|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| PROJECT PHP OOP minggu ke 15 (video 46-50 pemrograman web php) | 95 |

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

1. Pembuatan Home Page (Front End) pada bagian Menambah Checkout. Jadi ketika sudah memasukkan beberapa menu ke dalam keranjang, totalnya akan muncul dan selanjutnya bisa checkout. Nah checkout ini berada di bawah tabel berupa button yang nanti akan menghubungkan dengan file checkout.php dimana file ini akan menghubungkan ke tblorder untuk insert data berdasarkan checkout dari pelanggan.

2. Di dalam file produk.php terdapat button untuk beli, button ini bisa dicopas ke dalam file beli.php dan diletakkan dibawah sendiri diluar php karena bentuknya html. Lalu ganti nama BELI menjadi CHECKOUT untuk hyperlink nya ?f nya tetap sama yaitu di folder home lalu untuk file yang dipilih maka ganti dengan m=checkout&total=<?php echo $total ?> jadi dia akan menuju ke file checkout dan menampilkan url total dari menu yang dibeli. Karena $total ini berada di dalam function pada file beli.php, untuk mengeluarkan $total agar bisa digunakan diluar function maka tambahkan global $total; di dalam functionnya. Dengan begitu $total yang ada di link akan muncul jumlah total dari menu yg dibeli pelanggan.

3. Selanjutnya buat file checkout.php (namanya sesuai dengan m yang ada di url tadi). Di dalam file ini isi dengan php yang pertama untuk if isset,

if(isset($\_GET[‘total’])) {

$total = $\_GET[‘total’];

echo $total;

}

Jadi apabila total sudah terisi maka dia akan mengambil nilai totalnya dan ditampilkan ke dalam file checkout ini. Selanjutnya buat

function idorder($sql){

global $db;

$sql = “SELECT idorder FROM tblorder ORDER BY idorder DESC”;

$jumlah = $db->rowCOUNT($sql);

If($jumlah == 0){

$id = 1;

} else{

$item = $db->getITEM($sql);

$id = $item[‘idorder’]+1;

}

Return $id;

}

Jadi function ini berfungsi untuk mengaktifkan db yang didalam function agar bisa digunakan diluar function dengan adanya global, lalu sql nya berfungsi untuk mengambil data dari idorder pada tblorder dengan urutan data yang mana yang terakhir dibuat. Lalu ada $jumlah yang berfungsi untuk mengambil jumlah data dari idorder nya, apabila data dalam idorder masih kosong maka data baru yang ditambahkan akan ber id 1, dan apabila data dalam idorder sudah terisi lebih dari 0 maka akan mengambil id dari data yang terakhir dibuat lalu ditambahkan 1 dan hasilnya akan digunakan untuk id baru yang akan dibuat. Apabila function sudah selesai, maka di dalam if isset tambahkan echo ‘<br>’; agar memberi jarak lalu echo idorder(); yang akan menampilkan isi function pada bagian sesuai yang direturn yaitu $id nya. dibawah function idorder, buat function lagi

function insertOrder($idorder, $idpelanggan, $tgl, $total)

{

    global $db;

    $sql = "INSERT INTO tblorder VALUES ($idorder, $idpelanggan, '$tgl', $total, 0,0,0)";

    $db->runSQL($sql);

}

Jadi function ini digunakan untuk menginsert data ke dalam tblorder, data yang masuk akan terurut sesuai dengan values nya. lalu db runSQL untuk menjalankan sql nya. karena di dalam kurung functionnya masih berupa variable dan belum ditentukan isinya maka sql tidak akan jalan, karena itu balik lagi ke If isset di dalamnya tambahkan

$idorder = idorder();

$idpelanggan = $\_SESSION[‘idpelanggan’];

$tgl = date(Y-m-d);

Letakkan 3 variable tadi dibagian atas setelah $total; lalu tambahkan echo ‘<br>’; setelah echo idorder() agar berjarak lalu insertOrder($idorder, $idpelanggan, $tgl, $total); yang berarti dia akan mengisi function insertOrder dengan $idorder sesuai function idorder(), $idpelanggan sesuai dengan id pelanggan yang digunakan login, $tgl sesuai tanggal hari ini dengan urutan year-month-day, $total sesuai dengan jumlah totalnya. Lalu untuk 3 kolom selain 4 kolom itu tadi diisi dengan 0 untuk kolom bayar, 0 untuk kolom kembali, 0 untuk kolom status. Dengan begitu jika sql sudah terisi maka akan dijalankan oleh db runSQL nya.

