

## **TUGAS INDIVIDU**

### **LAPORAN PRAKTIKUM 7 PEMROGRAMAN WEB**

Disusun untuk memenuhi tugas mata kuliah Pemrograman Web

**Dosen Pengampu:**

**Dr. Eng. Agussalim, M.T.**



**Disusun oleh:**

**Muhammad Surya Adhi Setiawan (21082010167)**

**Program Studi Sistem Informasi**

**Fakultas Ilmu Komputer**

**Universitas Pembangunan Nasional "Veteran" Jawa Timur**

**2023**

## Tujuan Praktikum

1. Memahami langkah-langkah koneksi PHP dengan MySQL.
2. Memahami perbedaan penggunaan mysql dengan mysqli.

## Landasan Teori

- **PHP**

PHP (Hypertext Preprocessor) adalah bahasa pemrograman server-side yang digunakan untuk membuat aplikasi web dinamis. PHP dapat berinteraksi dengan berbagai jenis database, termasuk MySQL.

- **MySQL**

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang digunakan untuk menyimpan dan mengelola data. MySQL sangat populer di kalangan pengembang web karena mudah digunakan dan gratis.

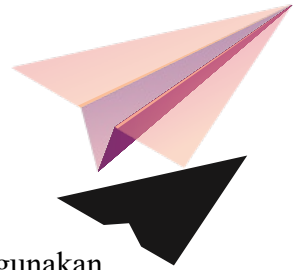
- **Koneksi antara PHP dan MySQL**

Untuk menghubungkan PHP dengan MySQL, Anda perlu membuat koneksi ke server MySQL menggunakan fungsi `mysqli_connect()` atau `mysql_connect()`. Setelah koneksi berhasil dibuat, Anda dapat melakukan operasi pada database seperti `SELECT`, `INSERT`, `UPDATE`, dan `DELETE` menggunakan fungsi `mysqli_query()` atau `mysql_query()`.

- **Perbedaan penggunaan mysql dengan mysqli**

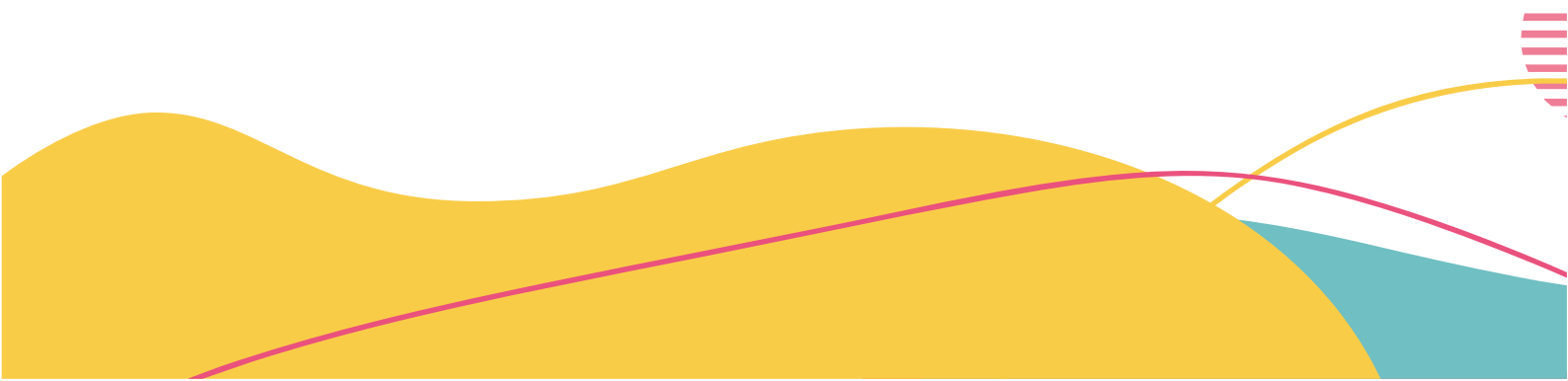
`mysql` adalah ekstensi PHP yang digunakan untuk mengakses database MySQL. Namun, ekstensi ini sudah tidak disarankan lagi karena sudah tidak dikembangkan lagi dan rentan terhadap serangan SQL injection. `mysqli` adalah ekstensi PHP yang lebih aman dan memiliki fitur yang lebih lengkap dibandingkan `mysql`. Ekstensi ini juga mendukung koneksi ke server MySQL melalui protokol TCP/IP dan protokol Unix socket. Selain itu, `mysqli` juga mendukung prepared statement, transaksi, dan prosedur tersimpan.

## **Tools dan Software yang Digunakan**



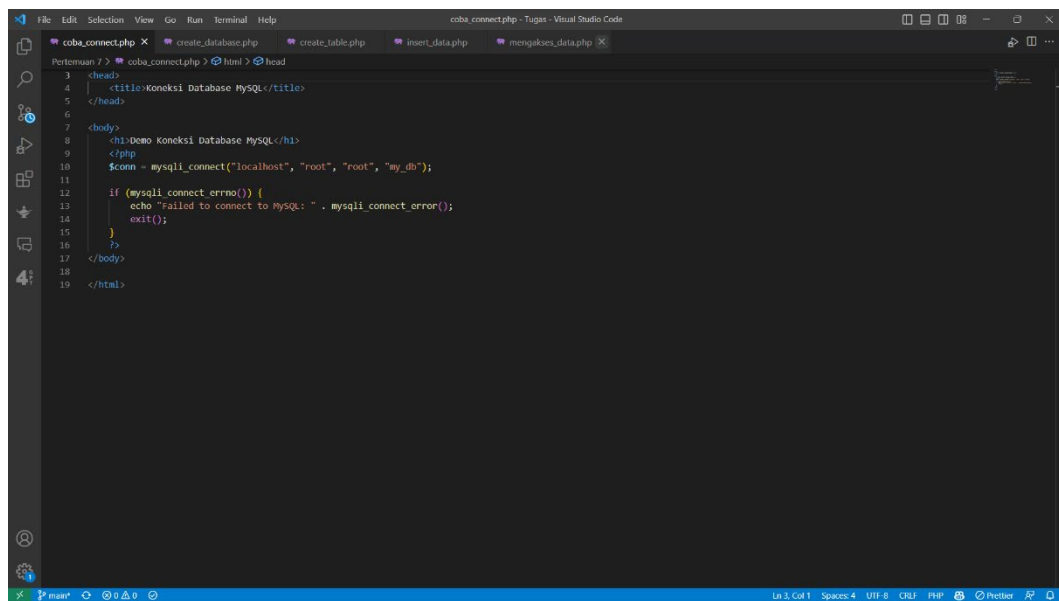
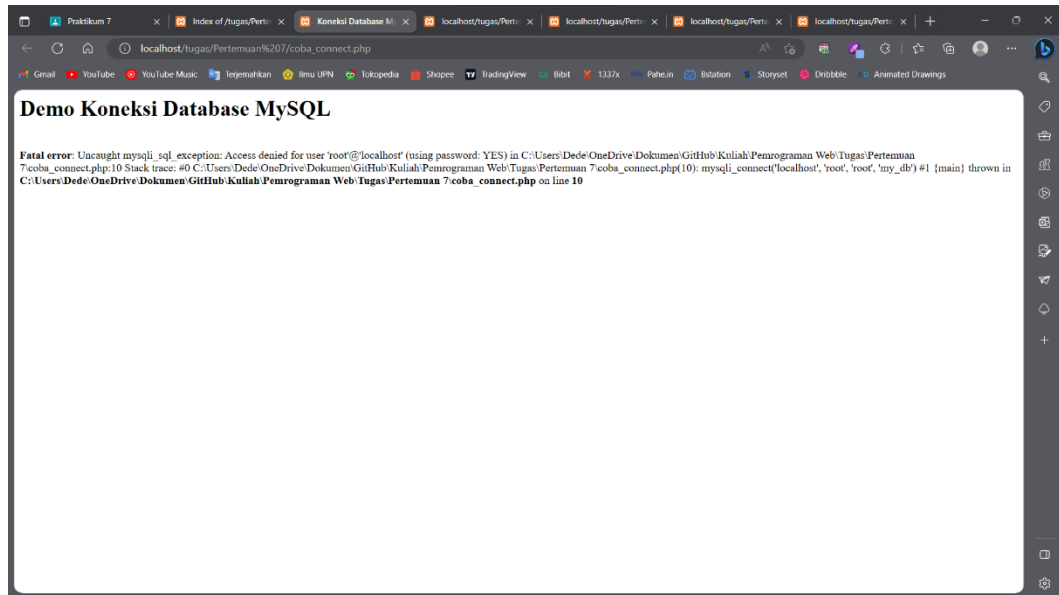
Untuk melakukan praktikum ini, beberapa tools dan software yang saya digunakan adalah sebagai berikut:

1. Text Editor: Seperti Visual Studio Code, Notepad++, Sublime Text, atau Brackets. Ini adalah alat utama untuk membuat dan mengedit file PHP.
2. Web Browser: Seperti Google Chrome, Mozilla Firefox, atau Microsoft Edge. Ini diperlukan untuk melihat hasil dari file PHP yang dibuat.



