**LAPORAN PRAKTIKUM**

**PEMOGRAMAN WEB**

Dosen Pengampu :

Syukhri, S.T., M.CIO.



Oleh :

Manja Fani Oktavia

22343056

**PROGRAM STUDI INFORMATIKA**

**JURUSAN TEKNIK ELEKTRONIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS NEGERI PADANG**

**2023**

1. **TOPIK PRAKTIKUM**

Topik : Query Dasar MySQL

1. **PROSES DAN HASIL PRAKTIKUM**
2. **Membuat Database**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP lalu mendefinisikan variabel $servername, $username, dan $password yang digunakan untuk membuat koneksi ke server MySQL.
3. Membuat koneksi ke server MySQL menggunakan fungsi mysqli\_connect() yang menerima tiga parameter, yaitu nama server, nama pengguna, dan kata sandi.
4. Setelah koneksi berhasil dibuat, kita memeriksa apakah koneksi berhasil dibuat atau tidak dengan menggunakan fungsi mysqli\_connect\_error(). Jika koneksi gagal, maka program akan berhenti dan menampilkan pesan kesalahan.
5. Membuat database baru dengan menggunakan perintah SQL CREATE DATABASE , dimana perintah ditempatka dalam varibel $sql.
6. Membuat fungsi mysqli\_query() untuk mengeksekusi perintah SQL pada server MySQL. Fungsi ini menerima dua parameter, yaitu koneksi MySQL dan perintah SQL yang akan dijalankan. Jika pembuatan database berhasil, maka program akan menampilkan pesan "database admin berhasil dibuat". Jika gagal, maka program akan menampilkan pesan kesalahan.
7. Setelah perintah SQL berhasil dijalankan, kita menutup koneksi ke server MySQL menggunakan fungsi mysqli\_close()
8. Menutup codingan dengan menggunakan tag PHP ?>

Source code :

<?php

$servername ="localhost";

$username="root";

$password="";

/\* create connection\*/

$conn=mysqli\_connect($servername,$username ,$password);

/\*check connection\*/

if(!$conn){

die("koneksi gagal:". mysqli\_connect\_error());

}

/\*create database \*/

$sql="create database admin";

if(mysql\_query($conn,$sql)){

echo "database admin berhasil dibuat";

}else{

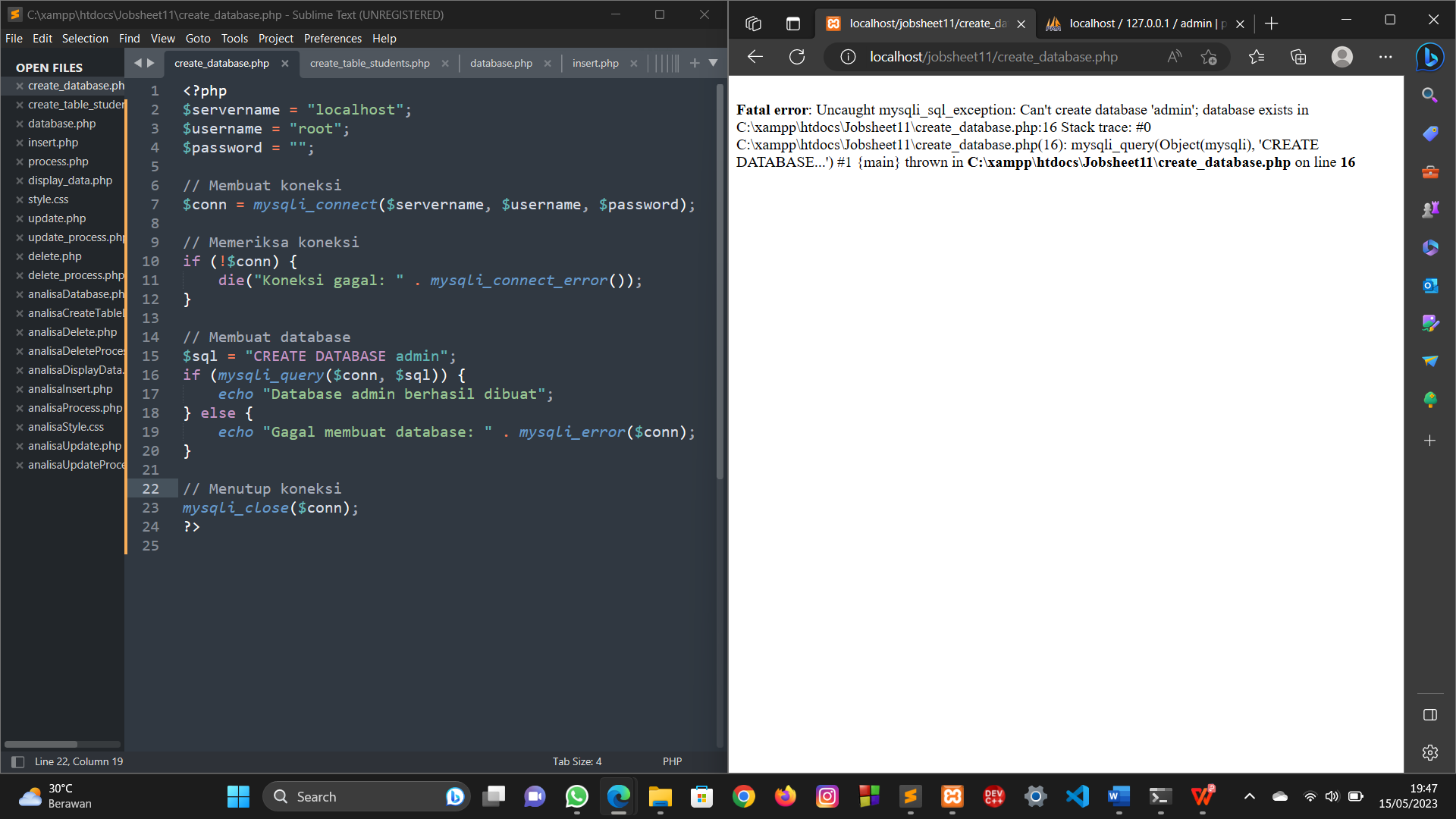
echo "gagal membuat database:". mysqli\_error($conn);

}

mysql\_close($conn);

?>

Hasil dan proses :



Penjelasan :

Code tersebut adalah script PHP yang berfungsi untuk membuat database baru di server MySQL menggunakan mysqli extension pada PHP.

Tujuan dari code tersebut adalah untuk membuat sebuah database baru bernama "admin" pada server MySQL yang diakses melalui localhost. Script ini juga melakukan pengecekan koneksi ke server MySQL dan menampilkan pesan kesalahan jika koneksi gagal.

1. **Membuat Tabel**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP , lalu mendefinisikan nilai dari variabel $servername, $username, $password, dan $dbname untuk membuat koneksi ke database.
3. Membuat fungsi mysqli\_connect() untuk membuat koneksi ke database menggunakan nilai variabel yang telah ditentukan sebelumnya.
4. Membuat pernyataan if untuk memeriksa apakah koneksi berhasil dibuat atau tidak. Jika koneksi gagal, maka kita akan menghentikan eksekusi program dan menampilkan pesan kesalahan.
5. Membuat membuat sebuah database bernama students menggunakan pernyataan SQL CREATE DATABASE.
6. Membuat tabel dengan nama students dalam database admin menggunakan pernyataan SQL CREATE TABLE. Tabel ini memiliki kolom-kolom ID, LastName, FirstName, Email, dan City.
7. Membuat kolom ID, kita memberikan atribut AUTO\_INCREMENT agar nilai kolom ini akan di-generate secara otomatis oleh sistem.
8. Membuat kolom ID juga ditentukan sebagai PRIMARY KEY sehingga setiap nilai yang disimpan di kolom ini harus unik.
9. Membuat pernyataan if untuk memeriksa apakah pembuatan tabel berhasil atau tidak. Jika berhasil, kita akan menampilkan pesan "tabel berhasil dibuat". Jika gagal, kita akan menampilkan pesan kesalahan.
10. Menutup koneksi ke database menggunakan fungsi mysql\_close()
11. Menutup codingan dengan menggunakan tag PHP ?>

Source code :

<?php

$servername ="localhost";

$username="root";

$password="";

$dbname="admin";

/\* create connection\*/

$conn=mysqli\_connect($servername,$username ,$password , $dbname);

/\*check connection\*/

if(!$conn){

die("connection failed:". mysqli\_connect\_error());

}

/\*create database \*/

$sql="create database students(

ID int NOT NULL AUTO\_INCREAMENT,

LastName varchar (50),

FirstName varchar(50),

Email varchar(50),

City varchar(50),

PRIMARY KEY(ID)

)";

if(mysql\_query($conn,$sql)){

echo "tabel berhasil dibuat";

}else{

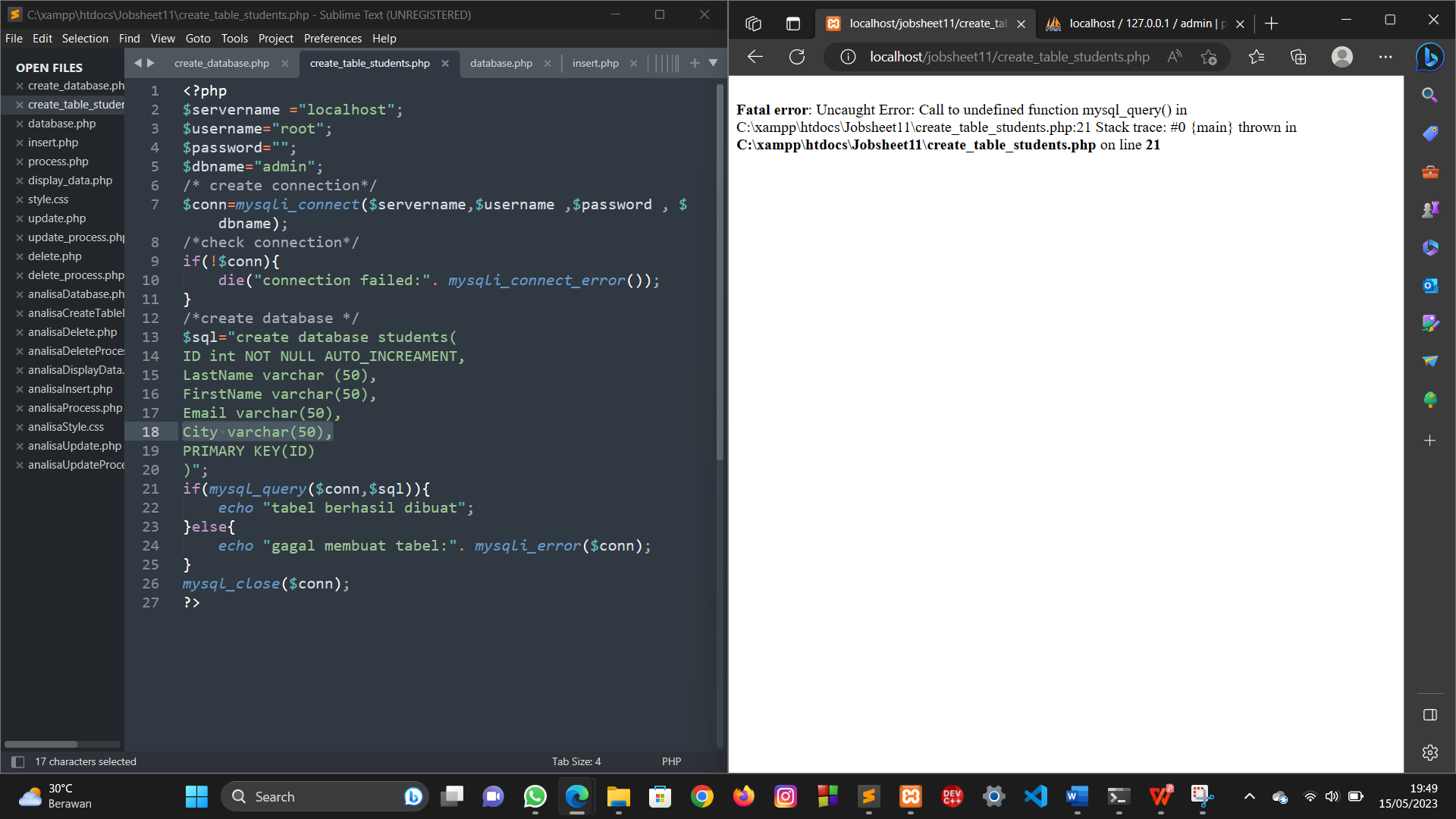
echo "gagal membuat tabel:". mysqli\_error($conn);

}

mysql\_close($conn);

?>

Hasil dan proses :



Penjelasan :

Code tersebut adalah script PHP yang berfungsi untuk membuat sebuah tabel baru pada database "admin" yang terletak di server MySQL menggunakan mysqli extension pada PHP.

Tujuan dari code tersebut adalah untuk membuat sebuah tabel baru bernama "Students" pada database "admin". Script ini juga melakukan pengecekan koneksi ke server MySQL dan menampilkan pesan kesalahan jika koneksi gagal.

1. **Membuat File Koneksi PHP dan MySQL**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP , lalu membuat sebuah variabel $servername dideklarasikan dan diinisialisasi dengan nilai "localhost". Variabel ini akan digunakan untuk menunjukkan nama server MySQL yang digunakan dalam koneksi.
3. Membuat sebuah variabel $username dideklarasikan dan diinisialisasi dengan nilai "root". Variabel ini akan digunakan untuk menunjukkan nama pengguna MySQL yang digunakan dalam koneksi.
4. Membuat variabel $password dideklarasikan dan diinisialisasi dengan nilai kosong. Variabel ini akan digunakan untuk menunjukkan kata sandi pengguna MySQL yang digunakan dalam koneksi.
5. Membuat variabel $dbname dideklarasikan dan diinisialisasi dengan nilai "admin". Variabel ini akan digunakan untuk menunjukkan nama database MySQL yang akan digunakan dalam koneksi.
6. Membuat fungsi mysqli\_connect () dipanggil dengan empat parameter: $servername, $username, $password, dan $dbname. Fungsi ini membuat koneksi ke server MySQL dan database yang ditentukan.
7. Jika koneksi tidak berhasil dibuat, maka fungsi die() akan dipanggil dengan pesan error "tidak bisa terhubung ke mysql:", diikuti dengan pesan error spesifik yang diberikan oleh fungsi mysql\_error().
8. Jika koneksi berhasil dibuat, variabel $conn akan berisi objek koneksi ke database MySQL yang dapat digunakan dalam eksekusi query dan operasi database lainnya.
9. Menutup codingan dengan menggunakan tag PHP ?>

Source code :

<?php

$servername ="localhost";

$username="root";

$password="";

$dbname="admin";

/\* create connection\*/

$conn=mysqli\_connect($servername,$username ,$password , $dbname);

