**LAPORAN PRAKTIKUM PEMROGRAMAN WEB & MOBILE I**



**Nama : JERI**

**NIM : 193030503055**

**Modul : III (PHP MySQL Database)**

**Kelas : A**

**JURUSAN TEKNIK INFORMATIKA FAKULTAS TEKNIK UNIVERSITAS PALANGKARAYA**

**2021**

**BAB I**

**TUJUAN DAN LANDASAN TEORI**

**1.1 Tujuan**

• Mahasiswa mampu membuat program yang bisa menyimpan data dalam jumlah yang banyak.

• Mahasiswa mampu membuat program yang bisa mengolah data yang tersimpan dalam database

**1.2 Landasan Teori**

Data dalam database MySQL disimpan dalam tabel-tabel. Sebuah tabel adalah koleksi dari data yang berelasi dan mengandung kolom dan baris. Database sangat bermanfaat untuk menyimpan informasi secara kategori. Contoh yang akan diberikan pada modul praktikum ini adalah tabel yang mengandung data Employees (Pekerja), Products (Produk), Customers (Pelanggan) dan Orders (Pesanan).

1. **Membuka Koneksi**

Sebelum mengakses data dalam database MySQL, kita harus terhubung ke server database MySQL. Berikut adalah contoh kode program agar terhubung dengan server mySQL:

<?php $servername = "localhost"; $username = "username"; $password = "password";

// Membuat hubungan $conn = new mysqli($servername, $username, $password);

// Memeriksa hubungan if ($conn->connect\_error){ die("Connection failed: " . $conn->connect\_error); } echo "Connected successfully"; ?>

Jika kode diatas tidak berhasil, kemungkinan variabel $connect\_error sudah tidak tersedia pada versi PHP yang kita gunakan, maka gunakan kode program berikut:

<?php

$servername = "localhost"; $username = "username"; $password = "password";

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

// Memeriksa Hubungan if (!$conn){ die("Connection failed: " . mysqli\_connect\_error()); } echo "Connected successfully"; ?>

Ketika selesai menggunakan data dari database, sebaiknya koneksi atau hubungan ke server ditutup, caranya dengan menggukan kode program berikut:

mysqli\_close($conn);

1. **Membuat Database**

Database pada MySQL bisa juga dibuat menggunakan kode program PHP. Program tersebut akan berisi statement SQL “CREATE DATABASE”. Berikut adalah contoh program yang digunakan untuk membuat database “myDB”:

<?php $servername = "localhost"; $username = "username"; $password = "password";

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

// Memeriksa Koneksi if (!$conn){ die("Connection failed: " . mysqli\_connect\_error()); }

// Membuat Database $sql = "CREATE DATABASE myDB"; if (mysqli\_query($conn, $sql)) {

echo "Database created successfully"; } else { echo "Error creating database: " . mysqli\_error($conn); }

mysqli\_close($conn); ?>

1. **Membuat Tabel**

embuatan tabel pada bahasa pemrograman PHP juga menggunakan statement SQL, yaitu statement “CREATE TABLE”. Contoh yang akan diberikan adalah pembuatan tabel MyGuests. Statement pembuatan tabel MyGuests adalah sebagai berikut:

CREATE TABLE MyGuests (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY, firstname VARCHAR(30) NOT NULL, lastname VARCHAR(30) NOT NULL,

email VARCHAR(50),

reg\_date TIMESTAMP

)

Berikut adalah contoh program untuk membuat tabel diatas:

<?php $servername = "localhost"; $username = "username"; $password = "password"; $dbname = "myDB";

// Membuat Koneksi $conn = mysqli\_connect($servername, $username, $password, $dbname); // Memeriksa koneksi if (!$conn){ die("Connection failed: " . mysqli\_connect\_error()); }

// SQL untuk membuat tabel $sql = "CREATE TABLE MyGuests (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY, firstname VARCHAR(30) NOT NULL, lastname VARCHAR(30) NOT NULL, email VARCHAR(50), reg\_date TIMESTAMP )";

if (mysqli\_query($conn, $sql)) { echo "Table MyGuests created successfully"; } else { echo "Error creating table: " . mysqli\_error($conn); }

mysqli\_close($conn); ?>

1. **Memasukkan Data ke Database**

Untuk memasukkan data kedalam tabel di database, terdapat beberapa aturan sintaks yang harus diikuti:

• Query SQL harus diberikan kutip dalam PHP.

• Nilai string didalam query SQL harus diberikan kutip.

• Nilai numeris tidak harus diberikan kutip.

• Nilai NULL tidak harus diberikan kutip.

Statement INSERT INTO digunakan untuk menambahkan baris pada tabel MySQL, query untuk menambahkan data bersebut adalah sebagai berikut:

INSERT INTO table\_name (column1, column2, column3,...)

VALUES (value1, value2, value3,…)

adapun contoh kode program PHP untuk menambahkan data tersebut adalah sebagai berikut:

<?php $servername = "localhost"; $username = "username"; $password = "password";

$dbname = "myDB";

// Membuat Koneksi $conn = mysqli\_connect($servername, $username, $password, $dbname); // Memeriksa Koneksi if (!$conn){ die("Connection failed: " . mysqli\_connect\_error()); }

$sql = "INSERT INTO MyGuests (firstname, lastname, email) VALUES ('John', 'Doe', 'john@example.com')";

if (mysqli\_query($conn, $sql)) { echo "New record created successfully"; } else { echo "Error: " . $sql . "<br>" . mysqli\_error($conn); }

mysqli\_close($conn); ?>

Jika kita melakukan perintah INSERT pada tabel yang menggunakan field dengan opsi AUTO\_INCREMENT, kita bisa mendapatkan ID dari baris yang terakhir diinputkan. Caranya menggunakan fungsi mysqli\_insert\_id($conn); nilai kembalian dari fungsi ini akan memberikan id dari record atau baris terakhir yang diinputkan.

Untuk menambahkan record dengan jumlah yang banyak, kita bisa menggunakan fungsi mysqli\_multi\_query($conn, $sql), berikut adalah contoh program untuk menambahkan record dengan jumlah yang banyak:

<?php $servername = "localhost"; $username = "username"; $password = "password"; $dbname = "myDB";

// Membuat Koneksi $conn = mysqli\_connect($servername, $username, $password, $dbname); // Memeriksa Koneksi if (!$conn){

die("Connection failed: " . mysqli\_connect\_error()); } $sql = "INSERT INTO MyGuests (firstname, lastname, email) VALUES ('John', 'Doe', 'john@example.com');"; $sql .= "INSERT INTO MyGuests (firstname, lastname, email) VALUES ('Mary', 'Moe', 'mary@example.com');"; $sql .= "INSERT INTO MyGuests (firstname, lastname, email) VALUES ('Julie', 'Dooley', 'julie@example.com')";

if (mysqli\_multi\_query($conn, $sql)) { echo "New records created successfully"; } else { echo "Error: " . $sql . "<br>" . mysqli\_error($conn); }

mysqli\_close($conn); ?>

1. **Mengambil Data dari Database**

Untuk mengambil data, statement SQL yang digunakan adalah SELECT nama\_kolom() FROM nama\_tabel, atau kita bisa menggunakan karakter \* untuk memilih semua kolom yang ada pada tabel. Pengambilan data dengan kriteria tertentu bisa dilakukan dengan menggunakan statement WHERE setelah nama\_tabel. Adapun contoh kode program untuk mengambil data dari database adalah sebagai berikut:

