunakanmetode GET dan menyimpannya ke dalam variabel`$id\_barang`.**

**`$include`../koneksi.php`;`:Meng-include file `koneksi.php` untuk menghubungkan ke database.**

**`$sql = “SELECT \* FROM barang WHERE id\_barang=’$id\_barang’”:`:**

**Membuat query SQL untuk memilih semua data dari tabel`barang` Dimana`id\_barang`sesuai dengan nilai`$id\_barang`.**

**`$query = mysqli\_query($koneksi, $sql);`:Menjalankan query SQL pada koneksi database yang telah dibuka dan menyimpan hasilnya dalam variabel `$query`.**

**`$data = mysqli\_fetch\_array($query);`:Mengambil hasil query sebagai array asosiatif dan menyimpannya dalam variabel`$data`.**

**`// var\_dump($data);`:Komentar yang berfungsi untuk mendebug isi dari `$data` jika diperlukan.**

**`?>`:Menutup blok kode PHP.**

**`<h5>Halaman Beli Barang.</h5>`:Menampilkan judul halaman “Halaman Beli Barang”.**

**`<a href=”?url=barang” class=”btn btn-primary”> KEMBALI </a>`:**

**Menampilkan tombol”KEMBALI” yang mengarahkan ke URL`? url=barang`.**

**`<hr>`:Menampilkan garis hoarizontal sebagai pemisah.**

**`<form method=”post” action=”?url=proses-beli-barang&id\_barang=<?= $id\_barang; ?>`.**

**1<div class=”form-group mb-2”>`:Membuka div untuk form-group dengan margin-bottom 2.**

**`<label>Nama Barang</label>`:Menampilkan label “Nama Barang”.**

**`<input value=”<?= $data[‘nama\_barang’]; ?> type=”text” name=”nama\_barang” class=”form-control” required>`: Membuat input field dengan nilai default dari `nama\_barang` yang diambil dari `$data`. Dan memberikan atribut`required` untuk validasi form.**

**`</div>`:Menutup div form-group.**

**`<div class=”form-group mb-2”>`:Membuka div baru utnuk form-group.**

**`<label>Harga</label>`:Menampilkan label “Harga”.**

**`<input value=”<?= $data[‘’harga] ?>” type=”text” name=”harga”**

**Class=”form-control” required>`:Membuat input field dengan nilai default dari `harga` yang diambil dari `$data`. Dan memberikan atribut`required`.**

**`</div>`:Menutup div form-group.**

**`<div class=”form-group mb-2”> digunakan untuk membungkus elemen – elemen form (seperti label dan input) dalam sebuah container yang terstruktur dengan baik.**

**Tombol `<button type="submit" class="btn btn-success"> SIMPAN </button>` adalah elemen HTML yang digunakan dalam konteks formulir web.**

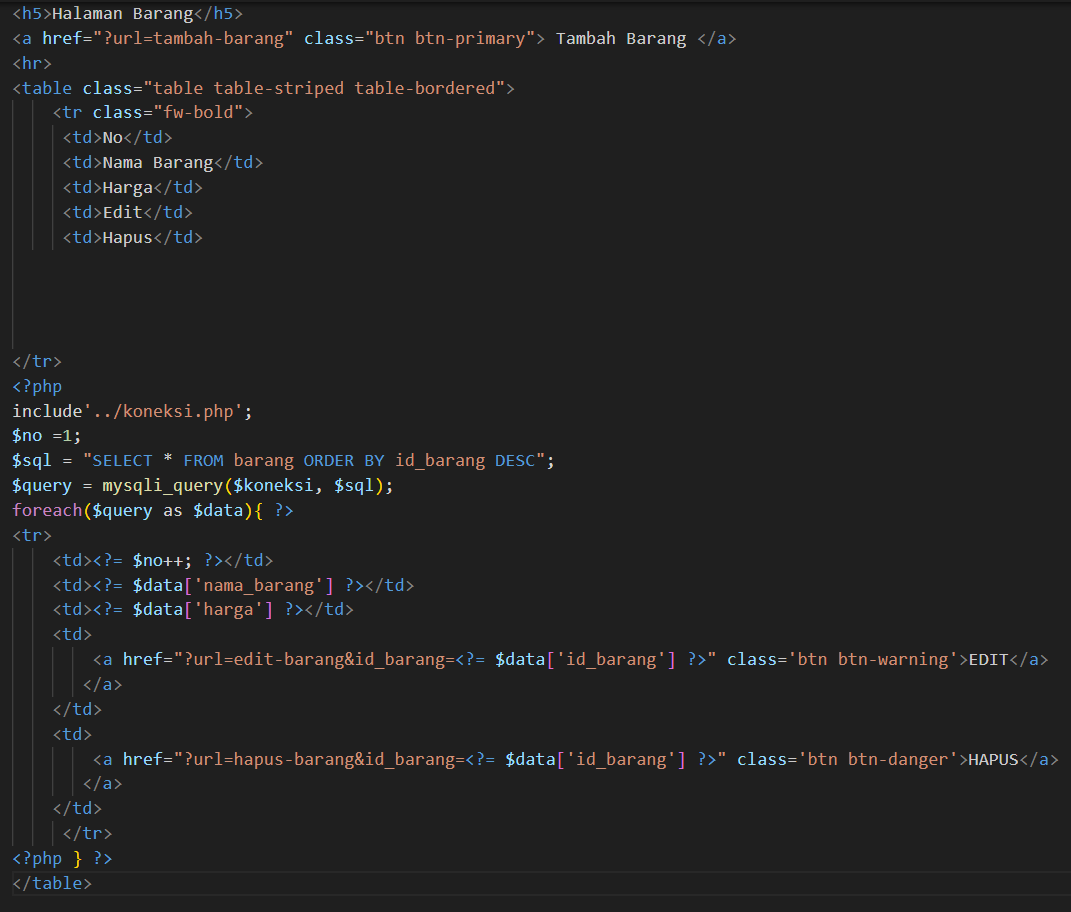
**`<button type=”reset” class=”btn btn-warning”> KOSONGKAN </button>` adalah elemen HTML yang digunakan untuk mereset atau mengosongkan formular web.**

