

PHP



LẬP TRÌNH PHP 1

BÀI 7: PHP & MYSQL

PHẦN 1

Nội dung bài học

- 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

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

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

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 trong kết quả, song chỉ những dữ liệu của dòng phù hợp trong bảng còn lại được thêm vào

❑ 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>

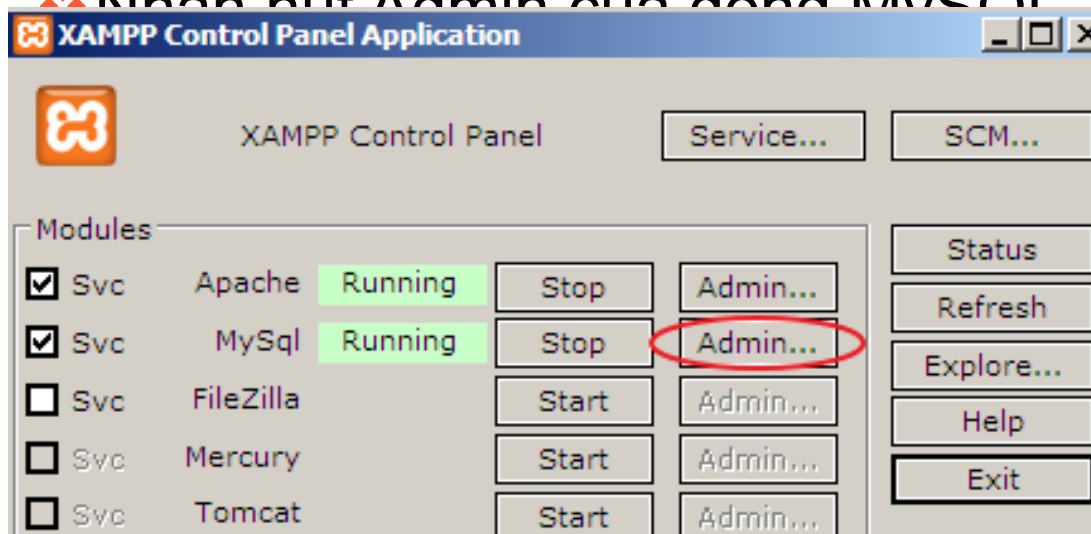
❖ Ví dụ: UPDATE products SET listPrice=610 WHERE
productName='Fender'

1.4. 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



□ 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 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".
- General Settings:** Shows the '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". A 'More settings' link is also present.
- MySQL Information:** A sidebar on the right provides details about the MySQL server:
 - Server: localhost via TCP/IP
 - Server version: 5.5.16
 - Protocol version: 10
 - User: root@localhost
 - MySQL charset: UTF-8 Unicode (utf8)
- Web server Information:** Another sidebar on the right provides details about the web server:
 - 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 \$
 - PHP extension: mysql

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

❑ Thực thi các câu truy vấn:

The screenshot shows the phpMyAdmin interface with the 'SQL' tab selected. The interface includes a navigation bar at the top with tabs for 'Databases', 'SQL', 'Status', 'Processes', 'Privileges', 'Export', 'Import', and 'More'. Below the navigation bar, there is a section titled 'Run SQL query/queries on' followed by a large text area for entering SQL queries. At the bottom, there are options to 'Bookmark this SQL query', a 'Clear' button, and a 'Go' button. Three red callouts provide instructions: 'Bước 1: Nhấn chọn tab SQL' (Step 1: Click the SQL tab), 'Bước 2: Viết các câu truy vấn SQL vào đây' (Step 2: Write the SQL queries here), and 'Bước 3: Nhấn nút này để thực thi câu truy vấn' (Step 3: Press this 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

❑ Import file chứa các câu truy vấn SQL:

Databases SQL Status Processes Privileges Export **Import** More

Importing into the current server

File to Import:

