|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| PHP Dasar video 31 s/d 40 | 90 |

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

1. Kita bisa menambahkan menu dalam tabel menu menggunakan form pada file insert.php dengan codingan yang sama dengan insert.php pada folder kategori dengan mengganti perintah yang berisi kategori dengan menu.
2. Pada form untuk harga kita menambahkan input number pada required placehorder agar bisa menginputkan angka, karena tipe data kolom harga pada data base adalah float.
3. Kita juga dapat menginputkan gambar menggunakan form dengan input typenya adalah file agar ketika kita klik akan langsung menuju ke explorer user dan name untuk formnya adalah gambar, lalu pada form kia masukkan perintah enctype=”multipart/form-data”.
4. Kita bisa membuat combobox dengan mengambil data dalam tabel kategori agar bisa memberikan pilihan data mana yang ingin ditambahkan.
5. Penggunaan if (empty($gambar)){ echo “<h3>Gambar Harus Terisi!</h3>”} agar jika variabel gambar tidak terisi akan muncul tulisan ‘Gambar Tidak Terisi!’ pada web. Lalu penggunaan

else { $sql = (‘’,$idkategori,’$menu’,’$gambar’,$harga)(karena $idkategori dan $harga bertipe data integer dan float maka tidak diberi tanda petik);

move\_uploaded\_file($temp,’../upload/’.$gambar)(file lokasi untuk menyimpan gambar yang diambil dari explorer);

$db -> runSQL($sql)(perintah untuk menjalankan sqlnya);

header(“location:?f=menu&m=select”)(lokasi web yang ingin dituju setelah selesai mengisi data tersebuy); } maka, file gambar yang diambil akan tersimpan ke file upload/file yang dituju lalu jika kita klik simpan akan kembali ke menu select.

1. Kita bisa memperbarui data yang sudah tersimpan di folder menu melalui file update.php dengan menggunakan codingan yang sama dengan insert.php sebelumnya dengan mengambil alamat alamat data yang tersimpan.
2. Cara mengambil masing masing data adalah memasukkan dulu perintah getITEM yang sudah dibuat sebelumnya,lalu membuat variabel untuk perintah tersebut misal ‘$item’. Kemudian, kita tambahkan pada input dalam setiap form, yaitu required value=”<?php echo $item[‘menu(nama setiap form nya)’]?>”.
3. Pada combobox agar menunjukkan dimanakah kategori pada menu yang kita pilih adalah dengan cara pada option kita tambahkan perintah <?php if($idkategori ==$r[‘$idkategori’] echo “Selected”) ?> .
4. Cara agar gambar yang sudah tersimpan bisa diganti adalah dengan menggunakan perintah if(!empty($temp)) { $gambar = $\_FILES[‘gambar’][‘name’]; move\_uploaded\_file($temp,”../upload/”.$gambar); } $sql = “UPDATE tblmenu SET idkategori=$idkategori, menu=’$menu’, gambar=’$gambar’, harga=$harga WHERE idkategori=$id”; echo $sql; $db -> runSQL($sql); Header(“location:?f=menu&m=select”); Maka, jika gambar sudah terisi maka akan terganti dengan gambar yang diinginkan lalu gambar akan tersimpan pada file lokasi yang dituju lalu jika di klik simpan akan kembali lagi ke menu select.
5. Kita juga bisa menghapus data yang tersimpan pada folder menu dengan menambahkan file delet.php pada folder menu.
6. Caranya sama dengan delete pada tabel kategori hanya saja mengganti perintah kategori dengan nama menu.
7. Kita dapat menampilkan nama pelanggan yang terdapat dalam database dengan menambahkan file select.php pada folder pelanggan. Dengan mencopy codingan yang sudah ada pada folder kategori dan menu. Lalu,mengisikan setaip kolom yang ada di setiap database.
8. Untuk menunjukkan pelanggan itu aktif atau tidak kita isi pada database dengan angka 1 atau 0 jika 1 maka pelanggan tersebut aktif, namun jika 0 maka pelanggan tersebut tidak aktif. Cara menampilkannya pada web adalah dengan mengguanakan fungsi if($r=[‘Aktif’]==1;){ } (maka jika dinyatakan 1 maka akan tampil pernyataan aktif). lalu penggunaan else { $status = ‘Tidak Aktif’;} (maka angka yang berada di database selain satu itu menunjukkan bahwa pelanggan tidak aktif).
9. Kita bisa mengubah pelanggan yang tidak aktif menjadi aktif begitu sebaliknya. Dengan cara, menggunakan fungsi sqlnya, yaitu $sql = “UPDATE tblpelanggan SET aktif=$aktif WHERE idpelanggan = $id”; kemudian jalankan sqlnya dengan menggunakan perintah $db->runSQL($sql); lalu, gunakan header(“location:?f=pelanggan&m=select”); agar ketika kita klik bagian Aktif atau Tidak Aktifnya akan terupdate lalu langusng akan kembali ke menu select pada folder pelanggan.
10. Kita bisa mengapus data pelanggaan yang sudah tersimpan dengan cara yang sama dengan codingan pada delete.php pada folder kategori dan menu hanya saja mengubah perintah yang bertuliskan menu/kategori diganti dengan pelanggan sesuai dengan yang ingin diubah.
11. Kita dapat mengambil data user/pengguna pada database dengan cara membuat file baru yaitu select.php isinya sama dengan select.php pada folder pelanggan hanya mengganti nama tabel sesuai yang ada dalam database lalu mengganti perintah pelanggan dengan user.
12. Kita juga dapat mengganti status user dengan codingan yang sama dengan folder pelanggan dengan mengganti perintah pelanggan dengan user untuk menunjukkan apakah user tersebut aktif atau banned.
13. Kita juga dapat menghapus data user yang terdaftar dalam database dengan menggunakan file delete.php codingannya sama dengan delete.php pada folder pelanggan.
14. Kita juga bisa juga bisa menambahkan data user dengan menambahkan file isert.php pada folder user, codingannya sama dengan insert.php pada folder sebelum sebelumnya hanya kita mengganti perintahnya menjadi user dan menambahkan form sesuai dengan isi yang ada dalam file select.php.
15. Dalam form untuk password dan konfirmasi kita menggunakan typenya adalah password agar otomatis jika kita mengisi password yang tampil hanya simbol khusus untuk menjaga privasi password.
16. Kita juga bisa menambahkan combobox untuk memasukkan level user.
17. Jika password yang diisikan ke konfirmasi password tidak sama atau sebaliknya kita bisa mengaturnya agar password tidak berantakan jadi jika tidak sama maka data tidak akan terisi dan muncul tulisan ‘Password Tidak Sama!’(sesuai dengan kalimat yang kita perintahkan), caranya adalah dengan menggunakan fungsi if seperti