# Hasil Praktikum

## Demo Connect





### Analisis:

Kode PHP tersebut merupakan contoh sederhana untuk melakukan koneksi ke database MySQL menggunakan fungsi `mysqli_connect()`. Pertama-tama, dilakukan pemanggilan fungsi `mysqli_connect()` dengan parameter sebagai berikut:

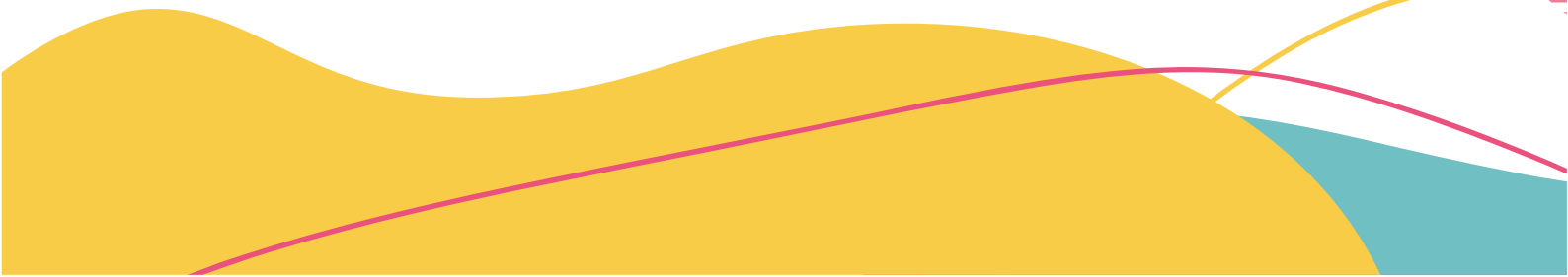
"localhost" : nama host atau alamat server database MySQL yang akan dihubungi. Dalam contoh ini, host yang digunakan adalah localhost atau server lokal.

"root" : username untuk mengakses database MySQL. Dalam contoh ini, username yang digunakan adalah root.

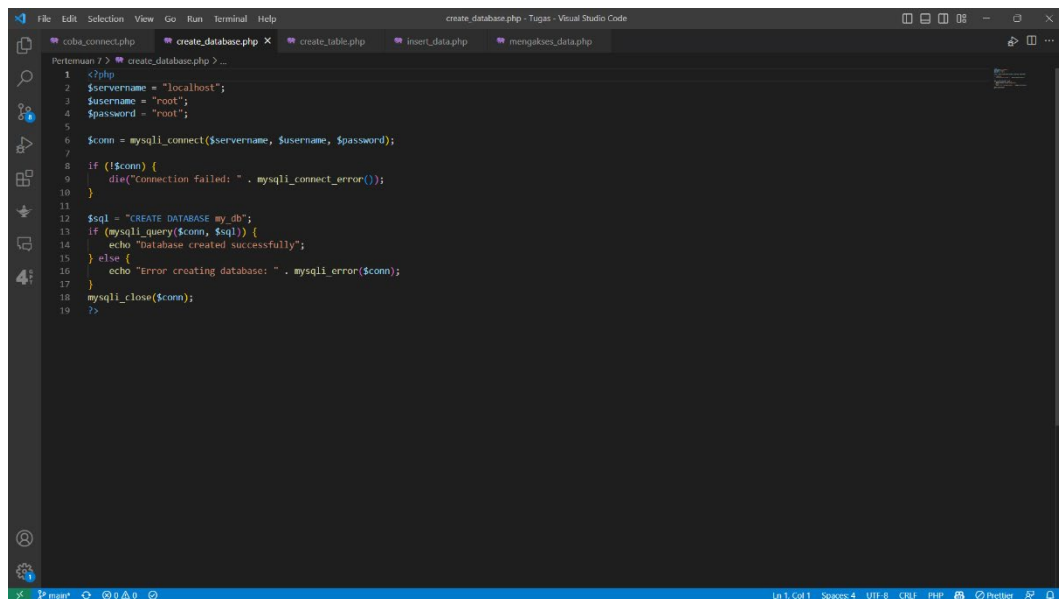
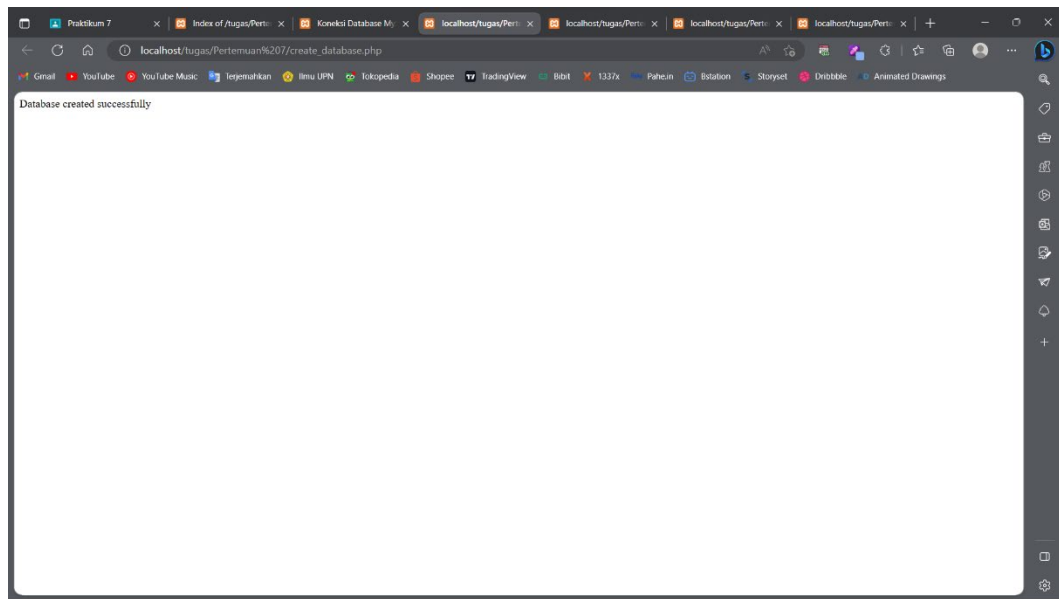
"root" : password untuk mengakses database MySQL. Dalam contoh ini, password yang digunakan adalah root.

"my\_db" : nama database yang akan diakses. Dalam contoh ini, database yang digunakan adalah my\_db.

Setelah itu, dilakukan pengecekan apakah koneksi ke database berhasil atau tidak menggunakan fungsi `mysqli_connect_errno()`. Jika koneksi gagal, maka akan ditampilkan pesan error dengan menggunakan fungsi `mysqli_connect_error()`.



