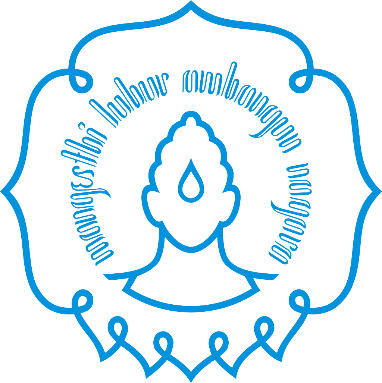
LAPORAN PRAKTIKUM PEMROGREAMAN WEB

PRAKTIKUM PEMROGRAMAN WEB

PRAKTIKUM IV CHECKBOX



# Disusun oleh :

Wahyu Bagas Dwi Prasetyo

V3922045

# Dosen

Masbahah, S.Pd., M.Pd

PS D-III TEKNIK INFORMATIKA SEKOLAH VOKASI UNIVERSITAS SEBELAS MARET 2023

# BAB I PENDAHULUAN

* 1. **Latar Belakang**

# Seiring dengan perkembangan teknologi internet, permintaan untuk aplikasi web terus meningkat. Aplikasi web digunakan untuk berbagai tujuan, mulai dari bisnis dan e-commerce hingga pendidikan dan hiburan.

# Dalam pembuatan aplikasi web, ada beberapa tahapan yang harus dilalui, seperti analisis kebutuhan, perancangan dan pengembangan, dan pengujian. Selain itu, ada juga beberapa faktor yang perlu diperhatikan dalam pembuatan aplikasi web, seperti performa, keamanan, dan skalabilitas.

# Pemrograman web juga melibatkan penggunaan database, seperti MySQL, PostgreSQL, atau MongoDB, untuk menyimpan data dan informasi dari aplikasi web. Selain itu, ada juga teknologi web seperti framework dan CMS (Content Management System) yang membantu dalam membangun aplikasi web dengan lebih efisien dan efektif.

# Dalam keseluruhan, pemrograman web adalah bidang yang berkembang pesat dan terus berubah seiring dengan perkembangan teknologi internet. Itu menjadi salah satu bidang yang sangat menarik dan menjanjikan bagi para pengembang dan pelajar yang ingin memperdalam keterampilan mereka dalam membangun aplikasi web yang kompleks dan inovatif

# .Tujuan

* + 1. Mahasiswa mampu memahami konsep pembuat pembuatan web
    2. Mahasiswa mampu memahami konsep dasar html css dan php
    3. Mahasiswa mampu memahami konsep form

# Manfaat

* + 1. Mahasiswa mampu membuat website
    2. Mahasiswa dapat menjadikan tugas ini sebagai portofolio
    3. Mahasisiswa dapat memahami konsep pembuatan design web