<?php $servername = "localhost"; $username = "username"; $password = "password"; $dbname = "myDB";

// Membuat koneksi $conn = mysqli\_connect($servername, $username, $password, $dbname); // Memeriksa Koneksi if (!$conn){ die("Connection failed: " . mysqli\_connect\_error()); }

$sql = "SELECT id, firstname, lastname FROM MyGuests"; $result = mysqli\_query($conn, $sql);

if (mysqli\_num\_rows($result) > 0) { // Menampilkan data pada setiap baris while($row = mysqli\_fetch\_assoc($result)) { echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>"; } } else { echo "0 results"; }

mysqli\_close($conn); ?>

1. **Menghapus Data dari Database**

Statement DELETE digunakan untuk menghapus baris data atau records dari tabel. Clause WHERE digunakan untuk menspesifikasikan baris yang akan dihapus. Jika statement DELETE digunakan tanpa menggunakan clausa WHERE, maka semua record yang ada pada tabel akan dihapus.

1. **Update Data dalam Database**

Untuk melakukan perubahan data di dalam database, statement UPDATE digunakan, yaitu sebagai berikut:

UPDATE nama\_tabel

SET kolom1=nilai1, kolom2=nilai2, …

WHERE kolom\_penentu=nilai\_penentu …

Jika statement ini digunakan untuk update data ke dua pada tabel “MyGuest” diatas untuk mengganti nama belakang dari “Moe” menjadi “Doe”, maka kode program untuk update tersebut adalah sebagai berikut:

<?php $servername = "localhost"; $username = "username"; $password = "password"; $dbname = "myDB";

// Membuat Koneksi $conn = mysqli\_connect($servername, $username, $password, $dbname); // Memeriksa Koneksi if (!$conn){

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

$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";

if (mysqli\_query($conn, $sql)) { echo "Record updated successfully"; } else { echo "Error updating record: " . mysqli\_error($conn); }

mysqli\_close($conn); ?>

**BAB II**

**PEMBAHASAN**

Membuat database pada MySql untuk menyimpan data-data pegawai beserta relasi tabelnya dan membuatr program untuk menyimpan, menghapus dan mengubah datadata pegawai. Berikut ini adalah penjelasan nya :

]

Langkah yang pertama adalam membuat koneksi, dengan nama Koneksi.php yang merupakan tempat dimana query dapat mengakses database, host, user, name yang ingin dihubungkan ke program.

<?php

$host = "localhost";    //nama host

$user = "root"; //username phpMyAdmin

$pass = ""; //password phpMyAdmin

$name = "pegawai";  //nama database

$koneksi = mysql\_connect($host, $user, $pass) or die("Koneksi ke database gagal!");

mysql\_select\_db($name, $koneksi) or die("Tidak ada database yang dipilih!");

?>

Gambar 2.1 Koneksi.php

Langkah selanjutnya adalah membuat index

<!DOCTYPE html>

<html>

<head>

    <title>TUGAS PEMROGRAMAN WEB & MOBILE I</title>

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

</head>

<body>

    <h1>DATA PEGAWAI</h1>

    <h2><a href="index.php">Beranda</a> / <a href="tambah.php">Tambah Data</a></h2>

    <table class="styled-table">

        <tr bgcolor="#f0ffff">

            <th>No.</th>

            <th>Nama Lengkap</th>

            <th>Bagian</th>

            <th>Gajih</th>

            <th>Opsi</th>

        </tr>

        <?php

        include('koneksi.php');

        $query = mysql\_query("SELECT \* FROM pegawai ORDER BY pegawai\_bagian DESC") or die(mysql\_error());

        if(mysql\_num\_rows($query) == 0){

            echo '<tr><td colspan="6">Tidak ada data!</td></tr>';

        }else{

            $no = 1;

            while($data = mysql\_fetch\_assoc($query)){

                echo '<tr>';

                    echo '<td>'.$no.'</td>';

                    echo '<td>'.$data['pegawai\_nama'].'</td>';

                    echo '<td>'.$data['pegawai\_bagian'].'</td>';

                    echo '<td>'.$data['pegawai\_gajih'].'</td>';

                    echo '<td><a href="edit.php?id='.$data['pegawai\_id'].'">Edit</a> / <a href="hapus.php?id='.$data['pegawai\_id'].'" onclick="return confirm(\'Yakin?\')">Hapus</a></td>';

                echo '</tr>';

                $no++;

            }

        }

        ?>

    </table>

</body>

</html>

Gambar 2.2 index.php

Penjelasan : Nama yang diberikan pada file php ini adalah Index.php merupakan tampilan yang akan diperlihatkan ketika program diakses, pada index nantinya akan ada tampilan table dan isinya serta opsi apakah data akan ditambah, dihapus dan diedit.

Pada bagian dibawah ini merupakan kode program edit.

<!DOCTYPE html>

<html>

<head>

    <title>TUGAS PEMROGRAMAN WEB & MOBILE I</title>

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

</head>

<body>

    <h1>DATA PEGAWAI</h1>

    <h2><a href="index.php">Beranda</a> / <a href="tambah.php">Tambah Data</a></h2>

    <h3>Edit Data Pegawai</h3>

    <?php

    include('koneksi.php');

    $id = $\_GET['id'];

    $show = mysql\_query("SELECT \* FROM pegawai WHERE pegawai\_id='$id'");

    if(mysql\_num\_rows($show) == 0){

        echo '<script>window.history.back()</script>';

    }else{

        $data = mysql\_fetch\_assoc($show);

    }

    ?>

    <form action="edit-proses.php" method="post">

        <input type="hidden" name="id" value="<?php echo $id; ?>">

        <table cellpadding="3" cellspacing="0">

            <tr>

                <td>Nama Lengkap</td>

                <td>:</td>

                <td><input type="text" name="nama" size="30" value="<?php echo $data['pegawai\_nama']; ?>" required></td>

            </tr>

            <tr>

                <td>Bagian</td>

                <td>:</td>

                <td>

                    <select name="bagian" required>

                        <option value="">Pilih Bagian</option>

                        <option value="SUB1" <?php if($data['pegawai\_bagian'] == 'SUB1'){ echo 'selected'; } ?>>SUB1</option>

                        <option value="SUB2" <?php if($data['pegawai\_bagian'] == 'SUB2'){ echo 'selected'; } ?>>SUB2</option>

                        <option value="SUB3" <?php if($data['pegawai\_bagian'] == 'SUB3'){ echo 'selected'; } ?>>SUB3</option>

                    </select>

                </td>

            </tr>

            <tr>

                <td>Gajih</td>

                <td>:</td>

                <td>

                    <select name="gajih" required>

                        <option value="">Pilih Gajih</option>

                        <option value="1000000" <?php if($data['pegawai\_bagian'] == '1000000'){ echo 'selected'; } ?>>1000000</option>

                        <option value="2000000" <?php if($data['pegawai\_bagian'] == '2000000'){ echo 'selected'; } ?>>2000000</option>

                        <option value="3000000" <?php if($data['pegawai\_bagian'] == '3000000'){ echo 'selected'; } ?>>3000000</option>

                    </select>

                </td>

            </tr>

            <tr>

                <td>&nbsp;</td>

                <td></td>

                <td><input type="submit" name="simpan" value="Simpan"></td>

            </tr>

        </table>

    </form>

</body>

</html>

Gambar 2.3 edit.php

Penjelasan : edit.php merupakan tampilan yang akan diperlihatkan ketika ingin mengedit data, proses mengambil data ke database untuk ditampilkan di form edit berdasarkan pegawai\_id yang didapatkan dari GET id -> edit.php?id=pegawai\_id, jika tidak ada data yg sesuai maka akan langsung di arahkan ke halaman depan atau beranda -> index.php, melakukan query ke database dg SELECT table pegawai dengan kondisi WHERE pegawai\_id = '$id'.