## Create Database





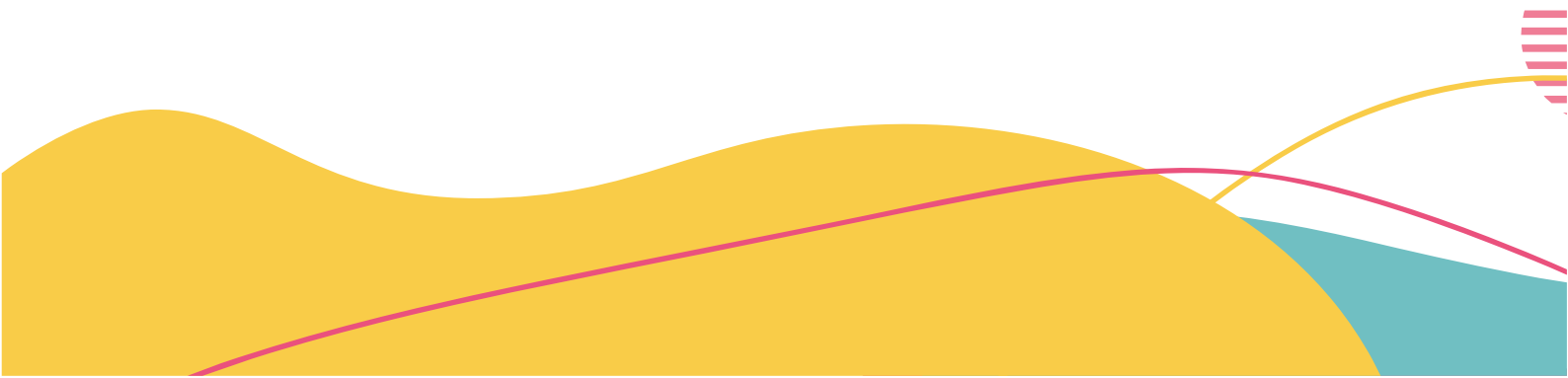
### Analisis:

Kode PHP tersebut merupakan contoh kode untuk membuat koneksi ke server MySQL dan membuat database baru dengan nama "my\_db". Berikut adalah analisis kode tersebut:

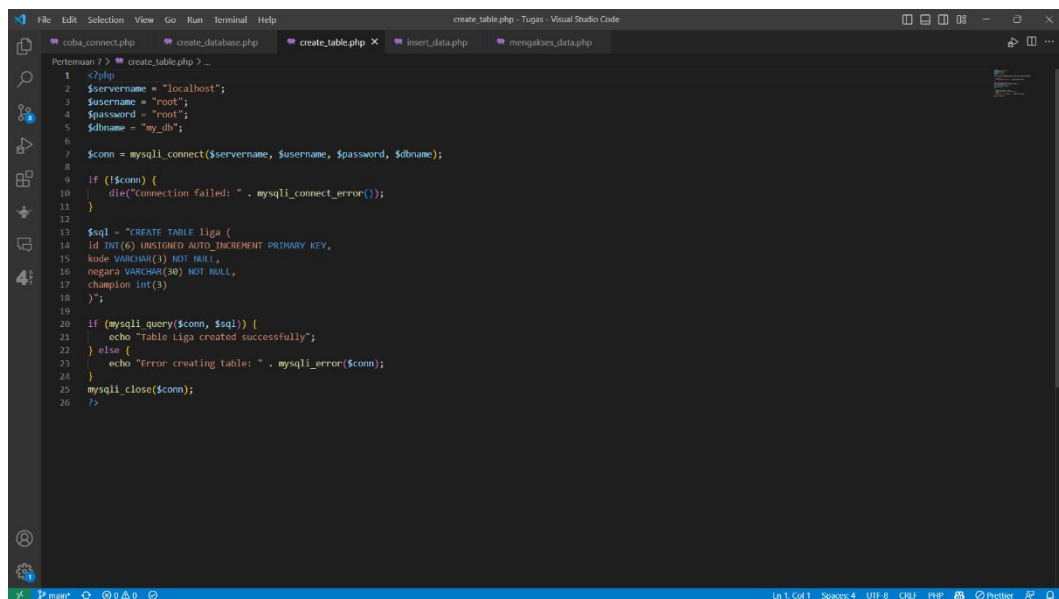
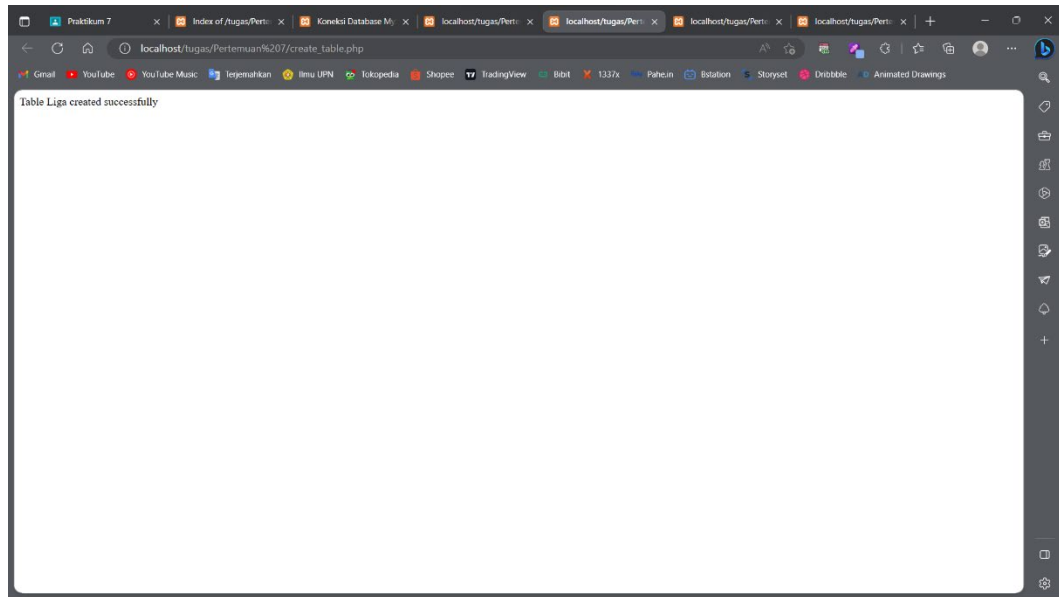
Variabel `$servername`, `$username`, dan `$password` digunakan untuk menyimpan informasi koneksi ke server MySQL. Variabel `$conn` digunakan untuk menyimpan koneksi ke server MySQL. Fungsi `mysqli_connect()` digunakan untuk membuat koneksi ke server MySQL dengan menggunakan informasi yang disimpan dalam variabel `$servername`, `$username`, dan `$password`. Jika koneksi berhasil dibuat, maka variabel `$conn` akan berisi objek koneksi. Jika koneksi gagal dibuat, maka fungsi `mysqli_connect_error()` akan mengembalikan pesan error dan program akan berhenti.

Setelah koneksi berhasil dibuat, variabel `$sql` digunakan untuk menyimpan perintah SQL untuk membuat database baru dengan nama "my\_db". Fungsi `mysqli_query()` digunakan untuk menjalankan perintah SQL yang disimpan dalam variabel `$sql`. Jika perintah SQL berhasil dijalankan, maka program akan menampilkan pesan "Database created successfully". Jika perintah SQL gagal dijalankan, maka fungsi `mysqli_error()` akan mengembalikan pesan error dan program akan menampilkan pesan "Error creating database: ".

Fungsi `mysqli_close()` digunakan untuk menutup koneksi ke server MySQL setelah program selesai dijalankan.



