

XÂY DỰNG WEBSITE SỬ DỤNG PHP & MYSQL



Nội dung bài học

- Hệ thống lại kiến thức đã học
- Các bước xây dựng một website động

1. Hệ thống kiến thức đã học

- Bài 1: Làm quen với môi trường PHP và MySQL (phần 1)
 - Giới thiệu về lập trình web với PHP & MySQL
 - Hướng dẫn viết mã cho ứng dụng PHP
 - Hướng dẫn kiểm thử và gỡ lỗi cho ứng dụng PHP
- Bài 2: Làm quen với môi trường PHP và MySQL (phần 2)
 - Giới thiệu về cơ sở dữ liệu quan hệ và MySQL
 - Hướng dẫn sử dụng PHP với MySQL
 - Giới thiệu về mô hình MVC
- Bài 3: Lập trình PHP (phần 1)
 - Viết câu lệnh điều khiển
 - Khởi tạo và sử dụng hàm
 - Khởi tạo và sử dụng đối tượng

Hệ thống kiến thức đã học

- Bài 4: Lập trình PHP (phần 2)
 - Làm việc với chuỗi và số
 - Làm việc với ngày tháng
 - Làm việc với mảng
- Bài 5: Lập trình PHP (phần 3)
 - Lấy dữ liệu từ form
 - Hiển thị dữ liệu lên trang web
- Bài 6: Lập trình PHP (phần 4)
 - Làm việc với cookie
 - Làm việc với session

Bài 1: làm quen với môi trường PHP & MySQL (phần 1)

1. Giới thiệu về lập trình web với PHP & MySQL
 - 1.1. Kiến trúc của một ứng dụng web
 - 1.2. Cách thức xử lý trang web tĩnh
 - 1.3. Cách thức xử lý trang web động
 - 1.4. Tổng quan về các phần mềm web
 - 1.5. Giới thiệu môi trường làm việc PHP
 - 1.6. Giới thiệu phần mềm NetBeans
 - 1.7. Quy trình xây dựng một ứng dụng PHP đơn giản
2. Hướng dẫn viết mã cho ứng dụng PHP
 - 2.1. Các kỹ năng lập trình PHP cơ bản
 - 2.2. Truyền dữ liệu từ yêu cầu
 - 2.3. Làm việc với dữ liệu
 - 2.4. Các câu lệnh điều khiển
3. Hướng dẫn kiểm thử và gỡ lỗi cho ứng dụng PHP
 - 3.1. Giới thiệu về kiểm thử và sửa lỗi
 - 3.2. Hướng dẫn sửa lỗi với xDebug & NetBeans

Bài 2: làm quen với môi trường PHP & MySQL (phần 1)

1. Giới thiệu về cơ sở dữ liệu quan hệ và MySQL
 - 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
2. Hướng dẫn sử dụng PHP với MySQL
 - 2.1. Sử dụng PHP để làm việc với MySQL
 - 2.2. Lấy dữ liệu từ tập kết quả
3. Giới thiệu về mô hình MVC
 - 3.1. Giới thiệu về mô hình MVC
 - 3.2. Hướng dẫn viết hàm
 - 3.3. Hướng dẫn chuyển hướng yêu cầu

Bài 3: lập trình PHP (phần 1)

1. Viết câu lệnh điều khiển

1.1. Viết mã cho biểu thức điều kiện

1.2. Viết cấu trúc lựa chọn

2. Khởi tạo và sử dụng hàm

2.1. Các kỹ năng cơ bản để làm việc với hàm

2.1.1. Khởi tạo và gọi hàm

2.1.2. Truyền tham số theo giá trị và tham chiếu

2.1.3. Tìm hiểu phạm vi hoạt động của biến

2.1.4. Gán giá trị mặc định cho tham số

2.1.5. Sử dụng danh sách tham số có độ dài biến đổi

2.2. Khởi tạo và sử dụng thư viện của hàm

Bài 3: lập trình PHP (phần 1)