Berikut ini adalah kode program proses untuk edit.

<?php

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

    include('koneksi.php');

    $id         = $\_POST['id'];

    $nama       = $\_POST['nama'];

    $bagian     = $\_POST['bagian'];

    $gajih      = $\_POST['gajih'];

    $update = mysql\_query("UPDATE pegawai SET pegawai\_nama='$nama', pegawai\_bagian='$bagian', pegawai\_gajih='$gajih' WHERE pegawai\_id='$id'") or die(mysql\_error());

    if($update){

        echo 'Data berhasil di simpan! ';

        echo '<a href="edit.php?id='.$id.'">Kembali</a>';

    }else{

        echo 'Gagal menyimpan data! ';

        echo '<a href="edit.php?id='.$id.'">Kembali</a>';

    }

}else{

    echo '<script>window.history.back()</script>';

}

?>

Gambar 2.4 prosesedit.php

Penjelasan : Pada bagian ini terdapat beberapa fungsi yaitu include untuk memasukan file koneksi ke database, $id untuk membuat variable $id dan datanya dari inputan hidden id, UPDATE, SET, WHERE, or, die untuk melakukan query dengan perintah UPDATE untuk update data ke database dengan kondisi WHERE siswa\_id='$id' <- diambil dari inputan hidden id, if dan else untuk query update sukses.

Berikut ini adalah kode program untuk menambahkan data

<!DOCTYPE html>

<html>

<head>

    <title>TUGAS PEMROGRAMAN WEB & MOBILE I</title>

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

</head>

<body>

    <h1>DATA PEGAWAI</h1>

    <h2><a href="index.php">Beranda</a> / <a href="tambah.php">Tambah Data</a></h2>

    <form action="tambah-proses.php" method="post">

        <table cellpadding="3" cellspacing="0">

            <tr>

                <td>Nama Lengkap</td>

                <td>:</td>

                <td><input type="text" name="nama" size="30" required></td>

            </tr>

            <tr>

                <td>Bagian</td>

                <td>:</td>

                <td>

                    <select name="bagian" required>

                        <option value="">Pilih Bagian</option>

                        <option value="SUB1">SUB1</option>

                        <option value="SUB2">SUB2</option>

                        <option value="SUB3">SUB3</option>

                    </select>

                </td>

            </tr>

            <tr>

                <td>Gajih</td>

                <td>:</td>

                <td>

                    <select name="gajih" required>

                        <option value="">Pilih Gajih</option>

                        <option value="1000000">1000000</option>

                        <option value="2000000">2000000</option>

                        <option value="3000000">3000000</option>

                    </select>

                </td>

            </tr>

            <tr>

                <td>&nbsp;</td>

                <td></td>

                <td><input type="submit" name="tambah" value="Tambah"></td>

            </tr>

        </table>

    </form>

</body>

</html>

Gambar 2.5 Tambah.php

Penjelasan : . Tambah.php merupakan tampilan program ketika di akses, sama seperti index.php dan edit.php sebelumnya.

Bsrikut ini adalah kode program proses penambahan data.

<?php

if(isset($\_POST[‘tambah’])){

    include(‘koneksi.php’);

    $nama       = $\_POST[‘nama’];

    $bagian     = $\_POST[‘bagian’];

    $gajih      = $\_POST[‘gajih’];

    $input = mysql\_query(“INSERT INTO pegawai VALUES(NULL, ‘$nama’, ‘$bagian’, ‘$gajih’)”) or die(mysql\_error());

    if($input){

        echo ‘Data berhasil di tambahkan! ‘;

        echo ‘<a href=”tambah.php”>Kembali</a>’;

    }else{

        echo ‘Gagal menambahkan data! ‘;

        echo ‘<a href=”tambah.php”>Kembali</a>’;

    }

}else{

    echo ‘<script>window.history.back()</script>’;

}

?>

Gambar 2.6 prosestambah

Penjelasan : Koneksi.php merupakan tempat dimana query dapat mengakses database, host, user, name yang ingin dihubungkan ke program.

Bagian terakhir adalah kode program untuk menghapus data.

<?php

if(isset($\_GET['id'])){

    include('koneksi.php');

    $id = $\_GET['id'];

    $cek = mysql\_query("SELECT pegawai\_id FROM pegawai WHERE pegawai\_id='$id'") or die(mysql\_error());

    if(mysql\_num\_rows($cek) == 0){

        echo '<script>window.history.back()</script>';

    }else{

        $del = mysql\_query("DELETE FROM pegawai WHERE pegawai\_id='$id'");

        if($del){

            echo 'Data pegawai berhasil di hapus! ';

            echo '<a href="index.php">Kembali</a>';

        }else{

            echo 'Gagal menghapus data! ';

            echo '<a href="index.php">Kembali</a>';

        }

    }

}else{

    echo '<script>window.history.back()</script>';