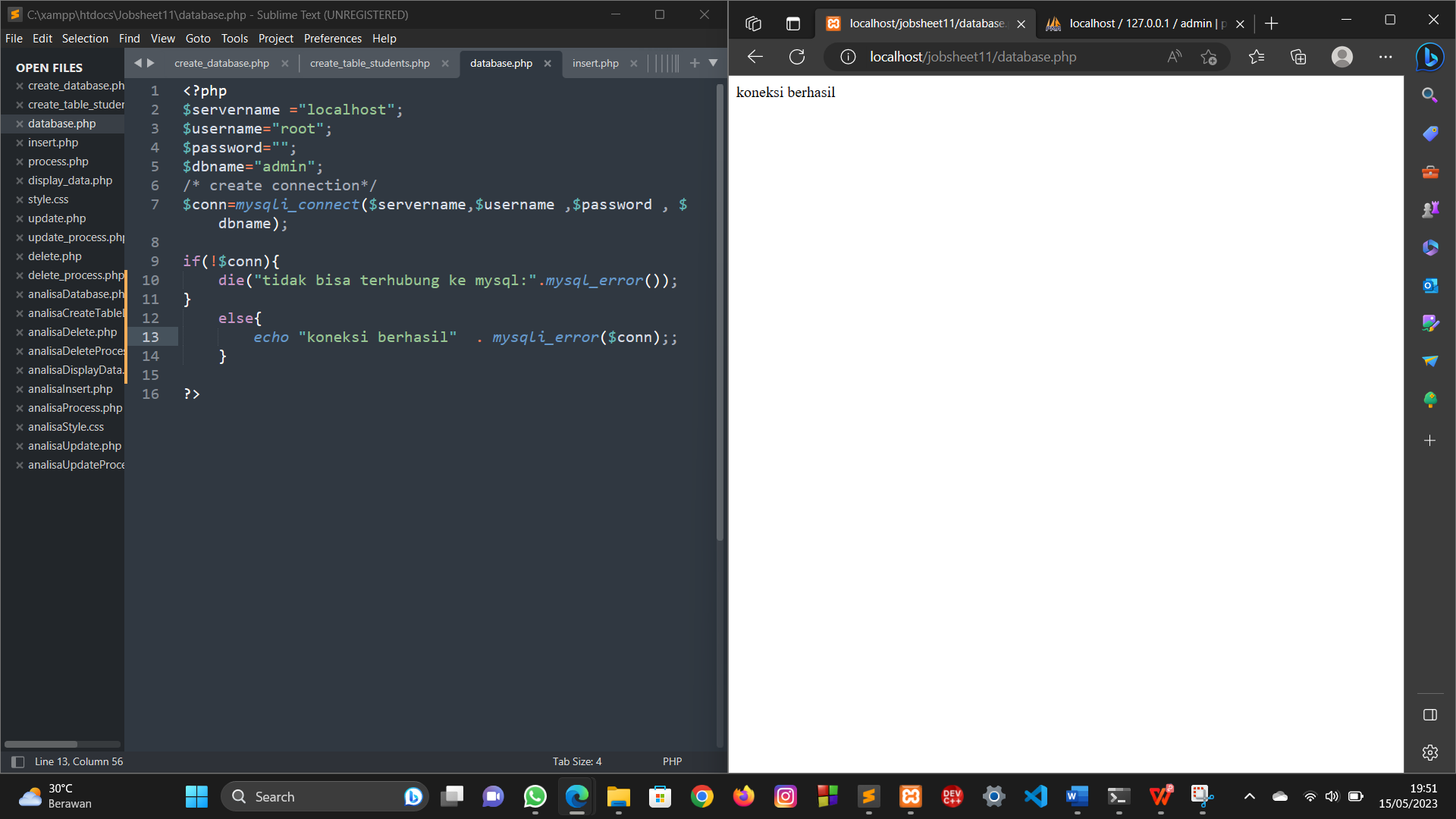
if(!$conn){

die("tidak bisa terhubung ke mysql:".mysql\_error())

}

?>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web tersebut terlihat bahwa koneksi sudah berhasil.

1. **Meng-Insert Data ke Tabel**
2. **insert.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag <html> , lalu membuat tag <body> didalam tag ini kita membuat form HTML dengan method "post" dan action"process.php" yang digunakan untuk mengirimkan data ke server dan memprosesnya dengan file "process.php".
3. Membuat label "Nama depan" dan input field dengan tipe "text" dan nama "first\_name" yang akan digunakan untuk memasukkan nama depan pengguna.
4. Membuat label "Nama belakang" dan input field dengan tipe "text" dan nama "last\_name" yang akan digunakan untuk memasukkan nama belakang pengguna.
5. Membuat label "Alamat email" dan input field dengan tipe "text" dan nama "email" yang akan digunakan untuk memasukkan alamat email pengguna.
6. Membuat label "Kota asal" dan input field dengan tipe "text" dan nama "city\_name" yang akan digunakan untuk memasukkan kota asal pengguna.
7. Membuat tombol "Simpan Data" dengan tipe "submit" dan nama "save" yang akan digunakan untuk mengirimkan data yang telah dimasukkan ke server untuk diproses.
8. Ketika tombol "Simpan Data" diklik, data dari input field akan dikirim ke server dengan method "post" dan akan diproses oleh file "process.php" yang diarahkan oleh atribut "action" pada form HTML.
9. Menutup codingan dengan menggunakan tag PHP ?>

Source code :

<html>

<body>

<form method="post" action="process.php">

Nama depan :<br>

<input type="text" name="first\_name">

<br>

Nama belakang:<br>

<input type="text" name="last\_name">

<br>

Alamat email :<br>

<input type="text" name="email">

<br>

Kota asal :<br>

<input type="text" name="city\_name">

<br><br>

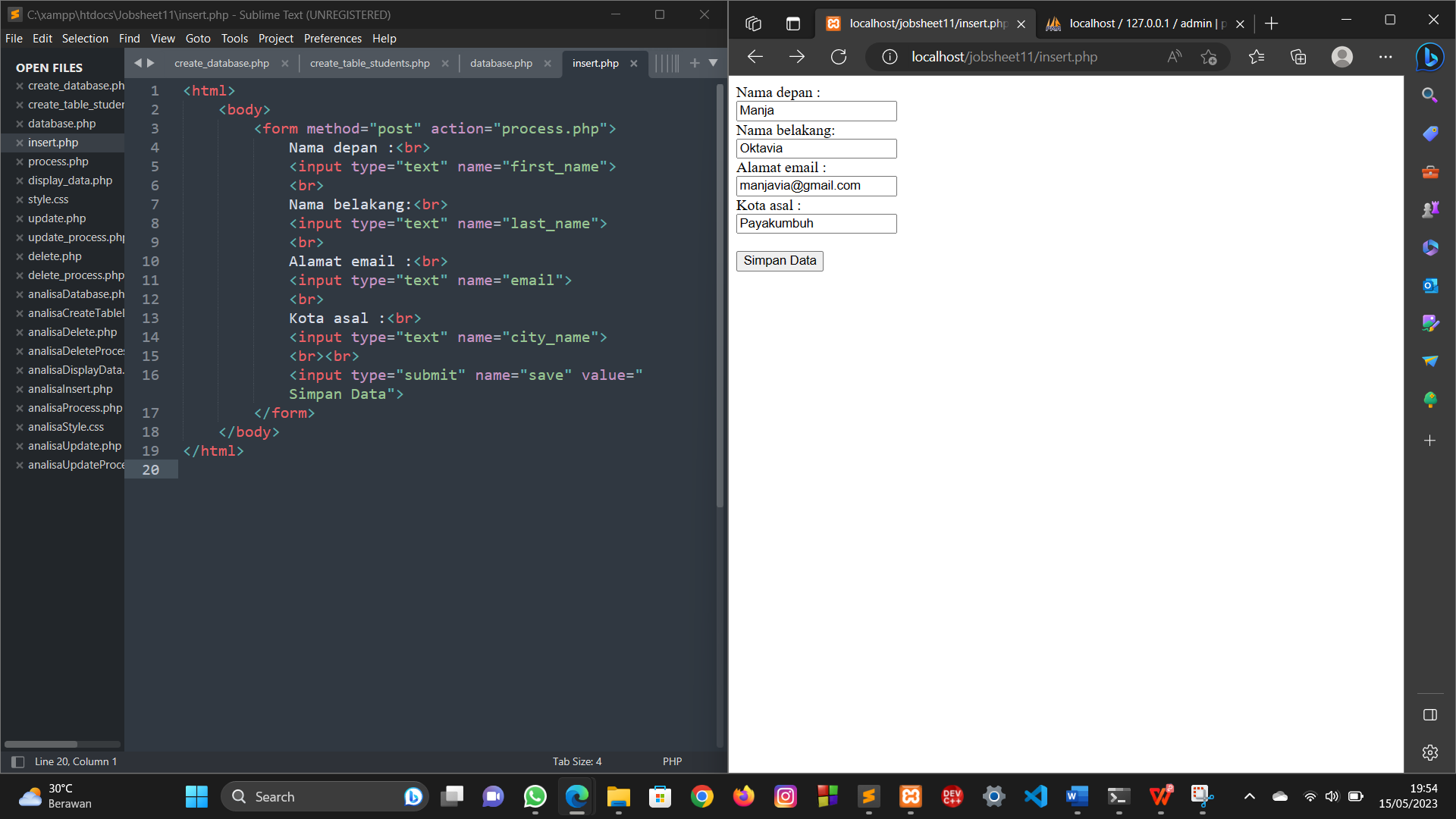
<input type="submit" name="save" value="Simpan Data">

</form>

</body>

</html>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web terdapat form yang terdiri dari Nama depan bernilai “ Manja “ , Nama belakang bernilai “ Oktavia” , Alamat email bernilai “ [manjavia@gmail.com](mailto:manjavia@gmail.com)” , Kota asal bernilai “ Payakumbuh “ . Setelah dat diisikan lalu pencet kolom Simpan Data untuk menyimpa data.

1. **process.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat fungsi include\_once untuk memasukkan file database.php yang berisi koneksi ke database MySQL.
3. Membuat perulangan if untuk melakukan pengecekan apakah form telah disubmit dengan menggunakan kondisi if(isset($\_POST['save'])). Jika iya, maka langkah selanjutnya adalah memproses data yang telah diisi pada form.
4. Memasukkan nilai dari input form ke dalam variabel $first\_name, $last\_name, $email, dan $city\_name.
5. Membuat query SQL untuk menambahkan data ke dalam tabel students menggunakan perintah INSERT INTO. Nilai-nilai dari variabel $first\_name, $last\_name, $email, dan $city\_name akan dimasukkan ke dalam kolom FirstName, LastName, Email, dan City, secara berurutan.
6. Menjalankan query SQL yang telah dibuat dengan menggunakan fungsi mysqli\_query. Jika query berhasil dijalankan, maka akan ditampilkan pesan "data berhasil disimpan!".
7. Jika query gagal dijalankan, maka akan ditampilkan pesan "gagal:" disertai dengan error message yang dihasilkan oleh MySQL.
8. Menutup koneksi ke database MySQL dengan menggunakan fungsi mysql\_close.
9. Membuat link untuk melihat data mahasiswa yang telah tersimpan di database dengan menggunakan elemen HTML <a> dan href yang mengarah ke file display\_data.php.

Source code :

<?php

include once 'database.php';

if(isset($\_POST['save'])){

$first\_name=$\_POST['first\_name'];

$last\_name=$\_POST['last\_name'];

$email=$\_POST['email'];

$city\_name=$\_POST['city\_name'];

$sql="INSERT INTO students

(FirstName,LastName,Email,City)

VALUES

('$first\_name','$last\_name','$email','$city\_name');

if (mysqli\_query($conn,$sql)){

echo "data berhasil disimpan!";

}else{

echo"gagal:". sql ."". mysql\_error()}

mysql\_close();

}

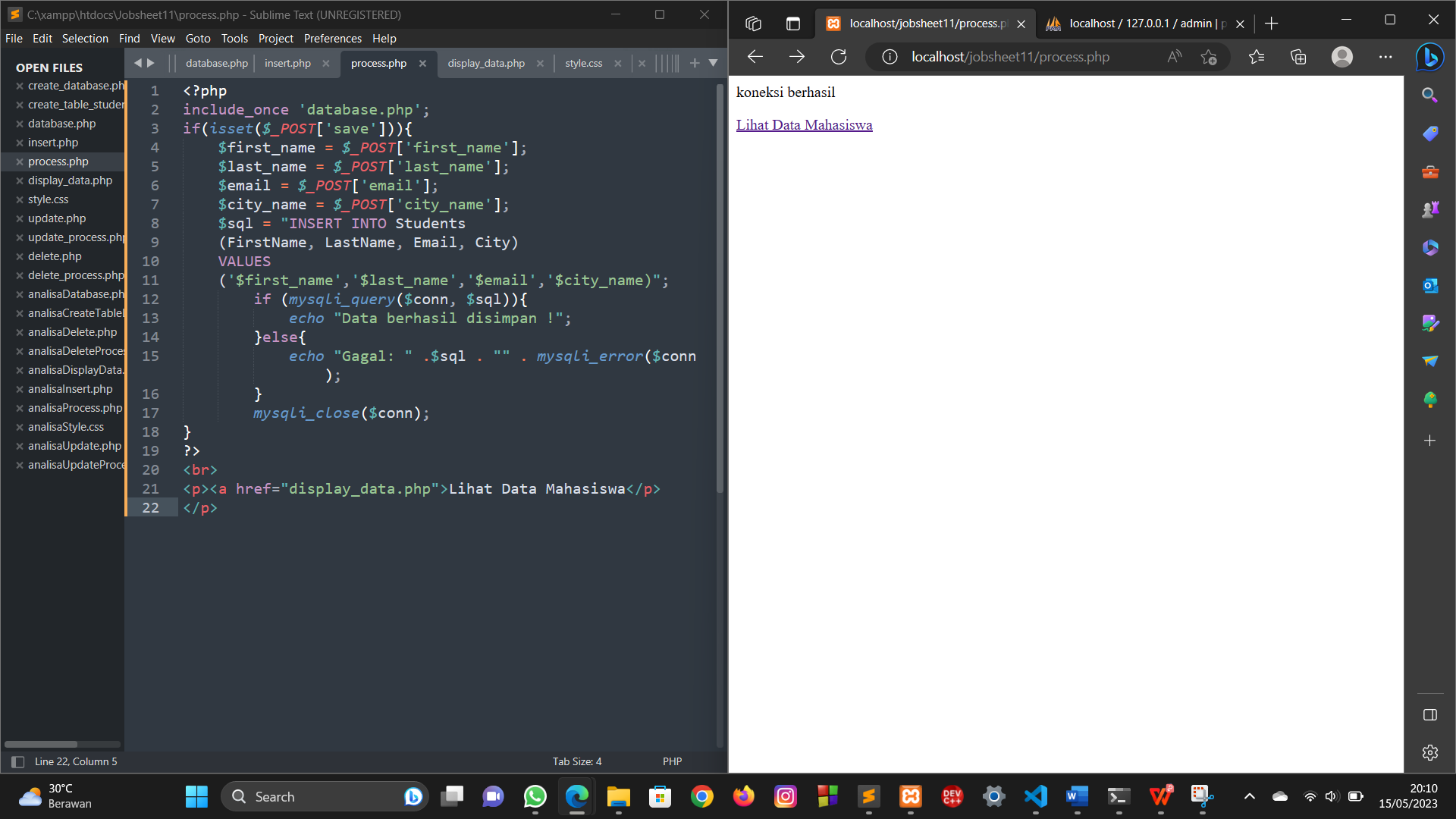
}

?>

<br>

<p><a href="display\_data.php">Lihat Data Mahasiswa</a></p>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web terlihat koneksi berhasil dan terdapat tautan berupa Lihat data mahasiswa.

