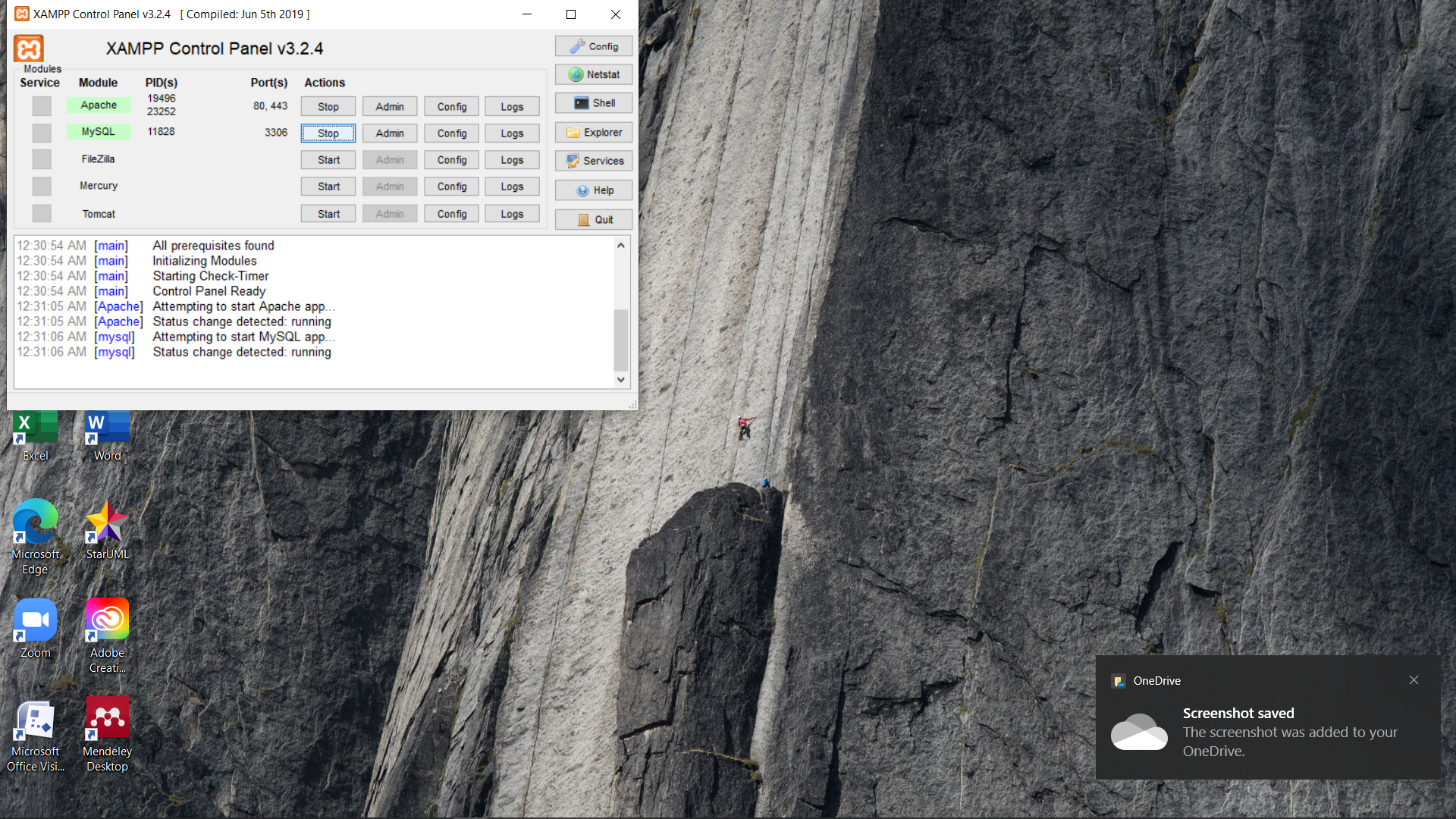
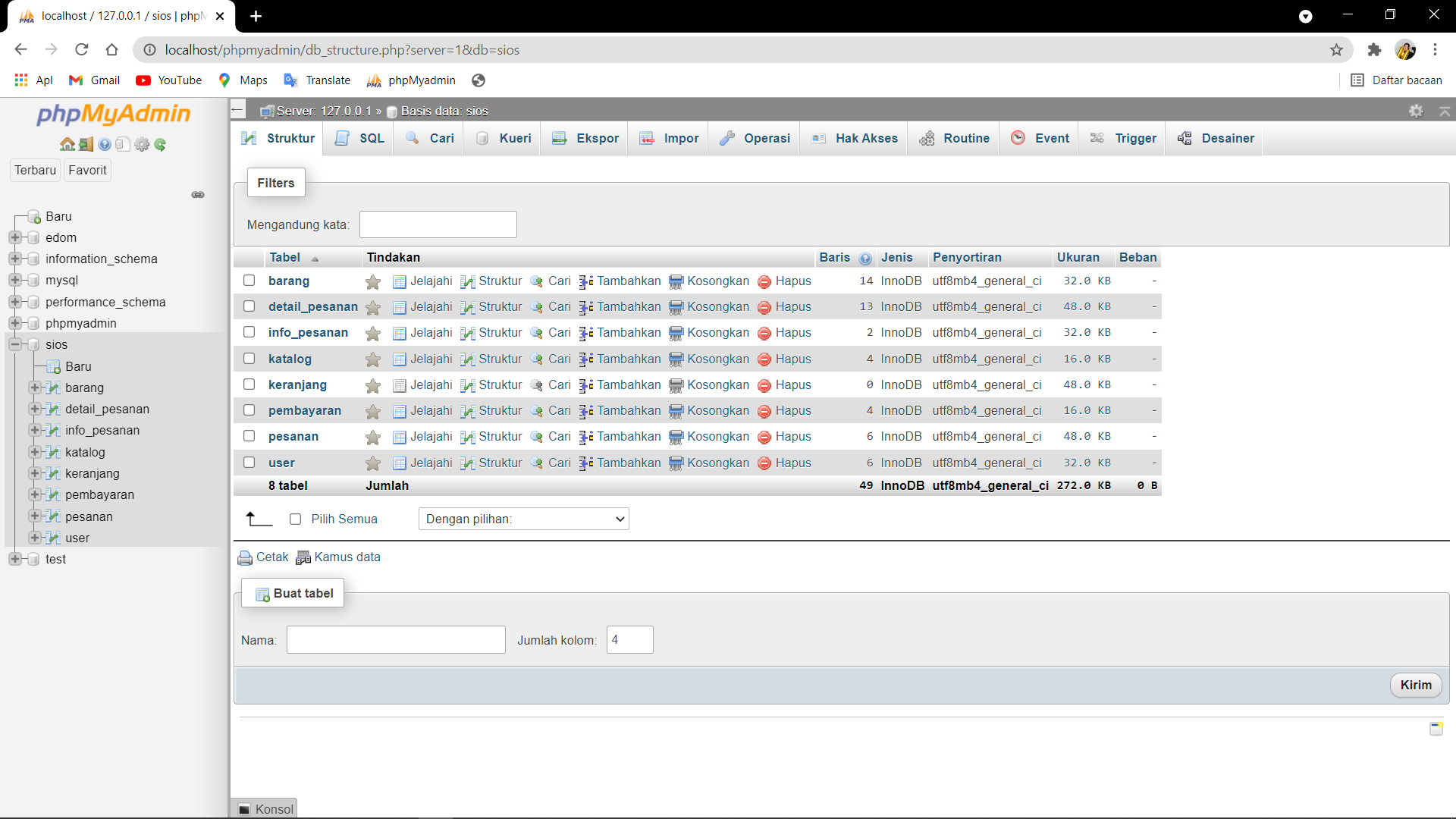
**STEP BY STEP**