if($password == $konfirmasi)(name dari form password dan untuk konfirmasi password){

$sql = “INSERT INTO tbluser VALUES(‘’,’$user’,’$email’,’$password’,’$level’,1)”(jika sama maka data akan tersimpan);

$db->runSQL($sql)(perintah untuk menjalankan sqlnya);

header(“location:?f=user&m=select”(lokasi yang dituju)); } maksudnya, jika isi dari form password dan konfirmasi password sama maka data setiap form akan di simpan sesuai tempatnya dan akan kembali ke menu select. Lalu penggunaan else { echo ‘<h3>Password Tidak Sama!</h3>’} tetapi jika isi dari form password dan konfirmasi tidak sama maka akan tampil tulisan ‘Password Tidak Sama!’;

1. Cara supaya mengatur agar web kita harus login/mengisi data terlebih dahulu adalah kita membuat file baru misal login.php pada suatu folder yang sama dengan file web utama dengan menambahkan perintah if pada web utama supaya ketika user membuka web maka akan langsung menuju pada bagian login. Caranya adalah menggunakan “session\_start();“ terlebih dahulu untuk memulai eksekusi session pada server dan kemudian menyimpannya pada browser. Lalu, menggunakan perintah if(!isset($\_SESSION[‘user’])) { header(”location:login.php”);} Artinya jika user tidak terisi/terdaftar maka akan diarahkan pada login.php untuk melakukan pendaftaran terlebih dahulu.
2. Pada login.php kita menambahkan sebuah form untuk tempat para user yang aktif dan terdaftar dalam database bisa mengakses index.php, maka menggunakan perintah form untuk nama email dan password untuk konfirmasi, lalu agar user yang terdaftar dapat langsung diarahkan ke dalam index.php adalah dengan menggunakan perintah if. Sebelum itu,kita harus mengambil data yang terdaftar dengan cara menggunakan perintah sqlnya yaitu $sql = “SELECT \* FROM tbluser email = ‘$email’ AND $password ”; lalu echo $sql untuk mengecek apakah sudah berhasil untuk mendapatkan datanya.
3. Untuk menunjukkan user itu apakah aktif apa tidak kita gunakan perintah rowCOUNT untuk menunjukkan angka 0/1(tidak aktif/aktif). Kita membuat satu variabel misal $count = $db -> rowCOUNT($sql);
4. Setelah kita mendapatkan data diatas selanjutnya kita mengambil email dan level user yang tersimpan di database kita menggunakan perintah getITEM caranya menggunakan if terlebih dahulu yaitu if($count == 0){ echo “<h3>Email dan Password Belum Terdaftar!</h3>” } artinya jika pada data status user adalah 0 maka email atau password tidak terdaftar dan tidak dapat mengakses index.php dan akan muncul tulisan ‘Email dan Password Belum Terdaftar!’(atau sesuai dengan yang kita perintahkan); lalu menggunakan fungsi else { $sql = “SELECT \* FROM tbluser email = ‘$email’ AND $password ”; $row = $db -> getITEM($sql); } yang berarti jika datanya bernilai 1 yang brarti aktif maka data akan tersimpan.
5. Untuk menyimpan setiap datanya kita menggunakan variabel ‘$\_SESSION’ dengan cara : $\_SESSION = $\_GET[‘email’] = $row[‘email’];