1. **Menampilkan Data Tabel**
2. **display\_data.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat kode include\_once 'database.php'; yang mengimpor file database.php yang berisi koneksi ke database.
3. Membuat baris kode $result = mysqli\_query($conn, "SELECT \* FROM Students"); menjalankan query SELECT pada tabel Students di database dan menyimpan hasilnya ke dalam variabel $result.
4. Melakukan pengecekan apakah jumlah baris yang dihasilkan dari query SELECT lebih besar dari 0 menggunakan mysqli\_num\_rows(). Jika iya, maka akan dilakukan penampilan data dalam bentuk tabel.
5. Jika ada data, maka akan ditampilkan dalam bentuk tabel menggunakan tag HTML <table> dan <tr> dengan menggunakan perulangan while untuk mengambil data satu per satu dari hasil query dengan mysqli\_fetch\_array(). Kemudian, data tersebut akan ditampilkan di dalam baris tabel menggunakan tag HTML <td>.
6. Jika tidak ada data yang ditemukan, maka akan ditampilkan pesan "Data tidak ditemukan".
7. Membuat hyperlink untuk menuju ke halaman insert.php, update.php, dan delete.php yang akan mengarahkan pada fungsi untuk menambahkan, mengedit, atau menghapus data di database.
8. Menutup kodingan dengan menggunakan tag </html>

Source code :

<?php

include\_once 'database.php';

$result = mysqli\_query($conn, "SELECT \* FROM Students");

?>

<!Doctype html>

<html>

<head>

<title>Menampilkan Data</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<?php

if (mysqli\_num\_rows($result) > 0) {

?>

<table>

<tr>

<td>Nama Depan</td>

<td>Nama Belakang</td>

<td>Alamat Email</td>

<td>Kota Asal</td>

</tr>

<?php

$i=0;

while($row = mysqli\_fetch\_array($result)) {

?>

<tr>

<td><?php echo $row["FirstName"]; ?></td>

<td><?php echo $row["LastName"]; ?></td>

<td><?php echo $row["Email"]; ?></td>

<td><?php echo $row["City"]; ?></td>

</tr>

<?php

$i++;

}

?>

</table>

<?php

}

else {

echo "Data tidak ditemukan";

}

?>

<br>

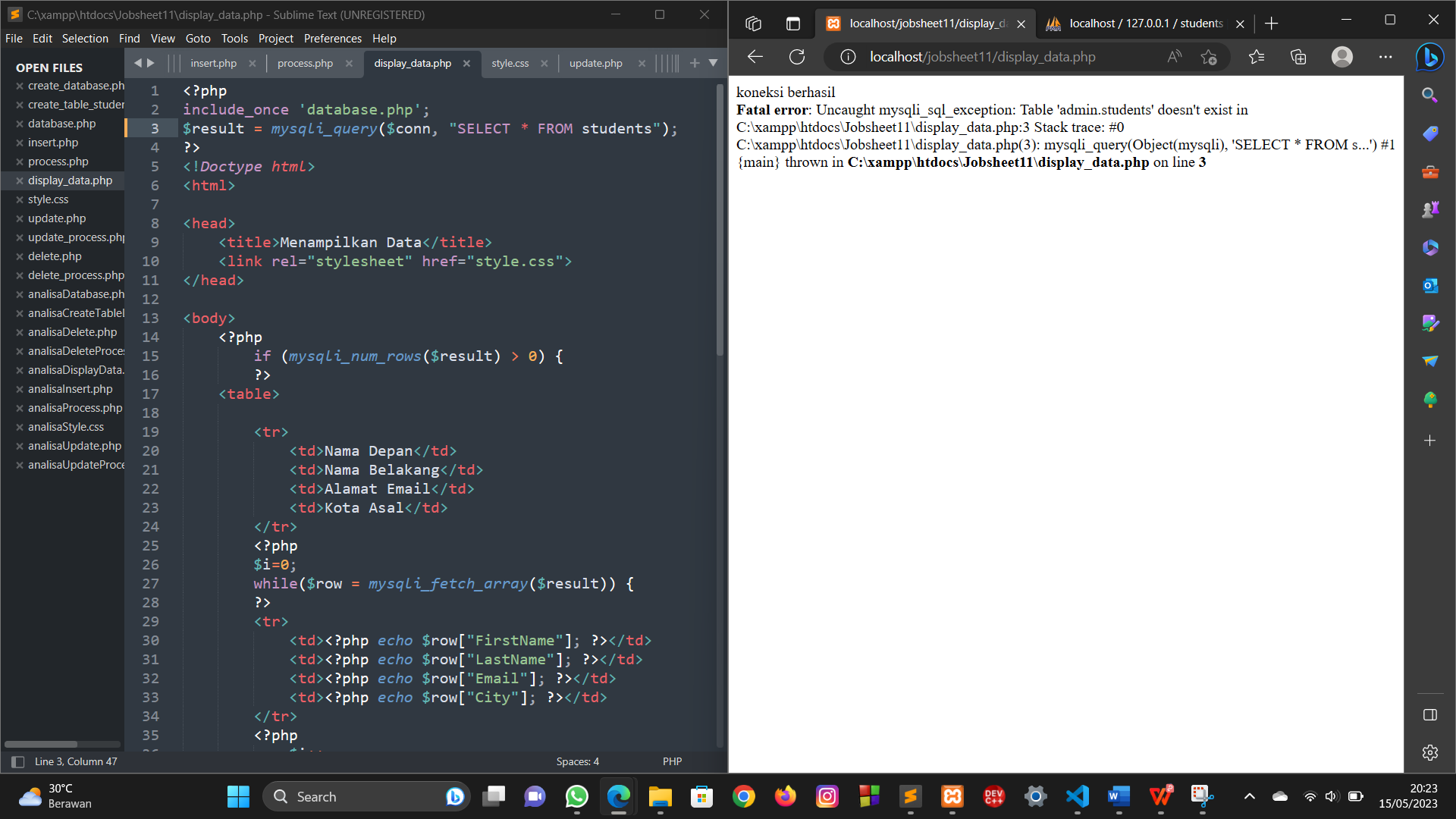
<a href="insert.php">Tambah Data Mahasiswa</a> | <a href="update.php">Edit Data Mahasiswa</a> | <a

href="delete.php">Hapus Data Mahasiswa</a>

</body>

</html>

Hasil dan proses :



Penjelasan :

Code tersebut adalah sebuah script PHP yang bertujuan untuk menampilkan data dari sebuah tabel database yang bernama "students".

1. **style.css**

Source code :

table {

font-family: arial, sans-serif;

border-collapse: collapse;

width: 100%;

}

td,th {

border: 1px solid #dddddd;

text-align: center;

padding: 8px;

}

tr:nth-child(even) {

border: 1px solid #dddddd;

text-align: left;

padding: 8px;

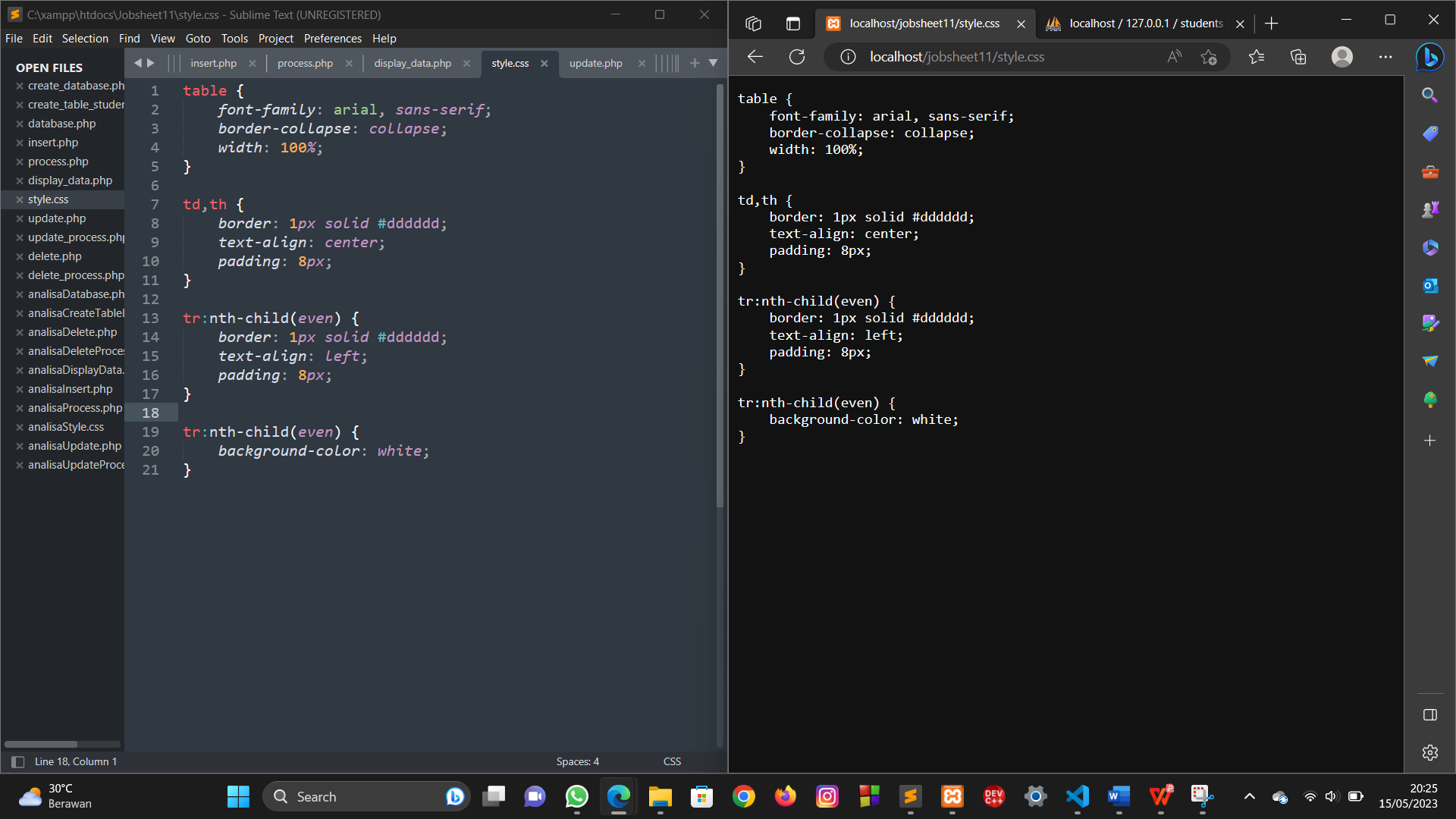
}

tr:nth-child(even) {

background-color: white;

}

Hasil dan proses :



Penjelasan :

1. table digunakan untuk memilih elemen tabel pada halaman web.
2. font-family : Menentukan jenis huruf yang digunakan untuk teks dalam tabel. Di sini, arial dan sans-serif digunakan sebagai jenis huruf alternatif jika jenis huruf utama tidak tersedia pada perangkat pengguna.
3. border-collapse: Menentukan apakah batas antar sel di tabel harus digabungkan menjadi satu atau tetap terpisah. Di sini, nilai collapse digunakan untuk menggabungkan batas sel sehingga tabel akan terlihat lebih rapi.
4. width: Menentukan lebar tabel dalam persen dari lebar halaman web.
5. Pada baris kedua dan ketiga, td dan th digunakan untuk memilih elemen sel pada tabel. Diatur dengan properti-properti CSS sebagai berikut:
6. border: Menentukan ketebalan dan warna batas antar sel.
7. text-align: Menentukan posisi teks dalam sel (dalam hal ini, teks dipusatkan).
8. padding: Menentukan ruang antara teks dan batas sel.
9. tr:nth-child(even) digunakan untuk memilih baris tabel yang genap. Diatur dengan properti-properti CSS sebagai berikut:
10. border: Menentukan ketebalan dan warna batas antar sel.
11. text-align: Menentukan posisi teks dalam sel (dalam hal ini, teks dipusatkan).
12. padding: Menentukan ruang antara teks dan batas sel.
13. tr:nth-child(even) digunakan lagi untuk memilih baris tabel yang genap dan diatur dengan properti background-color untuk memberikan warna latar belakang putih pada baris tersebut, sehingga membedakan baris genap dengan baris ganjil.
14. **Meng-update Data Tabel**
15. **update.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP , dan memanggil fungsi include\_once 'database.php'; untuk menghubungkan kode PHP dengan database.
3. Menjalankan query SQL untuk mengambil semua data mahasiswa dari tabel "Students" dengan menggunakan fungsi mysqli\_query($conn, "SELECT \* FROM Students");. Hasil query tersebut disimpan dalam variabel $result.
4. Mengecek apakah jumlah baris hasil query lebih dari nol dengan menggunakan fungsi mysqli\_num\_rows($result) > 0. Jika benar, maka akan menampilkan tabel dengan data mahasiswa, jika tidak, maka akan menampilkan pesan "Data tidak ditemukan".
5. Jika jumlah baris hasil query lebih dari nol, maka script akan menampilkan tabel dengan header kolom "No", "Nama Depan", "Nama Belakang", "Alamat Email", "Kota Asal", dan "Aksi".
6. Script akan mengambil setiap baris hasil query menggunakan fungsi mysqli\_fetch\_array($result), dan menampilkannya pada baris tabel dengan mengisi data pada masing-masing kolom.
7. Untuk setiap baris tabel, membuat kolom "Aksi" dengan menambahkan tautan "Update" yang mengarahkan pengguna ke halaman "update-process.php?ID=<id\_mahasiswa>" di mana <id\_mahasiswa> adalah ID unik dari mahasiswa.
8. Membuat tautan untuk mengarahkan pengguna ke halaman "insert.php" untuk menambahkan data mahasiswa baru, "update.php" untuk mengedit data mahasiswa, dan "delete.php" untuk menghapus data mahasiswa.
9. Menutup kodingan dengan menggunakan tag </html>

Source code :

<?php

include\_once 'database.php';

$result =mysqli\_query($conn, "SELECT \* FROM Students");

?>

<!DOCTYPE html>

<html>

<head>

<title>Menampilkan Data</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<?php

if (mysqli\_num\_rows($result) > 0){

?>

<table>

<tr>

<td>No</td>

<td>Nama Depan</td>

<td>Nama Belakang</td>

<td>Alamat Email</td>

<td>Kota Asal</td>

<td>Aksi</td>

</tr>

<?php

$i=0;

while($row = mysqli\_fetch\_array($result)) {

?>

<tr>

<td><?php echo $row["ID"];?></td>

<td><?php echo $row["FirstName"];?></td>

<td><?php echo $row["LastName"];?></td>

<td><?php echo $row["Email"];?></td>

<td><?php echo $row["City"];?></td>

<td><a href="update-process.php?ID=<?php echo $row["ID"]; ?>">Update</a></td>

</tr>

<?php

$i++;

}

?>

</table>

<?php

}

else

{

echo "Data tidak ditemukan";

