|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| PROJECT PHP OOP minggu ke 13 (video 31-40 pemrograman web php) | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Setelah mengatur folder kategori di dalam project aplikasi restoran, selanjutnya adalah mengatur pada bagian folder menu. Sebelumnya di dalam folder menu sudah diatur pada bagian menu nya, selanjutnya mengatur pada bagian insert menu seperti memasukkan(menginsert) atau upload datanya.

2. Insert.php di dalam folder menu diisi dengan insert.php dari folder kategori, jadi bisa dicopas lalu mengganti semua nama ‘kategori’ menjadi ‘menu’ dengan cara ctrl+shift+L. Bedanya insert.php di dalam menu dengan insert.php di dalam kategori adalah di dalam insert kategori hanya mengupload atau menginsert data pada kolom kategori saja dan idkategorinya otomatis menyesuaikan urutan data ke berapa yang diinsert, sedangkan insert menu ini digunakan untuk mengupload atau menginsert data dari kolom id yang menyesuaikan id kategori nya, menu (nama produknya), gambar, dan harga. Jadi di dalam insert menu ini akan mengupload 4 kolom untuk satu baris menu. Apabila salah satu kolom tidak diisi atau dilewati maka akan muncul tulisan kosong dengan nama sesuai kolomnya.

3. Sebelum mengatur insert pada menu ini agar terhubung dengan tabel kategori untuk menyesuaikan id nya dengan id menu maka tambahkan code:

<?php

$row = $db->getALL("SELECT \* FROM tblkategori ORDER BY kategori ASC");

?>

Yang artinya dia akan mengaktifkan function getALL yang sebelumnya sudah dibuat dalam dbcontrol.php yang berfungsi untuk mengambil semua isi data dari tabelkategori di bagian kategori dan mengurutkan sesuai urutan abjad namanya bukan urutan id nya. Untuk menyesuaikan id menu dengan id kategori ketika menginsert atau mengupload data pada menu maka di dalam form pada div yang pertama isi dengan code seperti ini

<label for="">Kategori</label><br>

            <select name="idkategori" id="">

                <?php foreach ($row as $r) : ?>

                    <option value="<?php echo $r['idkategori'] ?>"><?php echo $r['kategori'] ?></option>

                <?php endforeach ?>

            </select>

Yang artinya dia akan membentuk sebuah select dengan pilihan yang diambil dari data kategori pada tabel kategori untuk menyamakan id pada menu yang akan diinsert dengan id kategori yang sudah dibuat terlebih dahulu.

4. Yang kedua adalah insert untuk nama menu nya, sama halnya seperti insert pada kategori yang ada di dalam tabel kategori, letaknya ada di dalam div kedua setelah div id kategori tadi di dalam form. Codenya seperti ini:

<div class="form-group w-50">

            <label for="">Nama Menu</label>

            <input type="text" name="menu" required placeholder="isi menu" class="form-control">

</div>

Yang artinya dia akan membentuk sebuah kolom yang berisi input dengan type text dan placeholder atau nama yang diberikan di dalam kolom input nya yaitu tertulis isi menu.

5. Selanjutnya ada insert untuk harganya, insert harga ini sama dengan insert menu, bedanya insert harga ini bertype integer karena itu ditambahkan code number sebelum required placeholdernya. Codenya seperti ini:

<div class="form-group w-50">

            <label for="">Harga</label>

            <input type="text" name="harga" number required placeholder="isi harga" class="form-control">

</div>

Yang artinya dia akan membentuk sebuah kolom input dengan type text dan placeholder agar tertulis isi harga dan lebih ditujukan untuk diisi dengan angka karena ada tambahan numbernya.

6. Insert yang terakhir yaitu gambar, karena gambar ini type nya berupa file (bukan text atau yang lainnya) maka untuk input type nya yaitu file. Code nya seperti ini:

<div class="form-group w-50">

            <label for="">Gambar</label><br>

            <input type="file" name="gambar">

</div>

Yang artinya nanti akan muncul sebuah label dengan bertuliskan gambar lalu enter karena ada br dan akan ada seperti sebuah button yang bertuliskan choose file atau pilih file atau telusuri file. Jadi ketika ditekan button itu akan muncul pencarian file dan kita bisa memilih salah satu file gambarnya. Karena terdapat insert untuk mengambil sebuah file, maka tambahkan enctype=”multipart/form-data” yang berfungsi untuk dapat mengambil sebuah file melalui input type file.

7. Div yang terakhir yaitu untuk bagian submit nya yang nantinya akan diatur untuk mengambil tindakan setelah melakukan input atau insert. Jadi dituliskan input bertype submit dengan nama simpan dan value simpan lalu buttonnya akan berwarna primary atau biru.

8. Setelah membuat beberapa div di dalam form untuk menginsert berbagai macam type selanjutnya adalah membuat php setelah html, jadi letaknya dibawah sendiri yang codenya seperti ini:

<?php

if (isset($\_POST['simpan'])) {

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

    $menu = $\_POST['menu'];

    $harga = $\_POST['harga'];

    $gambar = $\_FILES['gambar']['name'];

    $temp = $\_FILES['gambar']['tmp\_name'];

    if (empty($gambar)) {

        echo "<h3>GAMBAR KOSONG</h3>";

    } else {

        $sql = "INSERT INTO tblmenu VALUES ('',$idkategori,'$menu','$gambar',$harga)";

        move\_uploaded\_file($temp, '../upload/' . $gambar);

        $db->runSQL($sql);

        header("location:?f=menu&m=select");