$\_SESSION = $\_GET[‘level] = $row[‘level];

Header(“location:index.php”);

Jika data sudah benar dan terdaftar maka user akan diarahkan ke menu index.php.

1. Kita juga bisa melogout akun dengan cara memberi hyperlink pada bagian logout dengan cara memberi perintah <a href=”?log=logout”>Logout</a> agar bisa terproses kita gunakan perintah ig yaitu if(isset($\_GET[‘log’])) { session\_destroy(); header(“location:login.php”) } artinya jika kita mengklik logout maka otomatis akan menuju ke menu login.
2. Kita bisa memperbarui data user yang sudah tersimpan dalam database. Caranya, kita membuat file baru yaitu updateuser.php pada folder user lalu kita membuat form untuk menampilkan data data yang tersimpan pada setiap bagian. Caranya sama dengan index.php pada folder user itu sendiri.
3. Cara untuk mengakses bagian update ini kita gunakan tulisan user yang berada pada bagian header yang sudah dibuat yaitu dengan memberikan hyperlink. Namun, pertama - tama kita tambahkan session untuk id user pada fungsi else yang kita buat pada file login.php seperti $\_SESSION[‘iduser’] = $row[‘iduser’];. Lalu, kita sambungkan pada hyperlink untuk user yang berada di header seperti <a href=”?f=user&m=updateuser&id= <?php echo $\_SESSION[‘iduser’] ?>”> agar ketika kita sudah login akun user akan tertampil di bagian user lalu ketika kita klik akan langung menuju menu update user.
4. Pada updateuser.php agar ketika kita klik pada bagian user langsung terlihat pada bagian mana sajakah yang terdaftar kita harus mengambil masing masing id bagian. Cara mengambilnya adalah dengan menggunakan fungsi if seperti

if(isset($\_GET[‘id’])) {

$id = $\_GET[‘id’];

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

$row = $db->getITEM($sql); } jika id dari masing masing bagian sudah terambil maka akan menjalankan sqlnya dengan menggunakan fungsi getITEM yang sudah pernah kita buat sebelumnya.

1. Lalu cara menginputkan masing masing id pada bagiannya di form adalah dengan menambahkan value pada form seperti misal pada form bagian nama user maka value =<?php echo $row[‘user(nama fungsi variabelnya)’] ?> begitu dengan setiap bagian form yang di perlukan.
2. Kita juga memerlukan combobox untuk menampilkan pada level apa user tersebut dengan menambahkan value pada setiap option combobox misal value = “admin”<?php if($row= [‘level’] === “admin”) echo “selected” ?> yang artinya jika level user yang terpilih adalah admin maka yang tampil dalam combobox adalah admin.
3. Jika kita sudah mendapatkan data yang diperlukan langkah selanjutnya adalah menyinpan data user pada database adalah dengan menggunakan fungsi if seperti

If($password == $konfirmasi) {

$sql = “UPDATE tbluser SET user=’$user’, email=’$email’, password=’$password’, level=’$level’

WHERE iduser = $id”;

$db->runSQL($sql);

header(“location:?f=user&m=select”);

} else {

echo ‘<h3>Password Tidak Sama!</h3>’

} yang artinya jika password dan konfirmasi password sudah sama maka data akan terupdate dan user akan diarahkan ke menu select user. Tetapi jika berbeda maka akan muncul tulisan ‘Password Tidak Sama!’ atau sesuai dengan apa yang kita perintahkan.