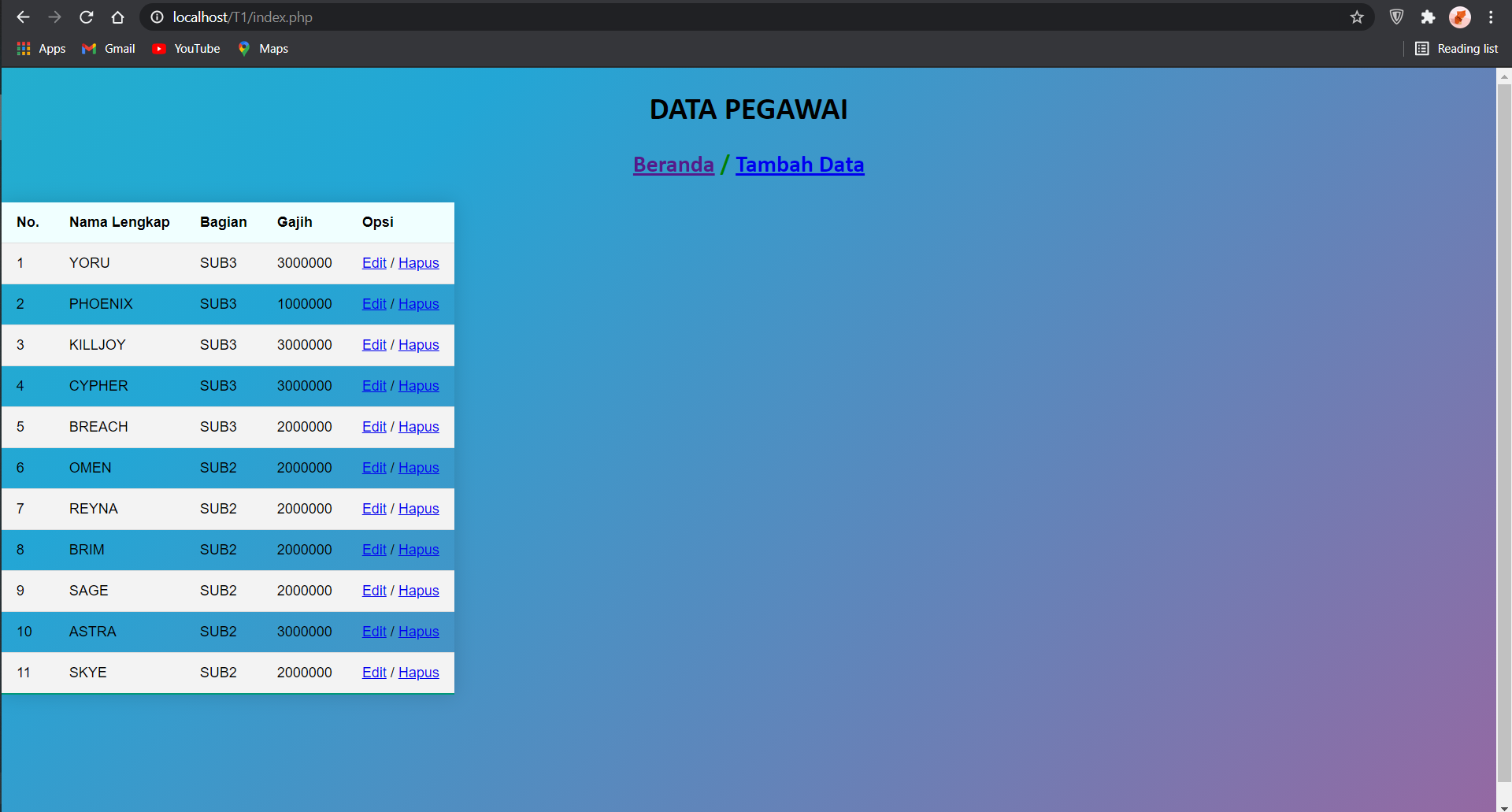
}

?>

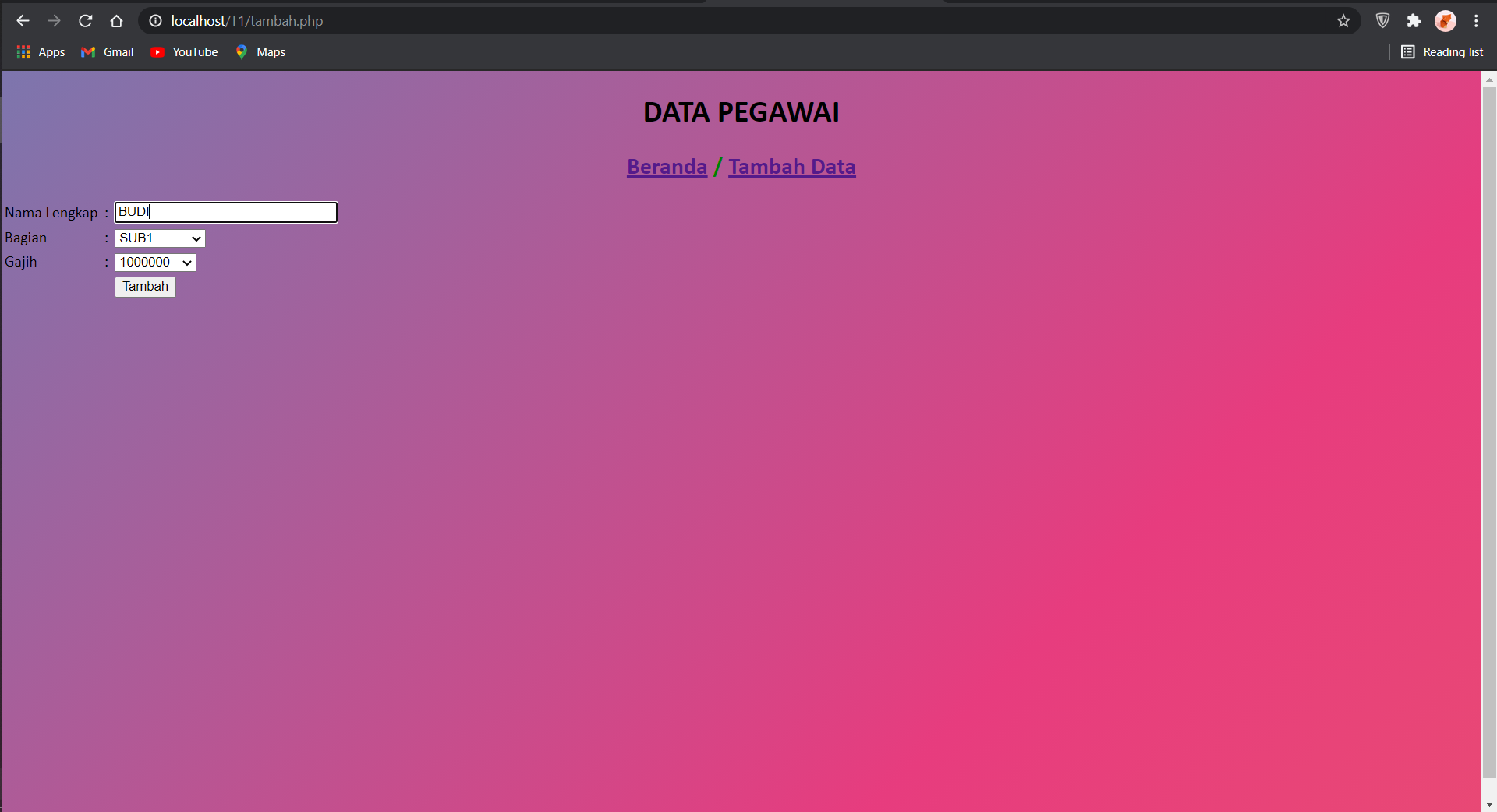
Gambar 2.7 hapus.php

Penjelasan : bagian ini adalah proses menghapus data yang dipilih ,jika data ada di database, maka melakukan query DELETE table pegawai dengan kondisi WHERE pegawai\_id='$id'.