**Bagian `<div>` yang ditutup dengan `</div> dan `</form>` adalah penutup untuk elemen-elemen HTML dalam formular. Di sini, `</div>`**

**Menutup sebuah div, sedangkan`</form>` menutup sebuah form.**

**Elemen`<div> adalah elemen blok yang digunakan untuk mengelompokkan elemen-elemen HTML ke dalam sebuah kelompok atau container. Ini sering digunakan untuk membuat tata letak dalam dokumen HTML dan untuk menerapkan gaya dengan menggunakan CSS.**

**Sementara itu, elemen <form> adalah elemen HTML yang digunakan untuk membuat formulir web. Formulir digunakan untuk mengumpulkan input dari pengguna, seperti teks, angka, checkbox, dan lainnya, dan mengirimkannya ke server untuk diproses.**

**Penjelasan detil dari tiap baris kode pada bagian `<h5>Halaman Barang</h5>`:**

**<h5>Halaman Barang</h5>**

**<a href="?url=tambah-barang" class="btn btn-primary"> Tambah Barang </a>**

**<hr>**

**<table class="table table-striped table-bordered">**

**<tr class="fw-bold">**

**<td>No</td>**

**<td>Nama Barang</td>**

**<td>Harga</td>**

**<td>Edit</td>**

**<td>Hapus</td>**

**</tr>**

**Kode ini menampilkan judul halaman “Halaman Barang” di level heading 5(`<h5>`), kemudain menambahkan sebuah tombol “Tambah Barang” yang mengarah ke URL `url=tambah-barang`. Setelah itu, kode ini menambahkan menambahkan sebuah garis horizontal (`<hr>`), dan sebuah tabel dengan class `table`,`table-striped`, dan `table-bordered`. Baris pertama dari tabel adalah header dengan judul “No”, “Nama Barang”, “Harga”,”Edit”, dan “Hapus”,**

**<?php**

**include'../koneksi.php';**

**$no =1;**

**$sql = "SELECT \* FROM barang ORDER BY id\_barang DESC";**

**$query = mysqli\_query($koneksi, $sql);**

**foreach($query as $data){ ?>**

**Kode ini melakukan pengambilan data dari database dengan mengimpor file`koneksi.php`. kemudian mendefinisikan variabel `$no` dengan nilai awal 1. Selanjutnya, dijalankan sebuah query SQL untuk mengambil semua data dari tabel`barang` dan mengurutkannya berdasarkan `id\_barang` secara menurun. Hasil query disimpan dalam variabel`$query`,kemudian dijalankan loop`foreach` untuk setiap data yuang ditentukan.**