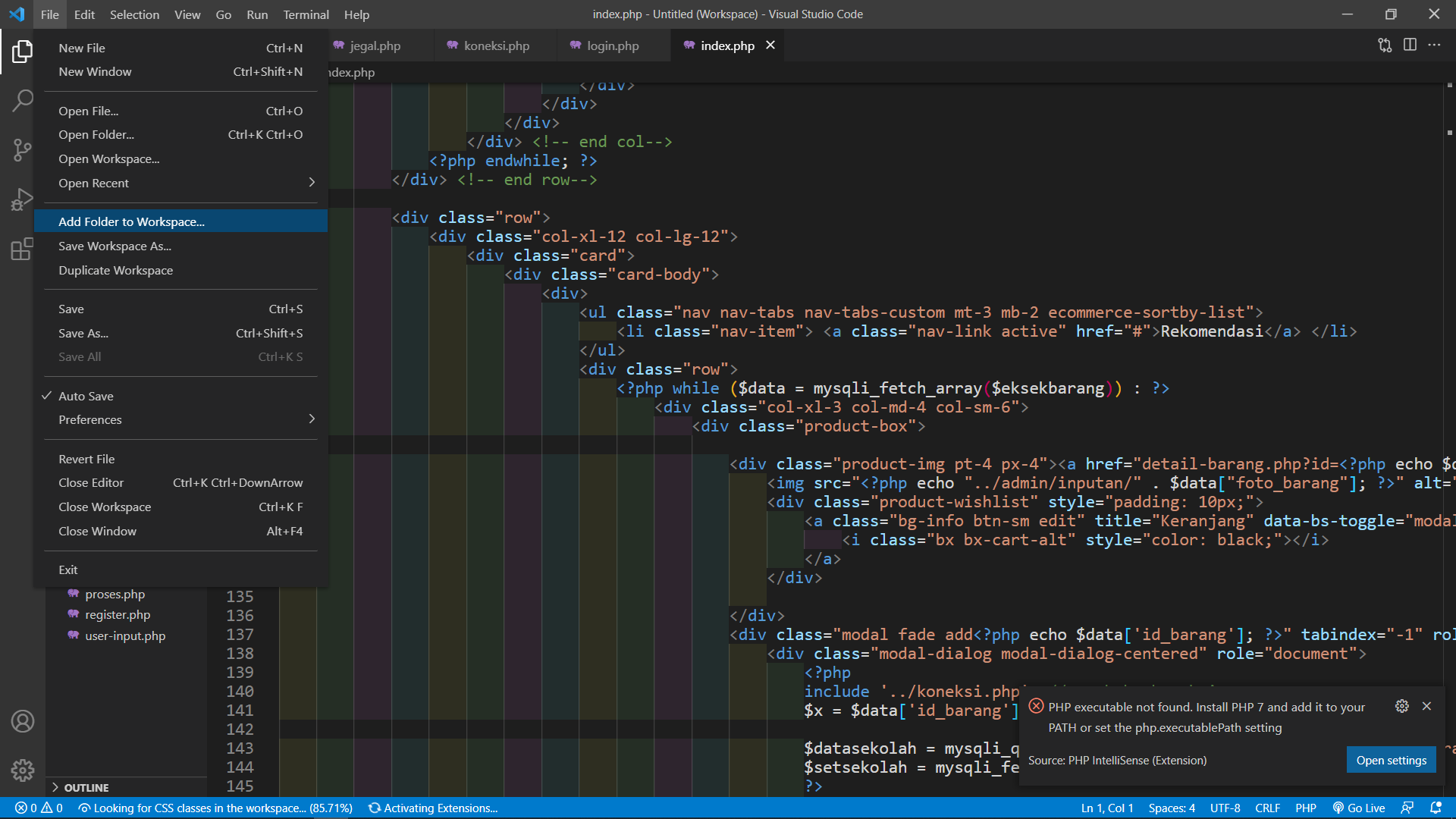
1. Install Xampp
2. Jalankan Xampp, nyalakan Apache dan MySQL



1. Kemudian masuk ke chrome dan buka <http://localhost/phpmyadmin/>



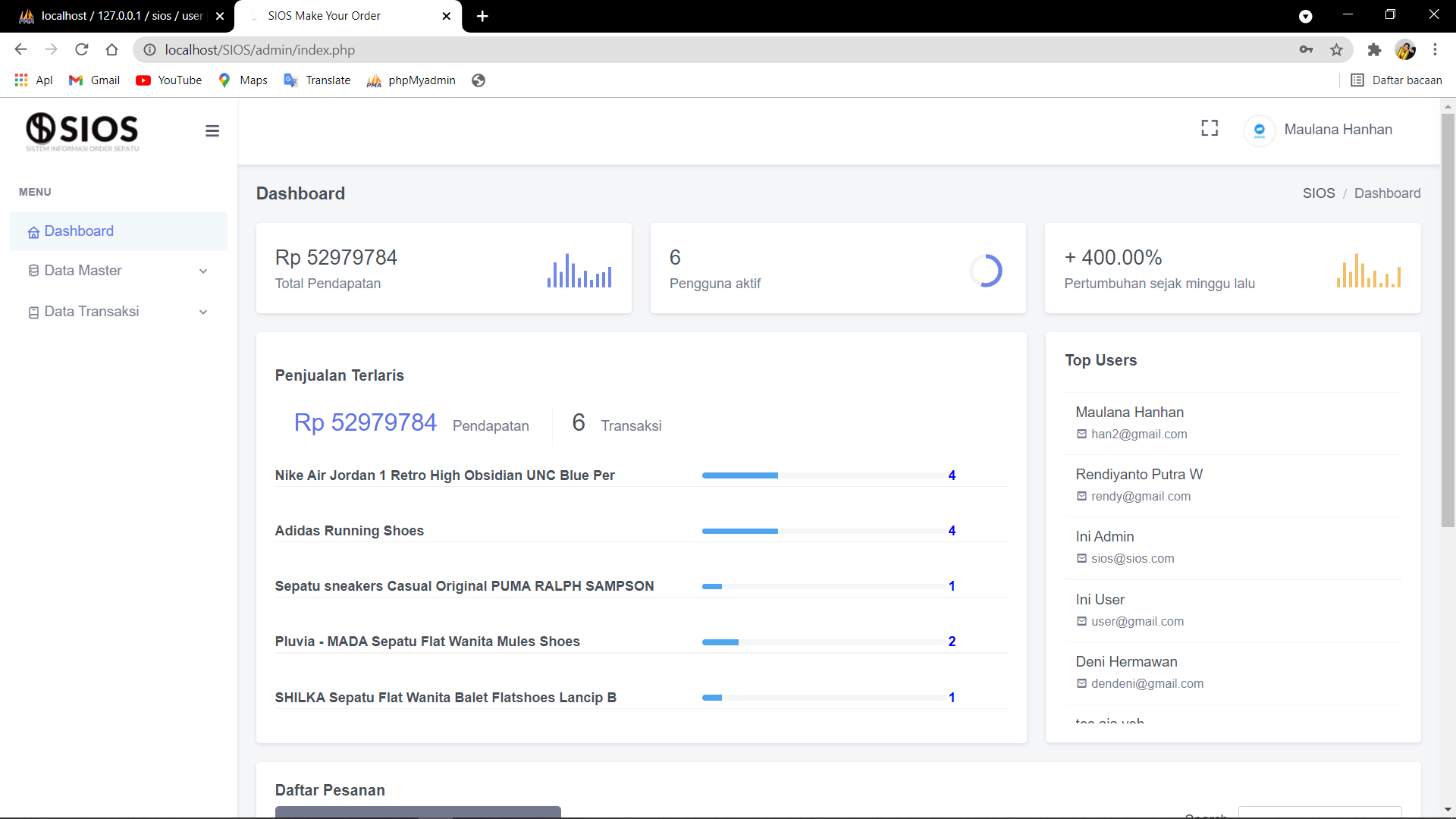
1. Setelah membuka php myadmin buat database baru dengan nama sios. Kemudian disini membuat tabel atau jika sudah memiliki database tinggal melakukan Ekspor.
2. Setelah database selesai dibuat, selanjutnya membuka directori C. kemudian masuk pada folder xampp dan masuk lagi pada folder htdocs. Setelah masuk pada folder htdocs disini saya membuat folder bernamakan SIOS untuk menyimpan file program website yang akan dibuat.
3. Selanjutnya install terlebih dahulu Visual Studio Code untuk mebuka program system dan membuat program.
4. Selanjutnya buka visual studio code kemudian pilih menu File dan pilih ada Folder to Workspace. Kemudian piih folder yang berada pada htdocs yang telah disimpan. Setelah dibuka maka file akan tampil pada visual studio code.



1. Setelah folder tampil pada visual studio code. Pada file koneksi digunakan untuk membuat koneksi program dengan database yang telah dibuat. Perintah yang digunakan sebagai berikut

|  |
| --- |
| <?php  $lokasi='localhost';  $db='sios';  $user='root';  $pass='';  $test = mysqli\_connect($lokasi,$user,$pass,$db);  if (!$test) {  echo "gagal nih: " . mysqli\_connect\_error();  exit();  }  /\*echo 'Koneksi berhasil';    $tampil="select \* from sekolah";  $eksekusi=mysqli\_query($test,"$tampil");  echo "<br>";  while($data =mysqli\_fetch\_array($eksekusi)){  echo $data['nama\_sekolah'],"<br>";  }  \*/  ?> |

