

Nama : Vebie Yoseva Theresia Pasaribu

NIM : 121140016

Pemrograman Web- RA

```
<?php
$host = "localhost";
$username = "root";
$password = "";
$database = "tugasdatamahasiswa";
$koneksi = new mysqli($host, $username, $password, $database);

if ($koneksi->connect_error) {
    die("Koneksi database gagal: " . $koneksi->connect_error);
}

if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['tambah'])) {
    $nim = $_POST['nim'];
    $nama = $_POST['nama'];
    $prodi = $_POST['prodi'];

    $query_tambah = "INSERT INTO datamhs (nim, nama, prodi) VALUES ('$nim', '$nama', '$prodi')
ON DUPLICATE KEY UPDATE nama = '$nama', prodi = '$prodi'";

    if ($koneksi->query($query_tambah) === TRUE) {
        echo "Data mahasiswa berhasil ditambahkan.";
    } else {
        echo "Error: " . $query_tambah . "<br>" . $koneksi->error;
    }
}
?>
```

Nama : Vebrie Yoseva

NIM : 121140016

PROGARAMAN WEB - RA

```
* $host = "localhost";  
* $username = "root";  
* $password = "";  
* $database = "tugasdata mahasiswa";  
* $koneksi = new mysqli($host, $username, $password, $database);
```

→ Untuk mendefinisikan variable \$host, \$username, \$password, \$database

→ \$koneksi = new mysqli(\$host, \$username, \$password, \$database) untuk membuat koneksi baru 'mysqli' dan disimpan dalam variabel \$koneksi.

```
if ($koneksi->connect_error) {  
    die("Koneksi database gagal ". $koneksi->connect_error);  
}
```

→ Untuk memeriksa apakah koneksi ke database berhasil atau tidak. Jika terjadi kesalahan dalam koneksi, pesan kesalahan akan dicetak dan program akan dihentikan.

```
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['tambah'])) {  
    $nim = $_POST['nim'];  
    $nama = $_POST['nama'];  
    $prodi = $_POST['prodi'];
```

→ untuk memeriksa request yang berasal dari metode POST dan memeriksa tombol tambah. Dimana jika data berasal dari POST, data dari formulir HTML akan diambil dan disimpan dalam variabel.

```
$query_tambah = "INSERT INTO datamhs (nim, nama, prodi) VALUES ('$nim', '$nama', '$prodi')  
ON DUPLICATE KEY UPDATE nama = '$nama', prodi = '$prodi'";
```

→ Untuk membuat query SQL untuk menambahkan data ke dalam tabel datamhs

```
if ($koneksi->query($query_tambah) === TRUE) {  
    echo "Data Mahasiswa berhasil ditambahkan.";  
} else {  
    echo "Error: ". $query_tambah. "<br>". $koneksi->error;  
}
```

→ menjalankan query SQL dengan menggunakan metode 'query()'.

→ Jika berhasil, pesan berhasil akan dicetak dan jika gagal, pesan kesalahan akan dicetak.

```

<?php
$query_tampilkan = "SELECT * FROM datamhs";
$result = $koneksi->query($query_tampilkan);

if ($result->num_rows > 0) {
    echo "<table border='1'>";
    echo "<tr><th>NIM</th><th>Nama</th><th>Program Studi</th><th>Aksi</th></tr>";
    while ($row = $result->fetch_assoc()) {
        echo "<tr>";
        echo "<td>".$row['nim']. "</td>";
        echo "<td>".$row['nama']. "</td>";
        echo "<td>".$row['prodi']. "</td>";
        echo "<td><a href='editdata.php?nim=".$row['nim']."'>Edit</a> | <a href='hapusdata.php?nim=".$row['nim']."'>Hapus</a></td>";
        echo "</tr>";
    }
    echo "</table>";
} else {
    echo "<p>Tidak ada data mahasiswa.</p>";
}
?>

```

• \$query_tampilkan = "SELECT * FROM datamhs";
→ menampilkan seluruh data dari tabel datamhs.

