|  |  |
| --- | --- |
| Materi | Nilai |
| PHP Video 25 – 30 #12 | 90 |

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

1. Menggunakan bantuan bootstrap untuk membuat tampilan web di file index dalam foldder admin
2. Membuat file insert.php dan menyambungkan file insert.php di dalam link pada file select.php.

* Cth : <a href="?f=kategori&m=insert">Tambah Data</a>

Artinya, pada saat di klik, otomatis file tersebut akan masuk ke dalam isi dari insert.php, karena ?f&m sudah disambungkan dengan require\_once pada file index.php . ORDER BY adalah fungsi mysql yang berguna untuk mengurutkan data.

1. Keyword ASC pada ORDER BY digunakan untuk mengurutkan data secara ascending ( dari kecil ke besar ).
2. Keyword DESC pada ORDER BY digunakan untuk mengurutkan data secara descending ( dari besar ke kecil ).
3. Apabila tidak diberi keyword ASC / DESC maka secara otomatis akan menggunakan keyword ASC, yaitu mengurutkan dari kecil ke besar.
4. Menggunakan dan menggabungkan fungsi SELECT dan ORDER BY pada mysql untuk memilih data dan mengurutkan data.

* Cth : $sql = "SELECT \* FROM tblkategori ORDER BY kategori ASC";

Artinya, pada $sql terdapat perintah untuk menselect data pada tblkategori dan mengurutkan data pada kategori dengan urutan dari ASC / dari kecil ke besar.

1. Memanggil objek $db->getALL agar data yang dipilih dapat dijalankan, dari file dbcontroller

* Cth : $sql = "SELECT \* FROM tblkategori ORDER BY kategori ASC";

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

Artinya, pada $sql terdapat perintah untuk memilih data dari tblkategori dan mengurutkan data pada kategori dengan urutan ASC / urut dari nilai kecil ke nilai besar, lalu $row, memanggil objek dengan function getALL untuk menjalankan fungsi $sql.

1. Membuat tabel di dalam php dengan menggunakan <?php foreach($row as $r): ?>, dan mengakhirinya dengan menggunakan <?php end foreach ?>
2. Memasukkan data yang ada pada html / tabel kategori dengan menggunakan variabel $r dan menambahkan <?php?> di dalam <td> tabel.

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

<tr>

<td><?php echo $r['kategori'] ?></td>

</tr>

<?php endforeach ?> Mengedit table dengan menggunakan class yang disediakan oleh bootstrap, yaitu table table-bordered.

1. Mengedit tampilan tabel dengan mengatur width / lebar pada class.

* Cth : <table class="table table-bordered w-75">

1. Paging pada database yaitu berfungsi menampilkan data dengan ketentuan beberapa data yang ditampilkan / data ditampilkan berpotong - potong, karena apabila data yang dibuat mencapai ribuan maka apabila ditampilkan semua akan memakan waktu lama untuk scrolling, maka dari itu data dipotong - potong penampilannya.
2. Function rowCOUNT berfungsi untuk menghitung berapa baris yang ada di dalam data, di dalam rowCOUNT terdapat perintah mysqli\_num\_rows untuk mengetahui total jumlah baris yang ada di dalam data.
3. Menambahkan $jumlahdata yang berisi function rowCOUNT.

* Cth : $jumlahdata = $db->rowCOUNT("SELECT idkategori FROM tblkategori");

1. Ceil pada PHP berfungsi untuk membulatkan data ke atas, karena pembagian pada data tidak selalu menghasilkan angka bulat, maka dari itu diperlukan pembulatan angka ke atas.

* Cth : $halaman = ceil($jumlahdata / $banyak);

Artinya, pada $halaman terdapat pembagian antara $jumlahdata dan $banyak, dan hasilnya akan dibulatkan karena menggunakan ceil.

1. Menampilkan $halaman menggunakan for.

* Cth : for ($i=1; $i <= $halaman; $i++) {

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

echo '&nbsp &nbsp &nbsp';

}

Artinya, terdapat perulangan $i yang dimulai dari 1 sampai <=$halaman, i akan mendapatkan penambahan. Terdapat hyperlink yang berisi $f=kategori&m=select&p, p berisi nilai link pada URL, &nbsp berarti untuk penambahan spasi.