    }

}

?>

Yang artinya dia akan mengambil tindakan yaitu di post atau di upload setelah memakai input bernama ‘simpan’ atau input bertype submit nya karena namanya simpan. Karena sebelum simpan terdapat beberapa insert, maka di dalam kurung kurawal if akan dibuat variable dengan isi variable post yang artinya data yang dimasukkan ke dalam input bernama sesuai dengan yang ada di dalam tanda [] nya akan dipost apabila setelah menginsert menekan button submit nya. Jika pada bagian gambar tidak diambil file nya atau tidak diisi maka akan muncul tulisan ‘GAMBAR KOSONG’ karena disitu terdapat code jika $gambar kosong maka dikeluarkan echo h3 gambar kosong. Setelah itu tujuan dipost nya data yang sudah diinsert tadi sesuai dengan $sql nya yaitu ke tabelmenu dengan diisi insert sesuai urutan kolomnya yaitu idkategori, menu, gambar, harga. Terdapat move\_uploaded\_file yang berisi ($temp, ‘../upload/’.$gambar) yang artinya gambar atau file yang dipilih pada input file gambar tadi akan diupload dari letak gambar sebelumnya yaitu $temp menuju ke folder upload lalu tersimpan sesuai dengan nama filenya ($gambar nya) tetapi bukan berarti file atau gambar pada letak sebelumnya hilang. Agar if isset post ini berhasil atau berjalan, maka ditambahkan objek $db->runSQL($sql) yang artinya dia akan menjalankan function runSQL yang berfungsi untuk menjalankan $sql nya, jadi apabila tidak ada objek $db ini maka if isset tadi tidak akan berjalan. Terakhir terdapat header yang berisi lokasi itu berfungsi untuk menampilkan tampilan browser dengan alamat tujuan sesuai lokasi di dalam tanda kurungnya setelah tindakan if isset sudah berhasil, jadi dia akan kembali ke select.php pada folder menu.

9. Setelah membuat insert.php pada menu, selanjutnya adalah membuat update.php pada menu. Update.php ini digunakan untuk mengupdate data atau mengubah isi data dari suatu baris yang terdapat di dalam tabel menu Untuk isi dari file update.php ini diisi sama dengan file insert.php, jadi bisa dicopas dari file insert ke file update ini.

10. Setelah mengcopas file insert.php ke file update.php, pada bagian if empty yang paling bawah sendiri bisa di comment terlebih dahulu atau dihapus agar dalam proses percobaan untuk update.php tidak langsung berjalan karena adanya objek db yang mengaktifkan function runSQL nya. Di dalam file ini terdapat php di bagian paling atas sendiri yang berisi $row yang digunakan untuk mengaktifkan function getALL dengan menggunakan objek db didalamnya, function getALL ini tanda kurungnya berisi select yang berfungsi untuk mengambil semua data dari tabelkategori di bagian kategorinya dengan mengurutkan namanya sesuai urutan abjad, code nya seperti ini $row = $db->getALL(“SELECT \* FROM tblkategori ORDER BY kategori ASC”); karena $row ini sudah ada, tambahkan if isset sebelum $row ini, codenya seperti ini:

if (isset($\_GET['id'])) {

    $id = $\_GET['id'];

    $sql = "SELECT \* FROM tblmenu WHERE idmenu=$id";

    $item = $db->getITEM($sql);

    $idkategori = $item['idkategori'];

}

Yang artinya ketika ditekan update pada salah satu menu di dalam tabel menu dia akan mengambil data dari baris itu untuk bisa disesuaikan id menu nya dengan id kategorinya. Setelah itu ganti placeholder pada div menu menjadi value=”<?php echo $item[‘menu’]?>” agar tulisan yang tampil pada input untuk nama menu sesuai dengan data nama menu sebelum diganti. Pada bagian div harga placeholder nya juga diganti dengan php yaitu <? php echo $item[‘harga’]?> agar tertulis harga sebelum diganti pada input harga nya. Untuk menampilkan tulisan pada select kategori agar sesuai dengan kategori pada menu nya dengan menggunakan id kategori untuk menyesuaikan namanya maka ganti option menjadi seperti ini <option <?php if ($idkategori == $r['idkategori']) echo "selected" ?> value="<?php echo $r['idkategori'] ?>"><?php echo $r['kategori'] ?></option> Artinya idkategori yang diambil dari menu itu sama dengan $r[‘idkategori’] yang nantinya digunakan untuk mengoutputkan (echo) atau menampilkan nama pada select sesuai dengan kategori nya menu yang deselect atau dipilih. Lalu untuk bagian gambarnya pada if isset terdapat $gambar yang isinya $\_FILES itu diganti isinya menjadi $gambar = $item[‘gambar’]; yang artinya dia akan memakai gambar sebelumnya sesuai dengan data atau menu yang dipilih. Setelah $temp tambahkan if yang code nya seperti ini if(!empty($temp)){$gambar=$\_FILES[‘gambar’][‘name’]; move\_uploaded\_file($temp, ‘../upload/’.$gambar)} yang artinya apabila gambar tidak kosong atau sudah ada isinya sebelumnya dengan mengetahui letak $temp nya maka ketika memilih gambar lagi dia akan mengupload atau mengganti gambar lama dengan gambar baru yang dipilih dan langsung dimasukkan ke dalam file upload, dan apabila ketika update hanya mengupdate bagian menu, harga, dll(tanpa gambar) maka gambar yang tampil akan tetap yaitu gambar yang lama.