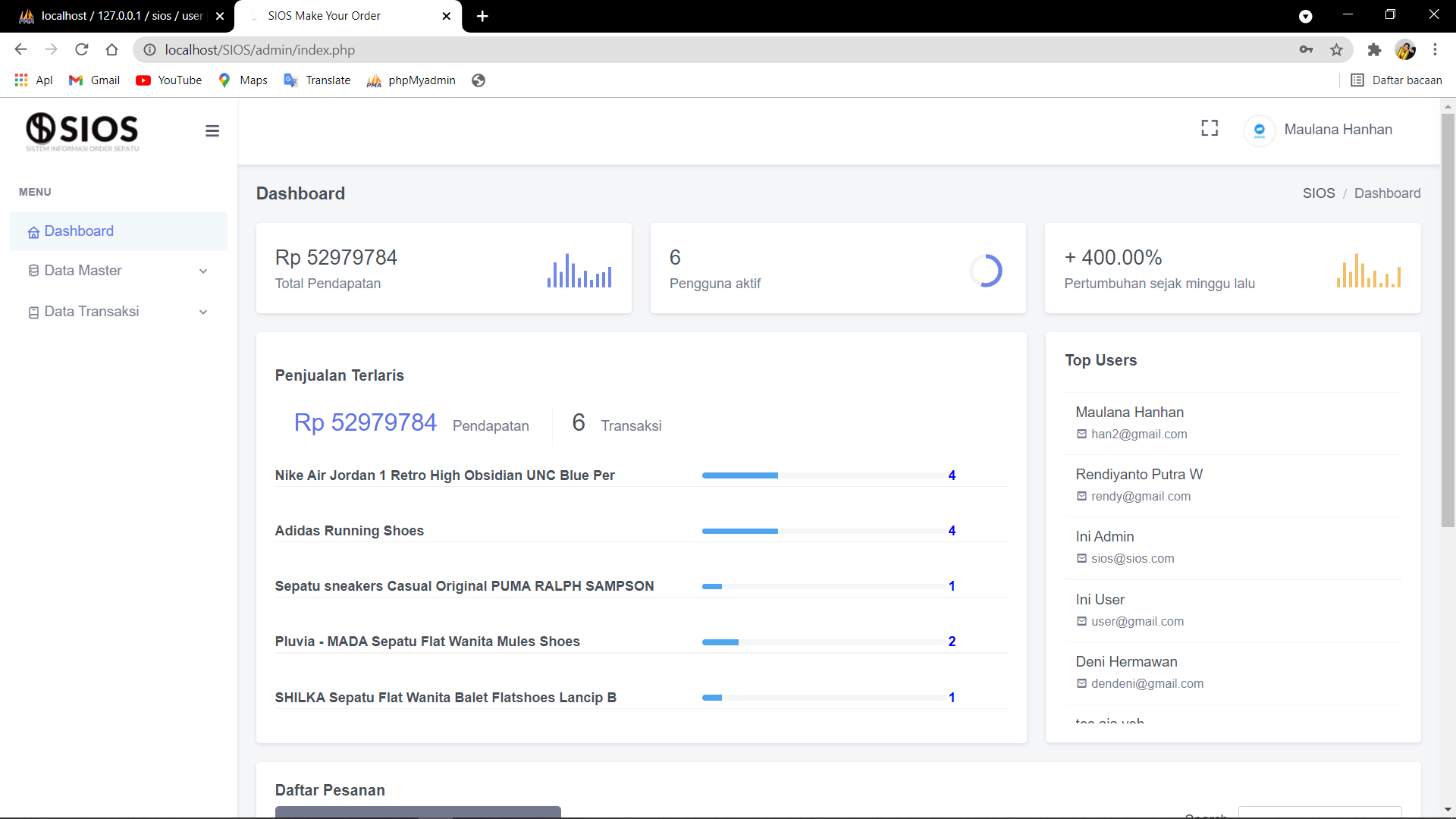
1. Kemudian pada file login.php dan register.php adalah program untuk membuat tampilan login dan administrator pada user serta administrator.
2. Pada file user-input.php digunakan untuk mengecek apakah email dan password yang dimasukan benar atau salah. Penjelasan lebih lanjutnya ada di dalam program.
3. Pada file proses.php digunakan untuk meyeleksi data user pada database. Pejelasan lebih lanjutnya didalam program.
4. Pada file jegal.php digunakan untuk membuat pertahanan pada system jika username yang dimasukan salah.
5. Disini ada folder Bernama admkomponen. Yang dimana berisi header, footer dan sidebar untuk menu administrator.



Pada file header.php adalah program dibawah digunakan untuk menempilkan profil user dan header sesuai dengan level user yang ada.

|  |
| --- |
| <?php  include "../jegal.php";  include "../koneksi.php";  @session\_start();  if($\_SESSION["level"]=='User'){  unset($\_SESSION["username"]);  session\_destroy();  header("location:../login.php");  }  ?> |

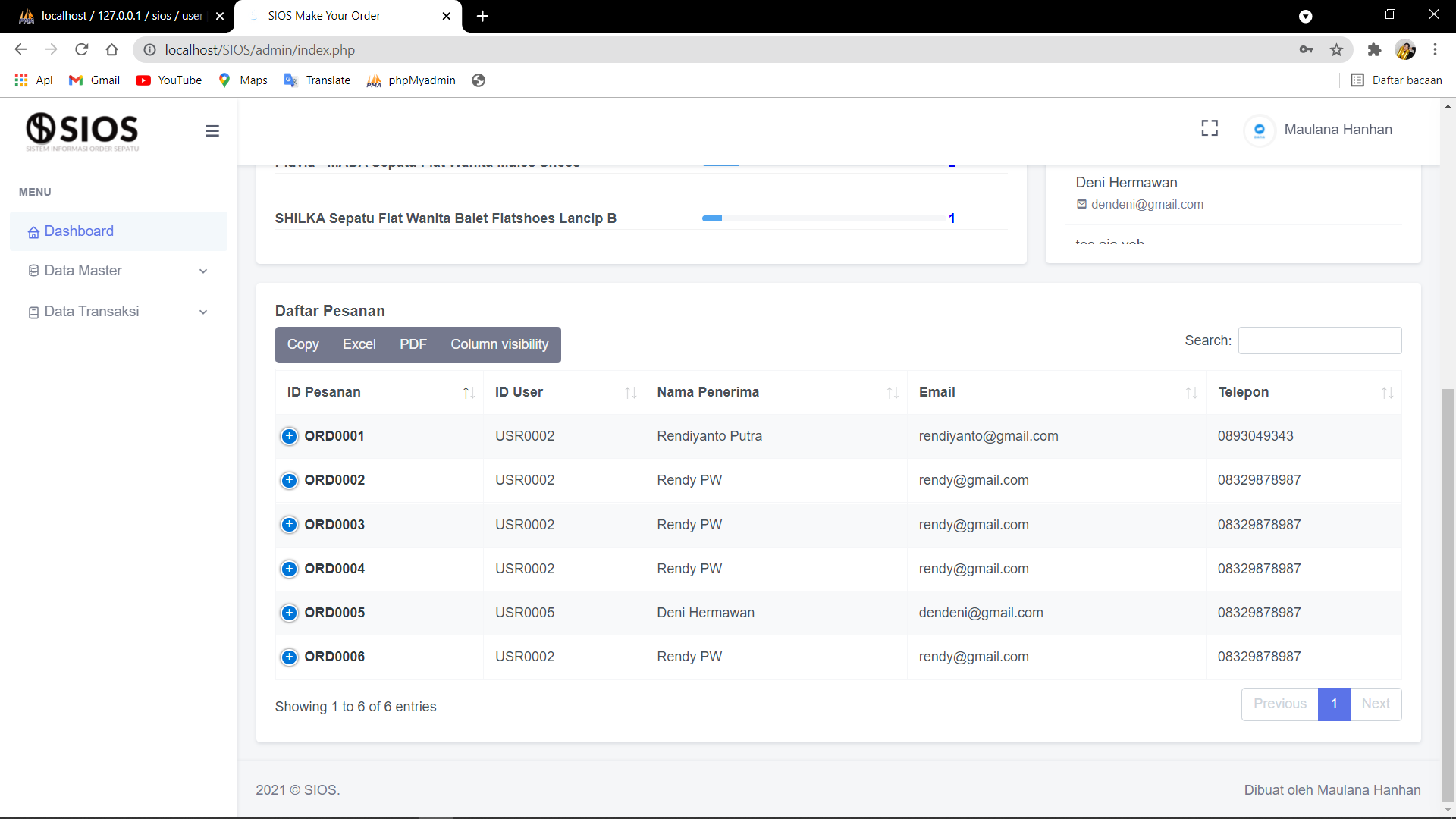
Header



Sidebar

Pada file sidebar.php adalah bagian Sidebar semua dan menu bisa diakses karena pada program telah menghubungkan link menggunakan a href seperti pada program berikut

|  |
| --- |
| <ul class="sub-menu" aria-expanded="true">  <li> <a href="barang.php">Data Barang</a></li>  <li> <a href="katalog.php">Data Katalog</a></li>  <li> <a href="pembayaran.php">Data Pembayaran</a></li>  <li> <a href="pengguna.php">Data Pengguna</a></li>  </ul> |