File may be compressed (gzip, bzip2, ...). A compressed file's name must end with [extension]. Example: .sql.zip

Browse your computer: **Choose File** No file chosen (Max: 8,192KiB)

Character set of the file: utf-8

Format:

SQL

Go

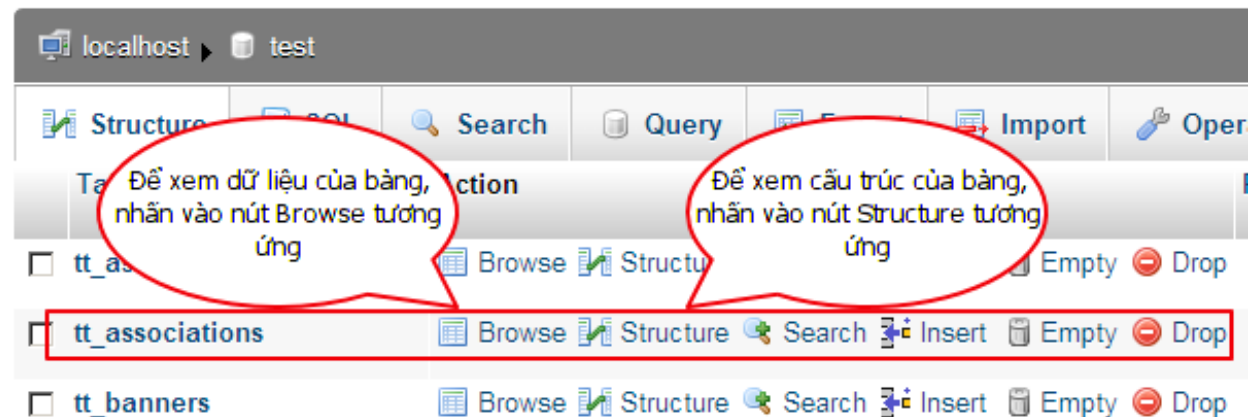
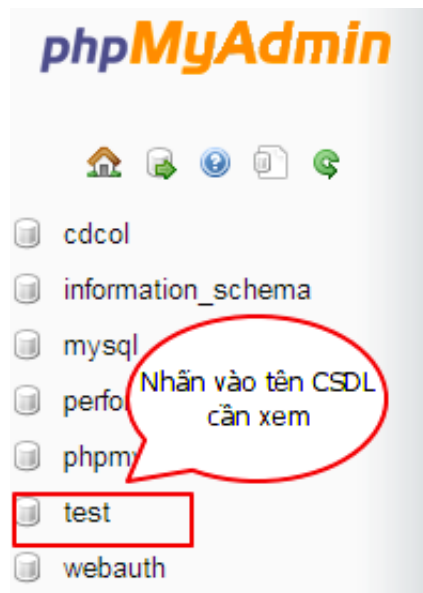
Bước 1: Nhấn chọn tab Import

Bước 2: Nhấn vào đây và chọn đường dẫn tới file cần import

Bước 3: Chọn format là SQL nếu muốn import câu truy vấn từ file SQL

Bước 4: Nhấn nút này để bắt đầu import

❑ Xem dữ liệu và cấu trúc bảng:



PHP



LẬP TRÌNH PHP 1

BÀI 7: PHP & MYSQL

PHẦN 2

2. Hướng dẫn sử dụng PHP với MySQL

Trong phần này có các nội dung:

- 2.1. Sử dụng PHP để làm việc với MySQL
- 2.2. Lấy dữ liệu từ tập kết quả

2.1. Sử dụng PHP để làm việc với MySQL

- ❑ Làm việc với MySQL: sử dụng đối tượng PDO (PHP Data Objects)
- ❑ Kết nối tới MySQL:
 - Tạo DSN (Data Source Name) xác định tên máy chủ và tên cơ sở dữ liệu: `mysql:host=host_address;dbname=database_name`
 - Tạo đối tượng PDO với hai đối số: DSN, tên tài khoản quản trị cơ sở dữ liệu và mật khẩu: `new PDO($dsn, $username, $password)`

```
$dsn = 'mysql:host=localhost;dbname=QLSV';
```

```
$username = 'root';
```

```
$password = '';
```

```
❑ $db = new PDO($dsn, $username, $password); //tạo đối tượng PDO $db  
việc.
```