}

?>

<br>

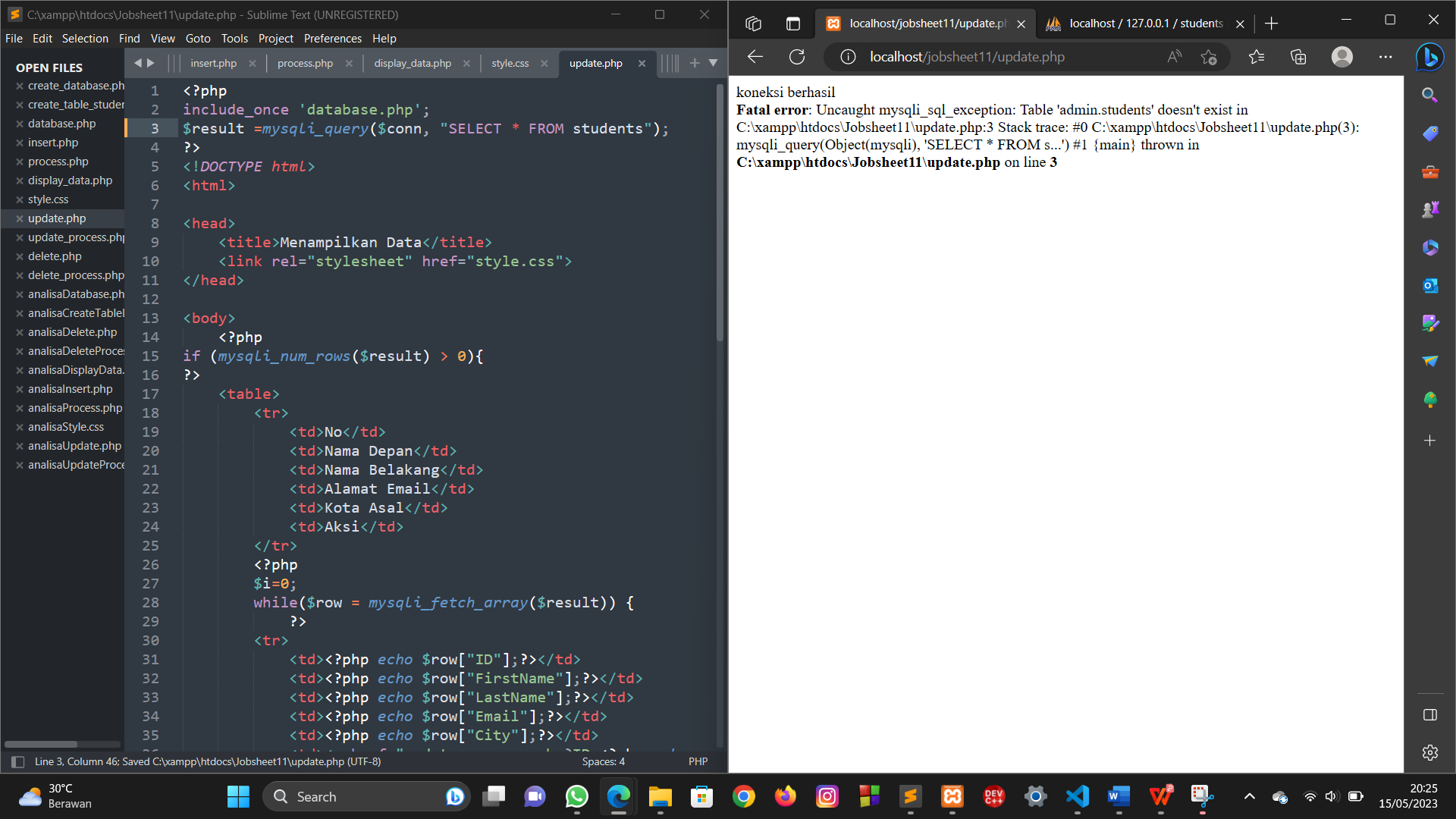
<a href="insert.php">Tambah Data Mahasiswa</a> | <a href="update.php">Edit Data Mahasiswa</a> | <a

href="delete.php">Hapus Data Mahasiswa</a>

</body>

</html>

Hasil dan proses :



Penjelasan :

Code tersebut adalah script PHP yang berfungsi untuk menampilkan data mahasiswa dari database yang telah dibuat sebelumnya dengan menggunakan mysqli extension pada PHP.

Tujuan dari code tersebut adalah untuk menampilkan data mahasiswa yang terdapat pada database "students" di server MySQL menggunakan mysqli\_query(). Data mahasiswa tersebut akan ditampilkan dalam bentuk tabel pada halaman web.

1. **update\_process.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP , lalu memasukkan file database.php ke dalam file saat ini menggunakan perintah include\_once.
3. Mengecek apakah terdapat data yang dikirimkan melalui method POST: Script selanjutnya adalah mengecek apakah terdapat data yang dikirimkan melalui metode POST dengan menggunakan fungsi count(). Jika terdapat data yang dikirimkan, maka script akan melanjutkan ke langkah berikutnya.
4. Melakukan pembaruan data pada tabel mahasiswa dengan menggunakan fungsi mysqli\_query(). Data yang diperbarui adalah ID, Nama Depan, Nama Belakang, Alamat Email, dan Kota Asal. Data baru tersebut diperoleh dari nilai input yang diisi oleh pengguna pada formulir update.
5. Jika data berhasil diperbarui, script akan menampilkan pesan konfirmasi "Data berhasil diperbaharui".
6. Mengambil data mahasiswa yang akan diperbarui dari tabel mahasiswa menggunakan fungsi mysqli\_query(). Data mahasiswa yang diambil ditentukan berdasarkan ID yang diterima melalui parameter GET pada URL.
7. Setelah data mahasiswa diperoleh, script akan menampilkan data tersebut pada formulir update. Data tersebut ditampilkan pada input form dengan menggunakan fungsi echo. Jika terdapat pesan konfirmasi, pesan tersebut juga akan ditampilkan di atas formulir.
8. Setelah pengguna mengisi formulir update dan mengirimkan data melalui metode POST, script akan kembali melakukan pembaruan data pada tabel mahasiswa menggunakan fungsi mysqli\_query(). Data yang diperbarui adalah ID, Nama Depan, Nama Belakang, Alamat Email, dan Kota Asal yang diambil dari nilai input pada formulir.
9. Setelah data mahasiswa berhasil diperbarui, script akan menampilkan tautan "Daftar Mahasiswa" yang akan mengarahkan pengguna kembali ke halaman display\_data.php.
10. Menutup kodingan dengan menggunakan tag </html>.

Source code :

<?php

include\_once 'database.php';

if (count($\_POST) > 0) {

mysqli\_query($conn, "UPDATE students set ID='" .

$\_POST['ID'] . "', FirstName='" . $\_POST['first\_name'] .

"', LastName='" . $\_POST['last\_name'] ."', Email='" .

$\_POST['email'] . "', City='" . $\_POST['city\_name'] . "'

WHERE ID='" . $\_POST['ID'] . "'");

$message = "Data berhasil diperbaharui";

}

$result = mysqli\_query($conn, "SELECT \* FROM Students WHERE ID='" . $\_GET['ID'] ."'");

$row = mysqli\_fetch\_array($result);

?>

<html>

<head>

<title>Update Data Mahasiswa</title>

</head>

<body>

<form name="frmUser" method="post" action="">

<div>

<?php if(isset($message)) { echo $message; } ?>

</div>

<div stlye="padding-bottom:5px;">

<a href="display\_data.php">Daftar Mahasiswa</a>

</div>

ID: <br>

<input type="hidden" name="ID" class="txtField" value="<?php

echo $row['ID']; ?>">

<input type="text" name="ID" class="txtField" value="<?php echo $row['ID']; ?>">

<br><br>

Nama Depan: <br>

<input type="text" name="first\_name" class="txtField" value="<?php echo $row['FirstName']; ?>">

<br><br>

Nama Belakang: <br>

<input type="text" name="last\_name" class="txtField" value="<?php echo $row['LastName']; ?>">

<br><br>

Alamat Email: <br>

<input type="text" name="email" class="txtField" value="<?php echo $row['Email']; ?>">

<br><br>

Kota Asal: <br>

<input type="text" name="city\_name" class="txtField" value="<?php echo $row['City']; ?>">

<br><br>

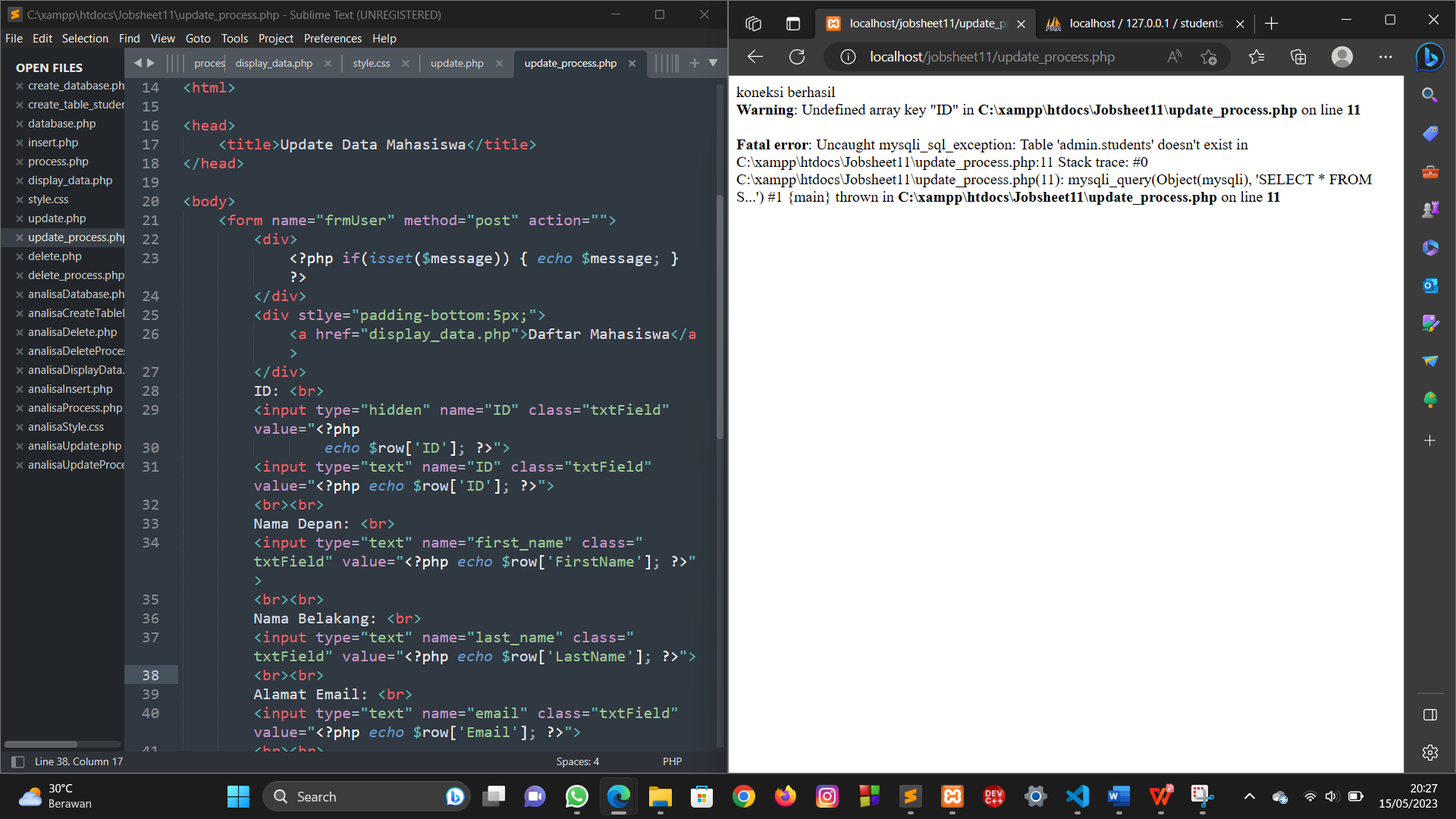
<input type="submit" name="submit" value="Update Data" class="button">

</form>

</body>

</html>

Hasil dan proses :



Penjelasan :

Code di atas adalah kode PHP yang mengakses dan mengubah data mahasiswa dari database.

1. **Menghapus Data Tabel**
2. **delete.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Memanggil file database.php menggunakan perintah include\_once.
3. Melakukan query ke database dengan menggunakan perintah mysqli\_query. Query tersebut akan mengambil seluruh data dari tabel Students pada database dan menyimpannya pada variabel $result.
4. Membuat struktur tabel pada halaman HTML dengan menggunakan tag <table>.
5. Menampilkan data mahasiswa pada tabel dengan menggunakan perintah mysqli\_fetch\_array. Perintah tersebut akan mengambil baris data satu per satu dari variabel $result dan menyimpannya pada variabel $row. Lalu, tampilkan data tersebut pada setiap kolom tabel dengan menggunakan perintah echo.
6. Pada setiap baris tabel, menambahkan kolom "Aksi" dengan menambahkan tag <td> dan sebuah hyperlink untuk menghapus data mahasiswa pada setiap baris. Hyperlink tersebut akan mengarah ke halaman delete-process.php dan akan mengirimkan parameter ID dari mahasiswa yang akan dihapus menggunakan metode GET.
7. Membuat tautan pada halaman HTML untuk menambahkan, mengedit, dan menghapus data mahasiswa.
8. Menutup halaman kodingan dengan menggunakan tag </html>

Source code :

<?php

include\_once'database.php';

$result = mysqli\_query($conn, "SELECT \* FROM Students");

?>

<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" href="style.css">

<title>Delete Data Mahasiswa</title>

</head>

<body>

<table>

<tr>

<td>No.</td>

<td>Nama Depan</td>

<td>Nama Belakang</td>

<td>Alamat Email</td>

<td>Kota Asal</td>

<td>Aksi</td>

</tr>

<?php

$i=0;

while($row=mysqli\_fetch\_array($result)) {

?>

<tr class="<?php if (isset($classname)) echo $classname;?>">

<td><?php echo $row["ID"]; ?></td>

<td><?php echo $row["FirstName"];?></td>

<td><?php echo $row["LastName"];?></td>

<td><?php echo $row["Email"];?></td>

<td><?php echo $row["City"];?></td>

<td><a href="delete-process.php?ID=<?php echo $row["ID"];?>">Delete</a></td>

</tr>

<?php

$i++;

}

?>

</table>

<br>

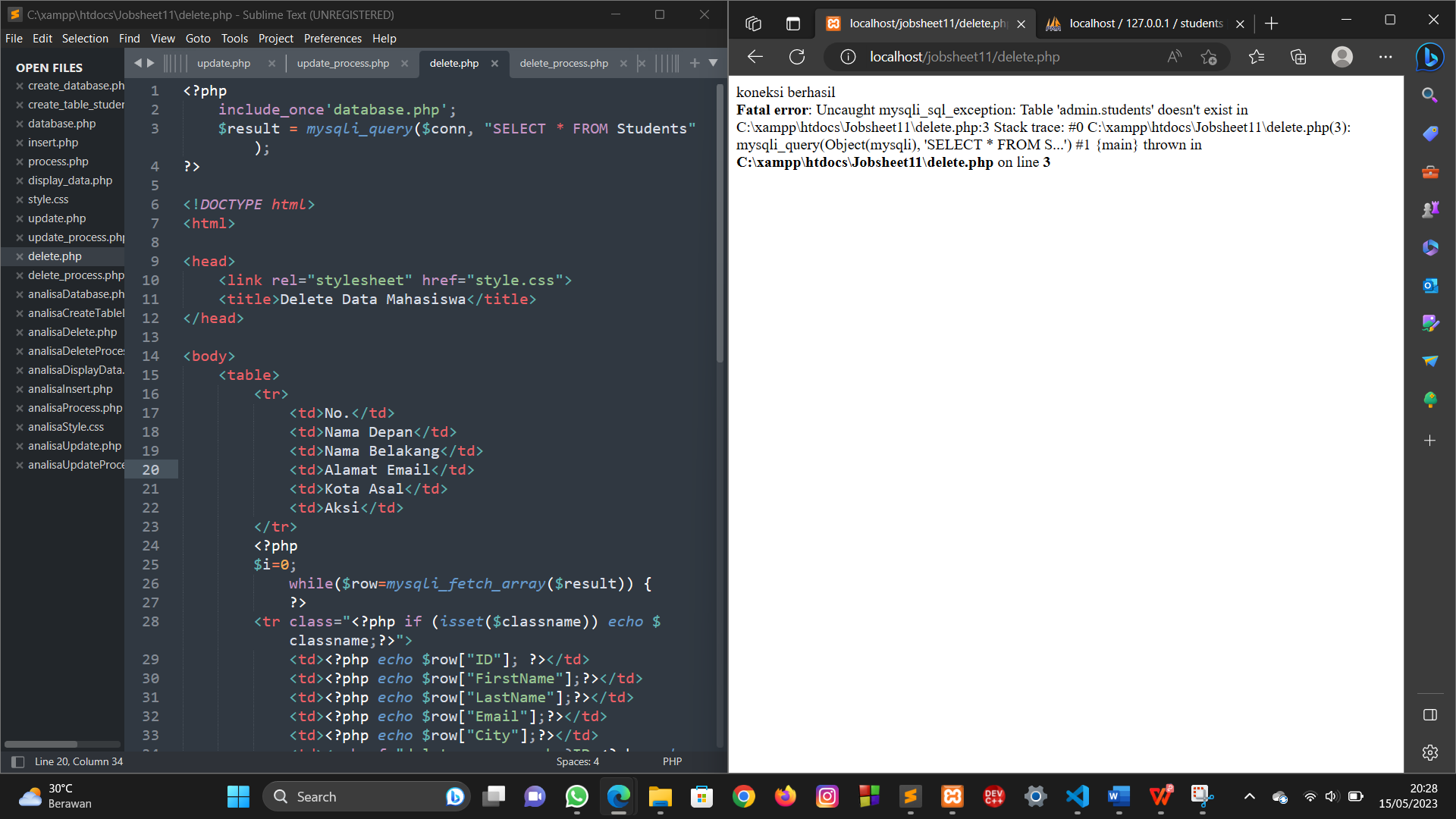
<a href="insert.php">Tambah Data Mahasiswa</a>|<a href="update.php">Edit Data Mahasiswa</a>|<a

href="delete.php">Hapus Data Mahasiswa</a>

</body>

</html>

Hasil dan proses :



Penjelasan :

Code tersebut adalah script PHP yang berfungsi untuk menampilkan data mahasiswa dari sebuah database bernama "students" yang telah dibuat sebelumnya melalui script database.php. Selain itu, script ini juga menyediakan link untuk menghapus data mahasiswa melalui file delete-process.php, serta menyediakan link untuk menambah dan mengedit data mahasiswa.