Footer

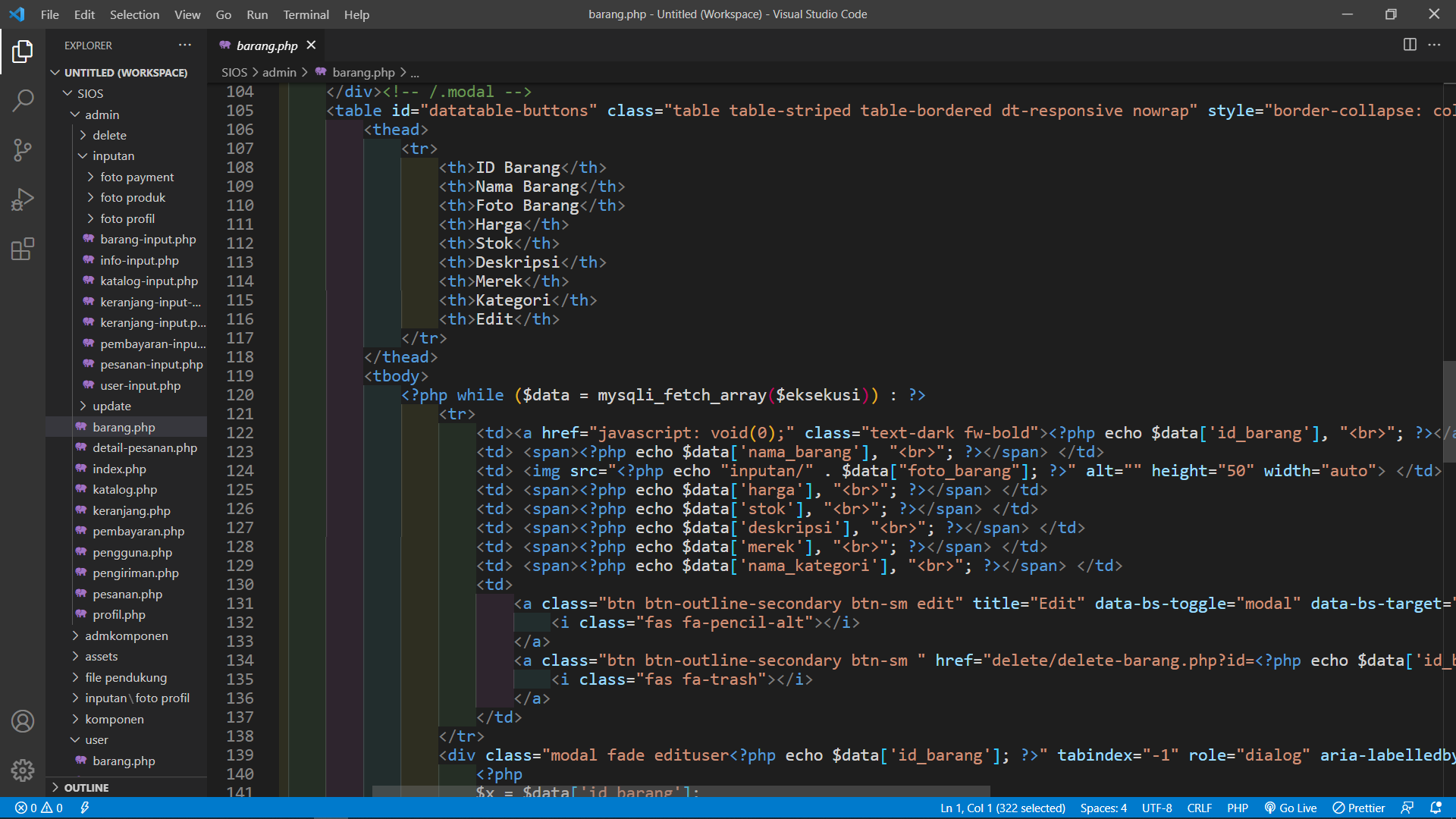
Pada file footer.php adalah program yang digunakan untuk membuat footer yang berisikan nama dan system. Tampilan footer pada user dan admin sama.

1. CRUD DATA ADMIN
2. Menampilkan Tabel

Untuk membuat tabel sesuai dengan database yang ada. Menggunakan perintah select \* from nama database.

|  |
| --- |
| <?php  //include "../jegal.php";  include "../admkomponen/header.php";  include "../admkomponen/sidebar.php";  include "../koneksi.php";  //@session\_start();  $tampil = "select \* from barang";  $eksekusi = mysqli\_query($test, "$tampil");  $bayar = "select \* from katalog";  $eksekbayar = mysqli\_query($test, "$bayar");  ?> |

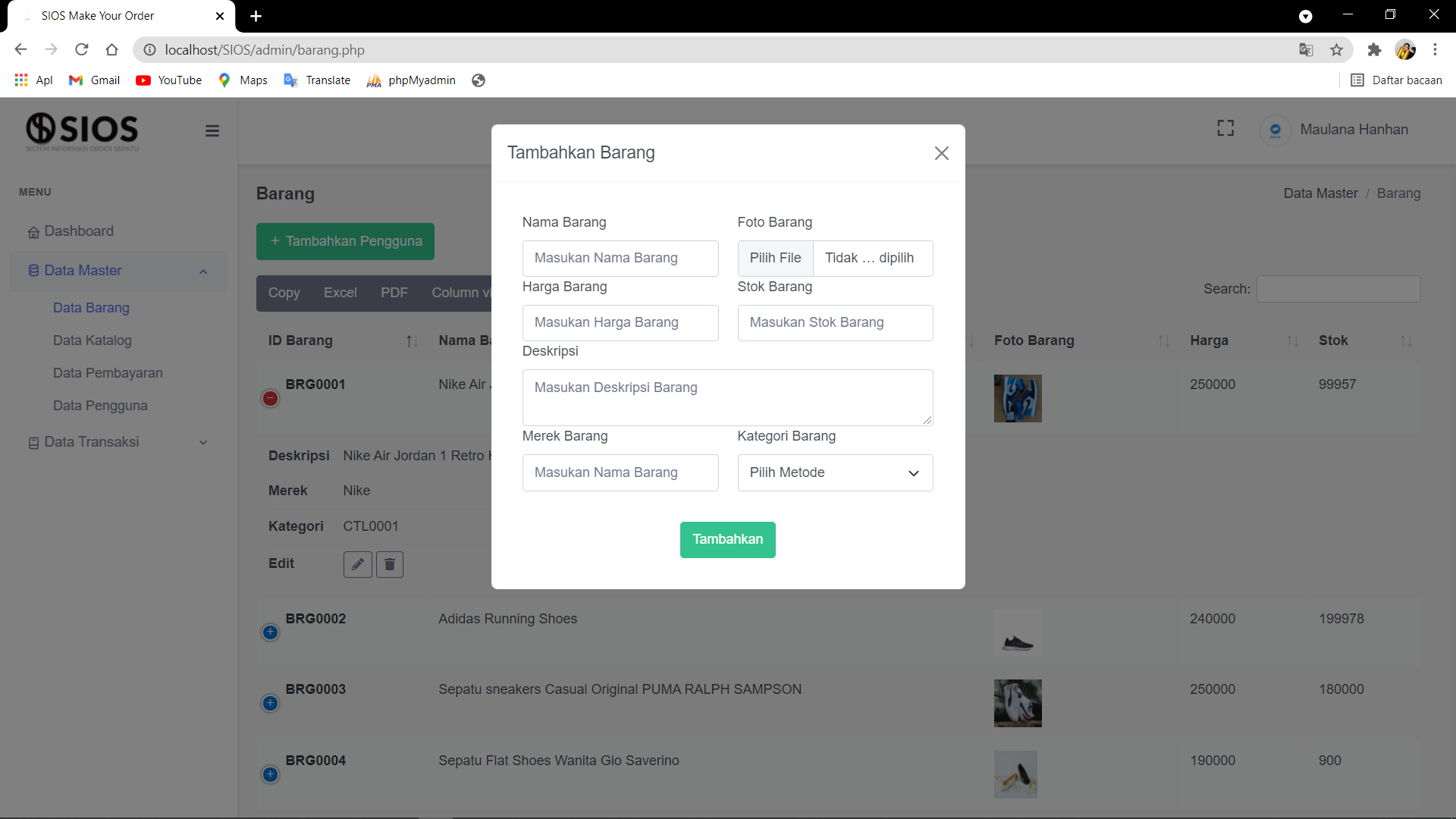
Kemudian seteah membuat perintah untuk menampilkan tabel, buat table dan panggil nama tabel atau data dengan menuliskan perintah echo.



Untuk menampilkan tabel prosesnya sama hanya mengganti nama tabel dan nama field yang digunakan

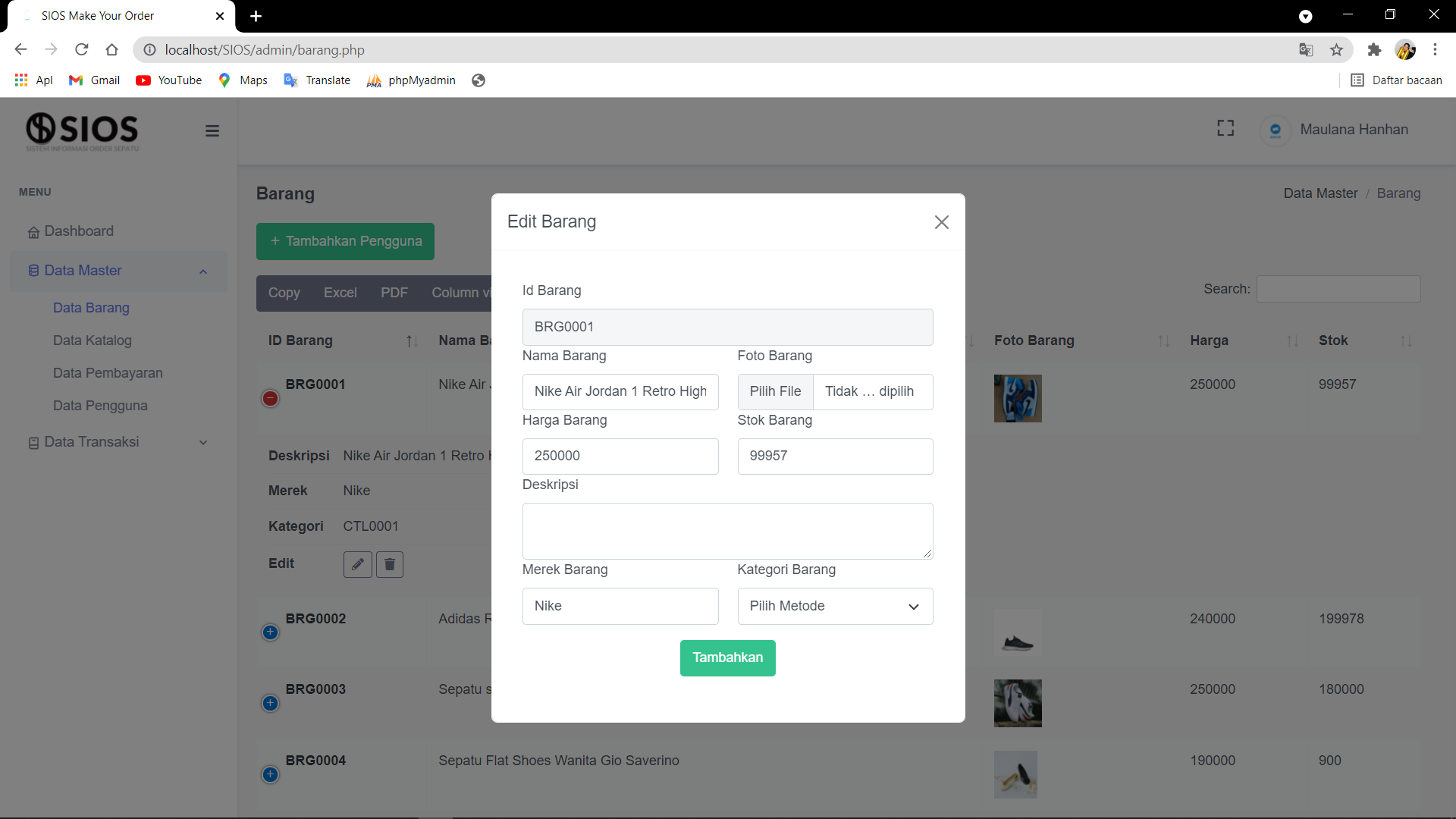
1. Input

Buka program pada folder admin kemudian masuk pada folder Inputan pada file barang-inputan.php sudah dijelaskan didalam programnya.