**<tr>**

**<td><?= $no++; ?></td>**

**<td><?= $data['nama\_barang'] ?></td>**

**<td><?= $data['harga'] ?></td>**

**<td>**

**<a href="?url=edit-barang&id\_barang=<?= $data['id\_barang'] ?>" class='btn btn-warning'>EDIT</a>**

**</a>**

**</td>**

**<td>**

**<a href="?url=hapus-barang&id\_barang=<?= $data['id\_barang'] ?>" class='btn btn-danger'>HAPUS</a>**

**</a>**

**</td>**

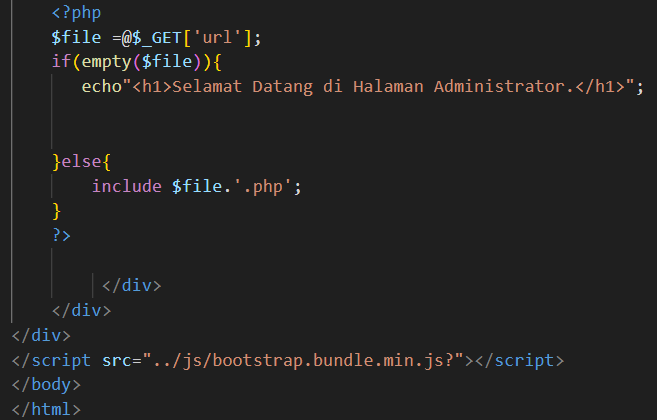
**</tr>**

**Kode ini menampilkan setiap baris data dari tabel barang. Setiap baris terdiri dari nomor urut yang diperbarui melalui variabel`$no++`, nama barang dari data, harga dari data, serta dau tombol aksi “Edit”  
 dan “Hapus” yang mengarah ke URL dengan parameter `id\_barang` yang sesuai dengan id barang dari data tersebut. Setiap tombol memiliki class yang sesuai untuk tampilan yang diinginkan.**

**<?php } ?>**

**</table>**

**Kode ini menutup loop `foreach` dan menutup tag tabel (`</table>`).**



**Laporan Aplikasi dan Penjelasan Kode:**

**<?php**

**session\_start();**

**if(empty($\_SESSION['id\_petugas'])){**

**echo"<script>**

**alert('Maaf Anda Belum Login');**

**window.location.assign('../index2.php');**

**</script>";**

**}**

**if($\_SESSION['level']!='admin'){**

**echo"<script>**

**alert('Maaf Anda Bukan Sesi Admin');**

**window.location.assign('../index2.php');**

**</script>";**

**}**

**?>**

**Penjelasan:**

**`session\_start();`:Memulai atau melanjutkan sesi yang ada.**

**`if(empty($\_SESSION[‘id\_petugas’])){ … }`:Memeriksa apakah variabel sesi `id\_petugas` ksoong atau tidak ada. Jika kosong, ini menandakan bahwa pengguna belum login. Jika benar, tampilkan pesan peringatan dan arahkan kekmbali pengguna ke halaman login ('../index2.php').**

**`if($\_SESSION[‘level’]!=’admin’){ … }`:Memeriksa apakah pengguna memiliki Tingkat akses ‘admin’. Jika tidak, tmapilkan pesan peringatan dan arahkan Kembali pengguna ke halaman lgoin**

**('../index2.php').**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1">**

**<title>Admin / Petugas - Aplikasi Barang Sederhana .</title>**

**<link href="../css/bootstrap.min.css" rel = "stylesheet">**

**</head>**

**<body>**

**<div class="container mt-5">**

**<h3>Aplikasi Barang Sederhana.</h3>**

**<div class="alert alert-info ">**

**Anda Login Sebagai <b>ADMINISTRATOR</b> Aplikasi Barang Sederhana.**

**</div>**

**<a href="admin.php"class="btn btn-primary"> Administrator</a>**

**<a href="admin.php?url=logout"class="btn btn-primary"> Logout</a>**

**<a href="admin.php?url=barang"class="btn btn-primary">Barang</a>**

**<a href="admin.php?url=pembelian"class="btn btn-primary">Pembelian</a>**

**<div class="car mt-3">**

**<div class="card-body">**

**<!-- ini isi web kita -->**

**<?php**

**$file =@$\_GET['url'];**

**if(empty($file)){**

**echo"<h1>Selamat Datang di Halaman Administrator.</h1>";**

**}else{**

**include $file.'.php';**

**}**

**?>**

**</div>**

**</div>**

**</div>**

**</script src="../js/bootstrap.bundle.min.js?"></script>**

**</body>**

**</html>**

**Penjelasan:**

**HTML markup untuk struktur dasar halaman web.**

**Bootstrap digunakan untuk styling.**

**Pada bagian PHP (`<?php … ?>`):**

**`$\_GET[‘url’]`:Mengambil nilai dari parameter URL ‘url’.**

**`if(empty($file)){ … }`:Memeriksa apakah nilai dari ‘url’ kosong. Jika iya, tampilkan pesan selamat datang.**

**`include $file.’.php’;`:Jika tidak kosong, termasuk file PHP yang sesuai dengan nilai ‘url’.**

**Terdapat link untuk menu Administrator, Logout, Barang, dan Pembelian.**

**Di bagian bawah, terdapat tag`<script>` yang mengarahkan ke file JavaScript Bootstrap. Namun, penutup tag script `</script>`**

**Tampaknya tidak diperlukan dan harus dihapus.**