4. Setelah data masuk ke dalam tblorder, selanjutnya akan ada rincian dari order tadi di dalam tblorderdetail nya. Sebelum menyambungkan dengan tblorderdetail, file checkout.php bisa kita kosongkan tampilannya dengan menghapus echo yang ada di dalam if isset seperti echo ‘<br>’; echo $total; dan echo idorder(); setelah itu dibawah function insertOrder buat function baru lagi dengan nama insertOrderDetail. Codenya seperti ini

function insertOrderDetail($idorder = 1)

{

    global $db;

    foreach ($\_SESSION as $key => $value) {

        if ($key <> 'pelanggan' && $key <> 'idpelanggan') {

            $id = substr($key, 1);

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

            $row = $db->getALL($sql);

            foreach ($row as $r) {

                $idmenu = $r['idmenu'];

                $harga = $r['harga'];

                $sql = "INSERT INTO tblorderdetail VALUES ('', $idorder, $idmenu, $value, $harga)";

                $db->runSQL($sql);

            }

        }

    }

}

Artinya dia akan memasangkan function orderdetail ini sesuai dengan idorder yang dipilih, maka di dalam function diisi dengan foreach untuk melooping menu yang dipilih oleh pelanggan sesuai session yang digunakan untuk login. Setelah itu data menu yang dipilih dari idmenu dan harga menunya akan diinsert ke dalam tblorderdetail. Yang diinsert ke dalam tblorderdetail sesuai dengan variable di dalam kurung values nya yaitu $idorder yang berarti orang keberapa yang order, $idmenu yang berarti id dari menu nya itu berapa, $value yang berarti berapa jumlah menu yang dipilih itu misalkan nasi goring value nya 3 atau beli 3 porsi, dan $harga berdasarkan harga dari permenunya. Setelah itu kembali lagi ke dalam if isset tambahkan code

if ($count == 0) {

        insertOrder($idorder, $idpelanggan, $tgl, $total);

        insertOrderDetail($idorder);

    } else {

        insertOrderDetail($idorder);

    }

Dan pindahkan insertOrder tadi ke dalam if. Jadi apabila jumlah yang order masih belum ada, maka data order yang akan ditambahkan(baru) akan menginsert mulai dari ke tblorder nya dan ke tblorderdetail nya. Dan apabila jumlah yang order sudah ada sebelumnya atau lebih dari 0 jumlahnya, maka data order yang akan ditambahkan hanya menginsert untuk tblorderdetail nya.

5. Pembuatan Home Page (Front End) pada bagian Mengosongkan Session Cart. Mula-mula buat function baru dengan nama kosongkanSession di dalam file checkout.php, letak function ini setelah function insertOrderDetail. Function isi berisi

function kosongkanSession()

{

    foreach ($\_SESSION as $key => $value) {

        if ($key <> 'pelanggan' && $key <> 'idpelanggan') {

            $id = substr($key, 1);

            unset($\_SESSION['\_' . $id]);

        }

    }

}

Jadi function ini terdapat unset yang berfungsi untuk menonaktifkan atau mengosongkan sessionnya dari id menu yang dipilih pelanggan satu persatu. Setelah membuat function ini kembali ke if isset tambahkan kosongkanSession(); agar function nya berjalan. Dengan begitu ketika pelanggan membeli beberapa menu dan dimasukkan ke keranjang, lalu dia melakukan checkout barang, nah menu-menu yang dipilih ini akan masuk ke dalam tblorder dan tblorderdetail. Sedangkan tabel keranjangnya tadi menjadi kosong karena di unset di dalam function kosongkanSession. Setelah itu pada bagian keranjang ini belum ada menu yang dipilih sama sekali dan masih ada button untuk checkout, karena belum ada barang yang dicheckout maka ketika barang di keranjang kosong button checkoutnya tidak akan tampil. Cara menghilangkan button ini yaitu dengan cara memberi php diantara hyperlink button checkout yang ada di dalam file beli.php. Seperti ini:

<?php

    if (!empty($total)) {

    ?>

        <a class="btn btn-primary" href="?f=home&m=checkout&total=<?php echo $total ?>" role="button">CHECKOUT</a>

    <?php

    }

    ?>

Jadi apabila $total nya tidak kosong atau sudah ada barang yang dimasukkan ke keranjang maka button checkout nya akan muncul dan apabila tidak ada barang maka button nya tidak muncul. Setelah itu tambahkan function di dalam file checkout.php dan letaknya ada dibagian bawah sendiri. Nama function ini yaitu info dan isinya berupa html h4 yang menuliskan TERIMAKASIH SUDAH BERBELANJA. Jika sudah membuat function, selanjutnya kembali ke if isset yang ada di paling atas dari file checkout.php di dalam if isset ini tambahkan header yang berisi location menuju ke file checkout, lalu pada bagian if isset ini biasanya ada if ada else, maka setelah if isset tambahkan else yang berisi info(); atau mengaktifkan function info. Jadi apabila sudah melakukan checkout sebelumnya dan checkout ini berhasil maka akan menampilkan tulisan yang ada di function info.

6. Pembuatan Home Page (Front End) pada bagian Menambahkan Jumlah Pembelian. Jadi disini akan ditambahkan navbar untuk cart dimana cart ini akan menampilkan jumlah baris dari menu yang sudah di beli atau menu yang ada di keranjang, dan ketika ditekan jumlahnya maka dia akan menampilkan isi dari keranjangnya, untuk itu pada bagian hyperlink email yang menyambungkan ke keranjang pada navbar pelanggan dihilangkan hyperlink nya saja, jadi hanya tersisa tulisan emailnya.

7. Untuk menambahkan navbar cart, di dalam file index.php di bagian navbar pelanggan bisa dicopas lalu diletakkan di bawahnya untuk diganti menjadi navbar cart. Lalu hyperlink yang ada di pelanggan dihapus dan untuk navbar cart nya ada perubahan menjadi seperti ini:

<div class="float-right mt-4 mr-4"> Cart : ( <a href="?f=home&m=beli">' . cart() . '</a> )</div>’;

Jadi dia akan berada di sebelah kiri dari navbar pelanggan bertuliskan Cart: lalu tanda kurung yang isinya sesuai dengan function cart dan ketika ditekan isi di dalam kurung nya akan menyambungkan ke folder home di file beli atau keranjangnya. Selanjutnya adalah bagian function cart(), function ini diletakkan di dalam index.php pada bagian php yang paling atas sendiri setelah if isset $\_GET[‘log’]. Function ini diberi nama cart() sesuai dengan yang ada di navbar cart tadi, untuk isinya bisa dicopas dari function insertOrderDetail() yang ada di dalam file checkout.php, setelah copas pada bagian foreach $row as $r isinya dihapus semua dan diganti dengan $cart++; nah $cart ini bisa diisi dan diletakkan dibawah global db nya, isi nya yaitu 0 agar cart ini melooping dari angka 0, setelah itu dibawah foreach nya ketik return $cart; Code cart() nya

function cart()

{

    global $db;

    $cart = 0;

    foreach ($\_SESSION as $key => $value) {

        if ($key <> 'pelanggan' && $key <> 'idpelanggan') {

            $id = substr($key, 1);

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

            $row = $db->getALL($sql);

            foreach ($row as $r) {

                $cart++;

            }

        }

    }

    return $cart;