1. Update

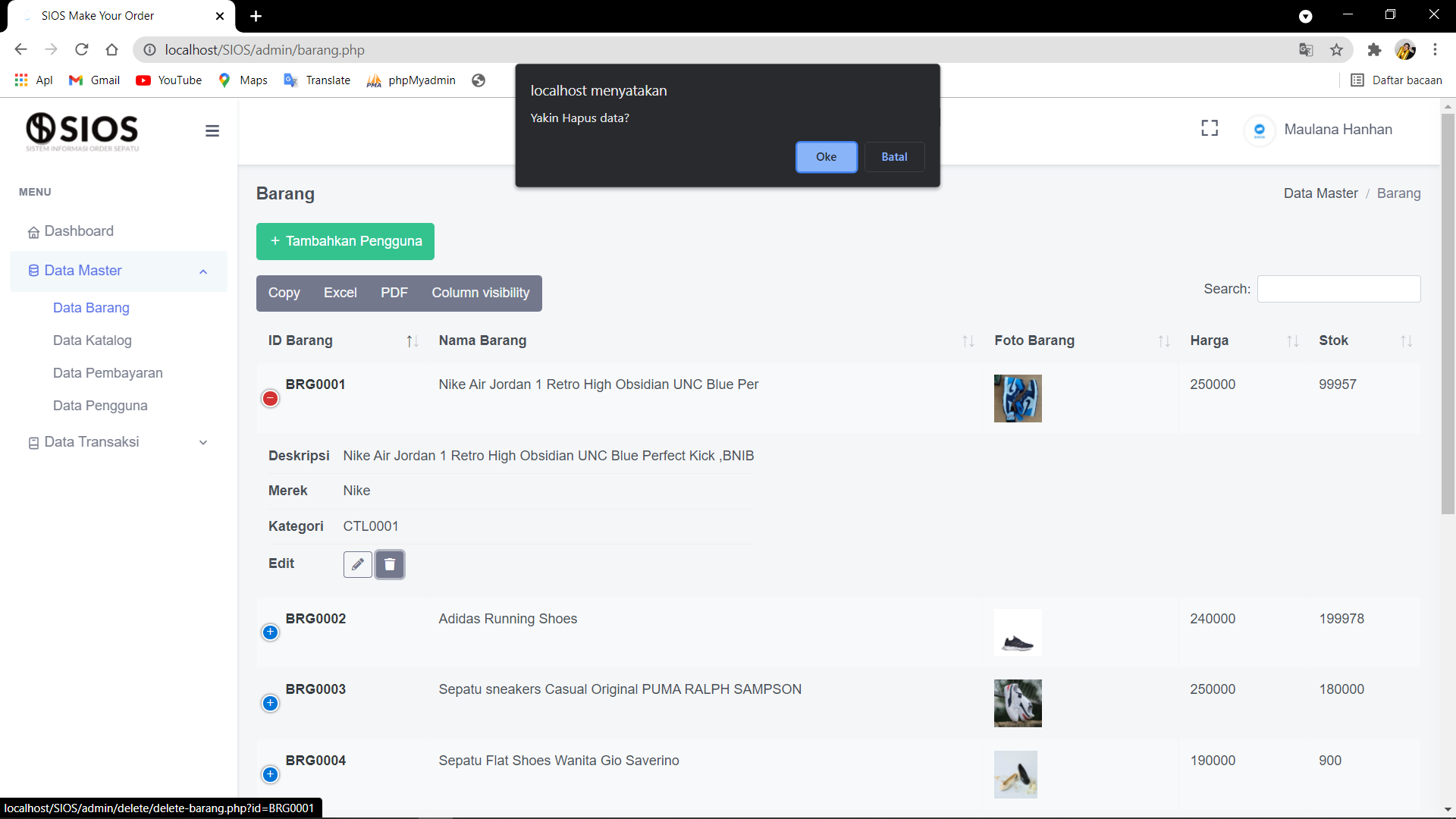
Buka program pada folder admin kemudian masuk pada folder Update pada file update-barang.php sudah dijelaskan didalam programnya. Pada program update-pesanan.php berbeda karena jika mengupdate pesanan dan pesanan dikemas maka akan otomatis memperbarui pada info-pemesanan.php



1. Delete

|  |
| --- |
| <?php  include "../../koneksi.php";  if (isset($\_GET['id'])) {  $x = $\_GET['id'];  $sql = "DELETE FROM barang WHERE id\_barang='$x'";  $query = mysqli\_query($test, $sql);  if ($query) {  echo "<script language=javascript> window.alert('Hapus Data Berhasil'); window.location='../barang.php';</script>";  //header('Location: ../pendaftaran.php');  } else {  die("gagal menghapus...");  }  } else {  die("akses dilarang...");  }  ?> |

Program diatas adalah program untuk proses delete atau menghapus data yang ada. Proses hapus data pada dasarnya sama cukup menggati nama tabel dan Primary key pada tabel dan menampilkan pada halaman yang dituju.

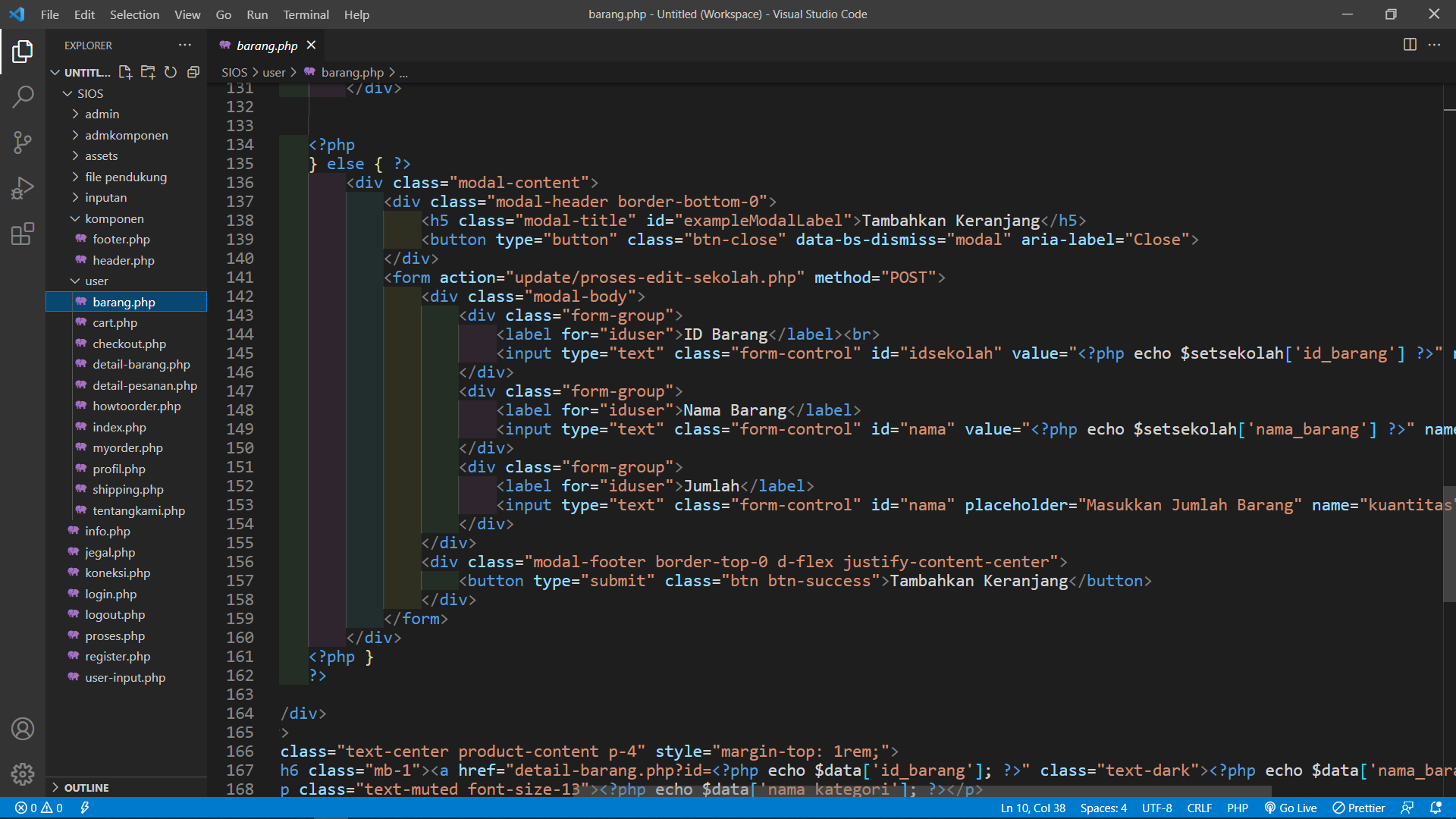


Program pada pembuatan menu admin pada dasarnya sama, proses CRUD data yang dilakukan juga sama, hanya dibedakan berdasarkan tabel pada database yang digunakan.