3. Khởi tạo và sử dụng đối tượng

3.1. Khởi tạo và sử dụng lớp

3.2. Viết hằng, thuộc tính và phương thức của lớp

3.3. Một số kỹ năng bổ sung

3.4. Làm việc với kế thừa

Bài 4: lập trình PHP (phần 2)

1. Làm việc với chuỗi và số
 - 1.1. Làm việc với chuỗi
 - 1.2. Làm việc với số
 - 1.3. Các kỹ năng khác
2. Làm việc với ngày tháng
 - 2.1. Sử dụng nhãn thời gian
 - 2.2. Sử dụng đối tượng
3. Làm việc với mảng
 - 3.1. Khởi tạo và sử dụng mảng
 - 3.2. Khởi tạo và sử dụng mảng liên kết
 - 3.3. Làm việc với hàng đợi và ngăn xếp
 - 3.4. Làm việc với mảng hai chiều

Bài 5: lập trình PHP (phần 3)

Làm việc với form:

1. Lấy dữ liệu từ form

- 1.1. Nhận dữ liệu từ textbox, password box, hidden field
- 1.2. Lấy dữ liệu từ radio button
- 1.3. Lấy dữ liệu từ checkbox
- 1.4. Lấy dữ liệu từ mảng checkbox
- 1.5. Lấy dữ liệu từ dropdown list
- 1.6. Lấy dữ liệu từ list
- 1.7. Lấy dữ liệu từ text area

2. Hiển thị dữ liệu lên trang web

- 2.1. Định dạng các ký tự đặc biệt
- 2.2. Định dạng ngắt dòng
- 2.3. Hiển thị dữ liệu với câu lệnh echo và print

Bài 6: lập trình PHP (phần 4)

1. Làm việc với cookie

- 1.1. Giới thiệu về cookie
- 1.2. Thiết lập và truy xuất cookie
- 1.3. Bật và tắt cookie

2. Làm việc với session

- 2.1. Nguyên nhân khó sử dụng session với HTTP
- 2.2. Cách thức hoạt động của session trong PHP
- 2.3. Bắt đầu một session
- 2.4. Gán và lấy giá trị của biến session
- 2.5. Kết thúc một session
- 2.6. Quản lý session

2. Các bước lập trình một website

- 2.1. Cài đặt các phần mềm cần thiết
- 2.2. Xây dựng cơ sở dữ liệu MySQL
- 2.3. Xây dựng phần front-end cho website
- 2.4. Xây dựng phần back-end cho website
- 2.5. Chạy thử và kiểm tra lỗi

2.1. Cài đặt các phần mềm cần thiết

- Cài đặt PHP, MySQL, Apache: chỉ cần cài đặt gói phần mềm XAMPP
- Cài đặt NetBeans để quản lý project, viết mã PHP, ...
- Cài đặt xDebug để dò lỗi

2.2. Xây dựng cơ sở dữ liệu

Sử dụng phpMyAdmin

- **Bước 1:** Khởi tạo cơ sở dữ liệu. Lưu ý chọn collation thích hợp để hỗ trợ tiếng Việt
- **Bước 2:** Khởi tạo các bảng với các trường có kiểu dữ liệu phù hợp
- **Bước 3:** Điền dữ liệu vào các bảng. Công đoạn này cũng có thể thực hiện sau bằng cách sử dụng PHP

2.3. Xây dựng phần front-end

- Front-end: phần website tương tác với khách truy cập website thông thường
- Lập trình viên thường có hai hướng xây dựng:
 - Xây dựng bình thường như các website HTML rồi thêm các đoạn mã PHP xử lý cần thiết vào
 - Xây dựng theo mô hình MVC: Trong thực tế, khi xây dựng mô hình MVC người ta có thể kết hợp tầng Controller và tầng View
- Hai kiểu lập trình:
 - Lập trình hướng đối tượng
 - Lập trình bình thường (hướng cấu trúc)