}

Jadi cart() ini akan melooping jumlah menu yang dipilih oleh pelanggan mulai dari 0 apabila belum ada menu yang dipilih lalu di return $cart yang nantinya ketika function cart dipanggil maka yang keluar adalah return $cart nya yaitu jumlah dari menu yang dipilih oleh pelanggan. Jadi di navbar akan tampil seperti ini Cart: (0) apabila belum ada menu yang dipilih, dan apabila sudah ada menu yang dipilih maka nilai 0 itu akan berubah sesuai jumlah menu yang dipilih. Selanjutnya di dalam tabel keranjang terdapat kolom untuk hapus dan seharusnya apabila satu menu dihapus maka jumlah cart di dalam navbar cart juga ikut berkurang 1, tetapi cart ini akan berkurang 1 apabila harus direfresh dulu, agar apabila ada menu yang dihapus dan otomatis mengurangi jumlah cart, untuk itu di dalam file beli.php pada bagian if isset $\_GET[‘hapus’] di dalamnya ditambahkan header location yang menyambungkan ke folder home di file beli.php, jadi ketika ditekan hapus dia akan kembali ke folder home file beli.php atau sama saja seperti merefresh secara otomatis.

8. Pembuatan Home Page (Front End) pada bagian Menambahkan Histori Pembelian. Histori pelanggan dibuat untuk mengetahui pelanggan telah melakukan pembelian apa saja dari aplikasi restoran ini. Untuk histori ini dibuatkan navbar sendiri yang letaknya berada di sebelah kiri chart, jadi di dalam file index.php ditambahkan div class untuk histori ini yang diletakkan dibawah div class chart, code nya

<div class="float-right mt-4 mr-4"><a href="?f=home&m=histori"> Histori </a></div>

Jadi ketika ditekan Navbar Histori dia akan masuk ke file histori.php yang ada di dalam folder home. Maka dari itu buat file baru yaitu histori.php. Setelah itu atur tblorder untuk digabung dengan tblpelanggan yang nantinya gabungan ini diberi nama vorder (view order). Dan gabung juga tblorderdetail dengan tblmenu yang nantinya gabungan ini diberi nama vorderdetail (view order detail). Ada cara tersendiri di dalam ms.acces yang menggabungkan 2 tbl menjadi 1 lalu diambil sql nya untuk dicopas ke dalam sql di phpmyadmin. Sql ini bisa dicopas melalu link ms.acces yang ada di deskripsi yt ke 50. Jadi ketika sudah mendownload link nya lalu buka di ms.acces dan pilih bagian vorder dulu, ketika sudah memilih vorder di dalam home bisa memilih view yang ada di navbar kiri atas lalu pilih sql dan copy, setelah itu kembali ke phpmyadmin pada bagian tblorder lalu pilih sql, hapus sql yang ada sebelumnya dan ganti dengan sql yang dicopy sebelumnya, sebelum paste tambahkan CREATE VIEW vorder AS baru di paste sql yang dicopy tadi. Begitu juga untuk vorderdetail maka copas sql nya ke dalam sql yang ada di tblorderdetail dengan menambahkan CREATE VIEW vorderdetail AS terlebih dahulu sebelum paste. Jika sudah melakukan penggabungan, di dalam phpmyadmin bisa dilihat ada tambahan view yang isinya vorder dan vorderdetail dan ketika dijelajahi merupakan isi dari penggabungan 2 tabel. Setelah menggabungkan tbl itu tadi, selanjutnya isi file histori.php tadi dengan mengcopas dari file select.php di folder kategori. Pada bagian $jumlahdata yang berisi objek db rowCOUNT, ubah codenya menjadi

$email = $\_SESSION['pelanggan'];

$jumlahdata = $db->rowCOUNT("SELECT idorder FROM vorder WHERE email= '$email' "); Jadi dia akan mengambil jumlah data sesuai dengan jumlah orderan yang sudah diorder pelanggan, pelanggan ini berdasarkan akun yang dibuat untuk login dan diambil dari vorder datanya. Pada bagian $sql nya isi dengan