## Create Table





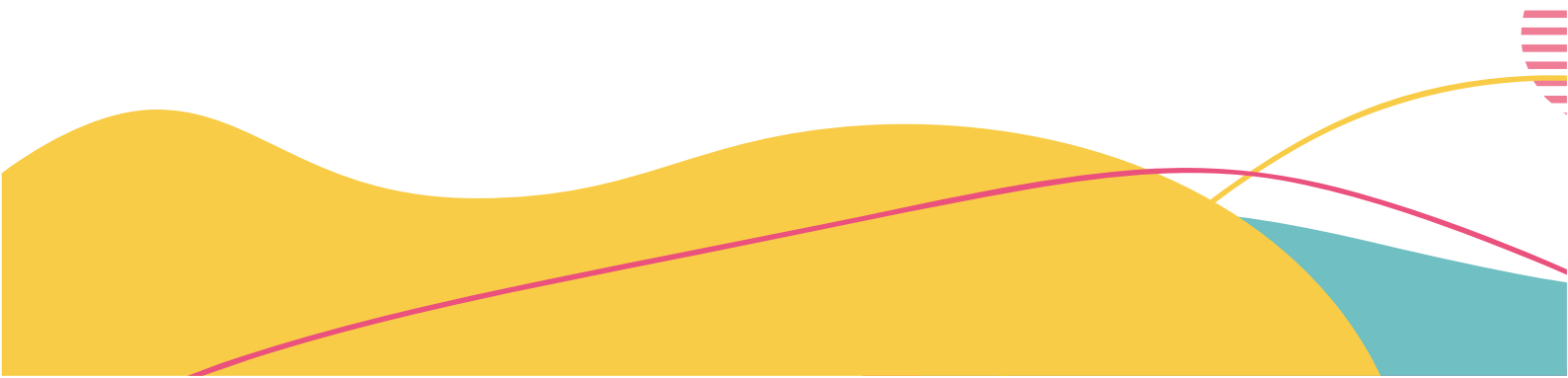


### Analisis:

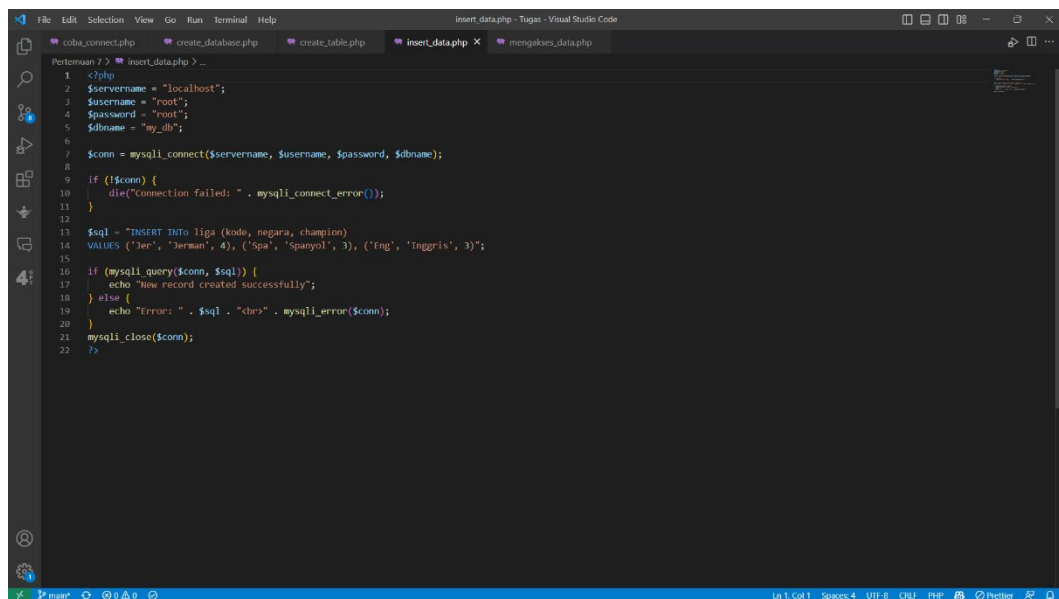
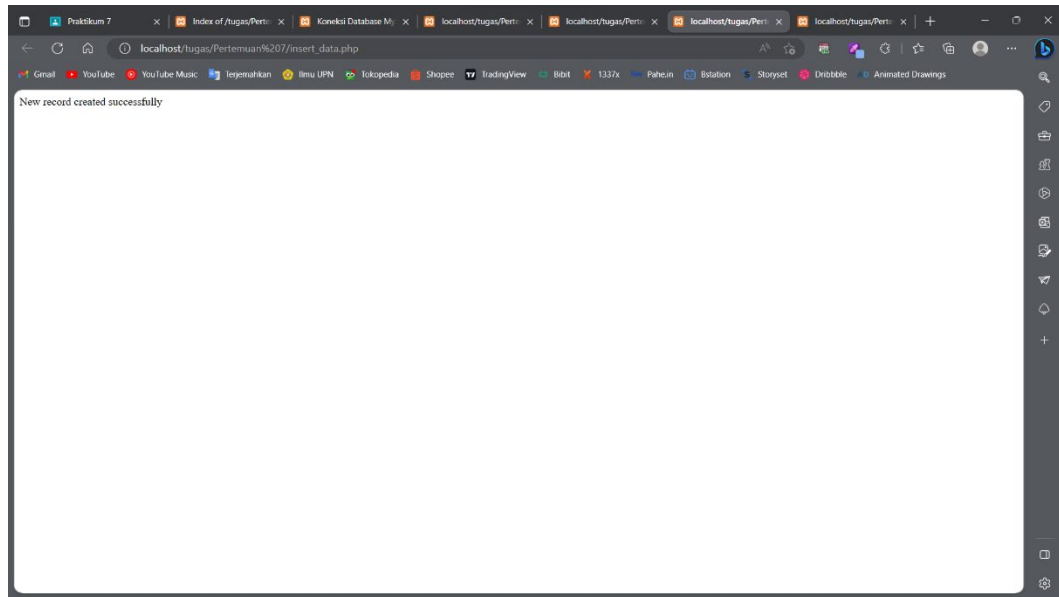
Kode PHP di atas merupakan sebuah script untuk membuat sebuah tabel pada database MySQL. Pertama-tama, script ini melakukan koneksi ke database dengan menggunakan fungsi `mysqli_connect()`. Jika koneksi berhasil, maka script akan menjalankan perintah SQL untuk membuat sebuah tabel dengan nama "liga". Tabel ini memiliki empat kolom, yaitu "id", "kode", "negara", dan "champion". Kolom "id" merupakan primary key yang di-generate secara otomatis dengan tipe data integer. Kolom "kode" dan "negara" memiliki tipe data varchar dan tidak boleh kosong (NOT NULL). Sedangkan kolom "champion" memiliki tipe data integer dengan panjang 3 digit.

Setelah perintah SQL dijalankan, script akan menampilkan pesan "Table Liga created successfully" jika tabel berhasil dibuat. Namun, jika terjadi kesalahan dalam pembuatan tabel, maka script akan menampilkan pesan "Error creating table" beserta pesan error yang dihasilkan oleh MySQL. Terakhir, script akan menutup koneksi ke database dengan menggunakan fungsi `mysqli_close()`.

Secara keseluruhan, script ini sangat berguna bagi pengembang web yang ingin membuat sebuah tabel pada database MySQL. Dengan menggunakan script ini, pengembang dapat membuat tabel dengan cepat dan mudah tanpa harus mengetik perintah SQL secara manual.



## Insert Data





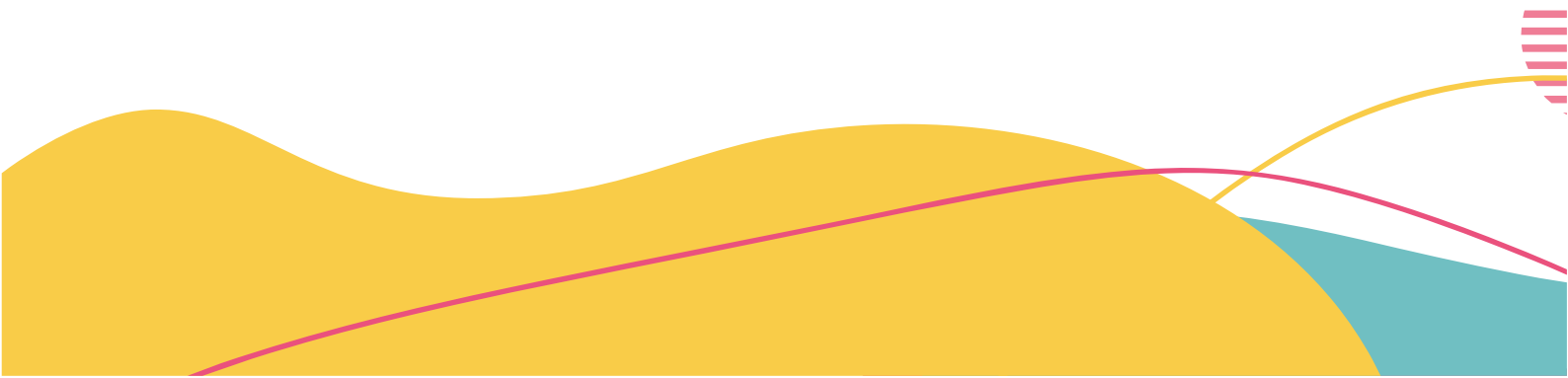
### Analisis:

Kode PHP di atas merupakan contoh penggunaan fungsi `mysqli_connect()` untuk melakukan koneksi ke database MySQL. Pada baris pertama hingga keempat, terdapat variabel yang berisi informasi mengenai server, username, password, dan nama database yang akan digunakan. Kemudian, pada baris keenam hingga kesepuluh, fungsi `mysqli_connect()` digunakan untuk melakukan koneksi ke database dengan menggunakan informasi yang telah didefinisikan sebelumnya.