11. Setelah melakukan langkah nomor 10, selanjutnya adalah membuat $sql di dalam if isset yang paling bawah sendiri (sebelum php empty, php empty nya bisa dihapus saja) dan $sql ini diisi dengan “UPDATE tblmenu SET idkategori = $idkategori, menu = ‘$menu’, gambar = ‘$gambar’, harga=$harga WHERE idmenu = $id”; yang artinya dia akan mengupdate data pada tabel menu dengan mengisi tiap kolom yang bagian idkategori diisi sesuai $idkategori, kolom menu diisi sesuai $menu, dst. Lalu dibagian data atau baris yang mana dia akan menyesuaikan dengan menu yang dipilih yaitu sesuai $id nya. Setelah itu buat objek db untuk diisi dengan runSQL($sql) yang berarti dia akan mengaktifkan function runSQL yang berfungsi untuk menjalankan $sql nya. dan terakhir menambahkan header yang berisi lokasi (“location:?f=menu&m=select”); yang bertujuan untuk ketika sudah selesai update dan menekan button simpan maka dia akan langsung menuju ke lokasi header nya yaitu file select.php pada menu.

12. Setelah membuat file update.php selanjutnya membuat file delete.php pada folder menu, file delete.php ini bisa diisi dengan mengcopas file delete.php yang ada dalam folder kategori, lalu mengganti semua kata kategori menjadi menu dengan memblok satu kata kategori lalu ctr+shift+L dan ketik menu, maka semua kata kategori yang ada di file itu berubah menjadi menu.

13. Setelah mengatur tabelkategori dan tabelmenu, selanjutnya adalah tabelpelanggan. Yang pertama adalah menampilkan data pelanggan, pada phpmyadmin sebelumnya sudah dibuat tblpelanggan yang berisi 4 kolom yaitu untuk idpelanggan, pelanggan, Alamat, dan telp. Selanjutnya bisa ditambahkan dengan 3 kolom lagi untuk bagian email, password dan aktif, kolom bagian email ini type data nya Varchar dengan panjang/nilai nya 150 lalu untuk kolom bagian password juga bertype data varchar dengan panjang / nilai maksimalnya 255 dan untuk kolom aktifnya bertype data integer. Setelah menambah kolom, kita bisa mencoba menginputkan 2 baris data untuk percobaan dengan mengisi kolom-kolom sesuai judulnya. Setelah mengatur phpmyadmin selanjutnya di dalam vs code. Di dalam folder pelanggan buat file seperti folder-folder sebelumnya yaitu select.php, file ini bisa diisi dengan mengcopas dari file select.php yang ada dalam folder kategori. Setelah mengcopas maka gunakan cara seperti sebelumnya yaitu ctrl+shift+L untuk mengubah nama kategori menjadi pelanggan. Apabila sudah selanjutnya cari kata pelanggan dibagian judul, ubah menjadi Pelanggan agar kata itu berawalan huruf capital sebagai judulnya yang tampil dalam browser(bukan codenya).

14. Karena file select.php pada folder pelanggan dicopas dari file select.php pada folder kategori, tabel yang tampil pada browsernya pasti juga sama seperti file select.php untuk kategori yaitu hanya memiliki 4 kolom (no, nama, delete ,dan update). Dan tabel pelanggan berbeda dengan tabel kategori, maka tambahkan th lagi untuk alamat, telp, dan email dibagian setelah nama(pelanggan) dan sebelum delete dan php. Untuk passwordnya dan keterangan aktifnya tidak ditampilkan pada tabel. Isi td dari kolom yang sudah ditambahkan ini codenya sama dengan td pada bagian nama(pelanggan) jadi bisa dicopas dan mengganti [‘pelanggan’] nya saja dengan nama sesuai kolomnya. Untuk th update, ganti dengan tulisan status, status ini akan menyatakan apabila pelanggan aktif maka dikolom status nya akan dituliskan aktif. Untuk menentukan status nya aktif apa tidak, maka akan menyesuaikan integer yang diinputkan dalam phpmyadmin sebelumnya, apabila integer yang diinputkan pada kolom aktif adalah 1 maka pelanggan ini aktif, dan apabila integer yang diinputkan bukan 1 maka pelanggan tidak aktif. Untuk itu tambahkan php di dalam tr sebelum td nya yang isi php nya codenya seperti ini:

if ($r['aktif']==1) {

                        $status = 'AKTIF';

                    }else{

                        $status = 'TIDAK AKTIF';

                    }

Yang artinya apabila isi data dari kolom aktif sama dengan 1 maka dia akan menggunakan $status berisi tulisan ‘AKTIF’ dan apabila isi data dari kolom aktif tidak sama dengan 1 maka dia akan menggunakan $status berisi tulisan ‘TIDAK AKTIF’. Selanjutnya adalah pada bagian td dari kolom status diisi dengan php yang mengecho atau mengoutputkan $status tadi. Karena status pelanggan bisa aktif bisa tidak, agar ketika ditekan status yang bertuliskan ‘TIDAK AKTIF’ menjadi ‘AKTIF’ begitu pula sebaliknya, maka kita buat file baru terlebih dahulu di dalam folder pelanggan dengan memberi nama file nya update.php.