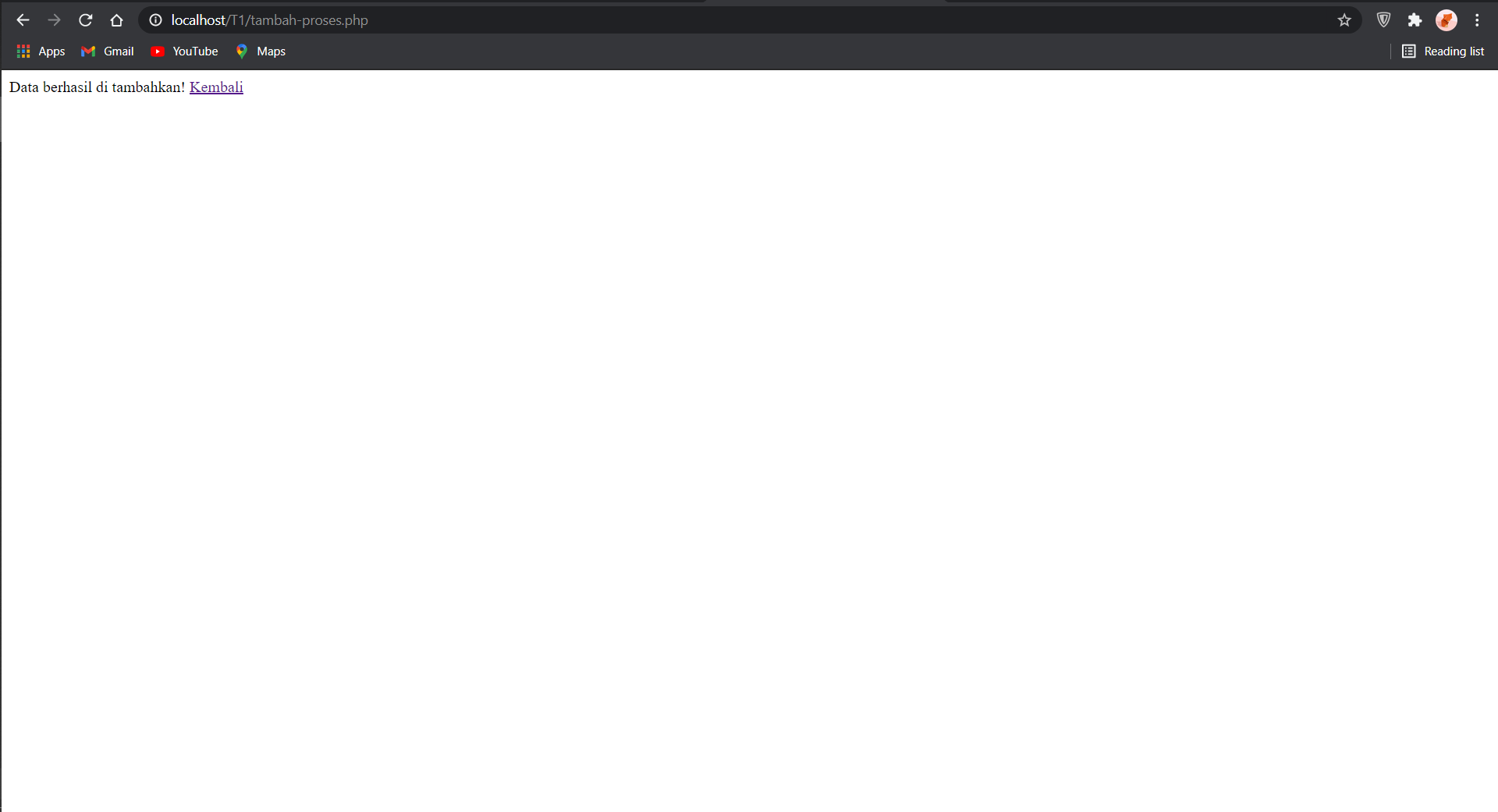
Berikut ini adalah output yang akan ditampilkan di dalam website :



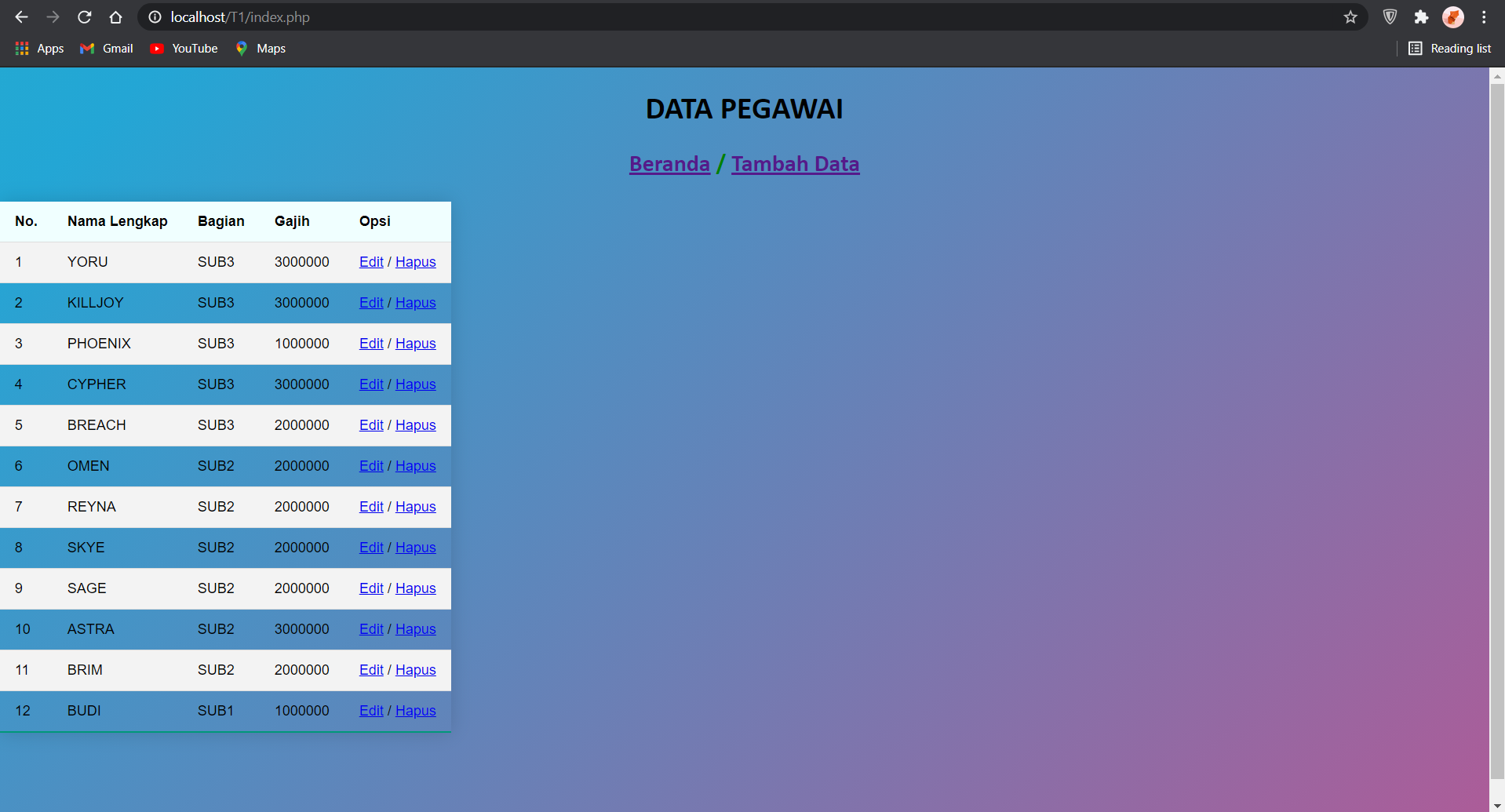
Gambar 2.8 Beranda atau tampilan awal website