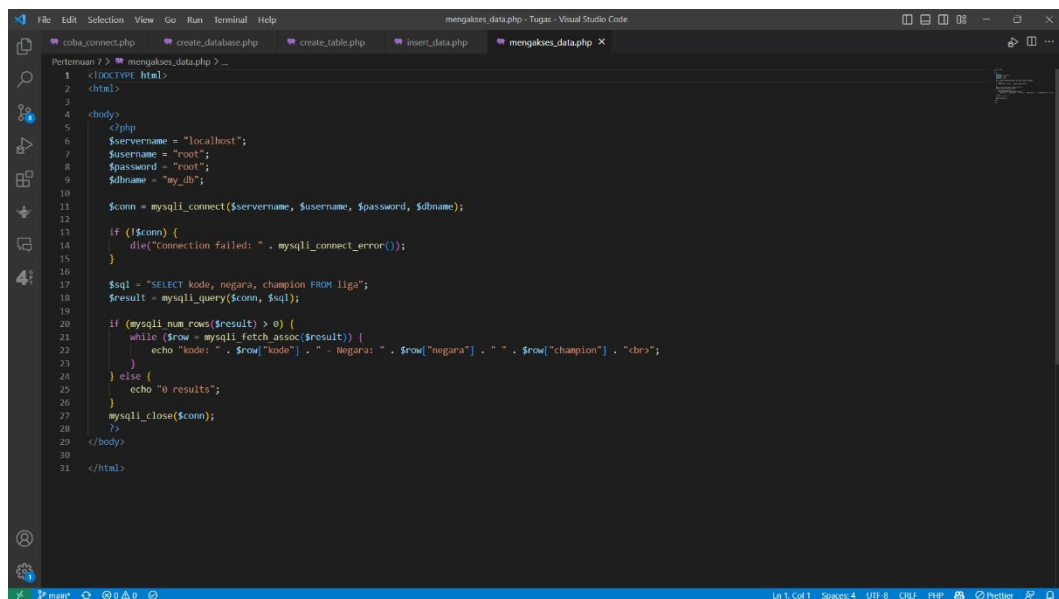
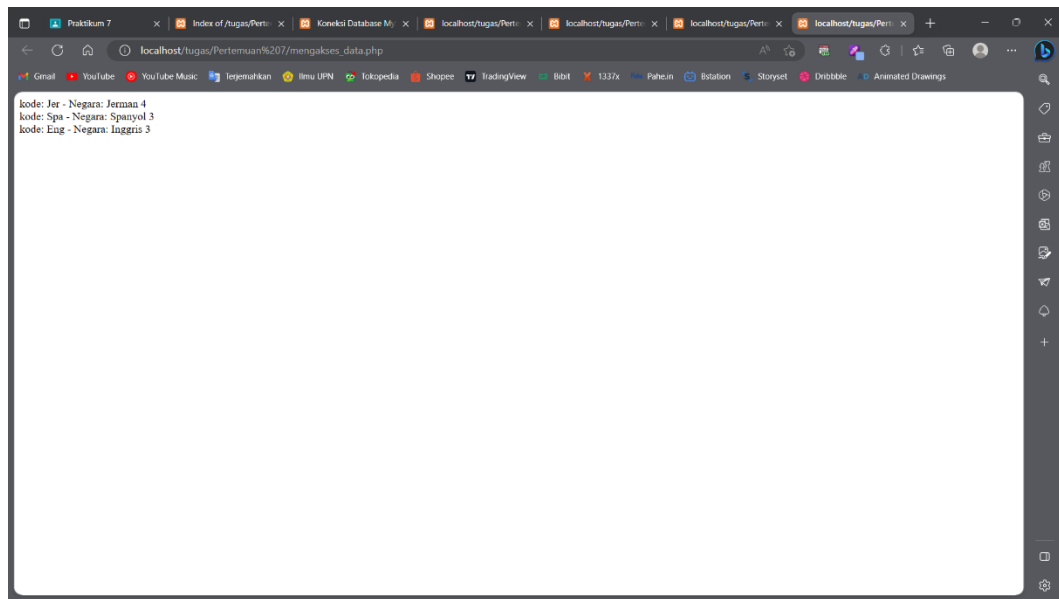
Setelah koneksi berhasil dilakukan, terdapat perintah SQL untuk melakukan operasi INSERT pada tabel liga dengan memasukkan beberapa data baru. Fungsi `mysqli_query()` digunakan untuk mengeksekusi perintah SQL tersebut dan mengembalikan nilai true jika berhasil atau false jika gagal.

Selanjutnya, terdapat kondisi if-else untuk mengecek apakah operasi INSERT berhasil atau tidak. Jika berhasil, maka akan ditampilkan pesan "New record created successfully". Namun, jika gagal, maka akan ditampilkan pesan "Error" beserta pesan error yang dihasilkan oleh MySQL.

Terakhir, terdapat fungsi `mysqli_close()` digunakan untuk menutup koneksi ke database. Dengan demikian, kode PHP di atas dapat digunakan sebagai contoh dasar untuk melakukan operasi INSERT pada database MySQL menggunakan PHP.



## Akses Data



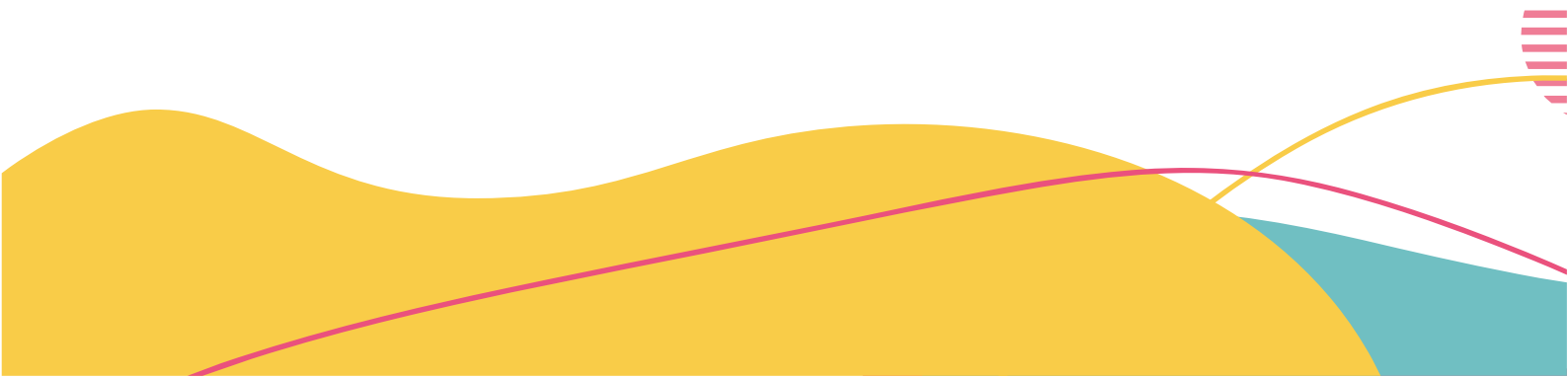


### Analisis:

Kode PHP tersebut merupakan sebuah script yang digunakan untuk mengambil data dari sebuah tabel pada database MySQL. Script tersebut terdiri dari beberapa variabel yang digunakan untuk menyimpan informasi mengenai server, username, password, dan nama database. Kemudian, script tersebut melakukan koneksi ke database menggunakan fungsi `mysqli_connect()` dan melakukan pengecekan apakah koneksi berhasil atau tidak menggunakan fungsi `mysqli_connect_error()`.

Setelah koneksi berhasil, script tersebut melakukan query untuk mengambil data dari tabel liga menggunakan perintah `SELECT`. Hasil query tersebut disimpan pada variabel `$result`. Kemudian, script tersebut melakukan pengecekan apakah terdapat data yang dihasilkan dari query menggunakan fungsi `mysqli_num_rows()`. Jika terdapat data, maka script tersebut akan melakukan perulangan menggunakan fungsi `mysqli_fetch_assoc()` untuk mengambil setiap baris data dan menampilkannya pada halaman web menggunakan fungsi `echo`.

