



LẬP TRÌNH PHP

MySQL

Th.S Nguyễn Đình Hoàng

Mail: hoangnd@itc.edu.vn

- Giới thiệu về cơ sở dữ liệu quan hệ và MySQL
- Hướng dẫn sử dụng PHP với MySQL

1. GIỚI THIỆU VỀ CƠ SỞ DỮ LIỆU QUAN HỆ VÀ MySQL

Trong phần này có các nội dung:

1.1. Giới thiệu về cơ sở dữ liệu quan hệ

1.2. Các kiểu dữ liệu thông dụng trong MySQL

1.3. Các câu lệnh dùng để thao tác dữ liệu trong SQL

1.4. Giới thiệu MySQL

1.5. Sử dụng phpMyAdmin

1.1. GIỚI THIỆU VỀ CƠ SỞ DỮ LIỆU QUAN HỆ

Tổ chức các bảng trong cơ sở dữ liệu:

- ❑ Hệ cơ sở dữ liệu quan hệ bao gồm các bảng. Bảng chứa dòng (bản ghi) và cột (trường)
- ❑ Cột biểu thị thuộc tính của thực thể
- ❑ Dòng chứa tập hợp các dữ liệu của thực thể
- ❑ Giao điểm giữa dòng và cột thường được gọi là ô
- ❑ Khóa chính dùng để phân biệt các dòng trong bảng
- ❑ Khóa ngoại dùng để liên kết giữa các bảng

1.1. GIỚI THIỆU VỀ CƠ SỞ DỮ LIỆU QUAN HỆ

Quan hệ giữa hai bảng trong cơ sở dữ liệu:

- ❑ Bảng trong cơ sở dữ liệu quan hệ được liên kết với nhau qua các cột
- ❑ Khi xác định khóa ngoại, muốn thêm dòng cho bảng có khóa ngoại, bạn phải định giá trị của cột đó từ một khóa chính trong bảng liên kết
- ❑ Một bảng có thể có quan hệ một - một và quan hệ nhiều - nhiều với bảng khác

1.2. CÁC KIỂU DỮ LIỆU THÔNG DỤNG TRONG MYSQL

Một số kiểu dữ liệu thông dụng:

- ❑ Char: chuỗi ký tự có độ dài cố định trong bộ ký tự ASCII
- ❑ Varchar: chuỗi ký tự có độ dài thay đổi trong bộ ký tự ASCII
- ❑ Int: giá trị nguyên nhiều kích cỡ
- ❑ Decimal: giá trị thập phân, chứa dấu phẩy động để ngăn cách phần nguyên và phần thập phân
- ❑ Date, Time: ngày, giờ

1.2. CÁC KIỂU DỮ LIỆU THÔNG DỤNG TRONG MYSQL

Ngoài ra còn có:

- ❑ Null: giá trị không xác định
- ❑ Giá trị mặc định (default): giá trị này được thay thế khi một hàng được thêm vào nhưng không được xác định giá trị
- ❑ Cột tự tăng: giá trị của nó là số tự động tăng khi một dòng được thêm vào bảng

1.3. CÁC CÂU LỆNH THAO TÁC DỮ LIỆU TRONG SQL

Chọn dữ liệu từ một bảng:

- ❑ Để xác định cột, sử dụng mệnh đề SELECT
- ❑ Để xác định bảng muốn truy xuất dữ liệu, sử dụng mệnh đề FROM
- ❑ Để xác định dòng, sử dụng mệnh đề WHERE
- ❑ Để xác định kiểu sắp kết quả, sử dụng mệnh đề ORDER BY

1.3. CÁC CÂU LỆNH THAO TÁC DỮ LIỆU TRONG SQL

Chọn dữ liệu từ nhiều bảng:

- ❑ Sử dụng mệnh đề JOIN để liên kết các bảng với nhau
- ❑ INNER JOIN: các dòng chỉ được trả về trong tập kết quả nếu khóa của dòng ở bảng thứ nhất bằng (khớp) với khóa của dòng ở bảng thứ hai
- ❑ OUTER JOIN: trả về các dòng từ một bảng trong liên nối ngay cả khi bảng kia không có dòng phù hợp
- ❑ LEFT/RIGHT OUTER JOIN: dữ liệu của tất cả các dòng trong bảng bên trái/phải được trả về bảng kết quả, song chỉ những dữ liệu của dòng phù hợp trong bảng còn lại được thêm vào

1.3. CÁC CÂU LỆNH THAO TÁC DỮ LIỆU TRONG SQL

❑ Thêm dữ liệu:

❖ Cú pháp:

INSERT INTO <Tên bảng> [<Danh sách cột>]
VALUES (<Danh sách giá trị tương ứng với cột>)

❖ Ví dụ: INSERT INTO products (categoryID,
productCode, productName, listPrice) VALUES (1,
'tele', 'Fender', 599.00)

❑ Sửa dữ liệu:

❖ Cú pháp: UPDATE <Tên bảng>
SET <Danh sách biểu thức>
WHERE <Các điều kiện>

❖ Ví dụ: UPDATE products SET listPrice=610 WHERE
productName='Fender'

1.3. CÁC CÂU LỆNH THAO TÁC DỮ LIỆU TRONG SQL

❑ Xóa dữ liệu:

❖ Cú pháp: UPDATE <Tên bảng>

WHERE <Các điều kiện>

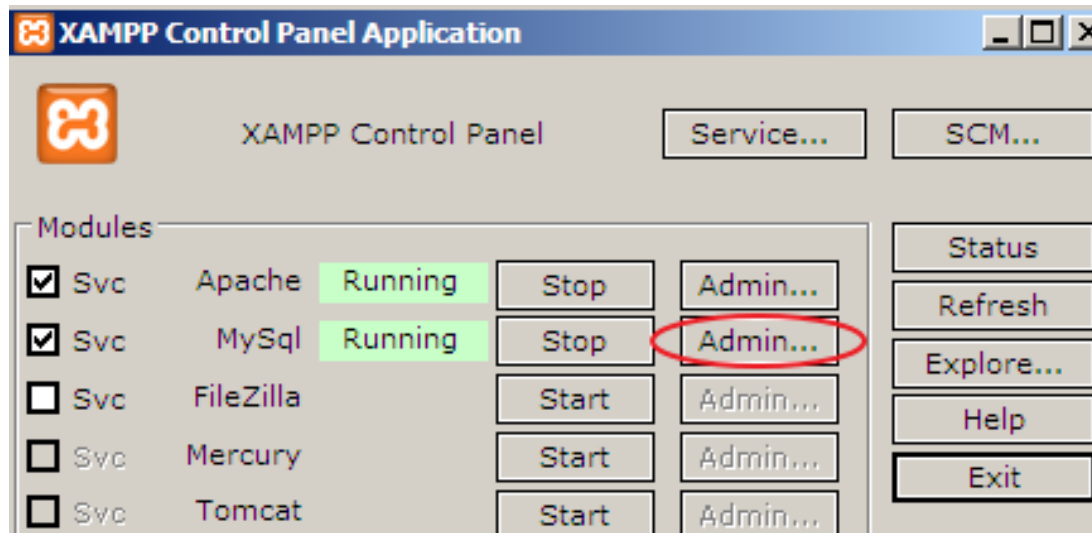
❖ Ví dụ: DELETE FROM products WHERE
productName='Fender'