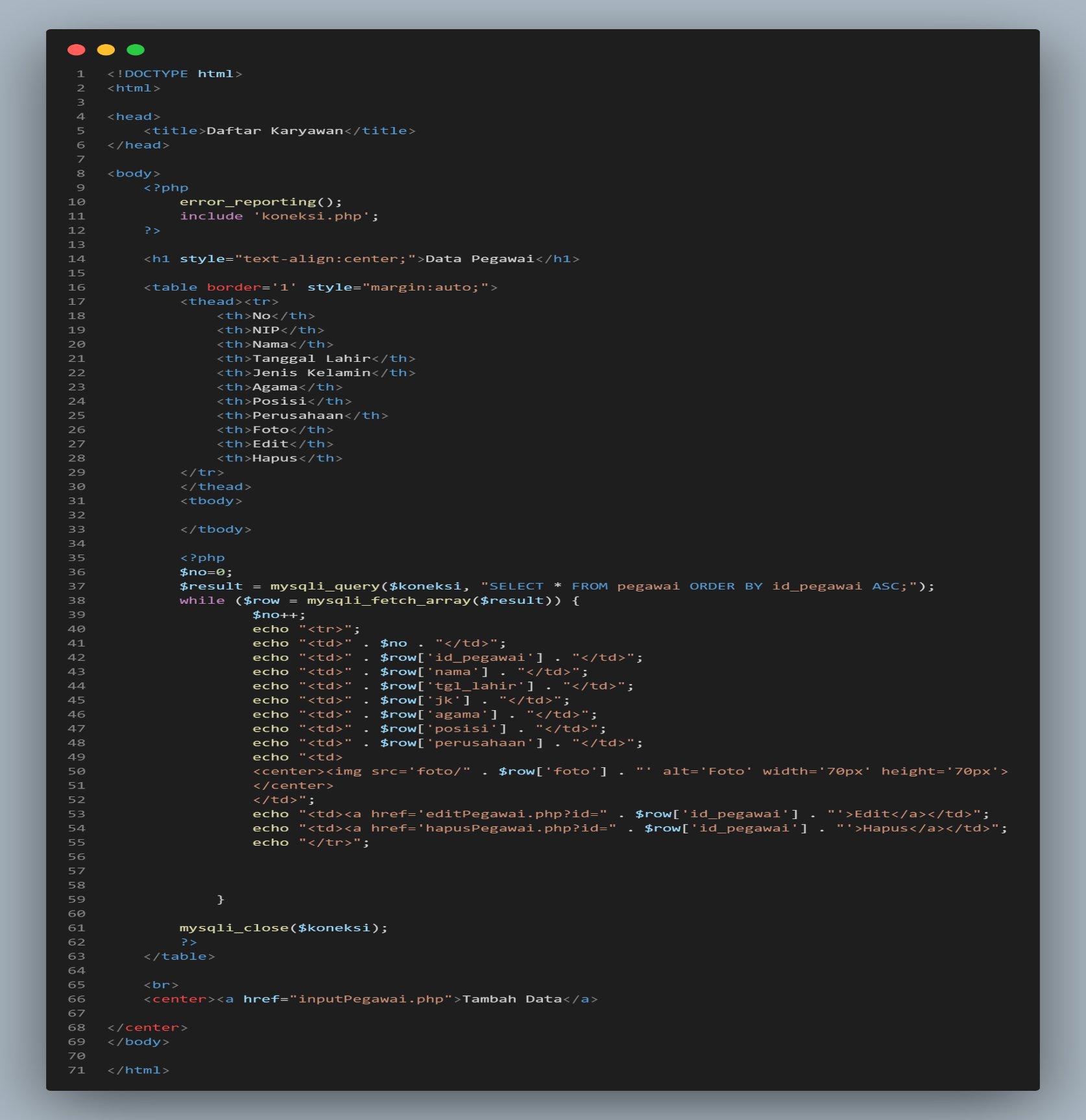
# Alat dan Bahan

* + 1. Komputer/laptop
    2. OS
    3. XAMPP
    4. Visual Code

# BAB II TINJAUAN PUSTAKA

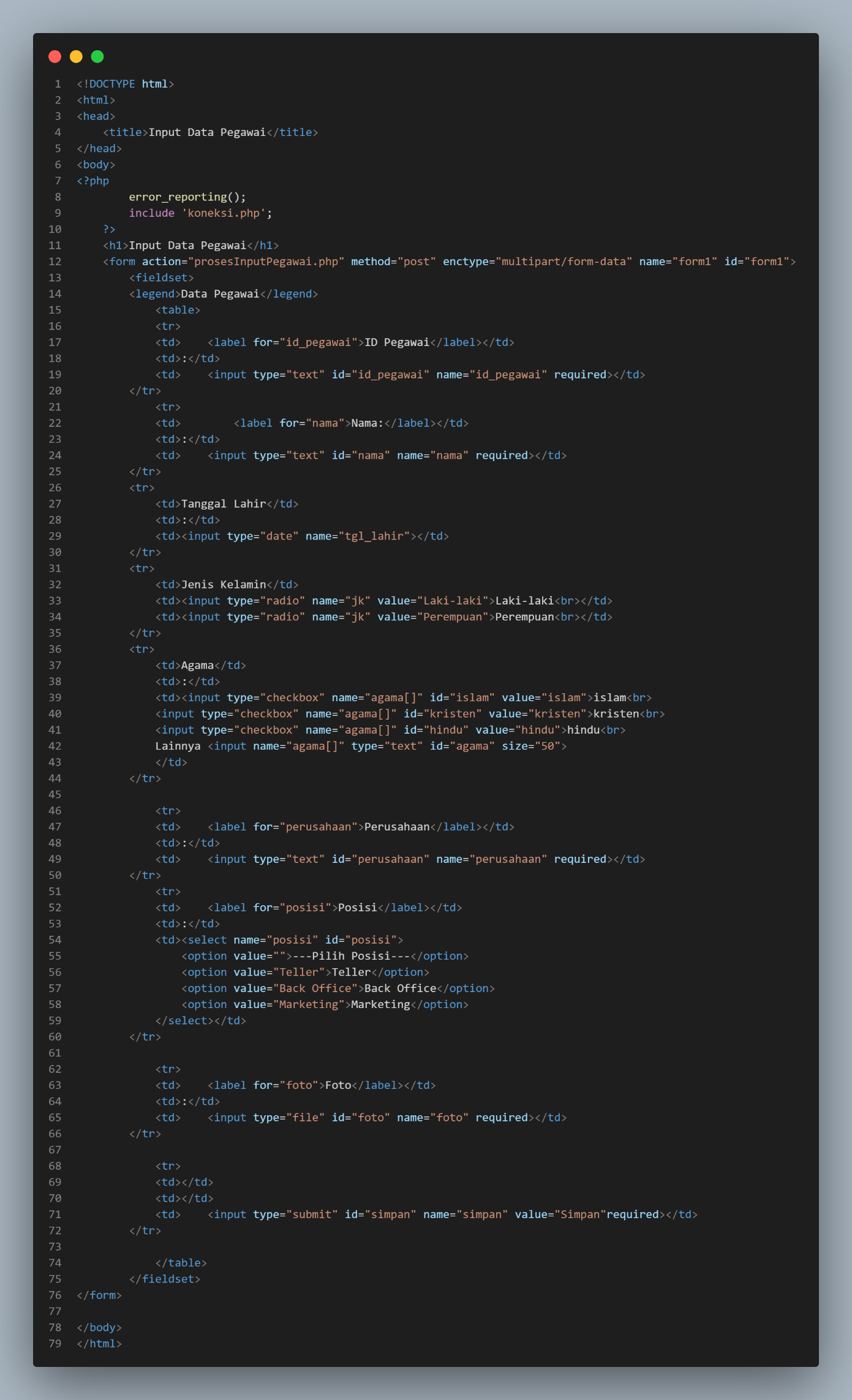
Pemrograman web adalah proses merancang, membuat, dan memelihara aplikasi atau situs web menggunakan bahasa pemrograman tertentu seperti HTML, CSS, dan JavaScript. Pemrograman web melibatkan penggunaan berbagai teknologi dan alat yang digunakan untuk mengembangkan berbagai jenis aplikasi web seperti website statis, website dinamis, aplikasi web, dan e-commerce. Hal ini melibatkan pembuatan dan pengaturan konten, struktur, tata letak, interaksi, dan fungsionalitas dari aplikasi web yang dirancang. Pemrograman web juga mencakup aspek keamanan, optimasi, dan pengujian dari aplikasi web yang dibuat.Salah satu cara membuat website bisa dengan menggunakan mysql. MySQL adalah sistem manajemen basis data hubungan (RDMS) yang cepat dan mudah digunakan untuk berbagai kebutuhan. MySQL dikembangkan oleh MySQL AB di Swedia. MySQL berlisensi open source dan menyediakan banyak fitur yang menjadi keunggulan MySQL.