Jika tidak terdapat data, maka script tersebut akan menampilkan pesan "0 results". Terakhir, script tersebut menutup koneksi ke database menggunakan fungsi `mysqli_close()`. Keseluruhan script tersebut merupakan contoh sederhana penggunaan PHP untuk mengambil data dari database MySQL dan menampilkannya pada halaman web.



# Tugas

## Tugas 1

Index of /tugas/pertemuan 7 x Buku Tamu x Data Pegawai x Data Pegawai x +

localhost/tugas/pertemuan707/tugas1.php

### Buku Tamu

Nama:

Email:

Isi:

ID	Nama	Email	Isi
1	Bambang Sujatmiko	bambang Sujatmiko@gmail.com	Halo, ini adalah pesan dari Bambang Sujatmiko
2	Budi Santoso	budisantoso@gmail.com	Halo, ini adalah pesan dari Budi Santoso
3	Dwi Cahyono	dwicahyono@gmail.com	Halo, ini adalah pesan dari Dwi Cahyono
4	Rudi Gunawan	rudigunawan@gmail.com	Halo, ini adalah pesan dari Rudi Gunawan

```
File Edit Selection View Go Run Terminal Help tugas1.php - Tugas - Visual Studio Code
tugas1.php x tugas2.php x tugas3.php x
Pertemuan 7 > tugas1.php > ...
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5 <title>Buku Tamu</title>
6 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css" integrity="sha384-xd0if...>
7 </head>
8
9 <body class="bg-dark text-light">
10 <div class="container">
11 <div class="text-center py-4">Buku Tamu</div>
12 <form method="post">
13 <div class="form-group">
14 <label for="nama">Nama</label>
15 <input type="text" class="form-control" id="nama" name="nama" required>
16 </div>
17 <div class="form-group">
18 <label for="email">Email</label>
19 <input type="email" class="form-control" id="email" name="email" required>
20 </div>
21 <div class="form-group">
22 <label for="isi">Isi</label>
23 <textarea class="form-control" id="isi" name="isi" rows="4" required></textarea>
24 </div>
25 <button type="submit" class="btn btn-primary">Kirim</button>
26 </form>
27 <br>
28 </body>
29 </html>
30
31 // konfigurasi database
32 $host = "localhost";
33 $username = "root";
34 $password = "root";
35
36 // Membuat koneksi ke database
37 $conn = mysqli_connect($host, $username, $password);
38
39 // Mengecek koneksi
40 if (!$conn) {
41     die("Koneksi gagal: " . mysqli_connect_error());
42 }
```



## Analisis:

Kode PHP di atas merupakan sebuah program sederhana untuk membuat buku tamu online. Program ini menggunakan teknologi web dengan memanfaatkan bahasa pemrograman PHP dan database MySQL. Program ini memiliki beberapa fitur, seperti menambahkan data tamu baru, menampilkan data tamu yang sudah ada, serta membuat database dan tabel jika belum ada.

Pada awal program, terdapat konfigurasi database yang harus diatur terlebih dahulu. Konfigurasi ini meliputi host, username, dan password untuk mengakses database. Setelah itu, program akan membuat koneksi ke database dengan menggunakan fungsi `mysqli_connect()`. Jika koneksi berhasil, maka program akan membuat database baru dengan menggunakan perintah SQL `CREATE DATABASE IF NOT EXISTS`. Jika database sudah ada, maka program akan menggunakan database tersebut dengan menggunakan fungsi `mysqli_select_db()`.

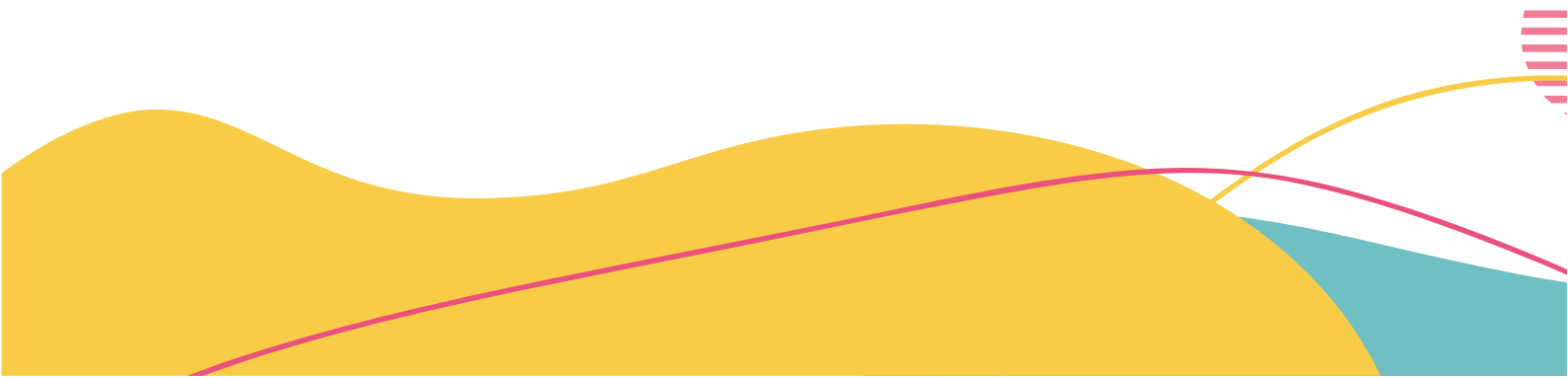
Setelah itu, program akan membuat tabel `buku_tamu` jika belum ada dengan menggunakan perintah SQL `CREATE TABLE IF NOT EXISTS`. Tabel ini memiliki beberapa kolom, seperti `ID_BT`, `NAMA`, `EMAIL`, dan `ISI`. Kolom `ID_BT` merupakan kolom yang digunakan sebagai primary key dan akan diisi secara otomatis oleh database. Kolom `NAMA`, `EMAIL`, dan `ISI` merupakan kolom yang akan diisi oleh data tamu.

Setelah tabel berhasil dibuat, program akan menambahkan data awal ke tabel `buku_tamu`. Namun, data awal ini di-comment terlebih dahulu agar tidak mengganggu saat program dijalankan. Data awal ini berisi beberapa data tamu yang sudah ada sebelumnya.

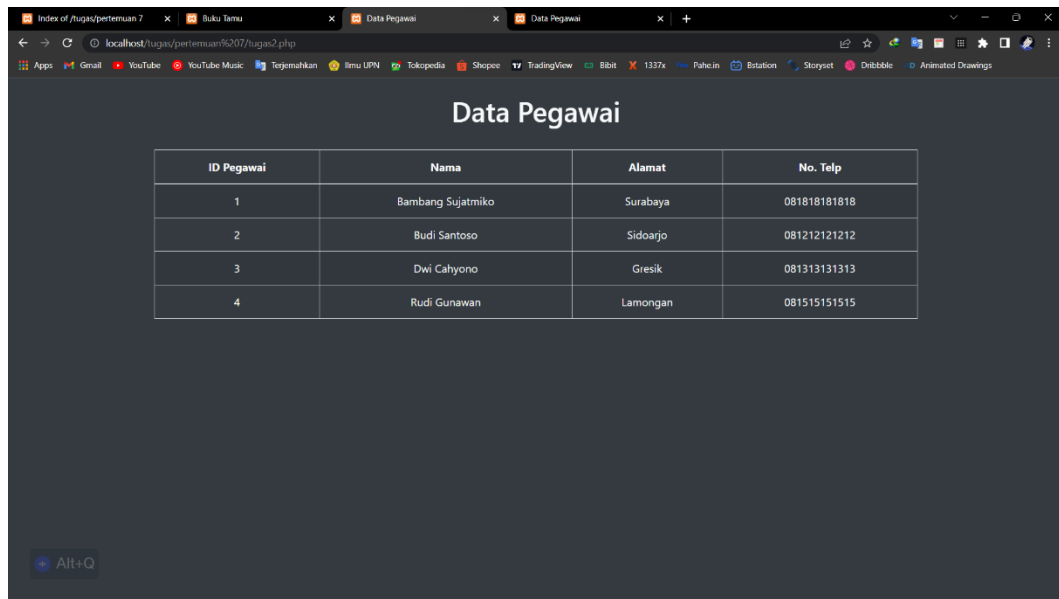
Setelah itu, program akan memproses form jika ada data yang dikirimkan. Data yang dikirimkan melalui form akan disimpan ke dalam tabel `buku_tamu` dengan menggunakan perintah SQL `INSERT INTO`. Data yang disimpan meliputi nama, email, dan isi pesan tamu.