1.4. GIỚI THIỆU MySQL

- ❑ MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở
- ❑ MySQL được cung cấp trong gói phần mềm XAMPP hoặc có thể cài riêng biệt
- ❑ Ưu điểm:
 - ❖ Rẻ: Hầu hết các tính năng của MySQL được cung cấp miễn phí, các tính năng khác tương đối rẻ so với các sản phẩm cùng dòng
 - ❖ Nhanh: MySQL là một trong những cơ sở dữ liệu nhanh nhất hiện nay
 - ❖ Dễ dùng: So với các hệ quản trị cơ sở dữ liệu khác, MySQL dễ cài đặt và sử dụng
 - ❖ Linh động: MySQL chạy trên hầu hết các hệ điều hành hiện đại như Windows, Unix, Solaris và OS/2

1.5. SỬ DỤNG PHPMYADMIN

- ❑ phpMyAdmin là ứng dụng quản trị hệ cơ sở dữ liệu MySQL
- ❑ Khởi động phpMyAdmin:
 - ❖ Khởi động XAMPP
 - ❖ Nhấn nút Admin của dòng MySQL



1.5. SỬ DỤNG PHPMYADMIN

□ Giao diện của phpMyAdmin:

The screenshot displays the phpMyAdmin web interface for a 'localhost' connection. The interface is divided into several sections:

- Left Sidebar:** Contains the 'phpMyAdmin' logo and a list of databases: 'cdcol', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'test', and 'webauth'. A red box highlights this list with the annotation: "Danh sách các cơ sở dữ liệu hiện tại".
- Top Navigation Bar:** Features tabs for 'Databases', 'SQL', 'Status', 'Processes', 'Privileges', 'Export', 'Import', and 'More'. A red box highlights these tabs with the annotation: "Các tab chức năng của phpMyAdmin".
- Main Content Area:**
 - General Settings:** Shows 'MySQL connection collation' set to 'utf8_general_ci'.
 - Appearance Settings:** Includes a 'Language' dropdown set to 'English', a 'Theme / Style' dropdown set to 'pmahomme', and a 'Font size' dropdown set to '82%'. A red box highlights this section with the annotation: "Thiết lập ngôn ngữ, giao diện cho phpMyAdmin". Below these are 'More settings' and a 'Language' icon.
- Right Panel:**
 - MySQL:** Displays server information: 'Server: localhost via TCP/IP', 'Server version: 5.5.16', 'Protocol version: 10', 'User: root@localhost', and 'MySQL charset: UTF-8 Unicode (utf8)'.
 - Web server:** Displays server details: 'Apache/2.2.21 (Win32) mod_ssl/2.2.21 OpenSSL/1.0.0e PHP/5.3.8 mod_perl/2.0.4 Perl/v5.10.1', 'MySQL client version: mysqlnd 5.0.8-dev - 20102224 - \$Revision: 310735 \$', and 'PHP extension: mysql'.

A red box at the bottom right highlights the 'phpMyAdmin' footer with the annotation: "Các thông tin về server, tài khoản, ...".

1.5. SỬ DỤNG PHPMYADMIN

❑ Thực thi các câu truy vấn:

The screenshot shows the PHPMyAdmin interface with the 'SQL' tab selected. The interface includes a top navigation bar with tabs: Databases, SQL, Status, Processes, Privileges, Export, Import, and More. Below the navigation bar is a section titled 'Run SQL query/queries on'. A large text area for writing SQL queries is present. At the bottom, there are options to bookmark the query, a delimiter selector, and a 'Go' button to execute the query.

Bước 1: Nhấn chọn tab SQL

Bước 2: Viết các câu truy vấn SQL vào đây

Bước 3: Nhấn nút này để thực thi câu truy vấn

1.5. SỬ DỤNG PHPMYADMIN

❑ Import file chứa các câu truy vấn SQL:

The screenshot shows the phpMyAdmin interface with the 'Import' tab selected. The page title is 'Importing into the current server'. The 'File to Import:' section includes a text input field, a 'Choose File' button, and a 'Character set of the file:' dropdown menu set to 'utf-8'. The 'Format:' section has a dropdown menu set to 'SQL' and a 'Go' button. Four red speech bubble annotations provide instructions in Vietnamese:

- Bước 1:** Nhấn chọn tab Import (Click to select the Import tab)
- Bước 2:** Nhấn vào đây và chọn đường dẫn tới file cần import (Click here and select the path to the file to be imported)
- Bước 3:** Chọn format là SQL nếu muốn import câu truy vấn từ file SQL (Select format as SQL if you want to import queries from a SQL file)
- Bước 4:** Nhấn nút này để bắt đầu import (Click this button to start import)

1.5. SỬ DỤNG PHPMYADMIN

❑ Xem dữ liệu và cấu trúc bảng:

The screenshot displays the phpMyAdmin interface. On the left, a sidebar lists databases: cdcot, information_schema, mysql, perform, phpmy, test, and webauth. The 'test' database is highlighted with a red box. A red speech bubble points to it with the text: "Nhấn vào tên CSDL cần xem" (Click on the name of the database you want to view).

The main panel shows the 'test' database selected. It features a toolbar with buttons: Structure, SQL, Search, Query, Import, and Operations. Below the toolbar, a table lists the databases in the 'test' database: tt_as, tt_associations, and tt_banners. The 'tt_associations' table is highlighted with a red box. A red speech bubble points to the 'Browse' button for 'tt_associations' with the text: "Để xem dữ liệu của bảng, nhấn vào nút Browse tương ứng" (To view the data of the table, click on the corresponding Browse button). Another red speech bubble points to the 'Structure' button for 'tt_associations' with the text: "Để xem cấu trúc của bảng, nhấn vào nút Structure tương ứng" (To view the structure of the table, click on the corresponding Structure button).

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

Mysql cải tiến hay còn gọi là Mysqli là một trong những trình quản lí cơ sở dữ liệu thông dụng nhất hiện nay trong phiên bản php 5.0 trở về trước mysql không còn được hỗ trợ. Mysqli có hai cách kết nối cơ sở liệu, kết nối bằng hàm, bằng hướng đối tượng.

- Phần mở rộng MySQLi (chữ "i" là viết tắt của cải tiến - improved).
- PDO (đối tượng dữ liệu PHP).

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

Các bước thao tác trên cơ sở dữ liệu

Thông thường thao tác trên cơ sở dữ liệu qua các bước sau:

- **Bước 1: Tạo kết nối**
- **Bước 2: Mở kết nối dữ liệu**
- **Bước 3: Tạo lệnh điều khiển truy vấn SQL**
- **Bước 4: Thực thi lệnh**
- **Bước 5: Đóng kết nối**
- **Bước 6: in kết quả**

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

1. Hàm mysqli_connect trong PHP

PHP `mysqli_connect()` thường dùng để kết nối cơ sở dữ liệu

Cú pháp:

resource `mysqli_connect` (server, username, password, databasename)

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
    if(!$conn){
        die('Kết nối thất bại:'.mysqli_connect_error());
    }else{
        echo"kết nối thành công"; }
?>
```

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

1. Hàm mysqli_connect trong PHP

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
    if(!$conn){
        die('Kết nối thất bại:'.mysqli_connect_error());
    }else{
        echo"kết nối thành công"; }
?>
```

Giải thích :

- \$servername: địa chỉ host của bạn
- \$username: là tên đăng nhập database
- \$password: đây là mật khẩu truy cập vào database
- \$dbname: tên database của bạn(demo)

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

2. Hàm mysqli_close trong PHP

PHP `mysqli_close()` thường được sử dụng để đóng kết nối database. Trả về true nếu đóng kết nối thành công.