1. **delete\_process.php**

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Mengimpor file "database.php" yang berisi koneksi database.
3. Membentuk sebuah string query SQL untuk menghapus data pada tabel "Students". Query SQL ini akan menghapus data yang memiliki ID yang sama dengan nilai dari variabel $\_GET["ID"]. Variabel $\_GET["ID"] diperoleh dari parameter URL yang dikirim melalui metode GET.
4. Mengeksekusi query SQL tersebut menggunakan fungsi mysqli\_query(). Jika query berhasil dieksekusi, maka kode akan mencetak pesan "Data Berhasil Dihapus". Namun, jika query gagal dieksekusi, maka kode akan mencetak pesan "Gagal Menghapus Data" dan pesan error yang dihasilkan oleh MySQL.
5. Menutup koneksi database menggunakan fungsi mysqli\_close().
6. Menutup kodingan dengan menggunakan tag PHP ?>

Source code :

<?php

include\_once'database.php';

$sql="DELETE FROM Students WHERE ID='" . $\_GET["ID"].

"'";

if(mysqli\_query($conn,$sql)) {

echo "Data Berhasil Dihapus";

} else {

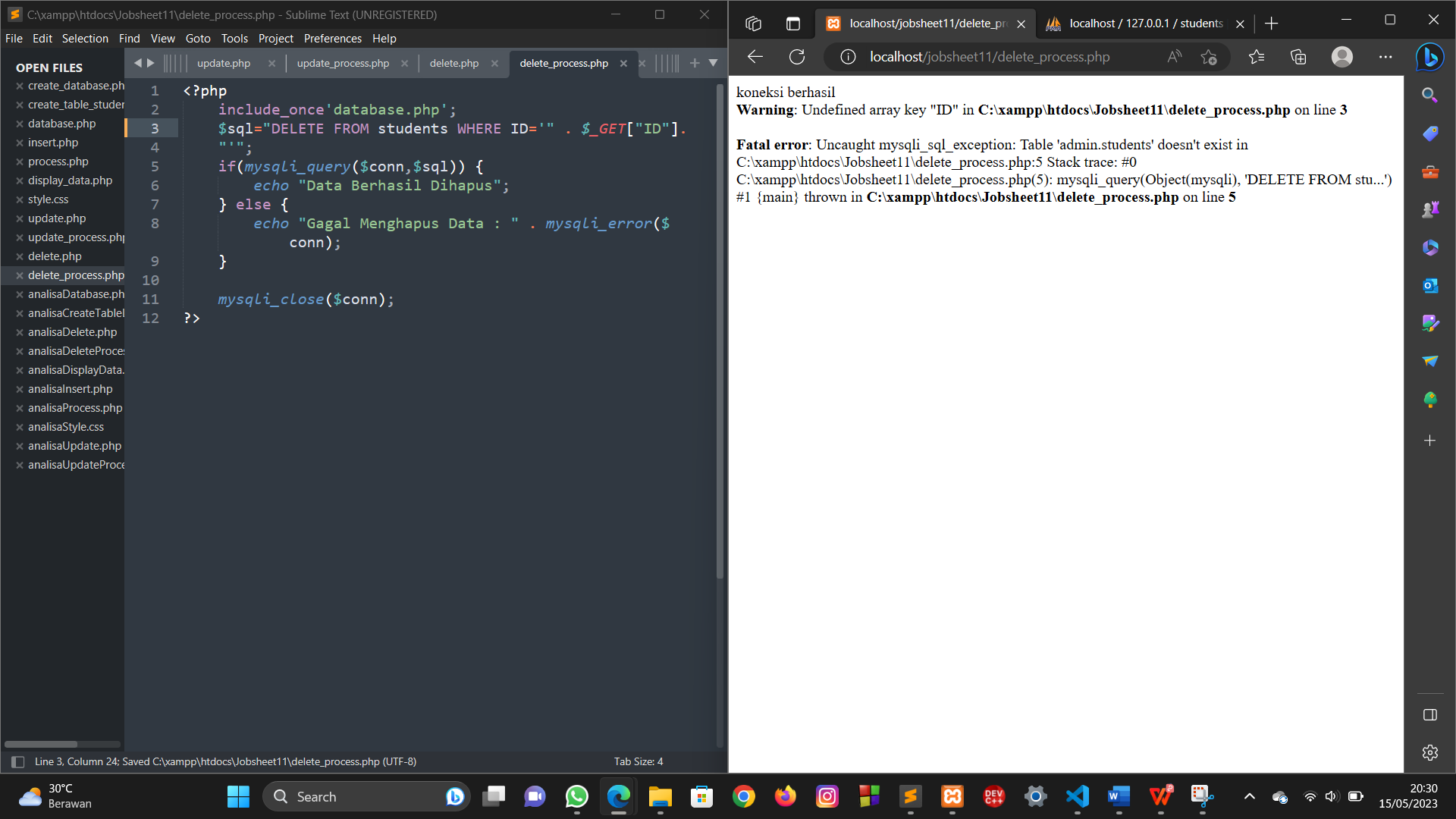
echo "Gagal Menghapus Data : " . mysqli\_error($conn);

}

mysqli\_close($conn);

?>

Hasil dan proses :

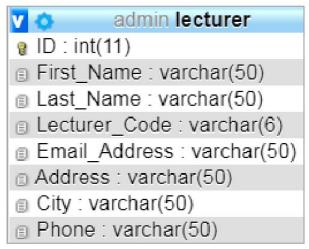


Penjelasan :

Code tersebut adalah script PHP yang berfungsi untuk menghapus data dari tabel "students" di database yang sedang aktif. Tujuan dari code ini adalah untuk menghapus data mahasiswa yang memiliki ID tertentu yang diterima melalui parameter URL ($\_GET).

1. **PROSES DAN HASIL EVALUASI/KASUS**

Membuat sebuah tabel baru bernama “lecturer” dengan struktur sebagai berikut:



Buat aplikasi web untuk select, insert, update dan delete seperti pada latihan diatas dan isi tabel minimal 20 data.

1. analisaCreateTableLecturer.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP , lalu mendefinisikan variabel $servername, $username, $password, dan $dbname dengan nilai yang sesuai.
3. Membuat koneksi ke database dengan menggunakan fungsi mysqli\_connect() dan menyimpannya pada variabel $conn. Kemudian, dilakukan pengecekan apakah koneksi berhasil atau tidak dengan menggunakan fungsi mysqli\_connect\_error().
4. Membuat query SQL untuk membuat tabel baru bernama "lecturer" dengan beberapa kolom seperti ID, FirstName, LastName, Lecturer\_Code, Email, Addres, City, dan Phone.
5. Menjalankan query SQL menggunakan fungsi mysqli\_query() dengan parameter $conn dan $sql. Kemudian, dilakukan pengecekan apakah query berhasil dijalankan atau tidak menggunakan fungsi mysqli\_error().
6. Menampilkan pesan "Tabel berhasil dibuat" jika query berhasil dijalankan atau pesan "Gagal membuat tabel:" beserta pesan error jika terdapat kesalahan.
7. Menutup koneksi ke database menggunakan fungsi mysqli\_close()
8. Menutup kodingan tag PHP ?>

Source code :

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "admin";

$conn = mysqli\_connect($servername, $username, $password, $dbname);

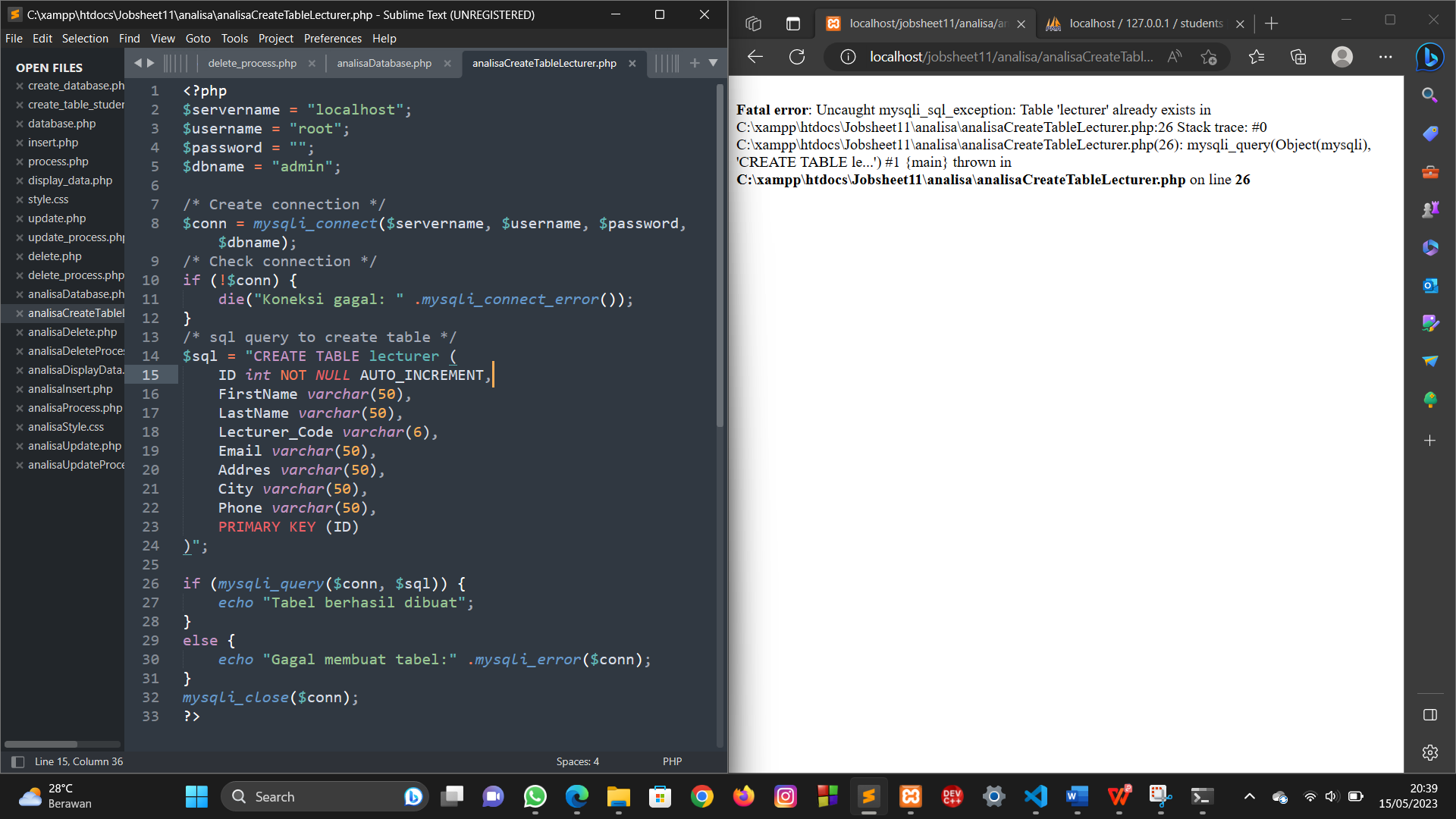
if (!$conn) {

die("Tidak bisa terhubung ke MySQL: " .mysqli\_connect\_error());

}

?>

Hasil dan proses :



1. analisaInsert.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag <html> , lalu membuat tag <body> didalam tag ini kita form dengan data yang diminta dan menekan tombol "Simpan Data", data akan disubmit ke file process.php.
3. Di dalam file process.php, data yang disubmit oleh pengguna akan diterima dan dapat diakses melalui $\_POST variable.
4. Setelah menerima data, file process.php dapat melakukan berbagai macam operasi terhadap data tersebut, seperti menyimpannya ke dalam database atau mengirimkan email konfirmasi ke pengguna.
5. Setelah selesai melakukan operasi terhadap data, file process.php dapat mengarahkan pengguna ke halaman lain, seperti halaman konfirmasi, atau kembali ke halaman utama dengan menggunakan fungsi header() pada PHP.
6. Menutup kodingan dengan menggunakan tag </html>

Source code :