Setelah data berhasil disimpan, program akan menampilkan data dari tabel `buku_tamu` dengan menggunakan perintah SQL `SELECT`. Data yang ditampilkan akan ditampilkan dalam bentuk tabel dengan menggunakan tag HTML. Jika belum ada data, maka program akan menampilkan pesan "Belum ada data".

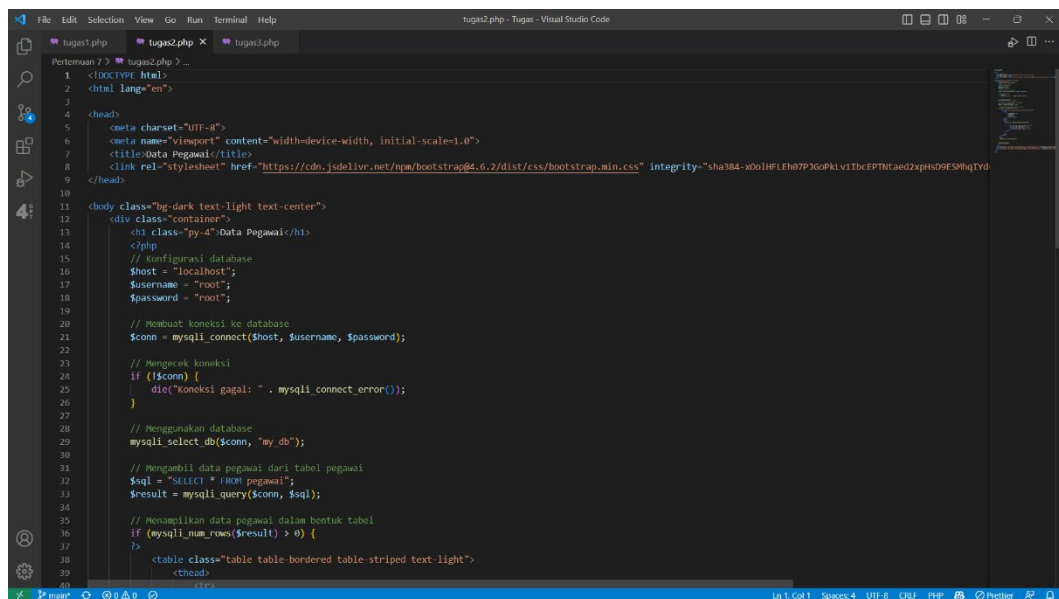
Terakhir, program akan menutup koneksi ke database dengan menggunakan fungsi `mysqli_close()`. Selain itu, program juga menggunakan beberapa library dari Bootstrap dan jQuery untuk mempercantik tampilan web.



## Tugas 2



ID Pegawai	Nama	Alamat	No. Telp
1	Bambang Sujatmiko	Surabaya	0818181818
2	Budi Santoso	Sidoarjo	0812121212
3	Dwi Cahyono	Gresik	0813131313
4	Rudi Gunawan	Lamongan	0815151515



```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Data Pegawai</title>
8     <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css" integrity="sha384-xd01HfLEh877P360K1V1IbEPTtaed2xphSD9ESh9qTYD"
9 </head>
10
11 <body class="bg-dark text-light text-center">
12     <div class="container">
13         <h1 class="py-4">Data Pegawai</h1>
14         <php
15             // Konfigurasi database
16             $host = "localhost";
17             $username = "root";
18             $password = "root";
19
20             // Membuat koneksi ke database
21             $conn = mysqli_connect($host, $username, $password);
22
23             // Mengecek koneksi
24             if (!$conn) {
25                 die("koneksi gagal: " . mysqli_connect_error());
26             }
27
28             // Menggunakan database
29             mysqli_select_db($conn, "my_db");
30
31             // Mengambil data pegawai dari tabel pegawai
32             $sql = "SELECT * FROM pegawai";
33             $result = mysqli_query($conn, $sql);
34
35             // Menampilkan data pegawai dalam bentuk tabel
36             if (mysqli_num_rows($result) > 0) {
37                 <table class="table table-bordered table-striped text-light">
38                     <thead>
```





## Analisis:

Kode PHP di atas merupakan sebuah program yang digunakan untuk mengambil data pegawai dari sebuah tabel di dalam database dan menampilkannya dalam bentuk tabel pada halaman web. Program ini menggunakan bahasa pemrograman PHP dan database MySQL untuk mengakses dan mengambil data dari tabel pegawai.

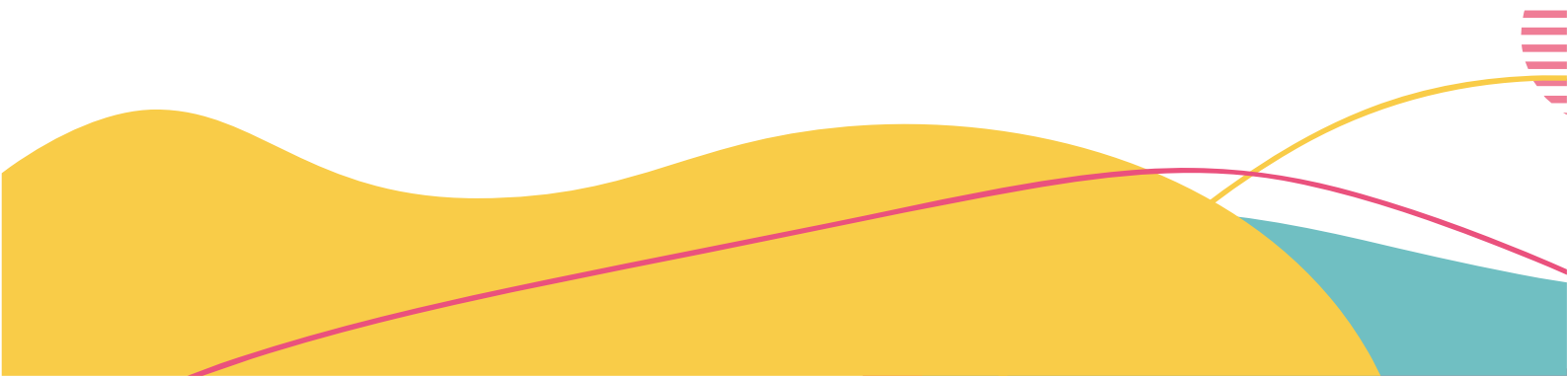
Pertama-tama, program ini melakukan konfigurasi database dengan mengatur host, username, dan password untuk membuat koneksi ke database. Kemudian, program ini menggunakan fungsi `mysqli_connect()` untuk membuat koneksi ke database dan fungsi `mysqli_select_db()` untuk memilih database yang akan digunakan.

Setelah koneksi berhasil dibuat, program ini mengambil data pegawai dari tabel pegawai dengan menggunakan perintah SQL `SELECT` dan menyimpan hasilnya dalam variabel `$result`. Kemudian, program ini mengecek apakah ada data pegawai yang ditemukan dengan menggunakan fungsi `mysqli_num_rows()`. Jika ada, program ini akan menampilkan data pegawai dalam bentuk tabel dengan menggunakan tag HTML dan setiap baris data pegawai ditampilkan dalam tag HTML dengan menggunakan fungsi `mysqli_fetch_assoc()` untuk mengambil data dari variabel `$result`.

Jika tidak ada data pegawai yang ditemukan, program ini akan menampilkan pesan "Tidak ada data pegawai". Setelah selesai menampilkan data pegawai, program ini menutup koneksi ke database dengan menggunakan fungsi `mysqli_close()`.

Selain itu, program ini juga menggunakan beberapa library dari Bootstrap untuk membuat tampilan halaman web yang lebih menarik dan responsif. Library tersebut antara lain adalah Bootstrap CSS, jQuery, dan Popper.js.