1. &nbsp adalah fungsi html yang berarti non-breaking space, jadi apabila memberikan fungsi ini maka akan terjadi penambahan spasi.
2. Untuk mengurutkan nomor di dalam tabel, maka menggunakan if untuk mengatur nya.

* Cth : if (isset($\_GET['p'])) {

$p = $\_GET['p'];

$mulai = ($p \* $banyak) - $banyak;

}else {

$mulai = 0;

}

Artinya, apabila $\_GET['p'] yang disimpan di dalam $p di klik, maka di dalam $mulai, $p akan dieksekusi dengan dikalikan \* $banyak lalu hasilnya dikurangi dengan \*banyak, maka setiap hyperlink yang diisi, akan memunculkan data yang berawal dari hasil operasi aritmatika tersebut. Cth : apabila nomor hyperlink yang di klik adalah nomor 3, maka 3\*3-3 = 6, jadi apabila di klik link nomor 3, maka akan ditampilkan data dari nomor 7 - 9.

1. Untuk menentukan berapa data yang ditampilkan, maka menggunakan fungsi LIMIT pada $sql.

* Cth : $sql = "SELECT \* FROM tblkategori ORDER BY kategori ASC LIMIT $mulai,$banyak";

menampilkan data dari fungsi LIMIT berawal dari $mulai, dan yang ditampilkan sebanyak $banyak, pada code di atas, $banyak adalah 3, jadi setiap link akan ditampilkan sebanyak 3 data.

1. Menyambungkan file select.php ke dalam insert.php dengan menggunakan link apabila button di klik.

* Cth : <a class="btn btn-primary" href="?f=kategori&m=insert" role="button">TAMBAH DATA</a>

di dalam button tersebut ditautkan link untuk insert.php, karena memanggil ?f=kategori&m=insert, yang sudah diatur apabila $\_GET['f'] dan $\_GET['m'] diklik maka otomatis akan menyambung pada ."./".$f."/".$m.".php", jadi terdapat file insert.php pada folder kategori.

1. Placeholder pada <input> berfungsi sebagai teks pemberitahuan di dalam kotak <input>.
2. Form-control digunakan untuk membuat panjang kotak form menjadi panjang browser / yang sudah diatur.
3. Penulisan perintah dari insert adalah dengan menggunakan INSERT INTO nama\_tabel VALUES(' ',' nama\_data\_yang\_ditambah');

* Cth : $sql = "INSERT INTO tblkategori VALUES('','$kategori')";

Artinya, di dalam $sql terdapat perintah untuk menambahkan data yait dengan INSERT INTO data di dalam tblkategori dengan VALUES / hasil (' ',$kategori) yang ada di dalam $kategori, pada values terdapat 2 pasang tanda petik, 1 pasang lain dikosongi karena terdapat 2 kolom pada tabel tersebut.

1. Menyambungkan data yang disimpan ke dalam database dengan menggunakan perintah function runSQL.

* Cth : $db->runSQL($sql);

Artinya, terdapat objek $db yang menjalankan perintah runSQL dan memasukkan data yang ditambahkan ke dalam database.

1. Header berfungsi untuk menampilkan halaman web yang ditautkan.
2. Mengembalikan tampilan web awal setelah terjadi insert data menggunakan header.

* Cth : header("location:?f=kategori&m=select");

Artinya, apabila selesai menambahkan data, maka akan kembali ke halaman ?f=kategori&m=select.

1. Membuat file update.php untuk tempat update data.
2. Menambahkan hyperlink yang berisi ?f=kategori&m=update&id.

* Cth : <td><a href="?f=kategori&m=update&id=<?php echo $r['idkategori'] ?>">Update</a></td>

Artinya, di dalam <td> Update terdapat hyperlink yang berisi ?f=kategori&m=update&id di dalam <php> yang membaca $r['idkategori'];

1. Menampilkan nilai id dengan menggunakan if.

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

$id = $\_GET['id'];

echo $id;

}

Artinya, isi dari $\_GET['id'] disimpan di dalam $id, dan apabila $\_GET['id'] di klik maka akan memunculkan nilai id.