<!DOCTYPE html>

<html>

<body>

<form method="post" action="process.php">

Nama depan: <br>

<input type="text" name="first\_name">

<br>

Nama belakang: <br>

<input type="text" name="last\_name">

<br>

Kode Pengajar: <br>

<input type="text" name="lecturer\_code">

<br>

Alamat email: <br>

<input type="email" name="email">

<br>

Alamat: <br>

<input type="text" name="addres">

<br>

Kota asal: <br>

<input type="text" name="city\_name">

<br>

Phone: <br>

<input type="text" name="phone">

<br><br>

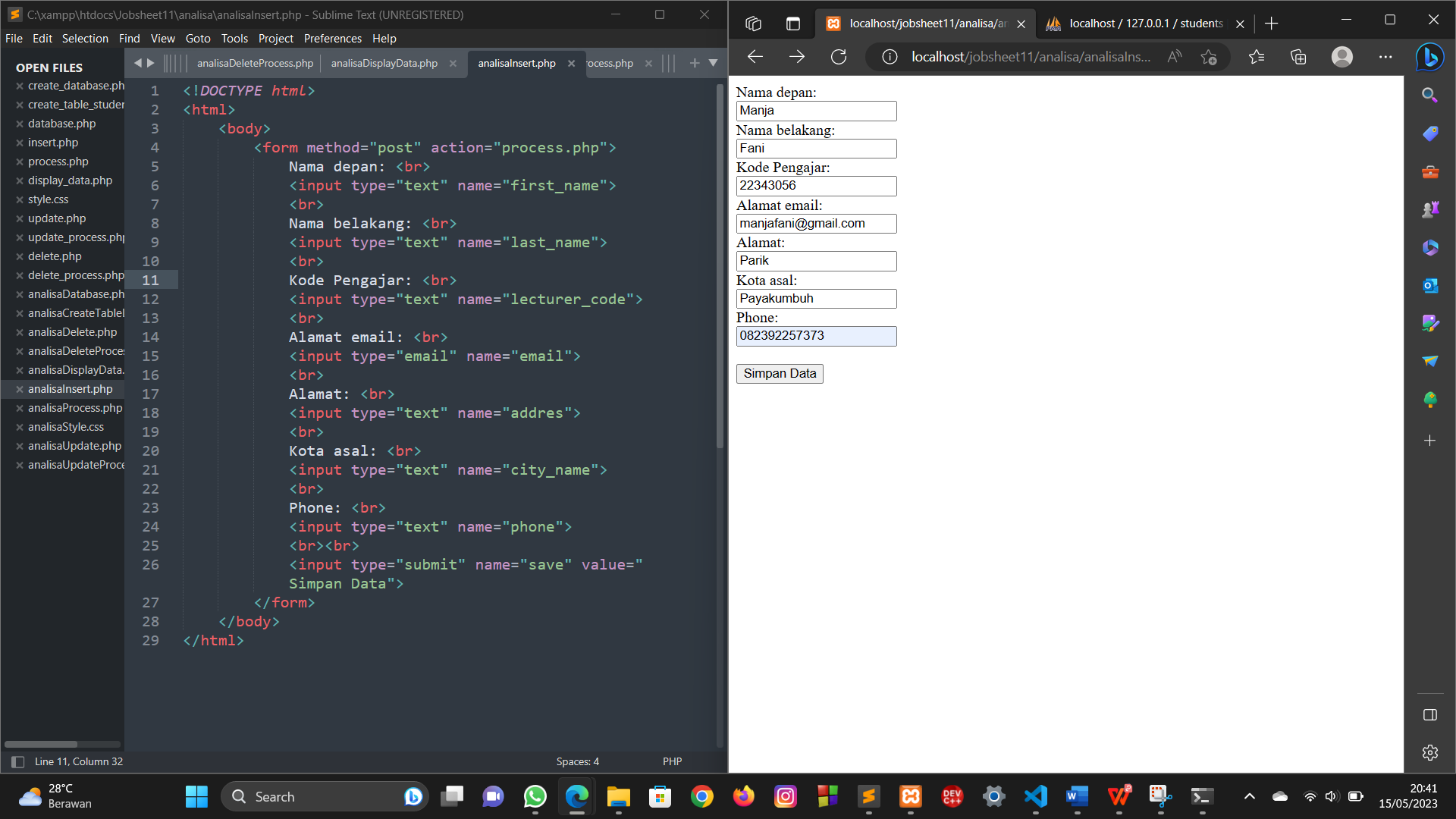
<input type="submit" name="save" value="Simpan Data">

</form>

</body>

</html>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web terdapat form yang terdiri dari Nama depan bernilai “ Manja “ , Nama belakang bernilai “ Fani” , Kode pengajar bernilai “ 22343056” ,Alamat email bernilai “ manjavia@gmail.com” , Alamat bernilai “ Parik “ , Kota asal bernilai “ Payakumbuh “ . Setelah dat diisikan lalu pencet kolom Simpan Data untuk menyimpa data.

1. analisaProcess.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP , lalu mengimport file analisaDatabase.php untuk menghubungkan ke database.
3. Mengecek apakah tombol "save" telah ditekan dengan menggunakan fungsi isset() pada variabel $\_POST.
4. Jika tombol "save" ditekan, maka ambil nilai dari form HTML menggunakan metode POST dan simpan di dalam variabel:
5. $first\_name untuk menyimpan nama depan pengajar.
6. $last\_name untuk menyimpan nama belakang pengajar.
7. $lecturer\_code untuk menyimpan kode pengajar.
8. $email untuk menyimpan email pengajar.
9. $addres untuk menyimpan alamat pengajar.
10. $city\_name untuk menyimpan nama kota pengajar.
11. $phone untuk menyimpan nomor telepon pengajar.
12. Memuat string SQL untuk memasukkan data ke dalam tabel "lecturer" dengan menggunakan perintah INSERT INTO.
13. Menjalankan perintah SQL menggunakan fungsi mysqli\_query() dengan parameter $conn dan $sql.
14. Jika berhasil, tampilkan pesan "Data berhasil disimpan !" menggunakan perintah echo.
15. Jika gagal, tampilkan pesan "Gagal: " beserta pesan kesalahan yang dihasilkan oleh mysqli\_error().
16. Menutup koneksi ke database menggunakan fungsi mysqli\_close().
17. Menampilkan link untuk melihat data pengajar menggunakan tag HTML <a> dan href "display\_data.php".

Source code :

<?php

include\_once 'analisaDatabase.php';

if (isset($\_POST['save'])) {

$first\_name = $\_POST['first\_name'];

$last\_name = $\_POST['last\_name'];

$lecturer\_code = $\_POST['lecturer\_code'];

$email = $\_POST['email'];

$addres = $\_POST['addres'];

$city\_name = $\_POST['city\_name'];

$phone = $\_POST['phone'];

$sql = "INSERT INTO lecturer

(FirstName, LastName, Lecturer\_Code, Email,Addres, City, Phone)

VALUES

('$first\_name','$last\_name', '$lecturer\_code','$email','$addres', '$city\_name', '$phone')";

if (mysqli\_query($conn, $sql)) {

echo "Data berhasil disimpan !";

} else {

echo "Gagal: " . $sql . "" . mysqli\_error($conn);

}

mysqli\_close($conn);

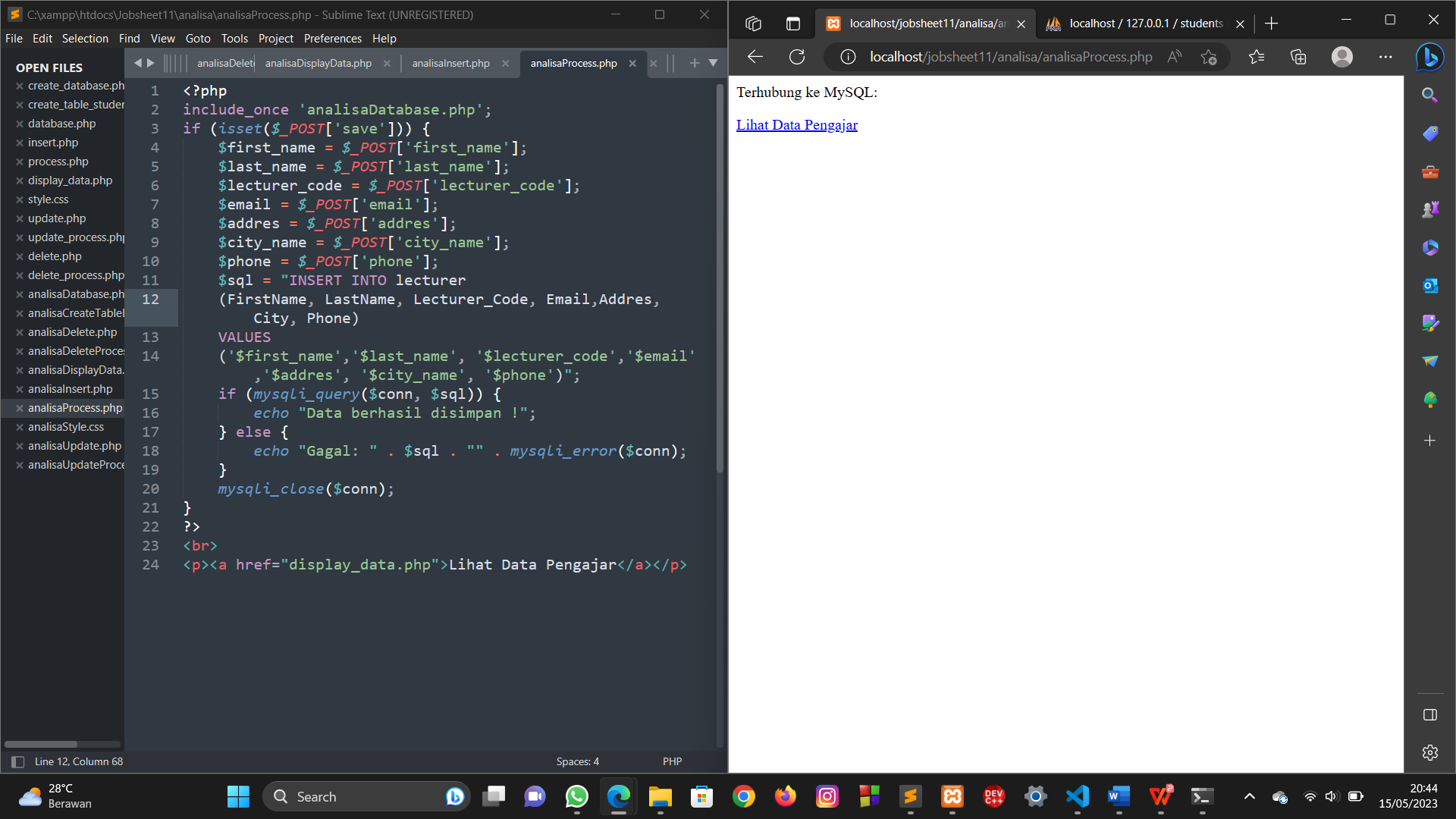
}

?>

<br>

<p><a href="display\_data.php">Lihat Data Pengajar</a></p>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web terdapat tautan berupa Lihat Data Mahasiswa.

1. analisaDisplayData.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Mengimpor file PHP analisaDatabase.php di-include agar koneksi ke database terbuka.
3. Membuat query SELECT \* FROM lecturer dieksekusi untuk mengambil semua data dari tabel pengajar.
4. Jika jumlah baris data yang diambil lebih dari 0, maka akan ditampilkan sebuah tabel dengan judul kolom "Nama Depan", "Nama Belakang", "Kode Pengajar", "Email", "Alamat", "Kota Asal", dan "Phone".
5. Setiap baris data dari tabel pengajar akan ditampilkan sebagai sebuah baris pada tabel yang telah dibuat pada langkah sebelumnya.
6. Jika tidak ada baris data yang ditemukan pada tabel pengajar, maka akan muncul pesan "Data tidak ditemukan".
7. Menampilkan tiga hyperlink yang mengarah ke halaman insert.php, update.php, dan delete.php yang masing-masing berfungsi untuk menambah, mengedit, dan menghapus data pada tabel pengajar.
8. Menutup kodingan dengan menggunakan tag </html>.

Source code :

<?php

include\_once 'analisaDatabase.php';

$result = mysqli\_query($conn, "SELECT \* FROM lecturer");

?>

<!DOCTYPE html>

<html>

<head>

<title>Menampilkan Data</title>

<link rel="stylesheet" type="text/css" href="style.css">

</head>

<body>

<?php

if (mysqli\_num\_rows($result) > 0) {

?>

<table>

<tr>

<td>Nama Depan</td>

<td>Nama Belakang</td>

<td>Kode Pengajar</td>

<td>Email</td>

<td>Alamat</td>

<td>Kota Asal</td>

<td>Phone</td>

</tr>

<?php

$i=0;

while ($row = mysqli\_fetch\_array($result)) {

?>

<tr>

<td><?php echo $row["FirstName"]; ?></td>

<td><?php echo $row["LastName"]; ?></td>

<td><?php echo $row["Lecturer\_Code"]; ?></td>

<td><?php echo $row["Email"]; ?></td>

<td><?php echo $row["Addres"]; ?></td>

<td><?php echo $row["City"]; ?></td>

<td><?php echo $row["Phone"]; ?></td>

</tr>

<?php

$i++;

}

?>

</table>

<?php

}

else {

echo "Data tidak ditemukan";

}

?>

<br>

<a href="insert.php">Tambah Data Pengajar</a>

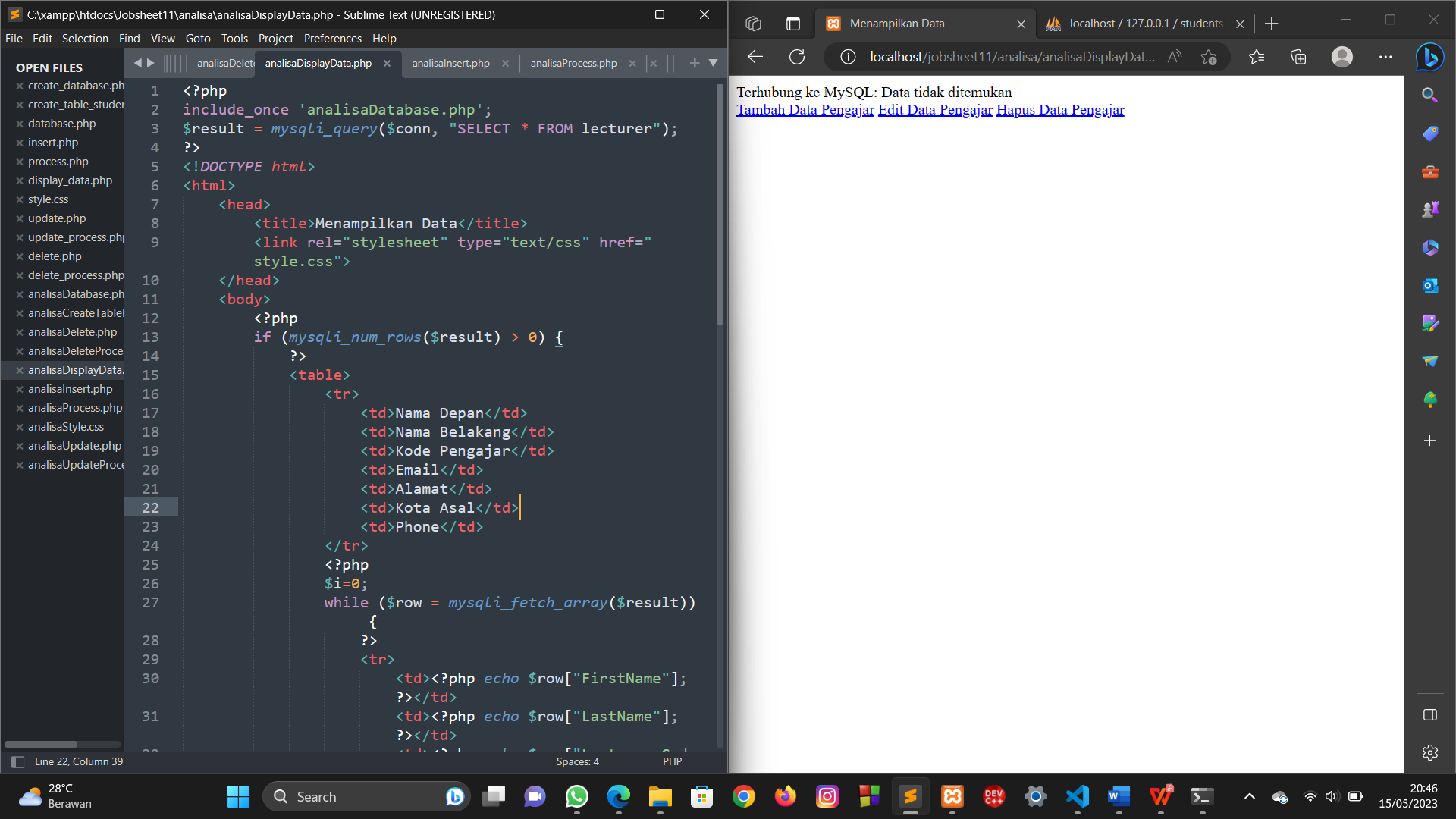
<a href="update.php">Edit Data Pengajar</a>

<a href="delete.php">Hapus Data Pengajar</a>

</body>

</html>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web tersebut terlihat bahwa data tidak ditemukan dan terdapat beberapa tautan berupa Tambah Data Pengajar , Edit Data Pengajar dan Hapus Data Pengajar.