15. Di dalam file update.php pada folder pelanggan, seperti biasa buat php open php close dulu yang isinya berupa if isset dengan $\_GET, codenya seperti ini:

if (isset($\_GET['id'])) {

    $id = $\_GET['id'];

    $row = $db->getITEM("SELECT \* FROM tblpelanggan WHERE idpelanggan=$id");

    if ($row['aktif'] == 0) {

        $aktif = 1;

    } else {

        $aktif = 0;

    }

    $sql = "UPDATE tblpelanggan SET aktif=$aktif WHERE idpelanggan=$id";

    $db->runSQL($sql);

    header("location:?f=pelanggan&m=select");

}

Yang artinya dia akan mengambil tindakan setelah menekan salah satu data pada bagian kolom status. Adanya $id yang diisi dengan $\_GET berfungsi untuk mengambil data id yang mana yang kita pilih, lalu $row yang berisi objek db yang berfungsi untuk mengaktifkan function getITEM dengan mengambil atau memilih data dari tabel pelanggan pada bagian sesuai $id nya yaitu sesuai yang kita pilih. Selanjutnya ada if yang berisi $row atau baris data yang dipilih ini dan pada bagian kolomnya apabila berisi 0(TIDAK AKTIF) maka ketika ditekan dia akan berubah menjadi 1(AKTIF) begitu juga sebaliknya karena ada else nya. Di dalam folder pelanggan ini tidak dibutuhkan lagi untuk bagian button insert data, karena bagian insert data atau tambah data pelanggan akan diletakkan dibagian front end dan saat ini masih proses pada bagian back end nya dulu, untuk itu bisa dihapus div yang berisi button nya saja.

16. Setelah file update.php selanjutnya adalah file delete.php pada bagian folder pelanggan, seperti sebelumnya pada bagian file delete.php pelanggan ini bisa dicopas dari file delete.php pada folder sebelumnya misalkan dari folder kategori, maka tinggal ubah semua kata kategori menjadi kata pelanggan.

17. Setelah mengatur pada bagian folder pelanggan, selanjutnya adalah folder user. Sama seperti folder pelanggan, di dalam phpmyadmin dibuatkan dulu tabel baru untuk tbluser karena sebelumnya belum dibuat. Di dalam tbluser ini berisi 6 kolom yaitu iduser, user, email, password, level, aktif. Untuk iduser nya bertype data integer dengan memakai auto\_increment diaktifkan, user bertype data varchar dengan panjang atau nilai maksimalnya 50, email bertype data Varchar dengan panjang atau nilai maksimalnya 150, password bertype data Varchar dengan panjang atau nilai maksimalnya 255 karena nanti ada proses dari enkripsi, level bertype data Varchar dengan panjang atau nilai maksimalnya 50, kemudian aktif bertype data integer dengan bawaan seperti yang didefinisikan yaitu 1. Setelah tabel beserta kolomnya jadi, selanjutnya bisa dicoba untuk membuat tiga baris data yang isinya sesuai dengan judul kolomnya. Baris data pertama untuk admin, kedua untuk koki, dan ketiga untuk kasir, Ketiga macam itu tadi dijadikan sebagai level nya nanti.

18. Setelah membuat tbluser pada phpmyadmin selanjutnya adalah menampilkan tabel user ini ke dalam browser dengan menggunakan vs code, untuk isi dari file select.php yang akan dibuat dalam folder user ini bisa diisi dengan mengcopas file select.php yang ada pada folder kategori. Setelah mengcopas file nya, ganti semua kata kategori menjadi user dengan menggunakan ctrl+shift+L dan untuk bagian judulnya ubah user menjadi User agar judulnya berawalan dengan huruf capital. Di dalam file select.php ini kolom pada tabelnya masih sesuai dengan jumlah tabel pada kategori, untuk mengubah tabel ini sesuai dengan tabel pada user, maka tambahkan th untuk email dan level setelah th user nya. Untuk td nya disamakan seperti td milik user yang memakai php lalu mengganti [‘user’] nya menjadi sesuai dengan nama kolomnya. Untuk td pada bagian update bisa diubah menjadi $status karena nanti akan disetting agar ketika ditekan berubah menjadi kebalikannya. Setelah itu ubah th update menjadi Status sebagai letaknya keterangan aktif atau tidak nya, hal ini sama seperti pada folder pelanggan sebelumnya yaitu dengan menambahkan php if isset sebelum td nya (di dalam foreach) yang dimana if isset ini berisi setting untuk mengatur apabila status aktif nya 1 maka tertulis aktif dan jika bukan 1 maka tertulis banned. Selain itu buat file baru untuk bagian update nya yaitu update.php yang berisi sama seperti update.php milik tabel pelanggan, jadi bisa dicopas saja lalu mengganti semua kata pelanggan menjadi user. Untuk delete.php pada folder user ini juga mengcopas dari file delete.php yang berasal dari folder pelanggan lalu mengganti semua kata pelanggan menjadi user.

19. Pada bagian ini adalah mengatur file insert.php pada tabel user, caranya dengan mengcopas file inser.php dari folder kategori karena isinya paling sedikit. Setelah mengcopas dari file insert.php kategori selanjutnya ubah semua kata kategori menjadi user dan untuk judulnya ubah menjadi User agar judul diawali dengan huruf capital. Di dalam file ini terdapat div untuk input nama user nya saja karena itu tambahkan div yang sama seperti input user untuk input email, password, konfirmasi password, dan level. Pada bagian div input untuk email type nya berupa email dan bukan text lalu ubah namanya menjadi email, dan placeholdernya juga diubah jadi email, pada bagian div input untuk passwordnya type nya berupa password dan nama juga placeholdernya diubah menjadi password, pada bagian div input konfirmasi sama seperti password hanya ubah name nya saja menjadi konfirmasi, terakhir pada bagian level ini input nya berupa select dengan 3 pilihan atau 3 option yaitu admin, koki, dan kasir. Untuk name dari select ini yaitu level. Karena sebelumnya sudah ada button bernama simpan, maka tidak usah diubah atau ditambahkan lagi. Jadi setelah form ini selesai buat sebuah php yang isinya if isset dengan variable post, codenya seperti ini:

<?php

if (isset($\_POST['simpan'])) {

    $user = $\_POST['user'];