1. Menambahkan perintah menampilkan nilai $id menggunakan SELECT.

* Cth : $sql = "SELECT \* FROM tblkategori WHERE idkategori=$id";

Artinya, di dalam $sql terdapat perintah SELECT untuk memilih data pada tblkategori yang ada dikategori urutan sesuai $id.

1. Untuk menjalankan perintah SELECT agar dapat menampilkan nilai $id, maka memanggil fungsi yang sudah dibuat di dalam objek $db.

* Cth : $row = $db->getITEM($sql);

Artinya, objek $db memanggil function getITEM untuk menjalankan $sql dan disimpan di dalam $row.

1. Menjalankan perintah UPDATE menggunakan if.

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

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

$sql = "UPDATE tblkategori SET kategori='$kategori' WHERE idkategori=$id";

$db->runSQL($sql);

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

}

Artinya, apabila $\_POST['simpan'] di klik, maka akan menjalankan $\_POST['simpan'] disimpan di dalam $kategori, dan terdapat $sql yang berisi perintah UPDATE tblkategori SET untuk mengatur kategori sesuai dengan $kategori terletak di idkategori sesuai dengan $id, menjalankan perintah runSQL($sql) di dalam objek $db dan jika selesai mengupdate data, maka akan kembali ke halaman awal yaitu ?f=kategori&m=select.

1. Membuat file delete.php yang berisi perintah untuk delte data.
2. Menambahkan hyperlink yang berisi ?f=kategori&m=delete&id.

* Cth : <td><a href="?f=kategori&m=delete&id=<?php echo $r['idkategori'] ?>">Delete</a></td>

Artinya, di dalam <td> Delete terdapat hyperlink yang berisi ?f=kategori&m=delete&id di dalam <php> yang membaca $r['idkategori'];

1. Menampilkan nilai $sql dengan menggunakan if.

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

$id = $\_GET['id'];

$sql = "DELETE FROM tblkategori WHERE idkategori=$id";

echo $sql;

}

Artinya, isi dari $\_GET['id'] disimpan di dalam $id, dan apabila $\_GET['id'] di klik maka akan menjalankan $sql yang terdapat perintah untuk DELETE data, lalu ditampilkan menggunakan echo.

1. Menambahkan perintah runSQL untuk menjalankan DELETE.

* Cth : $db->runSQL($sql);

Artinya, di dalam objek $db terdapat function runSQL dan menjalankan $sql.

1. header yang berisi link ke dalam halaman awal.

* Cth : header("location:?f=kategori&m=select");

Artinya, apabila selesai DELETE data, maka akan kembali ke halaman awal yaitu ?f=kategori&m=select.

1. Selected pada option digunakan untuk mengunci list yang terakhir kali digunakan sebelum di refresh, maka apabila list lain di klik, akan selalu kembali pada list terakhir yang diklik.

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

Artinya, pada option yang berisi $r['kategori'] terdapat perintah selected, yaitu untuk mengunci list terakhir yang di klik pada $['kategori'].

1. CTRL + SHIFT + L berfungsi untuk mengganti semua kata yang sama seperti kata yang di blok, jadi hanya mengubah satu kata dan kata lain yang otomatis ikut berubah. Mengganti semua kata kategori menjadi menu.
2. select di dalam form digunakan untuk membuat objek form menjadi list yang dapat dipilih.
3. option pada form digunakan untuk menampilkan list list yang dapat dipilih.
4. Menuliskan perintah fungsi SELECT di dalam satu variabel dengan function $db, tidak usah membuat variabel lain untuk menyimpan perintah fungsi.

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

Artinya, pada $row terdapat objek $db yang di dalamnya berisi getALL untuk memilih data dari tblkategori yang data pada kategori diurutkan secara ASC / dari nilai kecil ke besar dengan menggunakan ORDER BY.

1. Menampilkan nilai dari $row menggunakan var\_dump, agar mengetahui tipe data, pada tabel apa, dan nilai data.
2. Memasukkan semua data kategori ke dalam option menggunakan foreach untuk menampilkan nilai perulangan dengan menggunakan array.