\$result = \$koneksi → query(\$query_tampilkan);
→ menjalankan query dan menyimpan dalam variabel '\$result'.

if (\$result → num_rows > 0) {
→ memeriksa apakah hasil query mengembalikan lebih dari 0 baris.

echo "<table border='1'>";
echo "<tr><th>NIM </th><th>Nama </th><th>Program Studi </th><th>Aksi </th></tr>";
→ membuat sebuah tabel dengan border atau garis penggaris 1.
→ membuat kolom yang bernama NIM, Nama, Program Studi, dan Aksi.

while (\$row = \$result → fetch_assoc()) {
→ memulai loop while yang akan berjalan selama masih ada baris data yang diambil dari hasil query.
→ setiap baris data disimpan dalam variabel '\$row'.

echo "<tr>";
echo "<td>". \$row['nim']. "</td>";
echo "<td>". \$row['nama']. "</td>";
echo "<td>". \$row['prodi']. "</td>";
echo "<td>Edit | Hapus</td>";
echo "</tr>";
→ Mencetak setiap baris data kedalam tabel HTML.

→ setiap nilai kolom dicetak dalam tabel serta ditambahkan kolom Aksi dengan dua hyperlink, yaitu untuk mengedit data dan menghapus data dengan NIM sebagai parameter.

echo "</table>";
→ sebagai penutup setelah seluruh data ditampilkan.

} else {
echo "<p>Tidak ada data mahasiswa.</p>";
→ menampilkan teks pesan, jika hasil query tidak mengembalikan baris data / tidak terdapat data mahasiswa.

```

<?php
$query_program_studi = "SELECT DISTINCT prodi FROM datamhs";
$result_program_studi = $koneksi->query($query_program_studi);

while ($row = $result_program_studi->fetch_assoc()) {
    echo "<option value='" . $row['prodi'] . "'>" . $row['prodi'] . "</option>";
}
?>

```

• \$query-program-studi = "SELECT DISTINCT prodi FROM datamhs";
 → mendefinisikan variabel \$query-program-studi.
 → untuk memilih nilai unik dari kolom prodi pada tabel datamhs
 → DISTINCT digunakan untuk memastikan hanya nilai unik yang dikembalikan.
 \$result-program-studi = \$koneksi->query(\$query-program-studi);
 → menjalankan query sql dan disimpan dalam variabel \$result-program-studi.
 Hasil dari query ini berisi daftar program studi yang unik.
 while (\$row = \$result-program-studi->fetch_assoc()) {
 → memulai loop 'while' yang akan berjalan selama masih ada baris data yang diambil dari hasil query. Setiap baris data disimpan dalam variabel \$row.
 echo "<option value='" . \$row['prodi'] . "'>" . \$row['prodi'] . "</option>";
 → setiap baris data digunakan untuk membuat elemen <option> dalam HTML.
 → setiap nilai unik dari kolom 'prodi' menjadi nilai dari atribut 'value' dan juga teks yang ditampilkan di dalam elemen <option>.
 → digunakan untuk membuat pilihan pada elemen <select> di formulir HTML.
 }


```

<?php
if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['cari']) && isset($_GET['prodi'])) {
    $prodi = $koneksi->real_escape_string($_GET['prodi']);

    $query_cari = "SELECT * FROM datamhs WHERE prodi = '$prodi'";
    $result_cari = $koneksi->query($query_cari);

    if ($result_cari->num_rows > 0) {
        echo "<h2>Hasil Pencarian</h2>";
        echo "<table border='1'>";
        echo "<tr><th>NIM</th><th>Nama</th><th>Program Studi</th></tr>";
        while ($row = $result_cari->fetch_assoc()) {
            echo "<tr>";
            echo "<td>". $row['nim']. "</td>";
            echo "<td>". $row['nama']. "</td>";
            echo "<td>". $row['prodi']. "</td>";
            echo "<td><a href='editdata.php?nim=".$row['nim']."'>Edit</a> | <a href='hapusdata.php?nim=".$row['nim']."'>Hapus</a></td>";
            echo "</tr>";
        }
        echo "</table>";
    } else {
        echo "<p>Tidak ditemukan data mahasiswa pada program studi '$prodi'.</p>";
    }
}

$koneksi->close();
?>