Xây dựng phần front-end

- Xây dựng file kết nối tới cơ sở dữ liệu. Thông thường file này được sử dụng làm file thư viện
- **Ví dụ:** tạo file database.php chứa mã PHP kết nối tới cơ sở dữ liệu. Sau này, bất kỳ file nào cần kết nối thì chỉ việc gọi tới file database.php bằng câu lệnh `require/require_one`

```
<?php
    $dsn = 'mysql:host=localhost;dbname=my_guitar_shop1';
    $username = 'root';
    $password = '';

    //Sử dụng try catch để bắt lỗi nếu kết nối không thành công
    try {
        $db = new PDO($dsn, $username, $password);
    } catch (PDOException $e) {
        $error_message = $e->getMessage();
    }

?>
```


Xây dựng phần front-end

- Lấy dữ liệu từ tập kết quả: sử dụng cấu trúc lặp foreach và hàm fetch()

Ví dụ:

```
$query = 'SELECT * FROM product';  
$products = $db->query($query);  
$product = $products->fetch();  
echo $product[0];
```

- Xây dựng file index.php:
 - Đây là file người dùng sẽ truy cập đầu tiên khi vào website, là điểm khởi đầu để truy cập vào các mục khác
 - Thông thường, file này có các đoạn mã PHP làm nhiệm vụ điều hướng bằng cách sử dụng cấu trúc lựa chọn if else hoặc switch case
 - Sử dụng hàm header để chuyển hướng

Xây dựng phần front-end

Ví dụ:

```
switch ($action) {
    case 'listproducts':
        //lấy danh mục và sản phẩm hiện thời
        $category_id = $_GET['category_id'];
        include('product_list.php');
        break;
    case 'view_product':
        $categories = getcategoriss();
        $product_id = $_GET['product_id'];
        $product = get_product($product_id);
        include('product_view.php');
        break;
    case 'deleteproduct':
        $category_id = $_POST['category_id'];
        $product_id = $_POST['product_id'];
        delete_product($product_id);
        header("Location: .?category_id=$category_id");
        break;
}
```

Xây dựng phần front-end

- Xây dựng các hàm xử lý dữ liệu: dữ liệu trong cơ sở dữ liệu nhiều khi không hiển thị trực tiếp cho người dùng ngay được mà cần phải xử lý
- Vận dụng kiến thức:
 - Khởi tạo và sử dụng hàm
 - Làm việc với chuỗi và số
 - Làm việc với ngày tháng
 - Làm việc với mảng

Xây dựng phần front-end

- Xây dựng cookie/session cho website nếu cần lưu và theo dõi trạng thái của khách truy cập
- Vận dụng kiến thức: làm việc với cookie và session

Ví dụ:

```
//Thiết lập cookie trong trình duyệt
```

```
$name = 'userid';
```

```
$value = 'rharris';
```

```
$expire = strtotime('+1 year');
```

```
$path = ' / ';
```

```
setcookie($name, $value, $expire, $path);
```

```
//Lấy giá trị của cookie từ trình duyệt
```

```
$userid = $_COOKIE['userid']; // $userid là 'rharris'
```

```
//Xóa cookie khỏi máy chủ
```

```
$expire = strtotime('-1 year'); setcookie('userid', '', $expire, '/');
```

Xây dựng phần front-end

- Xây dựng giao diện người dùng với HTML, CSS, Javascript. Có thể sử dụng phương pháp nhúng mã PHP vào mã HTML để nhập/xuất dữ liệu cần thiết
- Vận dụng kiến thức: làm việc với form