$sql = "SELECT \* FROM vorder WHERE email = '$email' ORDER BY tglorder DESC LIMIT $mulai, $banyak"; Yang berarti dia akan mengambil data berdasarkan email di dalam vorder dengan mengurutkan dari tanggal order yang terakhir (DESC) ke tanggal yang lama. Setelah itu pada bagian tabelnya th atau judulnya ada 4 yaitu No, Tanggal, Total, dan Detail. Untuk td dari no sama dengan sebelumnya, td tanggal dan seterusnya seperti ini

<td><?php echo $r['tglorder'] ?></td>

<td><?php echo $r['total'] ?></td>

<td><a href="?f=home&m=detail&id=<?php echo $r['idorder'] ?>">Detail</a></td>

Jadi untuk tanggal dia akan menampilkan data tanggal order pelanggan itu, untuk total dia akan menampilkan dari data total harga yang telah diorder pelanggan itu, dan untuk detailnya berupa hyperlink yang menyambungkan ke folder home, file detail.php, dengan id sesuai detail yang dipilih. Ubah judul h3 dari kategori menjadi Histori Pembelian. Untuk paging dalam histori.php berada di paling bawah sendiri yaitu for, di dalam for ini ubah hyperlink pada echo menjadi ke folder home dan file histori.

9. Selanjutnya mengatur pada bagian file detail.php untuk detail dari histori yang telah dilakukan pelanggan. File detail.php bisa dicopas dari file histori.php, pada bagian $jumlah data yang paling atas sendiri codenya:

if (isset($\_GET['id'])) {

    $id = $\_GET['id'];

}

$jumlahdata = $db->rowCOUNT("SELECT idorderdetail FROM vorderdetail WHERE idorder= $id ");

Jadi file ini akan menampilkan jumlah data dari vorderdetail dengan mengambil idorderdetailnya sesuai dengan order yang dipilih untuk dilihat detailnya. Setelah itu dibagian $sql nya untuk bagian WHERE sama dengan WHERE yang ada di $jumlahdata yaitu WHERE idorder=$id; lalu ORDER BY nya idorderdetail ASC, yang artinya dia akan mengurutkan data nya berdasarkan idorderdetailnya dari yang paling awal (ASC). Di dalam tabel pada bagian judul yaitu th, ada 5 bagian untuk No, Tanggal, Menu, Harga, dan Jumlah. Setelah itu pada bagian td nya:

<tr>

<td><?php echo $no++ ?></td>

<td><?php echo $r['tglorder'] ?></td>

<td><?php echo $r['menu'] ?></td>

<td><?php echo $r['harga'] ?></td>

<td><?php echo $r['jumlah'] ?></td>

 </tr>

Jadi dia akan menampilkan data sesuai dengan isi indeksnya. Dengan begitu ketika menekan detail pada tabel histori, dia akan tampil sebuah tabel yang menampilkan rincian dari orderan yang dilihat detailnya. Lalu pada bagian paging dari file ini codenya seperti ini:

for ($i = 1; $i <= $halaman; $i++) {

    echo '<a href="?f=home&m=detail&id=' . $r['idorder'] . '&p=' . $i . '">' . $i . '</a>';

    echo '&nbsp &nbsp &nbsp';

}

Disitu terdapat hyperlink untuk id=' . $r['idorder'] . '. Yang berarti dia akan menampilkan sebuah halaman data yang merupakan atau sesuai dengan detail dari sebuah order yang ingin dilihat detailnya, jadi ketika dia ditekan dihalaman 1 maka yang tampil adalah beberapa baris (sesuai baris yang ditentukan perhalamannya) data detail dari satu order dengan urutan sesuai idorderdetailnya dari yang urutan angka yang paling awalnya. Jadi ketika dia di halaman 2 atau seterusnya dia akan menampilkan baris detail order yang belum tampil di halaman 1. Untuk mengecek apa berfungsi dengan baik tanpa ada erornya, maka bisa logout dan mencoba login dengan akun lain dan kembali ke halaman histori.

**Saya Belum Mengerti**

1.