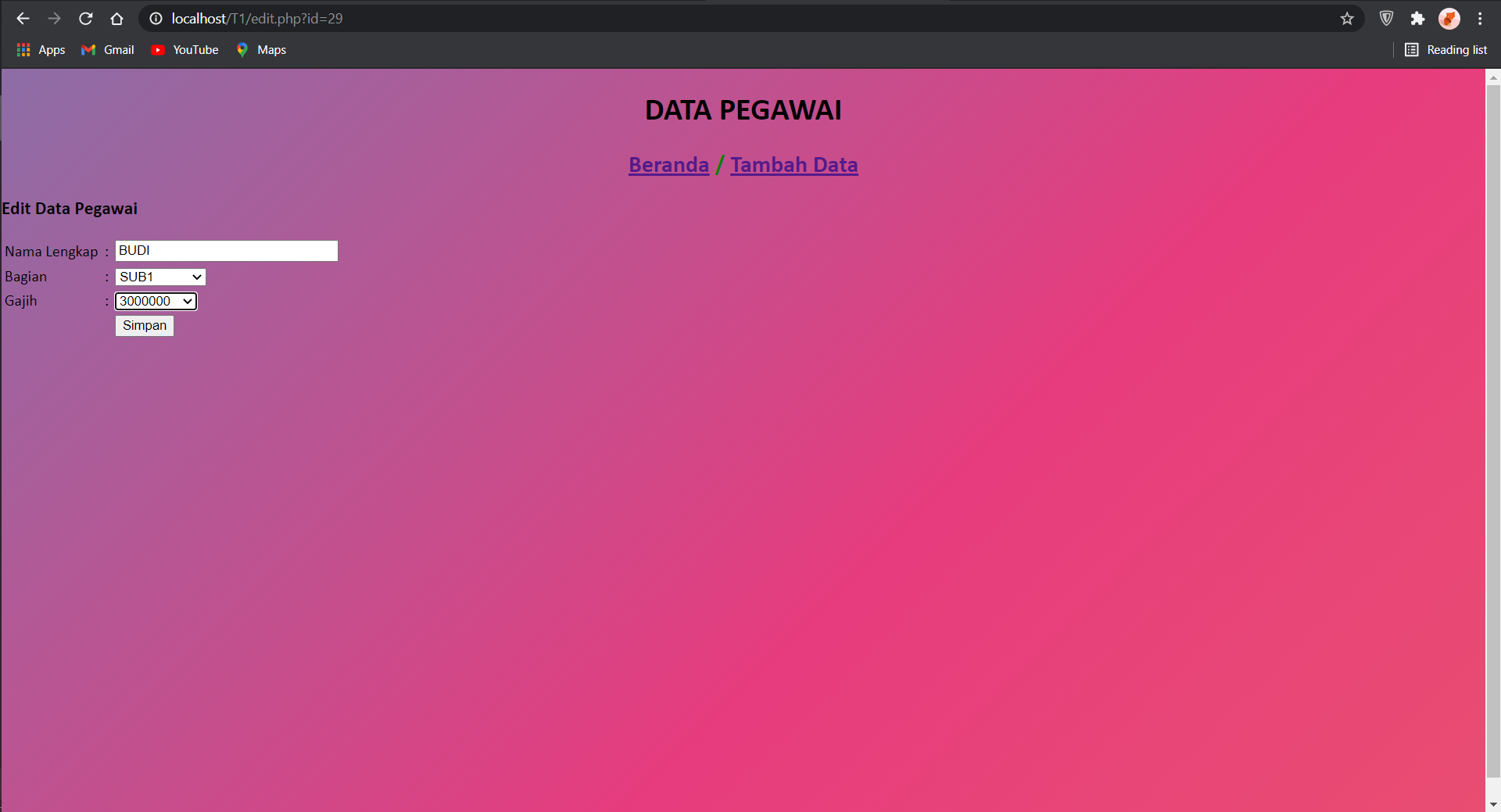
Gambar 2.9 Tambah Data



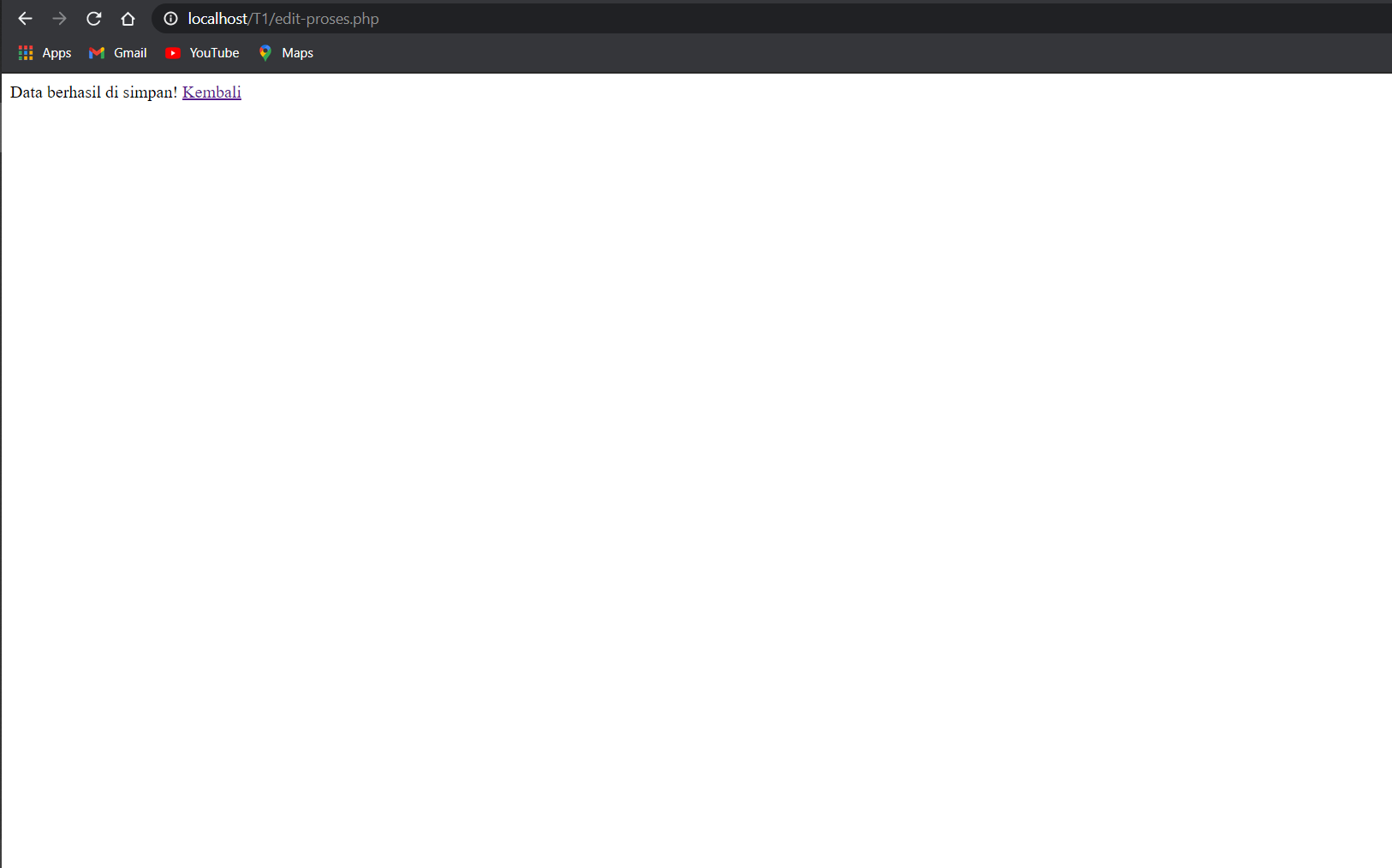
Gambar 2.10 Tambah Data berhasil dilakukan