❑ Thực thi câu lệnh SELECT và lưu kết quả vào một biến:

Phương thức	Mô tả
<code>query(\$select_statement)</code>	Thực thi câu lệnh SELECT và trả về đối tượng PDOStatement chứa tập kết quả. Nếu không có tập kết quả nào được trả về, phương thức này trả về giá trị FALSE.

```
$query = 'SELECT * FROM products';  
$products = $db->query($query); // $products chứa tập kết quả
```

❑ Thực thi câu lệnh INSERT, UPDATE, DELETE:

❖ Sử dụng phương thức exec của đối tượng PDO:

Phương thức	Chú thích
<code>exec(\$sql_statement)</code>	Thực thi một câu lệnh SQL cụ thể và trả về số hàng bị thay đổi. Nếu không có bản ghi nào bị ảnh hưởng, hàm trả về 0.

❑ Ví dụ:

```
//Hướng dẫn thực thi câu lệnh INSERT
$id = 1;
$name = 'Fender Stratocaster';
$query = "INSERT INTO products(ID,productName)
VALUES
($id, '$name')";
$insert_count = $db->exec($query); //lưu số bản ghi insert vào biến
```

❑ Xử lý ngoại lệ với try catch:

- ❖ Ngoại lệ là đối tượng chứa thông tin về lỗi xảy ra. Một số câu lệnh PHP sẽ vút ra ngoại lệ khi chúng gặp lỗi. Nếu một ngoại lệ không được xử lý thì ứng dụng PHP sẽ kết thúc ngay
- ❖ Để xử lý ngoại lệ ta dùng cấu trúc try catch với cú pháp:

```
try {  
    //câu lệnh có thể vút ra ngoại lệ  
}  
catch (Exception $e) {  
    $error = $e->getMessage(); //Biến $error dùng để xem thông báo lỗi  
}
```

2.2. Lấy dữ liệu từ tập kết quả

❑ Làm việc với mảng:

- Khởi tạo mảng: $\$Mang = \text{array}(<\text{Số phần tử}>);$
- Gán giá trị cho một phần tử của mảng:
 - Kiểu chỉ mục: $\$Mang[<\text{số thứ tự phần tử}>] = <\text{giá trị}>$
 - Kiểu chuỗi: $\$Mang[<\text{chuỗi tên phần tử}>] = <\text{giá trị}>$
- Lấy giá trị của một phần tử của mảng:
 - Kiểu chỉ mục: $\$Bien = \$Mang[<\text{số thứ tự phần tử}>]$
 - Kiểu chuỗi: $\$Bien = \$Mang[<\text{chuỗi tên phần tử}>]$

❑ Lấy dữ liệu từ hàng đầu tiên của tập kết quả:

- ❖ Sử dụng hàm fetch để lấy dữ liệu hàng đầu tiên của tập kết quả, sau đó dùng một biến để lưu dữ liệu trả về
- ❖ Cú pháp: `$Bien = $KetQua->fetch();`

❑ Lấy dữ liệu từ tất cả các hàng trong tập kết quả:

- ❖ Sử dụng cấu trúc vòng lặp foreach để lấy dữ liệu từ từng hàng của tập kết quả
- ❖ Dữ liệu từ từng hàng của tập kết quả được lưu vào một biến (cấu trúc fetch được tự động sử dụng)
- ❖ Cú pháp: `foreach ($KetQua as $Bien)`

```
{  
    //Lấy dữ liệu từ biến mảng $Bien  
}
```


Tổng kết bài học

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





KẾT THÚC