1. analisaUpdate.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Mengimport file analisaDatabase.php yang berisi fungsi koneksi ke database.
3. Melakukan query ke database untuk memilih semua data dari tabel lecturer.
4. Mencek apakah ada data yang ditemukan dengan menggunakan mysqli\_num\_rows(). Jika iya, maka tampilkan data dalam bentuk tabel dengan menggunakan perulangan while dan mysqli\_fetch\_array().
5. Menambahkan tag HTML untuk menampilkan data dalam tabel.
6. Menambahkan tautan untuk mengakses halaman insert.php, update.php, dan delete.php.
7. Menyimpan file dengan nama yang sesuai (misalnya, tampilanData.php).
8. Menjalankan program dengan mengakses file tersebut melalui web server yang terhubung dengan database yang sama.
9. Menutup kodingan dengan menggunakan tag </html>

Source code :

<?php

include\_once 'analisaDatabase.php';

$result = mysqli\_query($conn,"SELECT \* FROM lecturer");

?>

<! DOCTYPE html>

<html>

<head>

<title>Menampilkan Data</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<?php

if (mysqli\_num\_rows($result) > 0){

?>

<table>

<tr>

<td>No</td>

<td>Nama Depan</td>

<td>Nama Belakang</td>

<td>Kode Pengajar</td>

<td>Email</td>

<td>Alamat</td>

<td>Kota Asal</td>

<td>Phone</td>

<td>Aksi</td>

</tr>

<?php

$i=0;

while($row = mysqli\_fetch\_array($result)) {

?>

<tr>

<td><?php echo $row["ID"]; ?></td>

<td><?php echo $row["FirstName"]; ?></td>

<td><?php echo $row["LastName"]; ?></td>

<td><?php echo $row["Lecturer\_Code"]; ?></td>

<td><?php echo $row["Email"]; ?></td>

<td><?php echo $row["Addres"]; ?></td>

<td><?php echo $row["City"]; ?></td>

<td><?php echo $row["Phone"]; ?></td>

<td><a href="update-process.php?ID=<?php echo $row["ID"]; ?>">Update</a></td>

</tr>

<?php

$i++;

}

?>

</table>

<?php

}

else{

echo"Data tidak ditemukan";

}

?>

<br>

<a href="insert.php">Tambah Data Pengajar</a> |

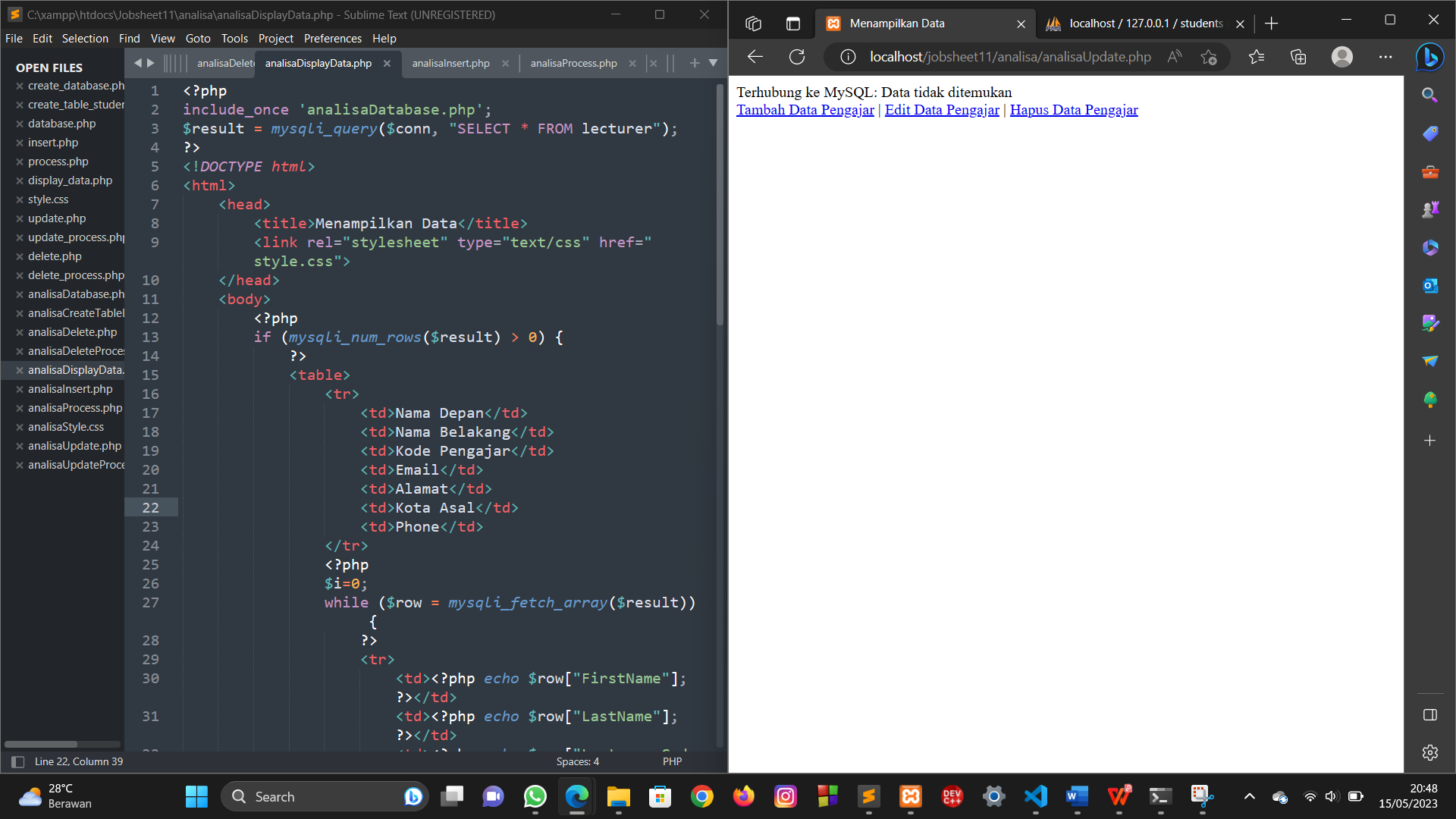
<a href="update.php">Edit Data Pengajar</a> |

<a href="delete.php"> Hapus Data Pengajar</a>

</body>

</html>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web tersebut terlihat bahwa data tidak ditemukan dan terdapat beberapa tautan berupa Tambah Data Pengajar , Edit Data Pengajar dan Hapus Data Pengajar.

1. analisaUpdateProcess.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text .
2. Membuat tag PHP , memanggil file "analisaDatabase.php" menggunakan fungsi "include\_once".
3. Setelah itu, kode PHP akan melakukan pengecekan apakah ada data yang dikirimkan melalui metode POST atau tidak. Jika ada, maka akan dilakukan proses update data di tabel "lecturer" dengan menggunakan perintah SQL "UPDATE".
4. Setelah proses update selesai dilakukan, maka akan ditampilkan pesan "Data Berhasil Diperbaharui".
5. Selanjutnya, kode PHP akan melakukan query ke tabel "lecturer" untuk mengambil data pengajar berdasarkan ID yang diterima melalui parameter GET.
6. Data pengajar yang ditemukan akan ditampilkan dalam form input yang dapat diubah.
7. Ketika pengguna menekan tombol "Update Data", maka data yang telah diubah akan disimpan ke dalam database menggunakan perintah SQL "UPDATE" yang sama seperti pada langkah 2.
8. Setelah proses update selesai dilakukan, maka akan ditampilkan pesan "Data Berhasil Diperbaharui" dan pengguna akan diarahkan kembali ke halaman "display\_data.php" yang menampilkan daftar pengajar yang ada di database.
9. Jika pengguna tidak melakukan perubahan pada data pengajar dan langsung menekan tombol "Update Data", maka tidak akan ada perubahan data yang disimpan ke dalam database.
10. Menutup kodingan dengan menggunakan tag </html>

Source code :

<?php

include\_once 'analisaDatabase.php';

if (count($\_POST) > 0) {

mysqli\_query($conn, "UPDATE lecturer set ID='" . $\_POST['ID'] . "'

,FirstName='" . $\_POST['first\_name'] . "'

,LastName='" . $\_POST['last\_name'] . "'

,Lecturer\_Code='" . $\_POST['lecturer\_code'] . "'

,Email='" . $\_POST['email'] . "'

,Addres='" . $\_POST['addres'] . "'

,City='" . $\_POST['city\_name'] . "'

,Phone='" . $\_POST['phone'] . "'

WHERE ID='" . $\_POST['ID'] . "'");

$message = "Data Berhasil Diperbaharui";

}

$result = mysqli\_query($conn, "SELECT \* FROM lecturer WHERE ID='" . $\_GET['ID'] . "'");

$row = mysqli\_fetch\_array($result);

?>

<html>

<head>

<title>Update Data Pengajar</title>

</head>

<body>

<form name="frmUser" method="post" action="">

<div>

<?php if (isset($message)) {

echo $message;

} ?>

</div>

<div style="padding-bottom:5px;">

<a href="display\_data.php">Daftar Pengajar</a>

</div>

ID : <br>

<input type="hidden" name="ID" class="txtField" value="<?php echo $row['ID']; ?>">

<input type="text" name="ID" value="<?php echo $row['ID']; ?>">

<br><br>

Nama Depan : <br>

<input type="text" name="first\_name" class="txtField" value="<?php echo $row['FirstName']; ?>">

<br><br>

Nama Belakang : <br>

<input type="text" name="last\_name" class="txtField" value="<?php echo $row['LastName']; ?>">

<br><br>

Kode Pengajar : <br>

<input type="text" name="lecturer\_code" class="txtField" value="<?php echo $row['Lecturer\_Code']; ?>">

<br><br>

Alamat Email : <br>

<input type="text" name="email" class="txtField" value="<?php echo $row['Email']; ?>">

<br><br>

Alamat : <br>

<input type="text" name="addres" class="txtField" value="<?php echo $row['Addres']; ?>">

<br><br>

Kota Asal : <br>

<input type="text" name="city\_name" class="txtField" value="<?php echo $row['City']; ?>">

<br><br>

Phone : <br>

<input type="text" name="phone" class="txtField" value="<?php echo $row['Phone']; ?>">

<br><br>

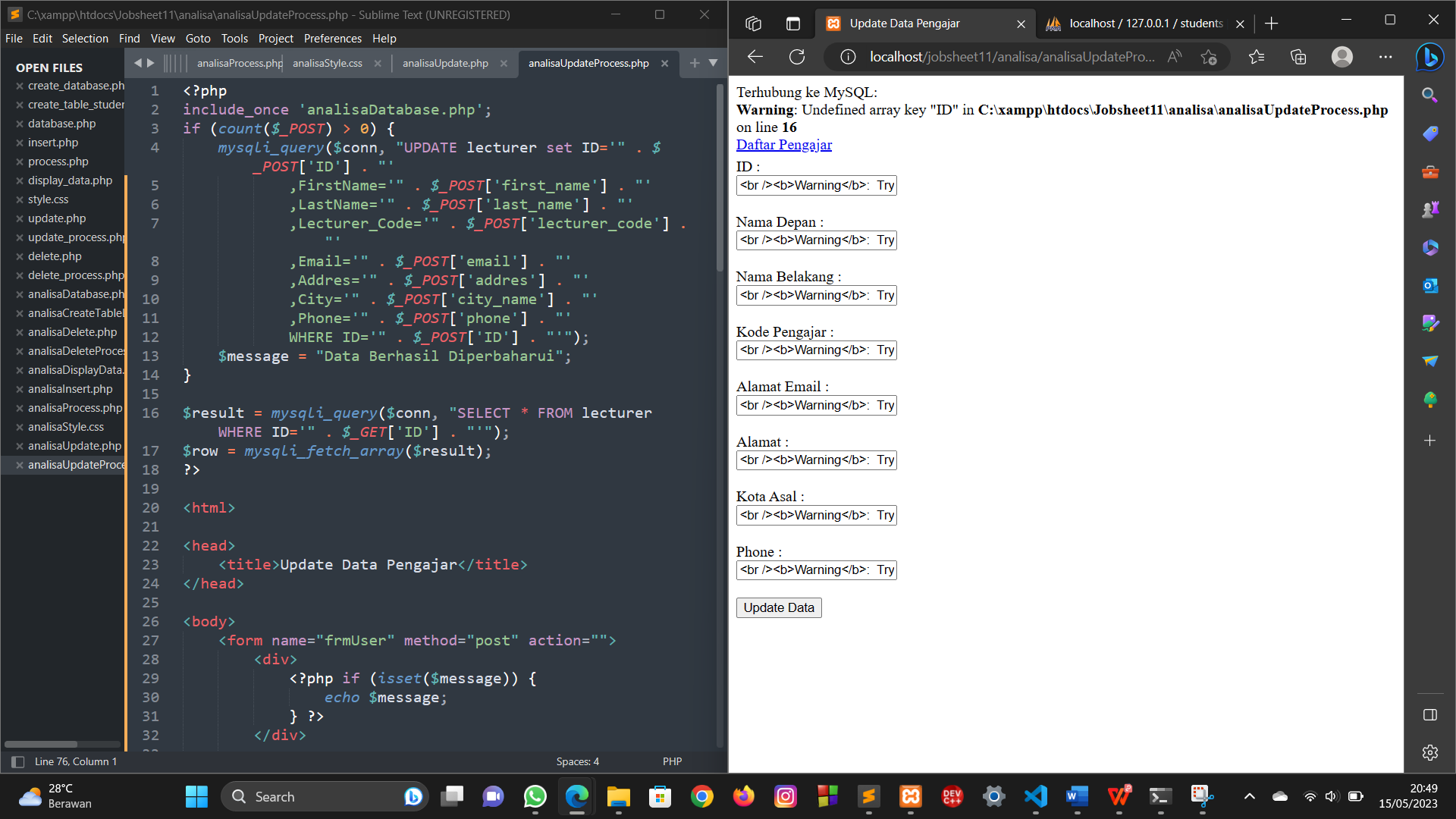
<input type="submit" name="submit" value="Update Data" class="button">

</form>

</body>

</html>

Hasil dan proses :