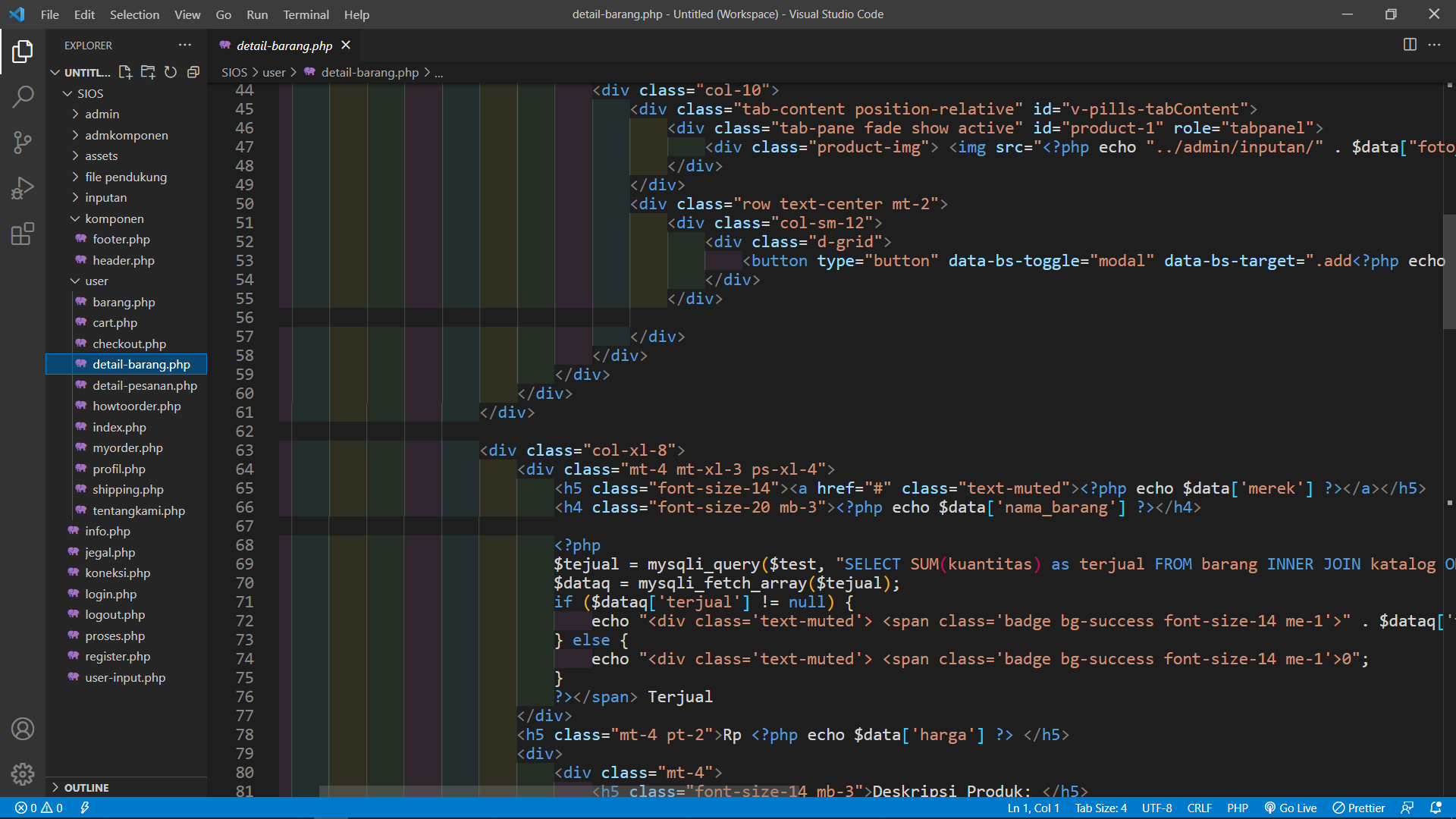
1. Kemudian selanjutnya pembuatan program untuk user. Pada pembuatan menu user masuk pada folder komponen dan terdapat file header dan footer untuk pembuatan tampilan user.
2. Pada file Header.php digunakan untuk tampilan user disini menampilkan data barang dan data katalog. Pada bagian SESSION digunakan untuk menampilkan header sebelum dan sesudah login berdasarkan username. Program lebih lanjutnya bisa dilihat pad file header.php pada folder komponen.
3. Pembuatan footer pada halaman admin dan user tidak jauh berbeda dan programnya sama. Bisa dilihat pada folder komponen kemudian masuk pada file footer.php
4. Pada file index.php disini digunakan untuk membuat tampilan dashboard pada user sebelum melakukan login. Ketika user menekan tombol keranjang, user terlebih dahulu harus melakukan login. Ketika sudah login maka akan muncul header baru dan tampilan dashboard baru untuk user, yaitu penambahan tombol keranjang dibagian header. Program dibawah digunakan untuk memanggil database atau tabel yang akan digunakan, jika sudah dideklarasikan nanti tinggal memanggil field yang akan digunakan atau akan ditampilkan pada halaman web.

|  |
| --- |
| <?php  //include "../jegal.php";  include "../komponen/header.php";  //include "../admkomponen/sidebar.php";  include "../koneksi.php";  //@session\_start();  $tampil = "select \* from barang";  $eksekbarang = mysqli\_query($test, "$tampil");  //$tampill = "select \* from katalog limit 4";  $tampill = "SELECT nama\_kategori,nama\_katalog, COUNT(id\_barang) as jumlah\_barang,foto\_barang  FROM barang JOIN katalog on barang.nama\_kategori=katalog.id\_katalog GROUP BY nama\_kategori limit 4  "; //ini yang bener  $eksekpend = mysqli\_query($test, "$tampill");  $tampilbarang = "select id\_barang,COUNT(\*) as jumlah\_pesanan,ROUND((COUNT(id\_barang)/(SELECT COUNT(\*) FROM detail\_pesanan))\*100,0) AS Prosentase from detail\_pesanan GROUP by id\_barang limit 4";  $eksebarange = mysqli\_query($test, "$tampilbarang");  ?> |

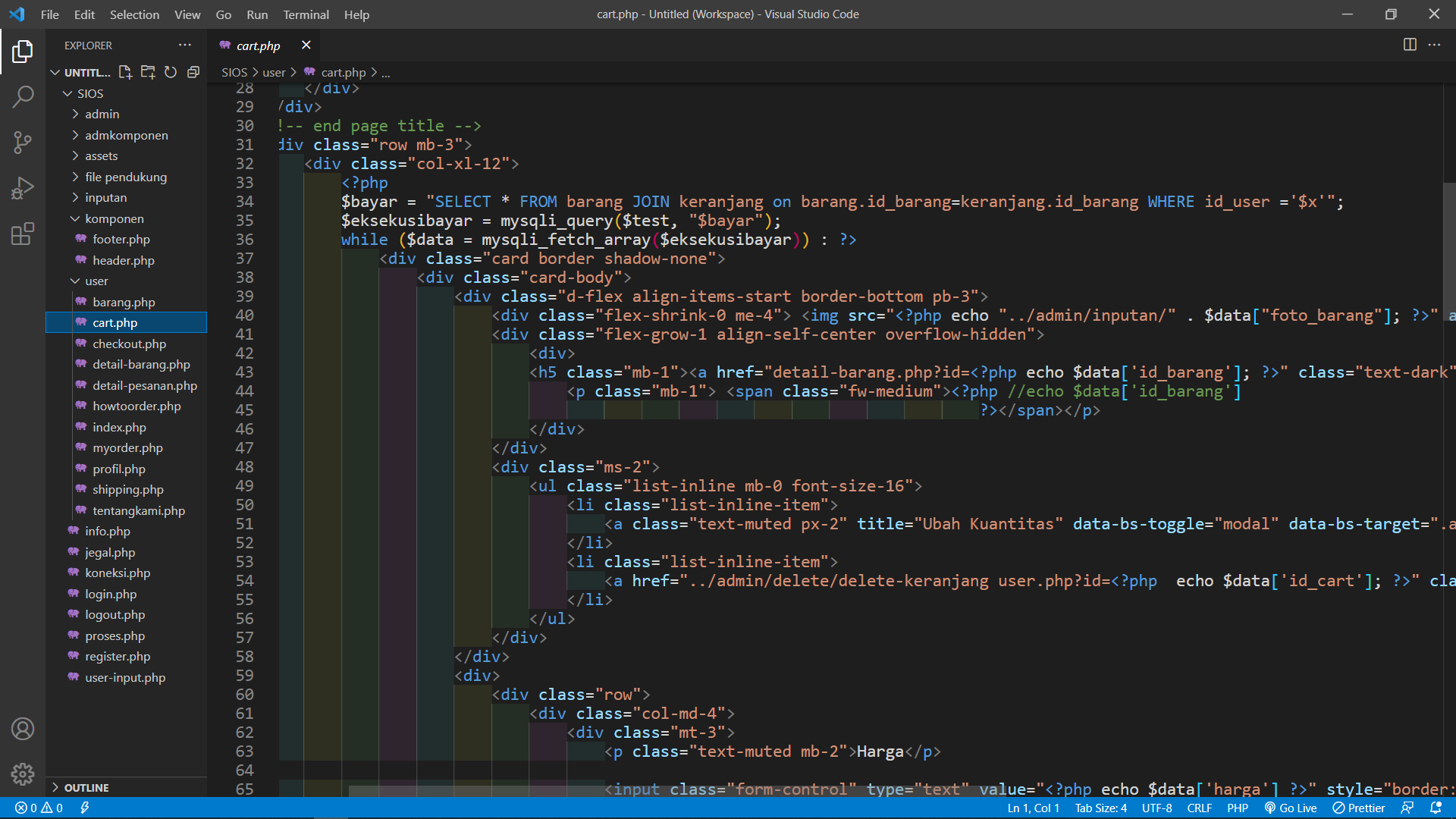
1. Pada file barang.php adalah tampilan jika user telah melakukan login, pemanggilan databasenya sama seperti pada file index.php hanya dibedakan pada bagian program berikut yang telah memanggil field-field yang akan digunakan