# BAB III

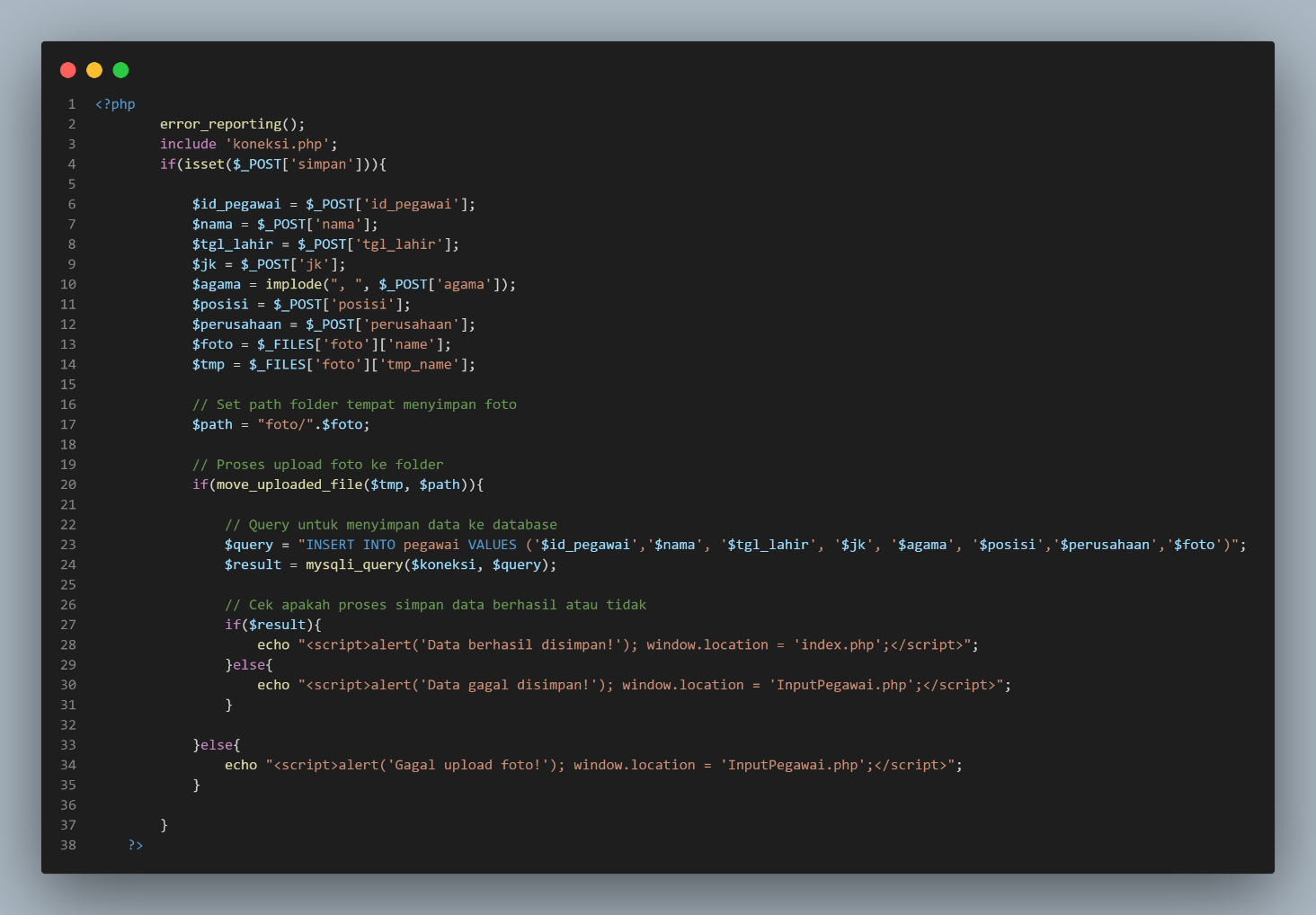
**HASIL DAN PEMBAHASAN**

****

****







****

# Penjelasan:

# Gambar 1:

* **<!DOCTYPE html>** : Menandakan bahwa dokumen ini adalah dokumen HTML
* **<html>...</html>** : Tag yang menandakan awal dan akhir dari dokumen HTML
* **<head>...</head>** : Tag yang berisi informasi tentang dokumen seperti judul halaman
* **<title>Daftar Karyawan</title>** : Tag untuk memberikan judul halaman web
* **<body>...</body>** : Tag yang berisi tampilan halaman web
* **<?php ... ?>** : Tag PHP yang memungkinkan penulis kode untuk menambahkan kode PHP di dalam dokumen HTML
* **error\_reporting();** : Kode PHP untuk menonaktifkan laporan kesalahan
* **include 'koneksi.php';** : Kode PHP untuk menyertakan file **koneksi.php** yang berfungsi untuk melakukan koneksi ke database
* **<h1 style="text-align:center;">Data Pegawai</h1>** : Tag HTML untuk menampilkan judul tabel pegawai
* **<table border='1' style="margin:auto;">...</table>** : Tag HTML untuk membuat tabel daftar karyawan
* **<thead>...</thead>** : Tag HTML untuk bagian kepala tabel
* **<tbody>...</tbody>** : Tag HTML untuk bagian isi tabel
* **$no=0;** : Variabel untuk menyimpan nomor urut pegawai
* **$result = mysqli\_query($koneksi, "SELECT \* FROM pegawai ORDER BY id\_pegawai ASC;");** : Kode PHP untuk melakukan query ke database dan mengambil data dari tabel pegawai
* **while ($row = mysqli\_fetch\_array($result)) { ... }** : Kode PHP untuk menampilkan setiap baris data dari tabel pegawai
* **mysqli\_close($koneksi);** : Kode PHP untuk menutup koneksi ke database setelah query selesai dilakukan