    $email = $\_POST['email'];

    $password = $\_POST['password'];

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

    $level = $\_POST['level'];

    if ($password === $konfirmasi) {

        $sql = "INSERT INTO tbluser VALUES ('','$user','$email','$password','$level',1)";

        $db->runSQL($sql);

        header("location:?f=user&m=select");

    } else {

        echo "<h3>PASSWORD TIDAK SAMA DENGAN KONFIRMASI</h3>";

    }

}

?>

Yang artinya dia akan mengambil tindakan berupa post setelah menekan input type submit bernama simpan, untuk isi dari input user, email, password, dan level akan dipost ke dalam tabel user dengan urutan seperti pada $sql itu dengan menggunakan variable terlebih dahulu yang mengisikan ambil tindakan untuk post input yang bernama sesuai di dalam tanda kurung staples[] nya. Ada angka 1 dalam $sql nya sebagai isi dari kolom aktif yang tidak perlu diinputkan dalam browsernya. Terdapat $konfirmasi sebagai keputusan untuk menggunakan password itu, jadi $konfirmasi ini digunakan di dalam if yang apabila $password sama dengan $konfirmasi maka $sql akan dijalankan lalu data akan terinsert atau tertambah dan langsung kembali ke halaman sesuai lokasi pada headernya. Dan apabila password tidak sesuai dengan konfirmasinya maka akan muncul tulisan PASSWORD TIDAK SAMA DENGAN KONFIRMASI dibagian bawah sendiri dan input dari halaman itu akan kosong sebagai tanda harus diinputkan kembali.

20. Berikutnya adalah membuat form login, pertama adalah menambahkan div user disebelah div logout, jadi di dalam sebuah div row ke 1 atau baris ke 1 di bagian col ke 2 atau kolom keduanya terdapat div untuk logout, lalu tambahkan div untuk user sebagai form login nya. Div row 1 col 2 ini menggunakan float-right yang berarti isinya berada di rata kanan, karena itu apabila logout berada di sebelah kanan maka user akan berada di sebelah kirinya karena diurutkan dari kanan ke kiri. Untuk mt atau margin-top pada user sama dengan mt pada logout, lalu tambahkan mr-4 atau margin right pada user agar berjarak dengan logout nya.

21. User ini digunakan sebagai login nya, jadi ketika ditekan user akan langsung menuju ke file login.php yang diletakkan pada folder admin, jadi file ini ada di dalam satu folder dengan file index.php. Di dalam file login ini bisa diisi dengan html dengan title login restoran. Setelah itu tambahkan link dibawah title ini yaitu link untuk menyambungkan dengan file bootstrap .css agar tampilan web nya menjadi bagus tertata sesuai bootstrapnya. Setelah itu di dalam body nya dibuat div container yang isinya berupa div row 1 lalu div col dengan letak ditenya yaitu mx-auto dan mt-4 agar berjarak dengan bagian atasnya. Div col ini diisi dengan form yang bisa dicopas dari form insert.php pada folder kategori. Code form nya seperti ini:

<div class="form-group">

                    <form action="" method="POST">

                        <div>

                            <h3>LOGIN RESTORAN</h3>

                        </div>

                        <div class="form-group">

                            <label for="">EMAIL</label>

                            <input type="email" name="email" required class="form-control">

                        </div>

                        <div class="form-group">

                            <label for="">PASSWORD</label>

                            <input type="password" name="password" required class="form-control">

                        </div>

                        <div>

                            <input type="submit" name="login" value="login" class="btn btn-primary">

                        </div>

                    </form>

                </div>

Yang berarti form ini menggunakan method post atau data form nya akan dipost, lalu terdapat div h3 sebagai judul form, div form-group yang pertama yaitu untuk email nya dengan input type email, div form-group yang kedua yaitu untuk password nya dengan input type password, dan div terakhir yaitu untuk input submit nya.

22. Setelah membuat form pada file login.php, di dalam file index.php pada bagian php yang paling atas sendiri tambahkan session\_start(); sebelum require\_once nya, session\_start ini artinya dia akan memulai atau mengaktifkan session. Setelah itu buat if not isset atau if(!isset) seperti ini

if (!isset($\_SESSION['user'])) {

    header("location:login.php");

}

Yang artinya apabila ketika belum mengambil tindakan apapun pada usernya atau belum mengisi form pada file login.php(jadi sessionnya masih kosong), dia akan menampilkan isi dari file login.php lagi. Jadi ketika masuk ke dalam folder admin dia tidak akan bisa menampilkan file index.php nya apabila belum mengisi form yang tampil dari file login.php nya.

23. Selanjutnya kembali lagi ke bagian file login.php, setelah html buat php yang berisi if isset untuk mempost form yang sudah terisi data oleh user nya (form sudah dibuat sebelumnya). Jadi if isset ini isinya $\_POST lalu [‘login’] karena nama dari submitnya login. Lalu untuk isi di dalam tanda kurung kurawal dari if isset ini dibuatkan variable dulu yang isinya $\_POST[‘email’]; lalu variable lagi yang isinya $\_POST[‘password’]; jadi dia akan mempost data dari input yang memiliki nama sesuai dengan isi dari tanda kurung staplesnya. Masih di dalam kurung kurawal, buat $sql yang berisi SELECT\*FROM seperti biasanya dengan mengambil data dari tbluser karena yang bisa login hanya user yang sudah terdaftar sebelumnya, untuk where nya karena mencocokkan benar tidaknya email dengan password yang sudah terdaftar maka WHERE email=’$email’ AND password=‘$password’; selanjutnya di dalam file login.php tambahkan php sebelum html nya lalu isi dengan code

session\_start();

require\_once "../dbcontrol.php";