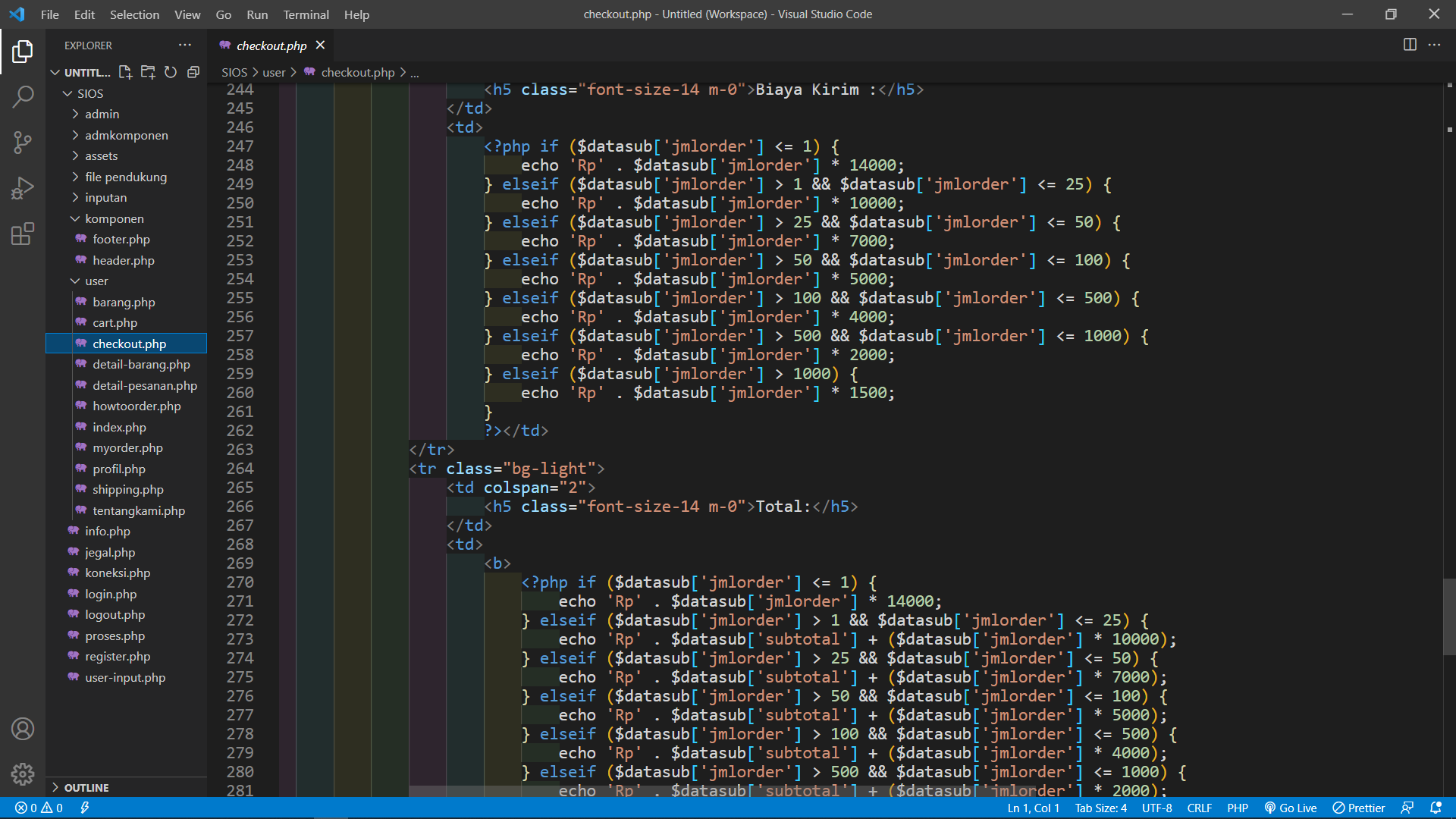
1. Pada file detail-barang.php adalah program untuk menampilkan tampilan detail pada barang. Deklarasikan tabel database yang akan digunakan kemudian panggil menggunakan echo.



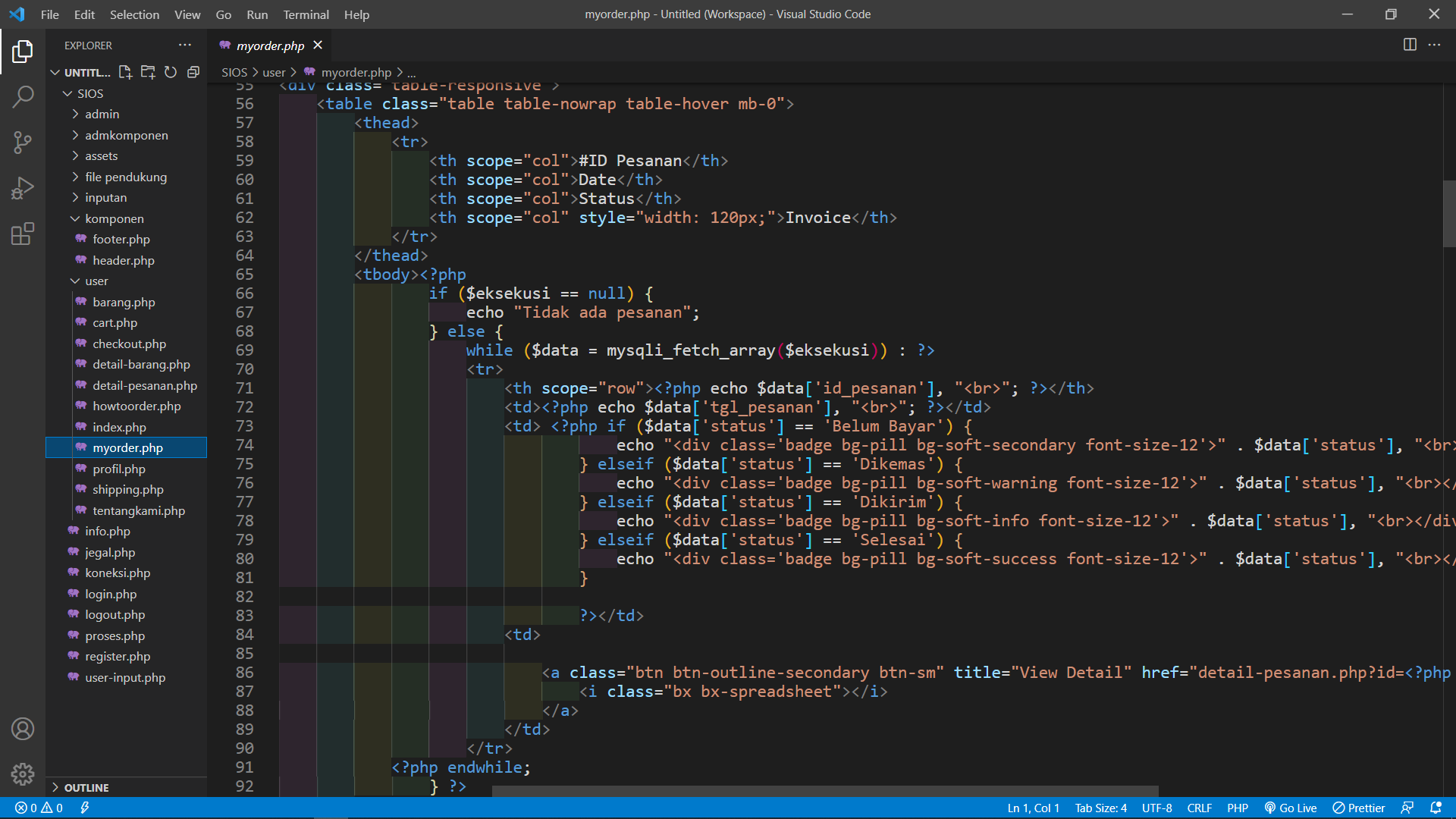
1. Pada file cart.php adalah program untuk tampilan keranjang belanjaan user, setelah melihat barang dan memasukan ke keranjang maka data akan masuk pada keranjang, deklarasi variable dan tabel database yang digunakan terlebih dahulu kemudian masukan data yang akan digunakan dengan perintah echo dan masukan nama field sseperti dibawah ini.