```

4. SELECT di formulir HTML.

- memulai blok kode jika request berasal dari metode GET, dan terdapat parameter 'cari' dan 'prodi' dalam URL.
- \$prodi = \$koneksi->real_escape_string(\$_GET['prodi']);
- mengambil nilai dari parameter 'prodi' yang dikirimkan melalui URL menggunakan \$GET dan membersihkan nilai tersebut dengan 'real_escape_string' untuk mencegah serangan SQL injection.
- \$query_cari = "SELECT * FROM datamhs WHERE prodi = '\$prodi'";
- \$result_cari = \$koneksi->query(\$query_cari);
- query SQL akan mengambil data mahasiswa dengan program studi tertentu.
- hasil query disimpan dalam variabel '\$result_cari'.
- if (\$result_cari->num_rows > 0) {
- echo "<h2>Hasil Pencarian</h2>";
- echo "<table border='1'>";
- echo "<tr><th>NIM</th><th>Nama</th><th>Program Studi</th></tr>";
- memeriksa apakah hasil query ~~men~~mengembalikan lebih dari 0 baris.
- jika ya, maka akan menampilkan judul "Hasil pencarian" dan memulai tabel HTML.
- echo "</table>";
- sebagai penutup setelah seluruh data ditampilkan.
- } else {
- echo "<p>Tidak ditemukan data mahasiswa pada program studi '\$prodi'.</p>";
- }
- jika hasil query tidak mengembalikan baris, maka akan mencetak teks pesan data tidak ditemukan.

```
if ($koneksi->connect_error) {  
    die("Koneksi database gagal: " . $koneksi->connect_error);  
}  
  
if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['nim'])) {  
    $nim = $_GET['nim'];  
  
    $query_edit = "SELECT * FROM datamhs WHERE nim = '$nim'";  
    $result_edit = $koneksi->query($query_edit);  
  
    if ($result_edit->num_rows > 0) {  
        $row = $result_edit->fetch_assoc();  
        $nim_edit = $row['nim'];  
        $nama = $row['nama'];  
        $prodi = $row['prodi'];  
    } else {  
        echo "Data mahasiswa tidak ditemukan.";  
        exit();  
    }  
}
```

```
if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['nim'])) {
```

→ akan memeriksa apakah request berasal dari metode GET dan apakah terdapat parameter 'nim' dalam URL.

```
$nim = $_GET['nim'];
```

→ mengambil nilai parameter 'nim' dari URL menggunakan \$_GET dan menyimpannya dalam variabel \$nim.

```
$query_edit = "SELECT * FROM datamhs WHERE nim = '$nim'";
```

→ Query SQL mengambil data mahasiswa berdasarkan NIM yang diberikan.

→ hasil query disimpan dalam variabel 'result_edit'.

```
if ($result_edit->num_rows > 0) {
```

```
    $row = $result_edit->fetch_assoc();
```

```
    $nim_edit = $row['nim'];
```

```
    $nama = $row['nama'];
```

```
    $prodi = $row['prodi'];
```

```
} else {
```

```
    echo "Data Mahasiswa tidak ditemukan.";
```

```
    exit();
```

```
}
```

→ memeriksa apakah hasil query mengembalikan lebih dari 0 baris.

→ Jika ya, data mahasiswa dengan NIM yang diberikan tdk ditemukan.

→ Nilai dari setiap kolom disimpan dalam variabel masing-masing.