$db = new DB;

Yang artinya file ini juga menggunakan session, lalu menyambungkan dengan file dbcontrol.php dan memakai class bernama DB untuk digunakan function-functionnya nanti. Setelah itu kembali lagi ke php yang paling bawah sendiri dalam file login.php, tambahkan $count yang diisi dengan $db->rowCOUNT($sql); rowcount ini berfungsi untuk mengetahui apakah data yang diinputkan dalam form tadi ada dalam tbluser(terdaftar) atau tidak. Apabila terdaftar angkanya 1, apabila tidak terdaftar maka angkanya 0. Berikutnya dibawah $count buat if yang isinya ($count == 0){ echo “<center><h3>Email atau Password salah</h3></center>”;} yang artinya apabila angka dari data yang dimasukkan tadi 0, maka email atau passwordnya salah. Selanjutnya tambahkan

else {

        $sql = "SELECT \* FROM tbluser WHERE email='$email' AND password='$password'";

        $row = $db->getITEM($sql);

        $\_SESSION['user'] = $row['email'];

        $\_SESSION['level'] = $row['level'];

        $\_SESSION['iduser'] = $row['iduser'];

        header("location:index.php");

    }

Yang apabila data yang diinputkan bukan 0, yaitu 1 maka login berhasil dan bisa masuk ke dalam file index.php. jadi ketika data dimasukkan dan angka nya 1 maka dia akan mencocokkan dengan data yang ada di tbluser bagian email ,password, dan idusernya.

24. Setelah mengatur login, selanjutnya adalah mengatur bagian logoutnya. Pada index.php div class untuk logout bisa ditambahkan a href dengan alamat tujuannya ?log=logout. Setelah itu tambahkan dulu if isset pada php yang paling atas sendiri dengan code:

if (isset($\_GET['log'])) {

    session\_destroy();

    header("location:index.php");

}

Yang artinya ketika kita menekan logout dia akan menonaktifkan sessionnya dan kembali ke index.php lalu secara otomatis dia akan mengecek lagi login.php nya apakah data nya sudah benar atau belum karena jika tidak ada session maka data yang dimasukkan tadi belum ada dan harus mengulang login lagi.

25. Untuk mengupdate user, kita bisa tambahkan hyperlink yang tampilannya adalah ‘email’ dari user pada div user yang ada di bagian atas kanan browser atau seperti navbarnya itu. Jadi email yang tampil ini akan sesuai dengan email yang kita gunakan untuk login. Untuk mencocokkan email yang dibuat login ini menjadi tampil pada navbar usernya maka di dalam div user pada bagian isi yaitu User tambahkan menjadi User: <a href=?f=user&m=updateuser&id=<?php echo $\_SESSION[‘iduser’]?>><?php echo $\_SESSION[‘user’]?></a> yang artinya dia akan menampilkan user sesuai dengan usernya email yang digunakan login lalu ketika ditekan nama email yang tampil itu dia akan mengarahkan ke folder user pada bagian file updateuser.php lalu mengambil data yang sebaris dengan id user nya.

26. Karena email yang ditekan tadi menuju file updateuser.php, maka di dalam folder user buat file baru yang bernama updateuser.php da nisi file ini dengan mengcopas dari file insert.php dan mengganti judulnya menjadi Update User. Setelah itu tambahkan php diatas judul Update Usernya da nisi php ini dengan

if (isset($\_GET['id'])) {

    $id = $\_GET['id'];

    $sql = "SELECT \* FROM tbluser WHERE iduser=$id";

    $row = $db->getITEM($sql);

}

Yang berarti apabila sudah mendapat id nya dia akan mengambil data yang sebaris dengan $id nya. Setelah itu di dalam form nya kan terdapat div yang berisi input-input dengan type yang berbeda, nah ubah placeholdernya menjadi value semua dan isi value ini dengan php yang mengecho $row[‘user’] sesuai dengan inpunya user, jika inputnya email maka di dalam kurung staplesnya email, Dst. Hal ini bertujuan agar tulisan yang tampil adalah data yang sudah diinput sebelumnya. Setelah itu pada bagian optionnya ubah menjadi seperti ini, misalkan pada option adminnya:

<option value="admin" <?php if ($row['level'] == "admin") echo "selected" ?>>Admin </option>

Yang berarti option ini tampilan awalnya sesuai dengan level pada user yang digunakan atau dipilih saat itu. Jadi apabila user ini levelnya admin, maka tampilan awal optionnya admin, dan apabila user ini levelnya koki, maka tampilan awal optionnya koki, dst.

27. Pada pembuatan updateuser.php yang terakhir adalah mengubah isi dari php yang ada dibawah sendiri dari file nya, untuk if $password === $konfirmasi di dalam tanda kurung kurawalnya $sql ini diubah dari INSERT INTO menjadi UPDATE, codenya seperti ini

$sql = "UPDATE tbluser SET user='$user', email='$email', password='$password', level='$level' WHERE iduser=$id";

Jadi dia akan mengupdate tabeluser pada kolom-kolomnya sesuai dengan data yang diinputkan atau diubah pada formnya. Untuk WHERE iduser berarti dia akan mengupdate data yang sebaris dengan id yang saat ini digunakan (id user yang menjalankan).