1. Kita bisa mengatur menu utama pada aplikasi restoran dengan membuat file baru pada folder utama yaitu restoran yang bernama index.php isinya hampir sama dengan index.php pada folder admin.
2. Dalam index.php pada folder restoran ini kita mengambil isi data dari tblkategori yang nantinya ditampilkan. Caranya adalah dengan menggunakan fungsu sql terlebih dahulu yaitu

$sql = “SELECET \* FROM tblkategori ORDER BY kategori”;

$row = $db->getAll($sql);

ketika kita sudah membuat perintah tersebut selanjutnya kita menampilkannya menggunakan fungsi if terlebih dahulu pada bagian nav kategori seperti

If(!empty($row)) { (lalu kita memasukkan datanya menggunakan foreach di dalam fungsi if seperti)

foreach($row as $r): (agar data yang ditampilkan satu per satu)

Lalu kita masukkan fungsi diatas pada bagian nav kategori karena data yang kita ambil adalah kategori terlebih dahulu. Dengan cara,

echo $r[‘kategori’]; }

1. Untuk menampilkan produk yang kita jual seperti makanan misalnya kita bisa menambahkan folder home pada restoran lalu kita menambahkan file produk.php untuk isinya sama dengan file select.php pada folder menu. Tetapi, ada beberapa yang harus diubah yaitu pada bagian combobox juga pada fungsi if untuk opsi/pilihan.
2. Cara untuk menampilkan produknya adalah kita menambahkan fungsi if terlebih dahulu pada file index.php pada folder restoran seperti

If(isset($\_GET[‘f’]) && ($\_GET[‘m’])){

$f = $\_GET[‘f’];

$m = $\_GET[‘m’];

} (kita mengambil folder dan menunya dulu kemudian menggunakan fungsi else) else { require\_once “home/produk.php”; } agar jika folder dan menu tersebut sudah terisi maka akan langsung menuju ke file restoran.

1. Setelah terbaca kita akan menyambungkan file index.php untuk ke menu produk dengan membuat perintah di dalam fungsi if seperti

If(isset($\_GET[‘f’]) && ($\_GET[‘m’])){

$f = $\_GET[‘f’];

$m = $\_GET[‘m’];

$file = $f.’/’.$m.’.php’;

echo $file;

} ketika lokasinya sudah benar lalu gunakan require\_once $file;

Jadi, file index.php akan menuju ke folder home lalu akan mencari file produk.php lalu akan menampilkannya menggunakan perintah require\_once. Lalu, agar tampilan produknya lebih rapi kita bisa menggunakan card dari bootstrap.

1. Penggunaan clear : both; pada style agar item tidak terkena flot dan agar tetap rapih.
2. Kita bisa mengatur kategori dari setiap menu. Jadi, ketika kita klik salah satu kategorinya maka akan muncul pada bagian manakah kategori makanan itu. Pada bagian kategori kita isi hyperlinknya dengan idnya seperti <a href=”?f=home&m=produk&id=<?php echo $r[‘kategori’] ?>”> lalu untuk emanmpilkan seluruh isi kategorinya adalah dengan menggunakan echo $r[‘idkategori’].
3. Lalu pada file produk kita tambahkan fungsi if agar kategori di index.php idnya bisa di ambil dan akan tersambung ke produk.php seperti if(isset($\_GET[‘id’])){ $id = $\_GET[‘id’]; }
4. Lalu kita tambahkan variabel where agar tahu dimanakah letak masing masing id kategori tersebut seperti $where = “WHERE idkategori = $id”; lalu kita tambahkan variabel tersebuta pada fungsi rowCOUNT dan sqlnya. Jika terjadi error itu karena idnya beum di isset atau belum diisi maka pada fungsi if kita tambahkan else {$where=””;} agar tetap ada isinya walaupun hanya tanda petik saja. Lalu untuk idnya kita tambahkan variabel pada fungsi if untuk id yaitu $id=”&id”.$id; lalu pada else kita tambahkan $id = “ ”; agar jika tidak terisi akan tidak terjadi error.
5. Agar setiap produk tampil sesuai kategorinya kita atur pagingnya dengan menambahkan .$id. Pada fungsi for untuk pagingnya.
6. Untuk membuat tombol kembali ke home index.php kita gunakan judul untuk tombol kembalinya dengan menambahkan hyperlink yang menuju ke index.php seperti <a href=””index.php”> karena file tujuan sama tempatnya maka tidak menggunakan tanda garis miring ‘/’.

**Saya Belum Mengerti**

1.

2.

3.