→ Jika tidak, maka akan menampilkan pesan kesalahan dan program akan dihentikan.


```

if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['update'])) {
    $nim_edit = $_POST['nim_edit'];
    $nim_baru = $_POST['nim'];
    $nama = $_POST['nama'];
    $prodi = $_POST['prodi'];

    $query_check_nim = "SELECT * FROM datamhs WHERE nim = '$nim_baru'";
    $result_check_nim = $koneksi->query($query_check_nim);

    if ($result_check_nim->num_rows > 0 && $nim_baru != $nim_edit) {
        echo "NIM sudah ada dalam database. Harap gunakan NIM yang berbeda.";
    } else {

        $query_update = "UPDATE datamhs SET nim = '$nim_baru', nama = '$nama', prodi = '$prodi' WHERE nim = '$nim_edit'";

        if ($koneksi->query($query_update) === TRUE) {
            echo "Data mahasiswa dengan NIM $nim_edit berhasil diperbarui.";

            header("Location: index.php");
            exit();
        } else {
            echo "Error: " . $query_update . "<br>" . $koneksi->error;
        }
    }
}
?>

```

Program akan

```

if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['update'])) {
    $nim_edit = $_POST['nim_edit'];
    $nim_baru = $_POST['nim'];
    $nama = $_POST['nama'];
    $prodi = $_POST['prodi'];

    → memeriksa apakah request berasal dari metode POST dan apakah terdapat parameter 'update'.
    → Jika ya, maka nilai dan variabel yang diambil dari formulir disimpan dalam variabel masing-masing.

    $query_check_nim = "SELECT * FROM datamhs WHERE nim = '$nim_baru'";
    $result_check_nim = $koneksi->query($query_check_nim);
    if ($result_check_nim->num_rows > 0 && $nim_baru != $nim_edit) {
        echo "NIM sudah ada dalam database. Harap gunakan NIM yang berbeda.";
    }
    → memeriksa apakah NIM baru yang diinginkan sudah ada di database.
    → Jika ya dan NIM baru tidak sama dengan NIM yang sedang diubah ($nim_edit), maka akan menampilkan pesan kesalahan.
    else {
        $query_update = "UPDATE datamhs SET nim = '$nim_baru', nama = '$nama', prodi = '$prodi' WHERE nim = '$nim_edit'";
        if ($koneksi->query($query_update) === TRUE) {
            echo "Data mahasiswa dengan NIM $nim_edit berhasil diperbarui.";
            header("Location: index.php");
            exit();
        } else {
            echo "Error: " . $query_update . "<br>" . $koneksi->error;
        }
    }
    → memperbaiki data dengan NIM yang telah diubah.
    → Jika berhasil akan dikembalikan ke halaman index.php.

```

```
<?php

$host = "localhost";
$username = "root";
$password = "";
$database = "tugasdatamahasiswa";
$koneksi = new mysqli('localhost', 'root', '', 'tugasdatamahasiswa');

if ($koneksi->connect_error) {
    die("Koneksi database gagal: " . $koneksi->connect_error);
}

if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['nim'])) {
    $nim = $_GET['nim'];

    $query_hapus = "DELETE FROM datamhs WHERE nim = '$nim'";

    if ($koneksi->query($query_hapus) === TRUE) {
        header("Location: index.php");
        exit();
    } else {
        echo "Error: " . $query_hapus . "<br>" . $koneksi->error;
    }
}

?>
```

```
if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['nim'])) {
```

→ memeriksa apakah request berasal dari metode GET dan apakah terdapat parameter nim dalam URL. guna untuk menghapus data dengan NIM tertentu.

```
$nim = $_GET['nim'];
```

→ mengambil nilai dari parameter nim dari URL dengan menggunakan \$_GET dan menyimpan dalam variabel \$nim.

```
$query_hapus = "DELETE FROM datamhs WHERE nim = '$nim'";
```

→ query SQL akan menghapus data mahasiswa berdasarkan NIM yang diberikan

```
if ($koneksi->query($query_hapus) === TRUE) {  
    header("Location: index.php");  
    exit();  
}
```

```
} else {
```

```
    echo "Error : ". $query_hapus. "<br>". $koneksi->error;  
}
```

→ menjalankan query SQL untuk menghapus data mahasiswa.

→ Jika berhasil dijalankan, maka program akan melakukan redirect ke halaman utama index.php dengan menggunakan header ("Location: index.php") dan akan keluar dari script dgn exit().

→ Jika tidak, maka akan menampilkan pesan kesalahan.