28. Untuk pembuatan Home page (Front End) yang pertama adalah mengatur menu. Langkah pertama adalah menghapus semua data kategori, apabila ada eror nya maka pada select.php pada folder menu ambil bagian <?php if (!empty($row)) { ?> yang ada di tabelbody nya file select.php menu, setelah itu paste ke dalam file select.php nya kategori pada bagian yang sama yaitu didalam tbody nya dan tambahkan php untuk menutup if tadi, jadi php nya berisi } saja letaknya php ini sebelum tbody tutup. Jadi php itu tadi digunakan agar ketika tidak ada dalam kategori dia tidak eror dan tetap menampilkan judul tabelnya saja. Setelah data kategori kosong, tambahkan atau insert data baru pada kategori ini, jadi tambahkan 4 kategori yaitu makanan, minuman, jajan, dan buah.

29. Setelah melakukan langkah nomor 28, selanjutnya adalah membuat file index.php yang digunakan untuk tampilan awal ketika membuka folder restoran. File index.php ini diletakkan di dalam folder restoran (jadi dia sejajar dengan folder tabel-tabel lainnya). Jadi ketika tujuan kita ke folder restoran, dia akan mengeluarkan file index.php yang baru ini, dan apabila pada link nya ditambahkan /admin yang artinya masuk ke folder admin setelah folder restoran dia akan mengeluarkan file login.php untuk login dulu baru bisa masuk ke index.php yang ada di folder adminnya. File index.php yang baru ini bisa diisi dengan mengcopas file index.php yang lama, lalu mengganti atau menghapus beberapa bagian agar terlihat sedikit berbeda dengan index.php yang lama.

30. File index.php yang baru ini pada bagian if isset di dalam div col kedua dari div row kedua, php if isset ini dihapus saja dan diganti dengan php echo h1 yang berisi DAFTAR MENU, jadi yang ditampilkan bukan lagi file dari file-file menu, kategori, dll tetapi hanya judul daftar menu saja. Lalu pada bagian navbar user dan logout yang tadinya ada hyperlink dan php nya, ini dihapus saja jadi tinggal div class untuk tulisan Logout dan User nya saja, dan User ini diubah menjadi pelanggan lalu tambahkan 2 div lagi untuk login dan daftar, code nya seperti ini

<div class="col-md-9">

                <div class="float-right mt-4"> Logout</div>

                <div class="float-right mt-4 mr-4"> Login</div>

                <div class="float-right mt-4 mr-4"> Pelanggan </div>

                <div class="float-right mt-4 mr-4"> Daftar </div>

</div>

Jadi dia akan menampilkan navbar dengan rata kanan yaitu logout, login, pelanggan, daftar dengan jarak mr-5 setiap divnya. Selanjutnya ada 2 if isset session yang ada di php paling atas itu dihapus saja. Setelah itu ubah title menjadi Restoran Mantab atau nama restoran yang lainnya dan tambahkan h2 yang berisi restoran di div row ke 1 div col 1 dengan nama restorannya. Lalu untuk link yang mengarahkan pada bootstrap css nya hilangkan tanda ../ yang paling awal yang artinya dia keluar dari folder dan masuk ke dalam folder lainnya karena letak file index.php ini sejajar dengan letaknya folder bootstrap, jadi langsung ke folder bootstrap nya tanpa harus keluar folder dan masuk kembali. Begitu juga dengan link yang mengarahkan pada file dbcontrol.php karena file index.php ini setara dengan file dbcontrol.php maka hilangkan tanda ../ nya juga.

31. Setelah menghapus dan mengganti beberapa bagian dalam file index.php yang baru, selanjutnya di dalam php yang paling atas sendiri di dalam file ini tambahkan $sql yang berisi “SELECT \* FROM tblkategori ORDER BY kategori”; yang berarti dia akan mengambil data dari tabelkategori yaitu bagian kolom kategorinya, setelah itu buat $row yang berisi objek db untuk mengaktifkan function getALL($sql), jadi dia akan mengambil semua data dari kolom kategori pada tabelkategorinya. Setelah itu tambahkan h3 untuk judul kategori pada div row kedua dan div col, di dalam div col ini lalu tambahkan hr setelah h3 sebagai penambah garis dibawahnya dan letakkan h3 juga hr ini sebelum nav item. Setelah itu untuk nav itemnya codenya jadi seperti ini:

<?php if (!empty($row)) { ?>

                    <ul class="nav flex-column">

                        <?php foreach ($row as $r) : ?>

                            <li class="nav-item"><a class="nav-link active" href="#"><?php echo $r['kategori'] ?></a></li>

                        <?php endforeach ?>

                    </ul>

<?php } ?>

Yang artinya apabila di dalam tabelkategori bagian kategorinya tidak kosong atau ada datanya maka dia akan membuat navbar yang mengarah kebawah atau flex dan foreach digunakan untuk melooping isi data dari kategori dengan masing-masing datanya memiliki hyperlink. Untuk mengecek apakah ketika ada data kategori yang ditambahkan dia akan tampil ke dalam file index.php yang baru tadi, maka di dalam file index.php yang lama insert kategori dan cek pada file index.php yang baru.

32. Selanjutnya mengatur Home Page (Front End) pada bagian Menampilkan Produknya, karena sebelumnya pada bagian menu dihapus, maka kita tambahkan dulu menu-menu dengan menginsert pada page menu dan menu isi sesuai dengan kategori yang ada dan isi menu ini hingga 5 halaman yang dimana 1 halamannya berisi 3 menu, jadi totalnya 15 menu, 5 kategori, dan tiap kategorinya ada 3 menu. Jadi misalkan ada kategori makanan maka menu yang ditampilkan adalah macam-macam makanan. Di dalam folder restoran dibuat folder baru yaitu folder home yang diisi dengan file produk.php. Untuk isi dari file produk ini bisa mengcopas dari select.php yang ada di folder menu. Dengan mengcopas dari select.php pada folder menu tidak semua akan digunakan, jadi ada bagian-bagian tertentu yang akan dibuang yaitu button tambah menu diatas h3 judul menu, form yang berisi select option, php yang berisi if isset dibawah h3 judul menu.