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

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

<?php endforeach ?>

Artinya, terdapat foreach yang melakukan perulangan pada nilai $r['kategori'] dalam option, jadi semua data pada $r['kategori'] akan muncul di dalam option sebagai list.

1. Onchange pada form berfungsi untuk mengirimkan file secara auto-kirim ketika file yang akan dikirim telah dipilih.

* Cth : <select name="opsi" id="" onchange="this.form.submit()">

Artinya, terdapat select yang mempunyai nama opsi dan ketika file yang ada di dalam select telah dipilih, maka file otomatis akan dipilih karena menggunakan onchange="this.form.submit()" .

1. Memanggil $r['idkategori'] dan menambahkannya sebagai value di dalam option yang berisi $r['kategori'] sebagai list, agar ketika salah satu list kategori diklik, maka akan muncul nomor id kategori nya.

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

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

<?php endforeach ?>

Artinya, terdapat foreach untuk melakukan perulangan, dan sebagai key adalah $row sama dengan $r, di dalam foreach terdapat option dengan value $r['idkategori'] yang akan muncul nilai idkategori ketika $r['kategori'] di klik dan diatur menggunakan if.

1. Mengatur if, jika name dari select di klik maka akan muncul nomor dari idkategori.

* Cth : <?php

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

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

echo $opsi;

}

?>

Artinya, terdapat if yang berisi apabila $\_POST['opsi'] di klik maka $\_POST['opsi'] akan disimpan ke dalam $opsi, dan $opsi ditampilkan menggunakan echo.

1. Untuk mengatur agar ketika list yang di klik terkunci dan tidak kembali pada list yang terakhir diklik sebelum halaman di refresh, maka dengan menambahkan php yang berisi if sebelum value pada option.

* Cth : <option <?php if($r['idkategori']==$opsi) echo "selected" ?> value="<?php echo $r['idkategori'] ?>">

<?php echo $r['kategori'] ?>

</option>

Artinya, di dalam option terdapat if yang berisi $r['idkategori']==$opsi dan dikunci sesuai pada list yang di klik dengan value $r['idkategori'] dan isi dari option adalah $r['kategori'].

1. Jika pertama kali mengklik menu, maka akan terjadi error karena $opsi belum terisi, maka untuk mengatasi error tersebut, dengan menambahkan else dibawah if yang berisi $opsi.

* Cth : }else{

$opsi = 0;

}

Artinya, jika if belum diisi, maka dia akan menjalankan else yang berisi $opsi 0.

1. Untuk menampilkan lokasi dimana urutan idkategori jika list di klik dengan membuat variabel baru $where.

* Cth : $where = "WHERE idkategori = $opsi";

Artinya, $where berisi perintah untuk menampilkan idkategori dengan nilai yang sama seperti $opsi.

1. Agar tabel menampilkan menu sesuai dengan data menu di dalam kategori maka di dalam tabel pada $sql diberi tambahan $where sebagai perintah.

* Cth : $sql = "SELECT \* FROM tblmenu $where ORDER BY menu ASC LIMIT $mulai,$banyak";

Artinya, di dalam $sql terdapat perintah untuk memilih data dari tblmenu dan $where pada menu yang diurutkan dengan urutan dari kecil ke besar dihitung dari $mulai sebanyak $banyak.

1. Setelah mengatur function nya, maka selanjutnya mengatur tabel yagn digunakan untuk menampilkan data, agar apabila data kosong tidak terjadi error.

* Cth : if(!empty($row)){ ?>

1. Memasukkan gambar di dalam tabel dengan menggunakan link <img>dan menuliskan tempat folder dimana sehingga folder dapat terbaca.

* Cth : <td><img style="width:50px " src="../upload/<?php echo $r['gambar'] ?>" alt=""></td>
* Untuk mengatur tampilan gambar, pertama gambar harus ditampilkan terlebih dahulu agar gampang untuk diatur, apabila mengatur dulu tanpa gambar, maka akan susah

**Saya Belum Mengerti**