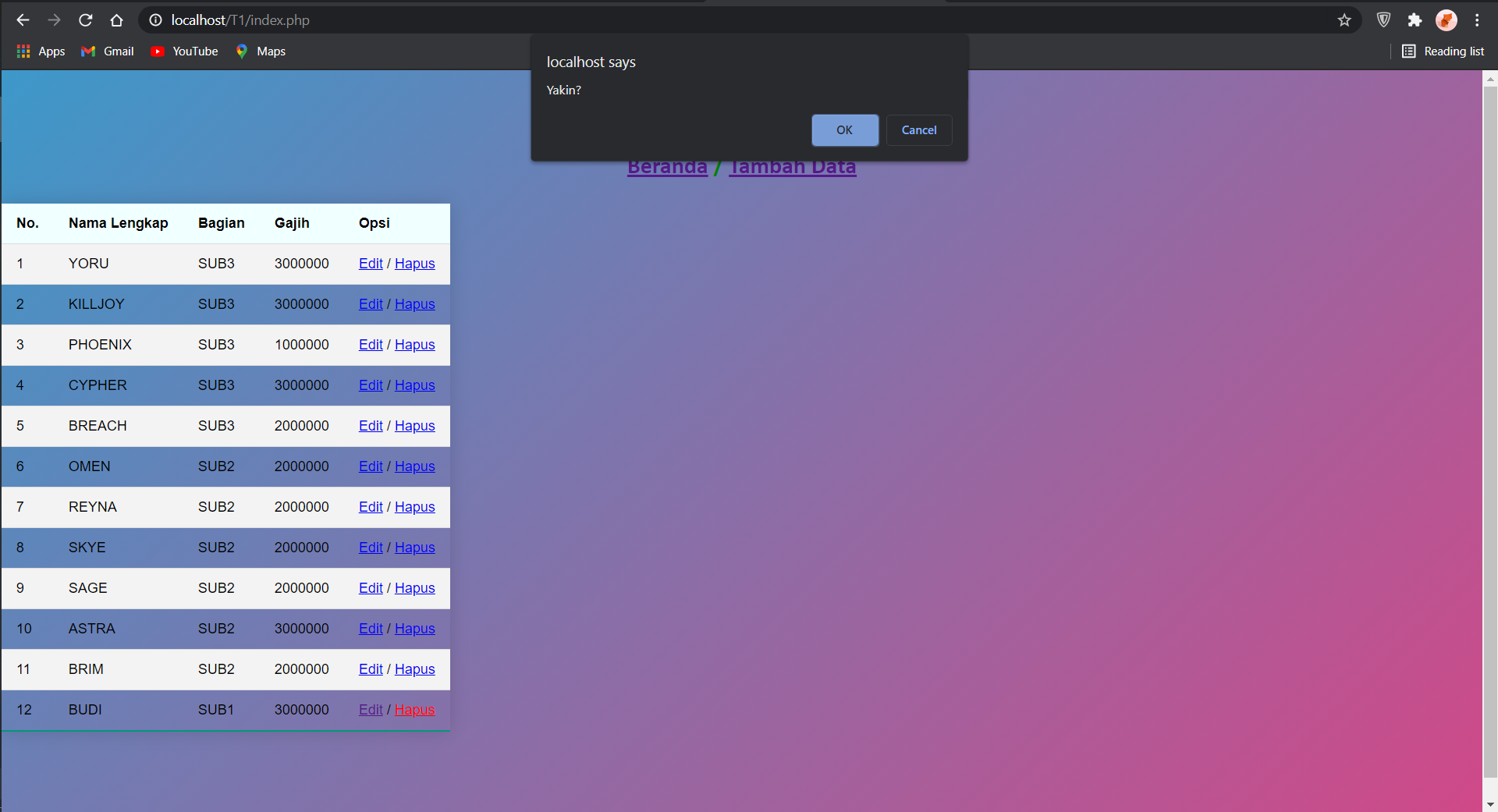
Gambar 2.11 Data Tampil



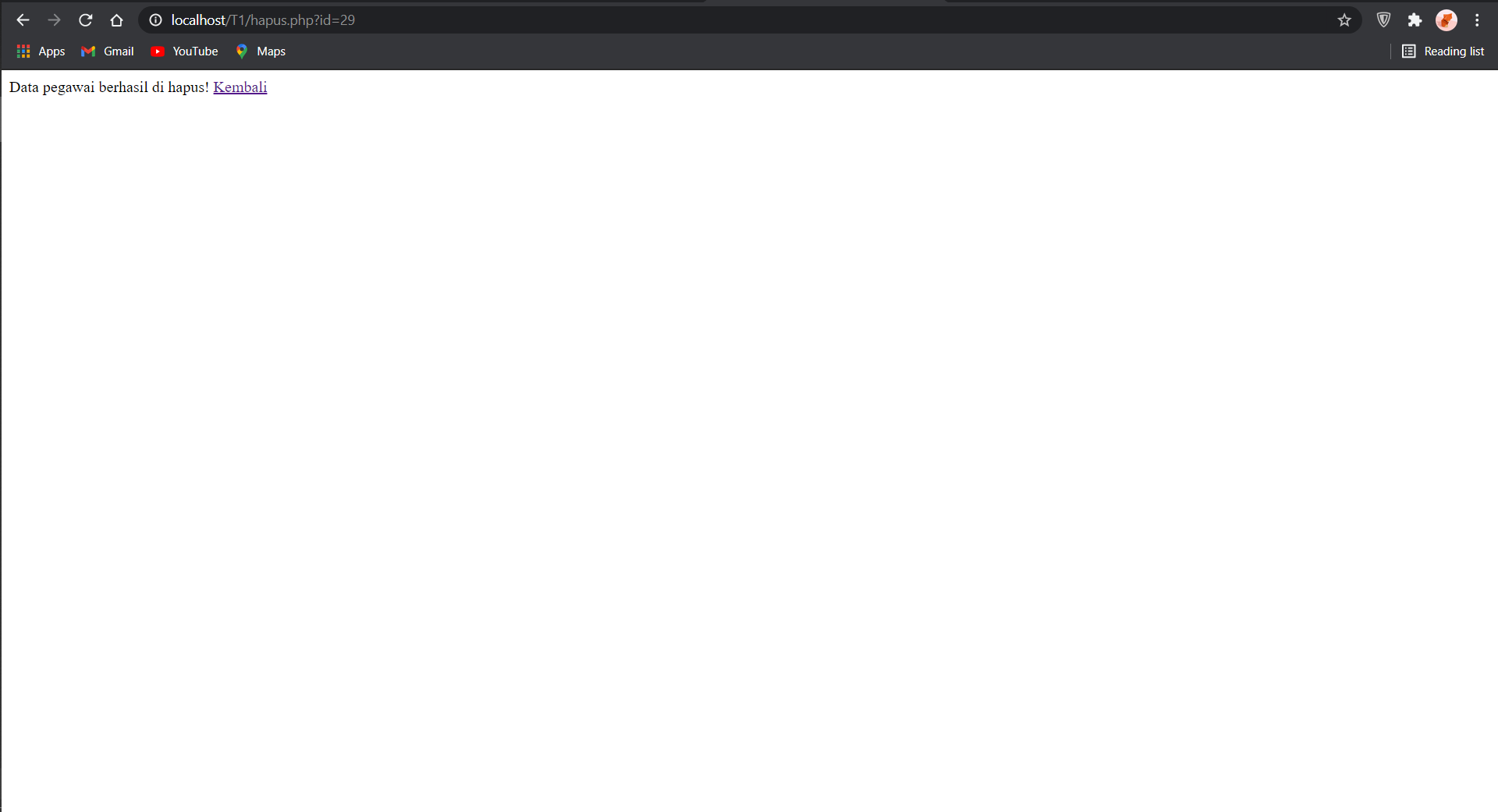
Gambar 2.12 Edit Data



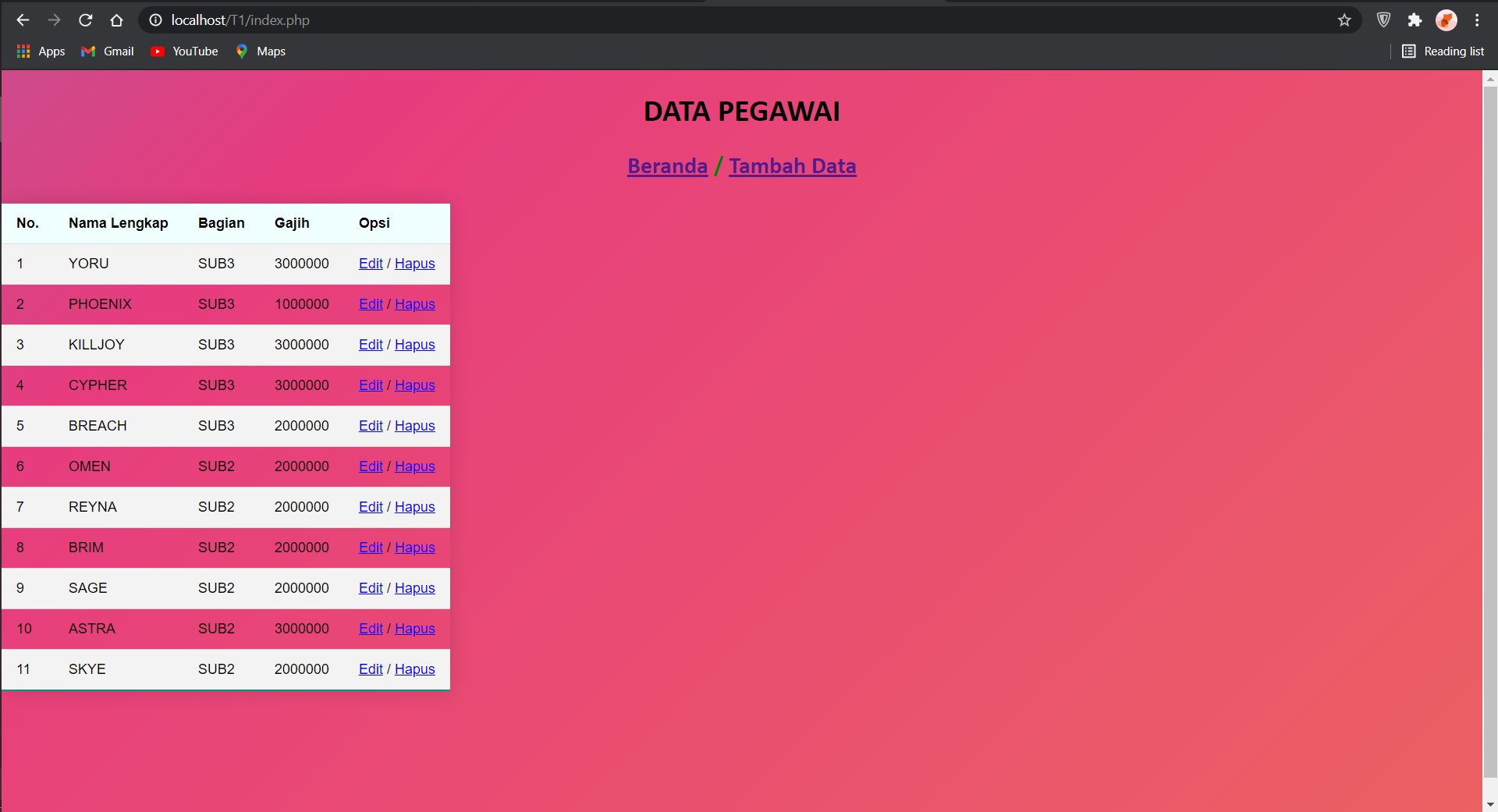
Gambar 2.13 Data Edit berhasil disimpan



Gambar 2.14 Mengapus Data



Gambar 2.15 Data berhasil dihapus



Gambar 2.16 Tampilan beranda data berhasil dihapus

**BAB III**

**KESIMPULAN**

Dapat disimpulkan bahwa Dalam pembuatan Web dengan penggunaan databaseMySQL, ada beberapa perintah yang wajib diketahui seperti mysqli\_connect yang digunakan untuk uji koneksi PHP ke MySQL, mysqli\_select\_db yang digunakan untuk koneksi kedatabase,mysqil\_fetch\_array(), mysqli\_fetch\_assoc(), atau mysqli\_fetch\_row() yang digunakan untuk menampilkan data yang ada didalam tabel, mysqli\_query digunakan untuk memanipulasi data pada database dan mysqli\_num\_rows untuk menghitung jumlah data yang ada didalam database.

Untuk menambahkan suatu database pada halaman web dibutuhkan koneksi untuk dapat meload data yang dimasukan sehingga dapat ditampilakan pada halaman web. Dengan interaksi database yang diterapkan pada pembuatan sebuah halaman web, dapat menerapkan banyak fungsi, seperti menambahkan data, menginputkan, dan menghapus. Tentunya dengan menggunakan berbagai metode dan funsi yang diterapkan di dalam file php dam html.

**DAFTAR PUSTAKA**

Asisten praktikum. 2021. *“Modul praktikum pemrograman web dan mobile I universitas palangka raya fakultas teknik jurusan teknik informatika”*

Docplayer. *“laporan praktikum database mysql”.*

<https://docplayer.info/108708770-Laporan-praktikum-6-database->mysql.html.