Cú pháp:

```
bool mysqli_close($conn);
```

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:'.mysqli_connect_error());
}else{
    echo"Kết nối thành công";
}
mysqli_close($conn);
?>
```

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

3. Hàm mysqli_query trong PHP

- Các hàm **mysqli_query()** thực hiện một truy vấn đến database

Cú pháp hàm mysqli_query():

```
mysqli_query(connection, query, resultmode);
```

Giải thích về hàm mysqli_query():

- connection** chỉ định kết nối MySQL sử dụng
- query** chỉ định chuỗi truy vấn
- resultmode** chỉ định chế độ của kết quả trả về (MYSQLI_STORE_RESULT là mặc định, còn nếu cần phải lấy số lượng lớn dữ liệu thì sử dụng MYSQLI_USE_RESULT)

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

3. Hàm mysqli_query trong PHP

```
//Tạo database demo
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:'.mysqli_connect_error());
}else{
    echo"kết nối thành công";
}
$sql = 'CREATE Database demo';
if(mysqli_query( $conn,$sql)){
    echo "Đã tạo thành công cơ sở dữ liệu.";
}else{
    echo "Không tạo được cơ sở dữ liệu ".mysqli_error($conn);
}
mysqli_close($conn);
?>
```


2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

4. Hàm mysqli_select_db trong PHP

hàm **mysqli_select_db** được sử dụng để chọn một cơ sở dữ liệu.

Hàm mysqli_select có cú pháp như sau:

```
mysqli_select_db(connection, dbname);
```

Giải thích hàm mysqli_select_db:

- **mysqli_select_db(...)** là hàm lựa chọn cơ sở dữ liệu, trả về **true** hoặc **false**
- **connection** Xác định kết nối MySQL để sử dụng
- **dbname** Xác định cơ sở dữ liệu để được sử dụng

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

4. Hàm mysqli_select_db trong PHP

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:'.mysqli_connect_error());
}else{
    echo"kết nối thành công.";
}
if(mysqli_select_db($conn, $dbname )==true)
    die("Database này đã tồn tại.")
$sql = 'CREATE Database demo';
if(mysqli_query( $conn,$sql)){
    echo "Đã tạo thành công cơ sở dữ liệu.";
}else{
    echo "Không tạo được cơ sở dữ liệu ".mysqli_error($conn);
}
mysqli_close($conn);
?>
```

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

5. Hàm `mysqli_num_rows` trong PHP

hàm `mysqli_num_rows()` trả về số lượng hàng trong một tập kết quả.

Cú pháp hàm `mysqli_num_rows()`: `mysqli_num_rows(result);`

❖ `result` là tập hợp các kết quả trả về từ các hàm

- `mysqli_query()`,
- `mysqli_store_result()`
- `mysqli_use_result()`

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

5. Hàm mysqli_num_rows trong PHP

```
<?php
$servername = "localhost"; $username = "root"; $password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:'.mysqli_connect_error());
}else{
    echo"kết nối thành công.";
}
$sql = "insert into user(name, user_salary) values('Thích thi lại', 90000)";
$result=mysqli_query($conn, $sql);
if($result){
    echo "Thêm thành công";
}else{
    echo "Không thể thêm dữ liệu vào bảng user: ". mysqli_error($conn);
}
$rows = mysqli_num_rows($result);
mysqli_close($conn);
?>
```

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MySQLi

6. Hàm mysqli_fetch_array trong PHP

Hàm **mysqli_fetch_array()** tìm nạp một hàng kết quả dưới dạng một mảng kết hợp

Cú pháp hàm mysqli_fetch_array:

```
mysqli_fetch_array(result);
```

❖ **result** là tập hợp các kết quả trả về từ các hàm

- mysqli_query(),
- mysqli_store_result()
- mysqli_use_result()

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

Một số hàm thông dụng cho việc truy vấn dữ liệu:

Hàm	Giải thích
array <code>mysqli_fetch_assoc (resource \$result)</code>	Trả về 1 mảng kết hợp (với các trường là key:), trả về false không có dữ liệu.
object <code>mysqli_fetch_object(\$result)</code>	Trả về dạng đối tượng
int <code>mysqli_num_rows</code>	Trả về tổng số dòng dữ liệu trong bảng.
array <code>mysqli_fetch_row (resource \$result)</code>	Hàm trả về mảng có chỉ số (0,1,2..).
<code>mysqli_fetch_array(resource \$result [, int \$result_type = MYSQL_BOTH])</code>	Trả về một mảng kết hợp, mảng chỉ số, hoặc cả 2. trong đó \$result_type có các giá trị:
	<ul style="list-style-type: none">• MYSQL_ASSOC: trả về associative array(giống <code>mysqli_fetch_assoc()</code>)• MYSQL_NUM: trả về enumerated array(giống <code>mysqli_fetch_row()</code>)• MYSQL_BOTH : (mặc định)

2. HƯỚNG DẪN SỬ DỤNG PHP VỚI MYSQLI

6. Hàm mysqli_fetch_array trong PHP

```
<?php
$servername = "localhost"; $username = "root"; $password = ""; $dbname = "demo";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "SELECT name, user_salary FROM user";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
    // output data of each row
    while($row = mysqli_fetch_array($result)) {
        echo "Họ tên: " . $row["name"]. " - Lương: " . $row["user_salary"]. "<br>";
    }
} else {
    echo "0 results";
}
mysqli_close($conn);
?>
```

3. VÍ DỤ INSERT MYSQLI

```
<?php
$servername = "localhost"; $username = "root"; $password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:<br>'.mysqli_connect_error());
}else{
    echo"Kết nối thành công <br>";
}
$sql = "insert into user(name, user_salary) values('Thích thi
lại', 90000)";
if(mysqli_query($conn, $sql)){
    echo "Thêm thành công";
}else{
    echo "Không thể thêm dữ liệu vào bảng user: ". mysqli_error($conn);
} mysqli_close($conn); ?>
```


3. VÍ DỤ UPDATE MYSQLI

```
<?php
$servername = "localhost"; $username = "root"; $password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:<br>'.mysqli_connect_error());
}else{
    echo "Kết nối thành công <br>";
}
$id=2; $name="Trần Thành Công"; $salary=80000;
$sql = "update user set name=$name , user_salary=$salary where
id=$id";
if(mysqli_query($conn, $sql)){
    echo "Cập nhật thành công";
}else{
    echo "Cập nhật không thành công: ". mysqli_error($conn);
}
mysqli_close($conn);
?>
```

3. VÍ DỤ DELETE MYSQLI

```
<?php
$servername = "localhost"; $username = "root"; $password = "";
$dbname = "demo";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:<br>'.mysqli_connect_error());
}else{
    echo"Kết nối thành công <br>";
}
$id=2;
$sql = "Delete user where id=$id";
if(mysqli_query($conn, $sql)){
    echo "Xóa thành công";
}else{
    echo "Xóa không thành công: ". mysqli_error($conn); }
mysqli_close($conn);
?>
```

3. VÍ DỤ QUERY MYSQLI

```
<?php
$servername = "localhost"; $username = "root"; $password = ""; $dbname =
"hiepsiit";
$conn = mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Kết nối thất bại:<br>'.mysqli_connect_error()); }else{
    echo"Kết nối thành công <br>";
}
$sql = 'SELECT * FROM user';
$retval=mysqli_query($conn, $sql);
if(mysqli_num_rows($retval) > 0)
{
    while($row = mysqli_fetch_assoc($retval))
    {
        echo "USER ID :{$row['id']} <br> ".
            "USER NAME : {$row['name']} <br> ".
            "USER SALARY : {$row['user_salary']} <br> ".
            "-----<br>";
    } //end of while
}else{
    echo "0 results";
}
mysqli_close($conn);
?>
```

TÓM TẮT CÁC HÀM MASQLI

Trong phần này có các nội dung:

1. Hàm `mysqli_connect` trong PHP
2. Hàm `mysqli_close()` trong PHP
3. Hàm `mysqli_query` trong PHP
4. Hàm `mysqli_select_db`
5. Hàm `mysqli_num_rows` trong PHP
6. Hàm `mysqli_fetch_array` trong PHP

KẾT THÚC