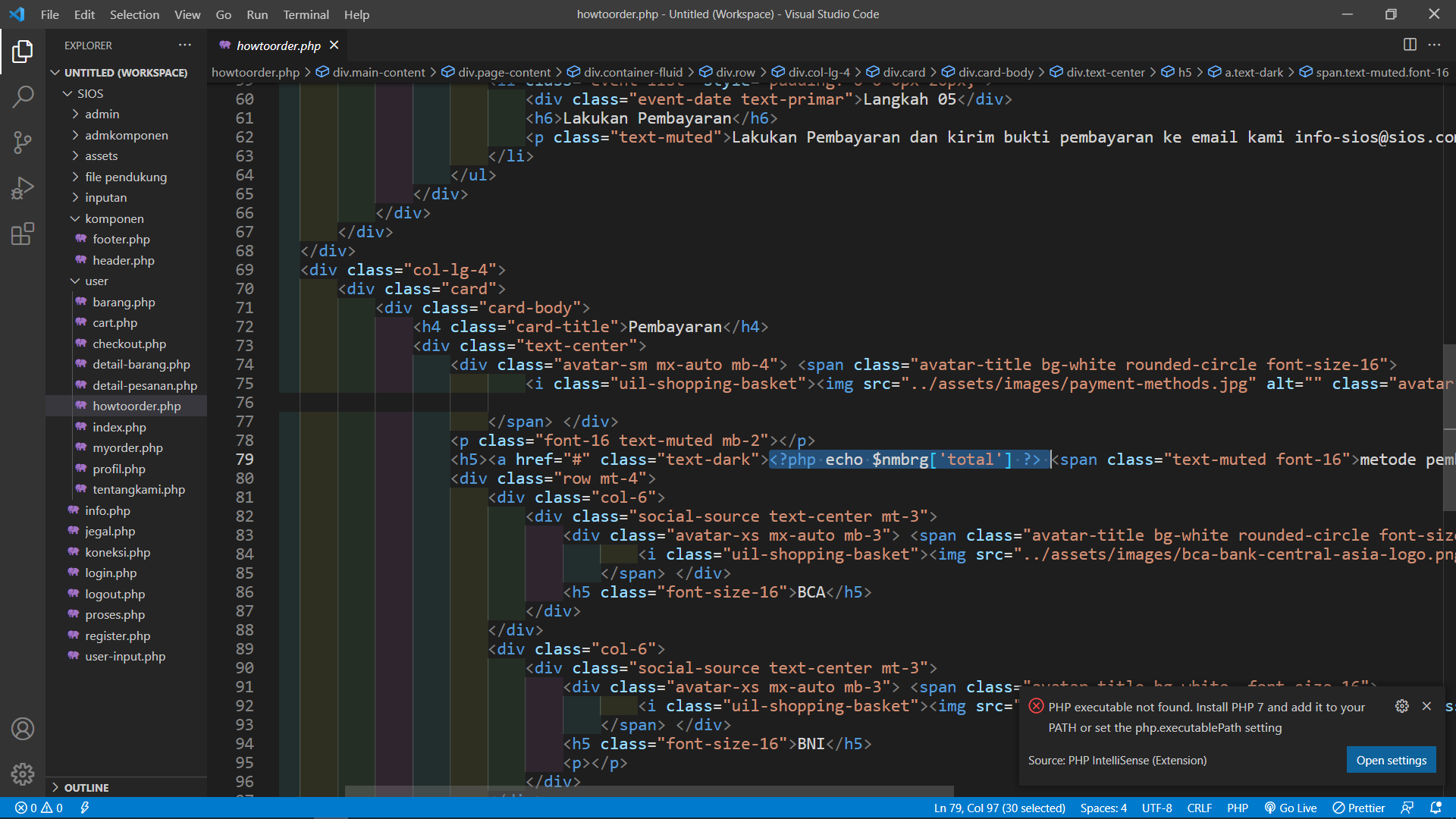
1. Kemudian setelah dari keranjang, user bisa melakukan cekout barang dengan menggunakan file cekout.php. Untuk melakukan cekout user memasukan data yang telah disediakan, kemudian data akan masuk pada tabel database yang telah di deklarasikan. Pada biaya kirim disini saya menggunakan percabangan untuk menentukan biaya kirim.



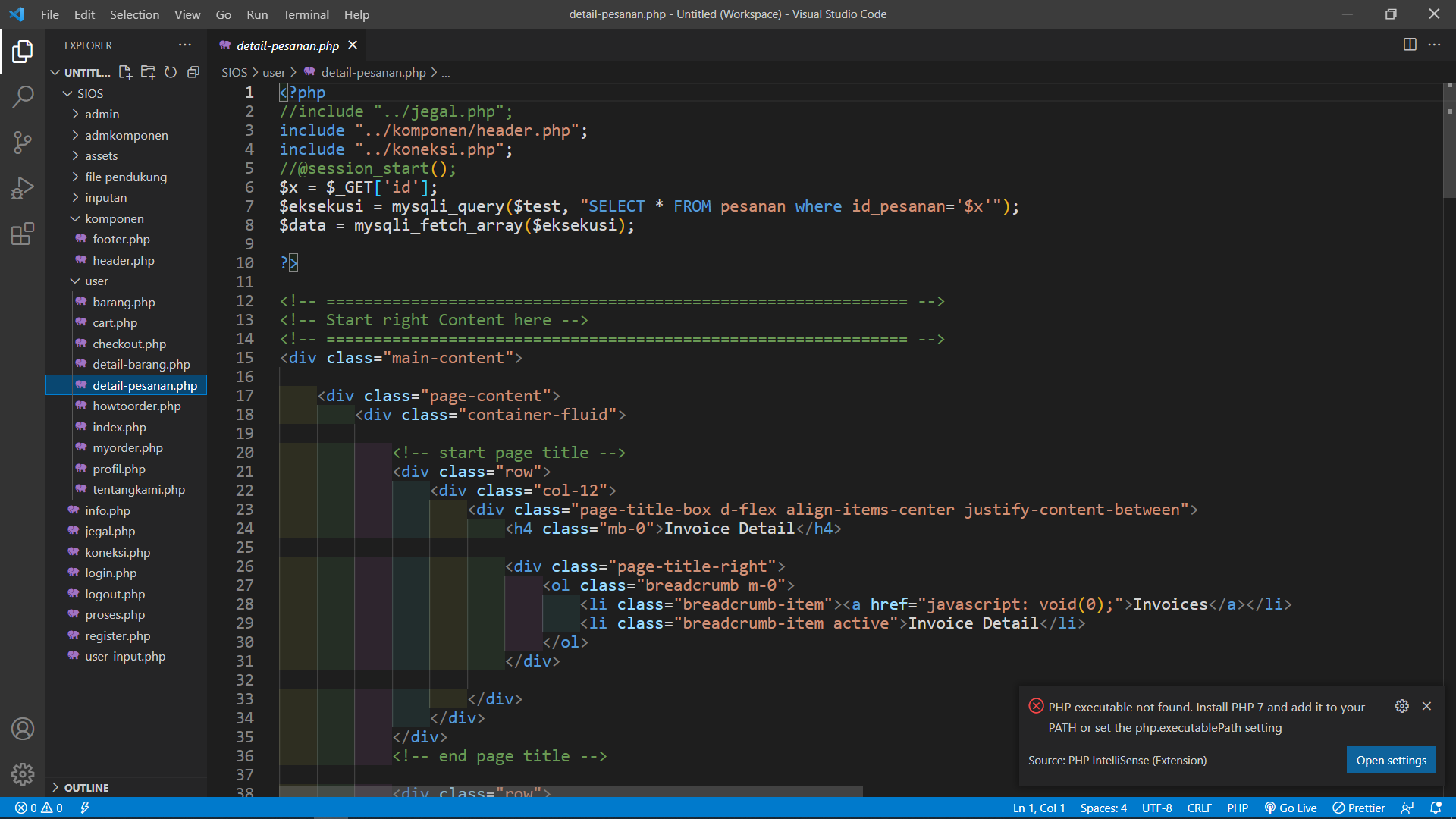
1. Setelah melakukan cekout, user bisa melihat data pesanan dan progress pesanan, pada file myorder.php sama halnya dengan program lainnya, deklarasikan terlebih dahulu variable untuk database kemudian disini saya menggunakan percabangan untuk kategori pesanan, dan memanggil data pada field status.



1. Pada file howtoorder.php adalah program untuk membuat tampilan cara pesan pada website dengan menggunakan data pembayaran, sisanya hanya menambahkan text dengan menggunakan ul dan li. Penggunaan database pada tabel pembayaran digunakan untuk memanggil harga, seperti pada program berikut.



1. Setelah melakukan pembayaran maka Langkah selanjutnya melihat detail-pesanan.php yang dimana berisikan invoice dengan menggunakan tabel pesanan untuk membuat invoice. Kemudian setelah mendeklarasikan tabel tinggal memanggil field yang akan digunakan.



1. Pada file tentangkami.php adalah program untuk menampilkan informasi website. Digunakan untuk menjelaskan seputar website penjualan tersebut dan tidak memanggil database atau data yang digunakan.
2. Pada File profil.php adalah program untuk membuat tampilan profil dan update profil. Penjelasan lebih lanjutnya di dalam program dengan komentar.

