

Modul 6 : MySQLi Object

KOMPETENSI :

Setelah menyelesaikan praktikum pada Modul 6, Praktikan diharapkan mampu memahami dasar penggunaan object MySQLi pada PHP untuk melakukan transaksi kedalam database.

TUJUAN INSTRUKSIONAL KHUSUS :

1. Praktikan dapat memahami tentang MySQLi Object.
2. Praktikan dapat memahami dan mengaplikasikan penggunaan method pada object MySQLi.

TEORI DASAR:

1. MySQLi Object

PHP MySQLi (PHP MySQL Improved) merupakan metode yang dapat digunakan untuk melakukan koneksi kedalam database MySQL yang didalamnya terdapat serangkaian fungsi yang dapat digunakan untuk melakukan operasi atau transaksi kedalam database. Untuk dapat menggunakan MySQLi, ekstensi MySQLi perlu diaktifkan pada PHP yang pada dasarnya telah aktif begitu PHP di install. Selain metode MySQLi terdapat metode lain untuk melakukan transaksi kedalam database diantaranya PDO (PHP Data Object) dan MySQL (Struktural). Namun yang digunakan adalah MySQLi dengan Object Oriented Style. Pada pemrograman PHP menggunakan paradigma Object Oriented Programming, untuk memanggil method atau property pada sebuah object digunakan notasi panah (->) pada objectnya. Sebagai perbandingan berikut perbedaan penggunaan object pada Java dan PHP.

PHP	Java
<pre><?php class Halo { public \$text; public function hello() { return "Hello World!"; } } \$obj = new Halo(); \$obj->text = 'PHP'; \$obj->hello(); echo \$obj->text;</pre>	<pre>public class Halo { String text; public String hello() { return "Hello World!"; } public static void main(String[] args) { Halo obj = new Halo(); obj.text = 'Java'; obj.hello(); System.out.println(obj.text); } }</pre>

1) Struktur Object MySQLi

Untuk membuat object koneksi dengan MySQLi sangat sederhana dengan membuat sebuah object dari kelas MySQLi dengan 4 parameter didalamnya secara berurut yaitu alamat Host Server MySQL, Username MySQL, Password MySQL dan Nama Database. Untuk melakukan pengecekan keberhasilan koneksi dapat dilakukan dengan menggunakan properti "connect_error" pada MySQLi. Berikut ini bentuk umum koneksi database dengan MySQLi beserta pengecekan koneksi.

Syntax

```
<?php
$host = 'localhost';
$user = 'root';
$pass = '';
$dbnm = 'nama_database';
$connection = new
mysqli($host,$user,$pass,$dbnm); if($connection-
>connect_errno > 0) {
    die('Unable to Connect to Database ['. $connection->connect_error. ']');
```

2) Transaksi Database

Untuk melakukan transaksi kedalam database dengan MySQLi Object baik itu operasi Insert, Update, Delete dan Select, dapat menggunakan method query() pada object Koneksi yang telah dibuat. Nilai kembali dari method ini adalah boolean. Dimana, apabila Query berhasil maka akan

mengembalikan nilai true dan sebaliknya. Berikut ini contoh Query yang dapat dilakukan.

Syntax

```
<?php
require_once 'koneksi.php';

$sql = "SELECT * FROM Nama_Table";
if($connection->query($sql)) {
    echo "<script>alert('Query Sukses');</script>";
} else {
    echo "<script>alert('Query Gagal!');</script>";
}
```

Khusus untuk perintah Select pada MySQL, terdapat method beberapa method yang dapat digunakan untuk melakukan pengambilan data pada database, method ini umumnya dikenal dengan method fetch. Beberapa yang umum digunakan untuk melakukan fetch data adalah method fetch_array(), fetch_assoc() dan fetch_object(). Berikut ini contoh penggunaan method fetch_object() untuk mengambil data dari table dalam database.

Syntax

```
<?php

require_once 'koneksi.php';

$sql = "SELECT * FROM Nama_Table";
if($result = $connection->query($sql)) {
    while($row = $result->fetch_object()) {
        echo $row->col1."<br>";
        echo $row->coln."<br>";
    }
} else {
    echo "<script>alert('Query Gagal!');</script>";
}
```

Perulangan while digunakan untuk melakukan perulangan hasil fetch data dimana dibuat sebuah object untuk menampung hasil pengambilan data dari query. Hal ini disebabkan perulangan while yang flexible terhadap kondisi dari segi kuantitas data yang belum diketahui jumlahnya didalam database.

PRAKTIKUM MANDIRI

1. Peralatan

Perangkat komputer / PC / Laptop / Notebook .
Sistem operasi Windows / Linux (optional Mac OS)
Aplikasi Visual Studio Code / Atom / SublimeText / phpStorm IDE
Web Server & MySQL Server (XAMPP)

2. Prosedur

- Baca dan pahami semua tahapan praktikum dengan cermat.
- Gunakan fasilitas yang disediakan dengan penuh rasa tanggung jawab.
- Rapikan kembali setelah menggunakan komputer (mouse, keyboard, kursi, dll).
- Perhatikan sikap anda untuk tidak mengganggu rekan praktikan lain.
- Pastikan diri anda tidak menyentuh sumber listrik.

3. Kegiatan Praktikum

- Buka texteditor / IDE yang akan digunakan.
- Jalankan Server XAMPP.

- 3) Buatlah sebuah database dengan nama 'web_XXX' XXX diganti dengan 3 digit stambuk terakhir.
- 4) Buatlah sebuah table dengan nama login yang didalamnya minimal terdapat 2 kolom yaitu username dan password.
- 5) Pada File Proses.php Praktikum Mandiri Modul 5, ubahlah proses login tersebut dengan pengecekan data username dan password yang telah dibuat pada database.
- 6) Setelah percobaan selesai, tutup semua perangkat lunak yang telah digunakan.
- 7) Matikan PC dan Rapihkan meja praktikum.

EVALUASI PRAKTIKUM

1. Pada Tugas Evaluasi Modul 5, lanjutkan dengan buatlah sebuah database dan sebuah table yang dapat menampung data-data di atas yang kemudian pada file proses di atas dapat melakukan operasi CRUD (Create, Retrieval, Read & Update) pada database.