Secara keseluruhan, program ini merupakan contoh sederhana dari penggunaan bahasa pemrograman PHP dan database MySQL untuk mengambil dan menampilkan data dari sebuah tabel pada halaman web. Program ini juga menunjukkan bagaimana menggunakan library dari Bootstrap untuk membuat tampilan halaman web yang lebih menarik dan responsif.



## Tugas 3

Index of /tugas/pertemuan 7 x Buku Tamu x Data Pegawai x Data Pegawai x +

localhost/tugas/pertemuan%207/tugas3.php






### Data Pegawai

Nama: Gatot Subroto

Alamat: Mojokerto

No. Telp: 081717171717

Simpan

ID Pegawai	Nama	Alamat	No. Telp	Aksi
1	Bambang Sujatmiko	Surabaya	0818181818	 
2	Budi Santoso	Sidoarjo	0812121212	 
3	Dwi Cahyono	Gresik	0813131313	 
4	Rudi Gunawan	Lamongan	0815151515	 

```
File Edit Selection View Go Run Terminal Help tugas3.php - Tugas - Visual Studio Code
tugas1.php tugas2.php tugas3.php X
Pertemuan 7 > tugas3.php > ...
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5 <title>Data Pegawai</title>
6 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css" integrity="sha384-xhol16/LH6P/P3oGKlvi11bcEPINTaed2xptsD9LSPhq1Yd
7 <style>
8   th,
9   td {
10     vertical-align: middle !important;
11   }
12 </style>
13 <script src="https://kit.fontawesome.com/42e7da2888.js" crossorigin="anonymous"></script>
14 </head>
15
16 <body class="bg-dark text-light">
17 <div class="container">
18 <h1 class="text-center py-4">Data Pegawai</h1>
19 <?php
20 // konfigurasi database
21 $host = "localhost";
22 $username = "root";
23 $password = "root";
24 $database = "my_db";
25
26 // Membuat koneksi ke database
27 $conn = mysqli_connect($host, $username, $password, $database);
28
29 // Mengecek koneksi
30 if (!$conn) {
31   die("Koneksi gagal: " . mysqli_connect_error());
32 }
33
34 // Menyimpan data pegawai
35 if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST["simpan"])) {
36   $nama = $_POST["nama"];
37   $alamat = $_POST["alamat"];
38   $no_telp = $_POST["no_telp"];
39
40   $sql = "INSERT INTO pegawai (nama, alamat, no_telp) VALUES ('$nama', '$alamat', '$no_telp')";
41   if (mysqli_query($conn, $sql)) {
42     echo "Data berhasil disimpan";
43   } else {
44     echo "Gagal menyimpan data: " . mysqli_error($conn);
45   }
46 }
```



## Analisis:

Kode PHP di atas merupakan sebuah program sederhana untuk mengelola data pegawai. Program ini menggunakan database MySQL untuk menyimpan data pegawai.

Pertama-tama, program ini melakukan koneksi ke database dengan menggunakan fungsi `mysqli_connect()`. Jika koneksi berhasil, maka program akan menampilkan pesan "Koneksi berhasil". Namun, jika koneksi gagal, maka program akan menampilkan pesan "Koneksi gagal" dan menghentikan program dengan menggunakan fungsi `die()`.

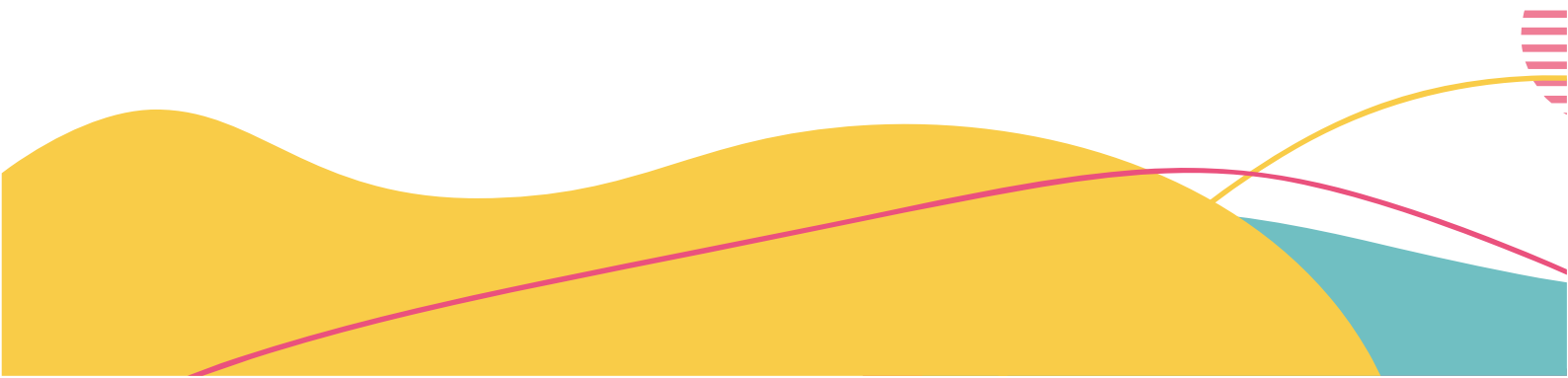
Selanjutnya, program ini memiliki tiga fungsi utama yaitu menyimpan data pegawai, menghapus data pegawai, dan mengubah data pegawai. Fungsi menyimpan data pegawai dilakukan dengan menggunakan metode POST dan `isset()`. Jika data berhasil disimpan, maka program akan menampilkan pesan "Data berhasil disimpan". Namun, jika terjadi error, maka program akan menampilkan pesan "Error" beserta pesan error dari MySQL.

Fungsi menghapus data pegawai dilakukan dengan menggunakan metode GET dan `isset()`. Jika data berhasil dihapus, maka program akan menampilkan pesan "Data berhasil dihapus". Namun, jika terjadi error, maka program akan menampilkan pesan "Error" beserta pesan error dari MySQL.

Fungsi mengubah data pegawai dilakukan dengan menggunakan metode POST dan `isset()`. Jika data berhasil diubah, maka program akan menampilkan pesan "Data berhasil diubah". Namun, jika terjadi error, maka program akan menampilkan pesan "Error" beserta pesan error dari MySQL.

Selain itu, program ini juga memiliki form untuk menyimpan atau mengubah data pegawai. Jika form digunakan untuk mengubah data pegawai, maka program akan menampilkan data pegawai yang akan diubah. Namun, jika form digunakan untuk menyimpan data pegawai, maka program akan menampilkan form kosong.

Terakhir, program ini menampilkan data pegawai yang tersimpan di database. Jika terdapat data pegawai, maka program akan menampilkan tabel yang berisi data pegawai beserta tombol untuk mengubah atau menghapus data pegawai. Namun, jika tidak terdapat data pegawai, maka program akan menampilkan pesan "Tidak ada data pegawai".





## Kesimpulan

Dalam praktikum ini, kita telah belajar bagaimana cara membuat koneksi antara PHP dan MySQL. Koneksi ini sangat penting karena memungkinkan kita untuk mengambil data dari database MySQL dan menampilkannya pada halaman web menggunakan PHP. Dalam praktikum ini, kita telah menggunakan fungsi `mysqli_connect()` untuk membuat koneksi ke database MySQL dan fungsi `mysqli_query()` untuk melakukan query pada database. Selain itu, kita juga telah belajar bagaimana cara menampilkan data dari database pada halaman web menggunakan PHP.

Pada saat membuat koneksi antara PHP dan MySQL, ada beberapa hal yang perlu diperhatikan, seperti memastikan informasi server, username, password, dan nama database yang digunakan sudah benar. Selain itu, kita juga perlu memastikan bahwa koneksi ke database berhasil terhubung dan query yang dilakukan menghasilkan data yang diinginkan.

Secara keseluruhan, pada praktikum ini kita telah berhasil membuat koneksi antara PHP dan MySQL dan menampilkan data dari database pada halaman web menggunakan PHP. Dengan memahami cara membuat koneksi antara PHP dan MySQL, kita dapat membuat aplikasi web yang lebih kompleks dan interaktif dengan menggunakan database sebagai sumber data.