33. Setelah menghapus beberapa bagian dalam file produk.php selanjutnya di dalam folder index.php yang baru atau yang di dalam folder restoran pada bagian echo h3 ‘DAFTAR MENU’ ganti dengan code seperti ini:

if (isset($\_GET['f']) && isset($\_GET['m'])) {

                    $f = $\_GET['f'];

                    $m = $\_GET['m'];

                    $file = $f . '/' . $m . '.php';

                    require\_once $file;

                } else {

                    require\_once "home/produk.php";

                }

Jadi ketika dia belum mengambil tindakan pada page index.php maka yang akan ditampilkan ketika membuka file index.php adalah isi dari file produk.php dari folder home yang pertama, dan jika ditekan page lainnya maka akan tampil sesuai urutan page selanjutnya. Pada produk.php hapus variable $where yang ada di dalam $jumlahdata dan $where yang ada di dalam $sql pada php setelah div class pertama. Apabila gambar belum muncul pada tampilan index.php restoran dibagian col browser letaknya file produk.php, maka pada bagian src gambarnya hapus ../ karena dari index.php menuju upload tidak perlu keluar folder. Pada tabel produk.php hapus bagian update dan delete nya baik bagian th maupun td nya. Setelah itu ubah echo yang pertama di dalam php for $i dibagian bawah sendiri dari file menjadi seperti ini

 echo '<a href="?f=home&m=produk&p=' . $i . '">' . $i . '</a>';

Jadi angka halamannya akan sesuai dengan yang sudah dibuat sebelumnya.

34. Langkah terakhir dari bagian menampilkan produk yaitu mengubah tabel yang tampil menjadi seperti sebuah card untuk setiap menunya, card ini bisa kita ambil codenya dari bootstrap lalu paste ke dalam vs code di dalam file produk.php bagian di dalam php if !empty lalu di dalam php foreach nya agar card ini terlooping sejumlah menu yang ada dan terbagi menjadi beberapa halaman seperti saat masih berupa tabel. Di dalam card ini terdapat src gambarnya, kita bisa ambil src gambar ini dari td gambar yang ada di tabel, lalu untuk card titlenya kita ambil dari td menu, dan untuk card-text nya kita ambil dari td harga. Untuk button pada cardnya ganti dengan code seperti ini

<a class="btn btn-primary" href="?f=kategori&m=insert" role="button">BELI</a>, Code itu tadi bisa diambil atau dicopas dari button tambah data pada folder kategori. Setelah itu tambahkan float left pada div card nya agar card tertata dari kiri ke kanan lalu beri margin 10px agar berjarak antara kanan kiri bawah atasnya. Lalu pada gambar nya tambahkan style dengan height nya 150px agar semua ukuran gambarnya sama lalu kurangi width pada div card yang sebelumnya 18rem menjadi 15rem, dengan begitu card yang tampil dalam browser satu barisnya berjejer 3 buah card. Apabila card sudah jadi maka hapus semua yang berkaitan dengan tabelnya karena tidak lagi digunakan. Apabila angka untuk memilih halaman nya tidak tertata dengan baik, maka atur dibagian for $i yang ada di bawah sendiri masukkan ke dalam div yang berstyle clear:both.

35. Mengatur Kategori Produk, karena sebelumnya di dalam file index.php sudah dibuat tampilan data dari kategori yang bisa ditekan atau memiliki hyperlink, agar ketika diklik salah satu data kategorinya akan keluar card menu sesuai kategorinya maka di dalam file index.php pada bagian li class letaknya code data kategori ditampilkan kan ada hyperlink dan disitu alamat tujuannya masih #, maka ubah # ini menjadi ?f=home&m=produk&id<?php echo $r[‘idkategori’]?> yang artinya dia akan menuju folder home lalu file produk dan memilih idkategori sesuai dengan kategori yang dipilih. Setelah itu di dalam produk.php pada bagian php di dalam div class setelah judul h3 menu nya hapus $row dan ganti dengan if isset yang codenya seperti ini:

if (isset($\_GET['id'])) {

        $id = $\_GET['id'];

        $where = "WHERE idkategori = $id";

        $id = "&id=".$id;

    } else {

        $where ="";

        $id="";

    }

Setelah itu tambahkan $where ke dalam $jumlahdata setelah FROM tblmenu nya, dan tambahkan lagi $where ke dalam $sql setelah SELECT \* FROM tblmenu $where. Dengan begitu ketika diklik atau dipilih salah satu kategori dia akan menampilkan menu apa saja yang ber idkategori sama dengan kategori yang dipilih. Dan apabila belum dipilih kategorinya dia tidak akan eror karena dialihkan ke else. Adanya $id itu nanti berfungsi pada bagian pengaturan page atau pembagian halamannya, jadi dibagian code untuk mengatur paging yaitu php yang paling bawah bisa ditambahkan $id setelah p= $i nya, jadi bacanya nanti &id=$id (sesuai id yang dipilih). Terakhir untuk judul Restoran Mantab nya agar menjadi sebuah hyperlink dan ketika ditekan kembali lagi ke menu awal nya maka tambahkan a href di dalam h2 untuk Restoran Mantabnya dan alamat tujuannya adalah index.php atau kembali ke halaman awal dari file nya , jadi judul Restoran Mantab ini sebagai tempat untuk kembali ke home nya.

**Saya Belum Mengerti**

1.