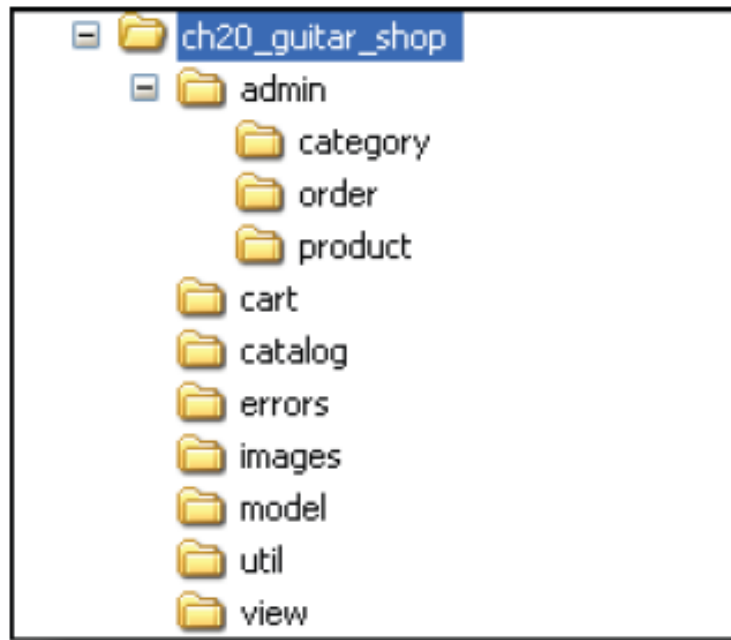
Ví dụ:

```
<div id="sidebar">
  <!-- display a list of categories -->
  <h2>Categories</h2>
  <ul class="nav">
    <?php foreach ($categories as $category) : ?>
      <li>
        <a href="?category_id=<?php echo $category['categoryID']; ?>">
          <?php echo $category['categoryName']; ?>
        </a>
      </li>
    <?php endforeach; ?>
  </ul>
</div>
```

2.4. Xây dựng phần back-end

- Back-end: phần website dành cho người quản trị
- Tùy vào nhu cầu quản trị mà một website có thể cần hoặc không cần phần back-end tách rời với front-end
- Cách thức xây dựng phần back-end tương tự như phần front-end, chỉ khác về mục đích xây dựng là để quản trị website
- Phần back-end thường được thiết kế gói trong một folder

Ví dụ:



2.5. Chạy thử và kiểm tra lỗi

- Bước cuối cùng là chạy thử project. Nếu xuất hiện lỗi thì dùng xDebug để theo dõi tìm hiểu nguyên nhân rồi xử lý
- Vận dụng kiến thức:
 - Sửa lỗi với NetBeans và xDebug
 - Kỹ năng lập trình PHP cơ bản: theo dõi lỗi

Ví dụ: đặt các câu lệnh echo để theo dõi giá trị của biến dưới dạng comment rồi kích hoạt khi cần:

```
$category_id = $product['categoryID'];  
$product_name = $product['productName'];  
$list_price = $product['listPrice'];  
$discount_percent = $product['discountPercent'];  
  
$discount_amount = $list_price * ($discount_percent / 100);  
//echo $discount_amount;  
$unit_price = $list_price - $discount_amount;  
//echo $unit_price;
```

Tổng kết bài học

- Để lập trình một website động với PHP và MySQL có 5 bước chính:
 1. Cài đặt các phần mềm cần thiết
 2. Xây dựng cơ sở dữ liệu MySQL
 3. Xây dựng phần front-end cho website
 4. Xây dựng phần back-end cho website
 5. Chạy thử và kiểm tra lỗi
- Hai hướng xây dựng:
 - Xây dựng bình thường như các website HTML rồi thêm các đoạn mã PHP xử lý cần thiết vào
 - Xây dựng theo mô hình MVC: Trong thực tế, khi xây dựng mô hình MVC người ta có thể kết hợp tầng Controller và tầng View
- Hai kiểu lập trình:
 - Lập trình hướng đối tượng
 - Lập trình bình thường (hướng cấu trúc)

XIN CẢM ƠN!