* **<a href="inputPegawai.php">Tambah Data</a>** : Tag HTML untuk membuat tautan ke halaman **inputPegawai.php** untuk menambahkan data baru ke dalam tabel.

# Gambar 2:

1. Pada baris ke-4 hingga 11, terdapat script PHP yang digunakan untuk memeriksa apakah parameter id sudah diterima dari URL atau belum. Jika sudah diterima, maka akan dieksekusi query untuk mengambil data pegawai yang sesuai dengan id tersebut. Jika tidak, maka akan dilakukan redirect ke halaman index.php.
2. Pada baris ke-16 hingga 39, terdapat form HTML yang digunakan untuk menampilkan data pegawai yang akan di-edit. Data pegawai yang diambil dari database tadi akan ditampilkan di dalam elemen form input seperti input text, radio button, checkbox, dan select. Data pegawai juga akan ditampilkan pada elemen file input untuk menampilkan foto pegawai.
3. Pada baris ke-41 hingga 53, terdapat tag input submit yang digunakan untuk mengirim data ke halaman prosesEditPegawai.php untuk melakukan update data pegawai yang di-edit. Ketika tombol Edit ditekan, maka data pada form akan dikirimkan ke halaman prosesEditPegawai.php untuk diolah lebih lanjut.

Gambar 3:

Pada baris ke-1, terdapat deklarasi DOCTYPE yang menentukan tipe dokumen HTML yang digunakan.

Kemudian, di dalam tag <html> pada baris ke-2, terdapat beberapa tag yang menentukan struktur halaman HTML seperti <head> yang berisi informasi tentang halaman dan <body> yang berisi isi dari halaman.

Pada baris ke-4, terdapat kode PHP yang digunakan untuk memanggil file koneksi.php yang menghubungkan dengan database. Kemudian, pada baris ke-7 terdapat tag <h1> yang menampilkan judul halaman.