1. analisaDelete.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Membuat tag PHP untuk menjalankan include\_once 'analisaDatabase.php'; yang akan memasukkan file analisaDatabase.php ke dalam script. File tersebut akan berisi fungsi koneksi ke database.
3. Membuat atribut $result =mysqli\_query($conn,"SELECT \* FROM lecturer"); akan mengeksekusi query SELECT \* FROM lecturer ke database menggunakan koneksi $conn yang sudah terbuka sebelumnya. Query tersebut akan mengambil semua data dari tabel lecturer dan hasil query akan disimpan dalam variabel $result.
4. Menampilkan data yang ada di dalam variabel $result ke dalam bentuk tabel. Data akan ditampilkan di dalam tag <table>.
5. Pada baris pertama dari tabel adalah header yang berisi judul kolom. Nama kolom yang ditampilkan di dalam header adalah No, Nama Depan, Nama Belakang, Kode Pengajar, Email, Alamat, Kota Asal, Phone, dan Aksi.
6. Membuat perulangan while ($row = mysqli\_fetch\_array($result)) akan mengeksekusi perulangan untuk setiap baris data yang ada di dalam variabel $result. Setiap baris data akan disimpan di dalam variabel $row.
7. Selama perulangan berlangsung, kode <?php echo $row ["ID"]; ?> akan menampilkan nilai dari kolom ID dari setiap baris data di dalam kolom No.
8. Membuat <?php echo $row ["FirstName"]; ?> akan menampilkan nilai dari kolom FirstName di dalam kolom Nama Depan.
9. Membuat <?php echo $row ["LastName"]; ?> akan menampilkan nilai dari kolom LastName di dalam kolom Nama Belakang.
10. Membuat <?php echo $row ["Lecturer\_Code"]; ?> akan menampilkan nilai dari kolom Lecturer\_Code di dalam kolom Kode Pengajar.
11. Membuat <?php echo $row ["Email"]; ?> akan menampilkan nilai dari kolom Email di dalam kolom Email.
12. Membuat <?php echo $row ["Addres"]; ?> akan menampilkan nilai dari kolom Addres di dalam kolom Alamat.
13. Membuat <?php echo $row ["City"]; ?> akan menampilkan nilai dari kolom City di dalam kolom Kota Asal.
14. Membuat <?php echo $row ["Phone"]; ?> akan menampilkan nilai dari kolom Phone di dalam kolom Phone.
15. Membuat <a href = "delete-process.php?ID=<?php echo $row ["ID"]; ?>">Delete</a> akan menampilkan tombol delete untuk setiap baris data. Tombol tersebut akan mengirimkan data ID ke halaman delete-process.php ketika tombol tersebut diklik.
16. Setelah perulangan selesai, kode </table> akan menutup tag <table>.
17. Membuat <br> digunakan untuk membuat baris kosong.
18. Membuat <a href="insert.php">Tambah Data Pengajar</a>| <a href="update.php">Edit Data Pengajar</a>| <a href="delete.php">Hapus Data Pengajar</a> digunakan untuk menampilkan link yang akan mengarahkan pengguna.
19. Membuat kodingan dengan menggunkan tag </html>

Source code :

<?php include\_once 'analisaDatabase.php';

$result =mysqli\_query($conn,"SELECT \* FROM lecturer");

?>

<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" href="style.css">

<title>Delete Data Pengajar</title>

</head>

<body>

<table>

<tr>

<td>No</td>

<td>Nama Depan</td>

<td>Nama Belakang</td>

<td>Kode Pengajar</td>

<td>Email</td>

<td>Alamat</td>

<td>Kota Asal</td>

<td>Phone</td>

<td>Aksi</td>

</tr>

<?php

$i=0;

while ($row = mysqli\_fetch\_array($result)){

?>

<tr class="<?php if (isset ($classname) ) echo $classname;?>">

<td><?php echo $row ["ID"]; ?></td>

<td><?php echo $row ["FirstName"]; ?></td>

<td><?php echo $row ["LastName"]; ?></td>

<td><?php echo $row ["Lecturer\_Code"]; ?></td>

<td><?php echo $row ["Email"]; ?></td>

<td><?php echo $row ["Addres"]; ?></td>

<td><?php echo $row ["City"]; ?></td>

<td><?php echo $row ["Phone"]; ?></td>

<td><a href = "delete-process.php?ID=<?php echo $row ["ID"]; ?>">Delete</a></td>

</tr>

<?php

$i++;

}

?>

</table>

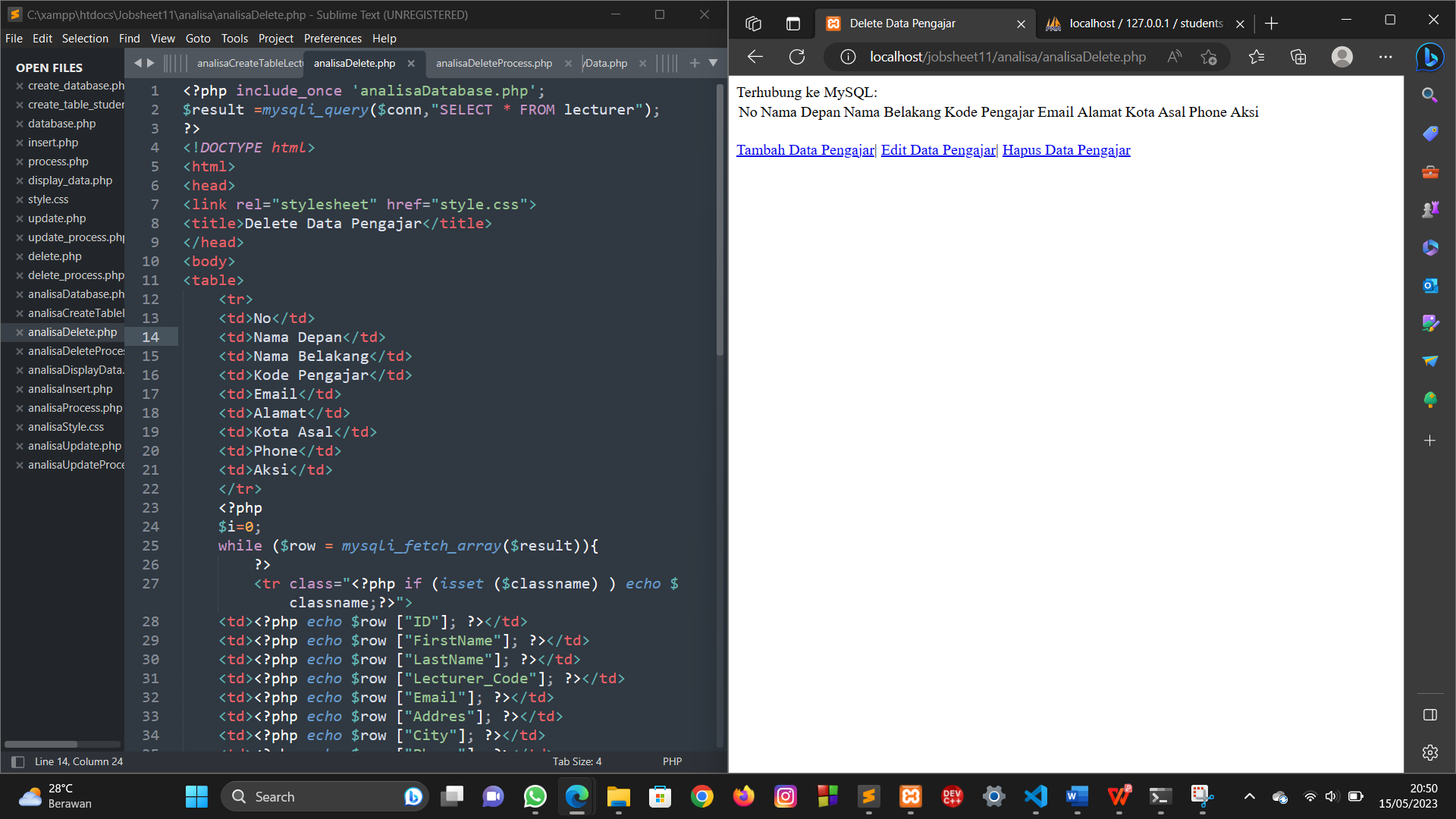
<br>

<a href="insert.php">Tambah Data Pengajar</a>| <a href="update.php">Edit Data Pengajar</a>| <a href="delete.php">Hapus Data Pengajar</a>

</body>

</html>

Hasil dan proses :



Penjelasan :

Berdasarkan program yang dilakukan , output dari program yang dilakukan seperti gambar diatas . Pada halaman web tersebut terdapat tabel beruisi No , Nama depan , Nama belakang , Kode pengajar , , Email , Alamat , Kota Asal , Phone dan Aksi bahwa data tidak ditemukan dan terdapat beberapa tautan berupa Tambah Data Pengajar , Edit Data Pengajar dan Hapus Data Pengajar.

1. analisaDeleteProcess.php

Langkah kerja :

1. Membuka teks editor yaitu Sublime Text.
2. Mengimpor atau memuat file yang berisi fungsi analisaDatabase.
3. Membuat sebuah variabel $sql, yang berisi perintah SQL untuk menghapus data pada tabel lecturer. Perintah SQL ini mengambil data berdasarkan ID yang dikirimkan melalui metode GET.
4. Menjalankan perintah SQL untuk menghapus data dengan menggunakan fungsi mysqli\_query. Jika perintah SQL berhasil dijalankan, maka pesan "Data berhasil dihapus" akan ditampilkan pada halaman web.
5. Jika terdapat kesalahan dalam menjalankan perintah SQL, maka akan muncul pesan "Gagal menghapus data" dan pesan kesalahan spesifik dari mysqli\_error.
6. Membuat mysqli\_close untuk menutup koneksi database setelah perintah SQL dijalankan.
7. Menutup tag PHP untuk menutup kodingan .

Source code :

<?php

include\_once 'analisaDatabase';

$sql = "DELETE FROM lecturer WHERE ID='" . $\_GET["ID"] ."'";

if (mysqli\_query($conn, $sql)) {

echo "Data berhasil dihapus";

} else {

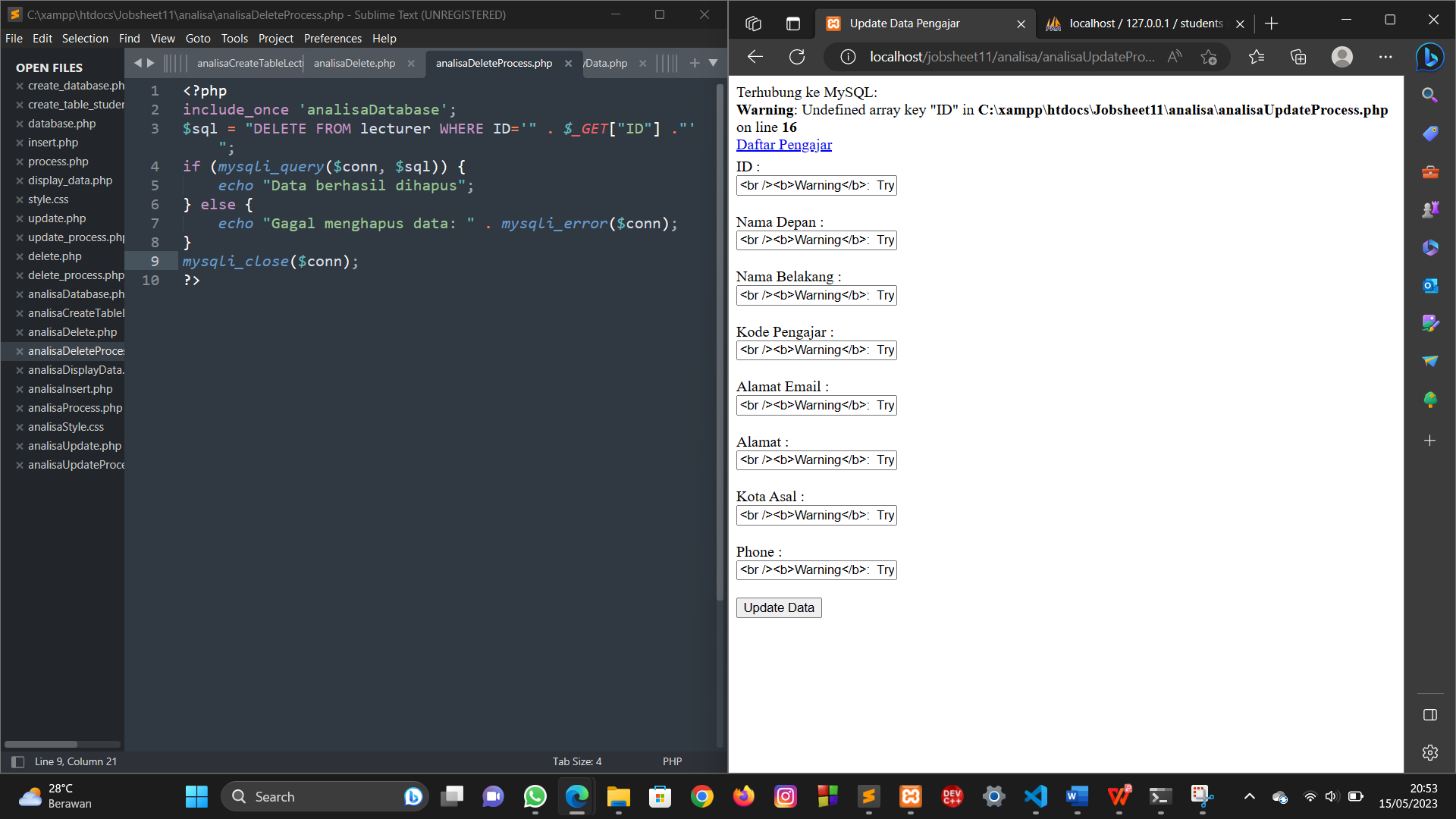
echo "Gagal menghapus data: " . mysqli\_error($conn);

}

mysqli\_close($conn);

?>

Hasil dan proses :



1. **SIMPULAN**

Mempelajari query dasar MySQL merupakan langkah awal yang penting dalam menguasai penggunaan MySQL sebagai sistem manajemen basis data (DBMS). Dalam mempelajari query dasar MySQL, kita dapat mencapai beberapa kesimpulan sebagai berikut:

1. Query adalah perintah-perintah yang digunakan untuk berinteraksi dengan basis data MySQL. Dengan mempelajari query dasar, kita dapat memahami cara mengambil, menyisipkan, memperbarui, dan menghapus data dalam basis data.
2. Struktur dasar query MySQL meliputi perintah SELECT, INSERT, UPDATE, DELETE, dan beberapa perintah lainnya. Memahami sintaks dan fungsionalitas masing-masing perintah ini akan memungkinkan kita untuk melakukan operasi yang berbeda pada basis data.
3. Query SELECT digunakan untuk mengambil data dari satu atau lebih tabel dalam basis data. Dalam mempelajari query SELECT, kita dapat memahami konsep penulisan klausa SELECT, FROM, WHERE, ORDER BY, dan banyak lagi. Hal ini memungkinkan kita untuk mengambil data dengan kriteria yang spesifik dan mengurutkannya sesuai kebutuhan.
4. Query INSERT digunakan untuk menyisipkan data baru ke dalam tabel. Dalam mempelajari query INSERT, kita akan belajar cara menyisipkan data dalam bentuk tunggal maupun dalam jumlah yang lebih besar.
5. Query UPDATE digunakan untuk memperbarui data yang sudah ada dalam tabel. Dalam mempelajari query UPDATE, kita akan belajar cara mengubah nilai-nilai dalam kolom-kolom tertentu berdasarkan kriteria yang ditentukan.
6. Query DELETE digunakan untuk menghapus data dari tabel. Dalam mempelajari query DELETE, kita akan belajar cara menghapus data dengan kondisi tertentu atau menghapus seluruh baris dalam tabel.
7. Dalam mempelajari query dasar MySQL, kita juga akan belajar tentang beberapa fungsi dan klausa penting lainnya, seperti fungsi agregat (SUM, COUNT, AVG, dll.), pengelompokan data dengan GROUP BY, penggabungan tabel dengan JOIN, pembatasan hasil dengan LIMIT, dan lain sebagainya.

Menguasai query dasar MySQL merupakan langkah penting dalam mengembangkan keterampilan pengelolaan basis data. Dengan memahami dan menguasai konsep dan sintaks dasar ini, kita akan dapat membuat query yang lebih kompleks dan efisien untuk mengambil, menyisipkan, memperbarui, dan menghapus data dalam basis data MySQL