Pada baris ke-8 sampai ke-43 terdapat form input data pegawai yang terdiri dari beberapa field, seperti ID Pegawai, Nama, Tanggal Lahir, Jenis Kelamin, Agama, Perusahaan, Posisi, dan Foto.

Setiap field dilengkapi dengan label dan elemen HTML seperti input text, radio, checkbox, select, dan file.

Pada baris ke-45 sampai ke-50, terdapat tombol submit yang digunakan untuk menyimpan data pegawai yang diinputkan ke dalam database melalui file prosesInputPegawai.php yang ditentukan pada atribut action pada tag form.

Gambar 4;

* Baris 2: Memanggil file koneksi.php yang berisi script untuk menghubungkan ke database.
* Baris 5-12: Mendapatkan data yang di-submit dari form dengan menggunakan variabel $\_POST. Variabel $id\_pegawai, $nama, $tgl\_lahir, $jk, $agama, $posisi, $perusahaan, $foto, dan $tmp masing-masing menyimpan data dari input dengan nama yang sama pada form.
* Baris 15-33: Cek apakah ada foto yang diupload atau tidak. Jika ada, maka script akan menghapus foto lama dan menyimpan foto baru yang diupload pada direktori "foto/". Proses menghapus foto lama dilakukan dengan mengambil data dari database untuk memeriksa apakah ada foto lama yang harus dihapus dan menghapusnya dengan fungsi unlink(). Setelah itu, nama file baru akan diubah dengan menambahkan prefix unik pada nama file menggunakan fungsi uniqid(). Foto yang baru diupload akan disimpan pada direktori "foto/" menggunakan fungsi move\_uploaded\_file().
* Baris 35-41: Jika tidak ada foto yang diupload, maka script akan mengambil data foto lama dari database dan menyimpannya dalam variabel $foto.
* Baris 44-50: Data pegawai akan diupdate ke database dengan menggunakan fungsi mysqli\_query(). Query UPDATE akan dijalankan dengan mengubah data pada kolom nama, tgl\_lahir, jk, agama, posisi, perusahaan, dan foto pada baris dengan id\_pegawai tertentu.
* Baris 53-59: Jika query berhasil dijalankan, maka akan ditampilkan pesan sukses menggunakan fungsi alert() pada javascript dan halaman akan diredirect ke index.php. Jika query gagal dijalankan, maka script akan memberikan pesan error yang berisi informasi tentang error yang terjadi pada query.

Gambar 5:

Berikut adalah penjelasan detail dari setiap baris kode tersebut:

1. **error\_reporting();** adalah sebuah fungsi PHP yang digunakan untuk menonaktifkan tampilan error atau pesan kesalahan pada halaman web. Jika tidak diberikan parameter, maka fungsi ini akan menonaktifkan semua jenis error.
2. **include 'koneksi.php';** adalah sebuah perintah untuk memasukkan file koneksi.php ke dalam script PHP. File ini berisi kode untuk membuat koneksi ke database MySQL.
3. **if(isset($\_POST['simpan'])){...}** adalah sebuah kondisi untuk mengecek apakah tombol "simpan" pada halaman sebelumnya telah ditekan atau tidak. Jika tombol tersebut ditekan, maka script akan melanjutkan eksekusi.
4. **$id\_pegawai = $\_POST['id\_pegawai'];** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "id\_pegawai".
5. **$nama = $\_POST['nama'];** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "nama".
6. **$tgl\_lahir = $\_POST['tgl\_lahir'];** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "tgl\_lahir".
7. **$jk = $\_POST['jk'];** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "jk".
8. **$agama = implode(", ", $\_POST['agama']);** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "agama". Jika nilai yang dikirimkan berupa array, maka perintah **implode** akan menggabungkan semua elemen array tersebut menjadi sebuah string dengan pemisah koma dan spasi.
9. **$posisi = $\_POST['posisi'];** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "posisi".
10. **$perusahaan = $\_POST['perusahaan'];** adalah sebuah perintah untuk mengambil nilai yang dikirimkan melalui form dengan method POST dan memiliki nama "perusahaan".
11. **$foto = $\_FILES['foto']['name'];** adalah sebuah perintah untuk mengambil nama file foto yang diupload oleh user melalui form dengan method POST dan memiliki atribut "name" bernama "foto".
12. **$tmp = $\_FILES['foto']['tmp\_name'];** adalah sebuah perintah untuk mengambil lokasi sementara file foto yang diupload oleh user melalui form dengan method POST dan memiliki atribut "name" bernama "foto".
13. **$path = "foto/".$foto;** adalah sebuah perintah untuk menentukan path atau lokasi folder tempat menyimpan foto. Dalam hal ini, foto akan disimpan dalam folder "foto" yang berada pada root direktori web.
14. **if(move\_uploaded\_file($tmp, $path)){...}** adalah sebuah kondisi untuk mengecek apakah proses upload foto ke dalam folder berhasil atau tidak. Jika berhasil, maka script akan melanjutkan eksekusi.
15. `$query = "INSERT INTO pegawai VALUES ('$id\_pegawai','$nama', '$tgl\_lahir', '$jk', '$agama',

Gambar 6:

tersebut:

1. **$host = "localhost";** - variabel **$host** menyimpan nilai string "localhost" yang akan digunakan untuk mengidentifikasi host server MySQL.
2. **$user = "root";** - variabel **$user** menyimpan nama pengguna yang digunakan untuk mengakses database MySQL.
3. **$password = "";** - variabel **$password** menyimpan kata sandi yang digunakan untuk mengakses database MySQL. Dalam contoh ini, nilai kosong menunjukkan bahwa tidak ada kata sandi yang diperlukan.
4. **$dbname = "idcard";** - variabel **$dbname** menyimpan nama database yang akan digunakan.
5. **$koneksi = mysqli\_connect($host, $user, $password, $dbname);** - fungsi **mysqli\_connect()** digunakan untuk membuat koneksi ke database MySQL menggunakan nilai variabel yang telah didefinisikan sebelumnya. Fungsi ini mengembalikan objek koneksi, yang kemudian disimpan dalam variabel **$koneksi**.
6. **if (!$koneksi) { die("Koneksi gagal: " . mysqli\_connect\_error()); }** - blok **if** digunakan untuk memeriksa apakah koneksi ke database berhasil dibuat. Jika koneksi gagal, fungsi **mysqli\_connect\_error()** akan mengembalikan pesan kesalahan yang dihasilkan oleh MySQL, yang kemudian ditampilkan dengan menggunakan fungsi **die()**. Fungsi **die()** digunakan untuk menghentikan eksekusi program jika koneksi gagal.

# HASIL OUTPUT PROGRAM:

# 

# BAB IV KESIMPULAN

* 1. **Kesimpulan**

# Pemrograman web sangat berperan dalam perkembangan teknologi dengan penggunaan Data yang dikelola oleh database diharapkan dapat terorganisasi dengan baik. Data yang tertata dengan baik dapat mempermudah orang untuk melihat data tersebut. MySQL adalah tempat yang baik untuk menyimpan data.

# Saran

Saat mengetik di cmd prompt pada vs code dan cmd ada baiknya memperhatikan besar kecilnya kalimat agar tidak terjadi salah input yang menyebabkan eror program

# DAFTAR PUSTAKA

Pamungkas, C. A. (2017). *Pengantar dan Implementasi Basis Data*. Deepublish

Novendri, M. S., Saputra, A., & Firman, C. E. (2019). Aplikasi Inventaris Barang Pada Mts Nurul Islam Dumai Menggunakan Php Dan Mysql. *lentera dumai*, *10*(2).