**Bài tập thực hành 04**

# Lập trình PHP nâng cao

**Mục tiêu cần đạt**:

Sau buổi các bạn cần nắm thao tác sử dụng PHP nâng cao

* Hiểu và biết cách dùng Cookies và Session
* Tạo form đăng nhập và dùng cookies, session để quản lý các đăng nhập, thực hiện chức năng log out (thoát) khỏi hệ thống ● Biết về SQL Injection.
* Upload file, lưu thông tin về tập tin trong CSDL
* Đọc, và xử lý nội dung trong tập tin

Các bạn đọc và thực hiện viết code như hướng dẫn để làm quen, sau đó đọc trả lời các **Yêu cầu** ở dưới bài hướng dẫn để thực hiện yêu cầu bài thực hành.

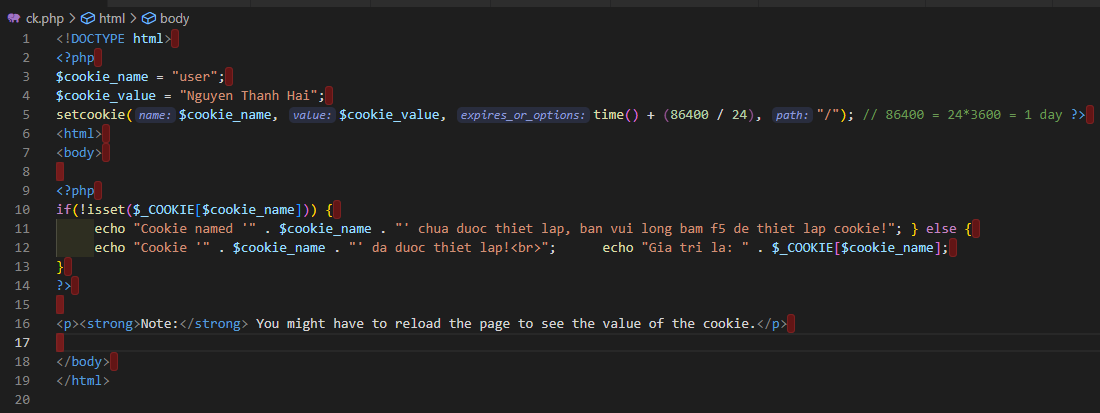
**Gợi ý Tham khảo:**

* <https://www.w3schools.com/php/>
* và các nguồn khác mà sinh viên tìm được

Yêu cầu bài thực hành:

1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

## Tạo Cookies



## Tạo 1 trang đăng ký người dùng với mật khẩu được mã hóa

Tạo csdl qlbanhang, với bảng customers đầu tiên như sau:



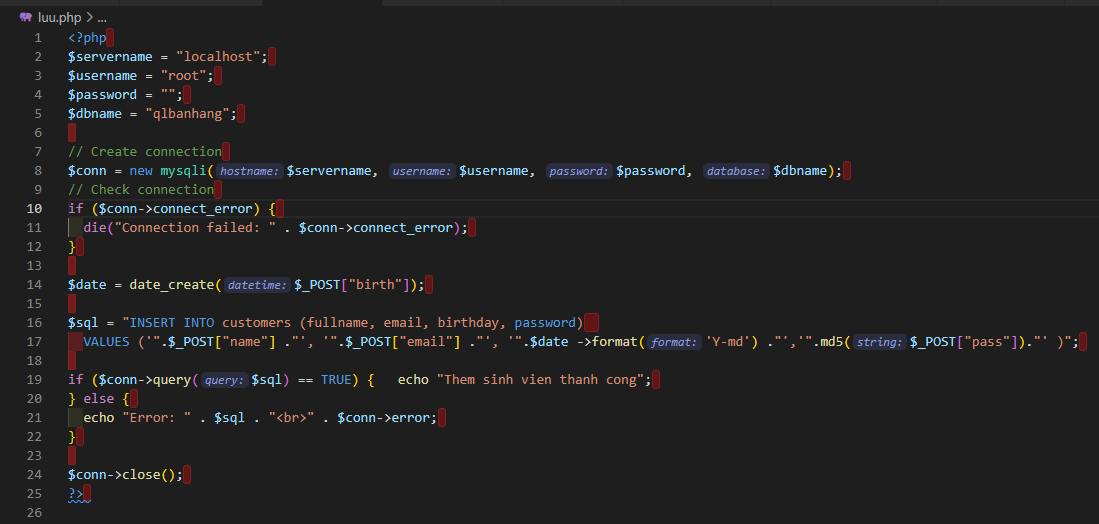
Tạo formnhap.php

### formnhap.php

### 

Và luu.php, chú ý lúc này chúng ta sẽ mã hóa mật khẩu với giải thuật md5 để bảo mật mật khẩu:

### luu.php



Sau khi chúng ta nhập xong mật khẩu với 12345 nhưng khi quan sát trong CSDL ta sẽ thấy mật khẩu đã được mã hóa ra chuỗi 827ccb0eea8a706c4c34a16891f84e7b:



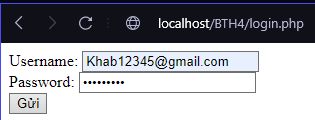
Trong trường hợp chúng ta muốn dùng email là username để đăng nhập ta nên đảm bảo 2 student không có email trùng nhau. Để đảm bảo điều này ta có thể chọn email là unique key (vì ta đã chọn id la primary key, và 1 bảng không thể có 2 khóa chính)

[ALTER TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) `customers` ADD UNIQUE( `email`);

## Tạo form đăng nhập

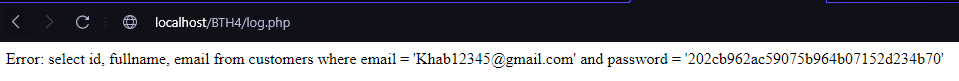
Sau khi tạo người dùng xong, ta có thể chạy form đăng nhập với username là email và password lấy từ CSDL

### Login.php



### Log.php

Khi chúng ta nhập sai, nhìn vào thông báo lỗi ta thấy



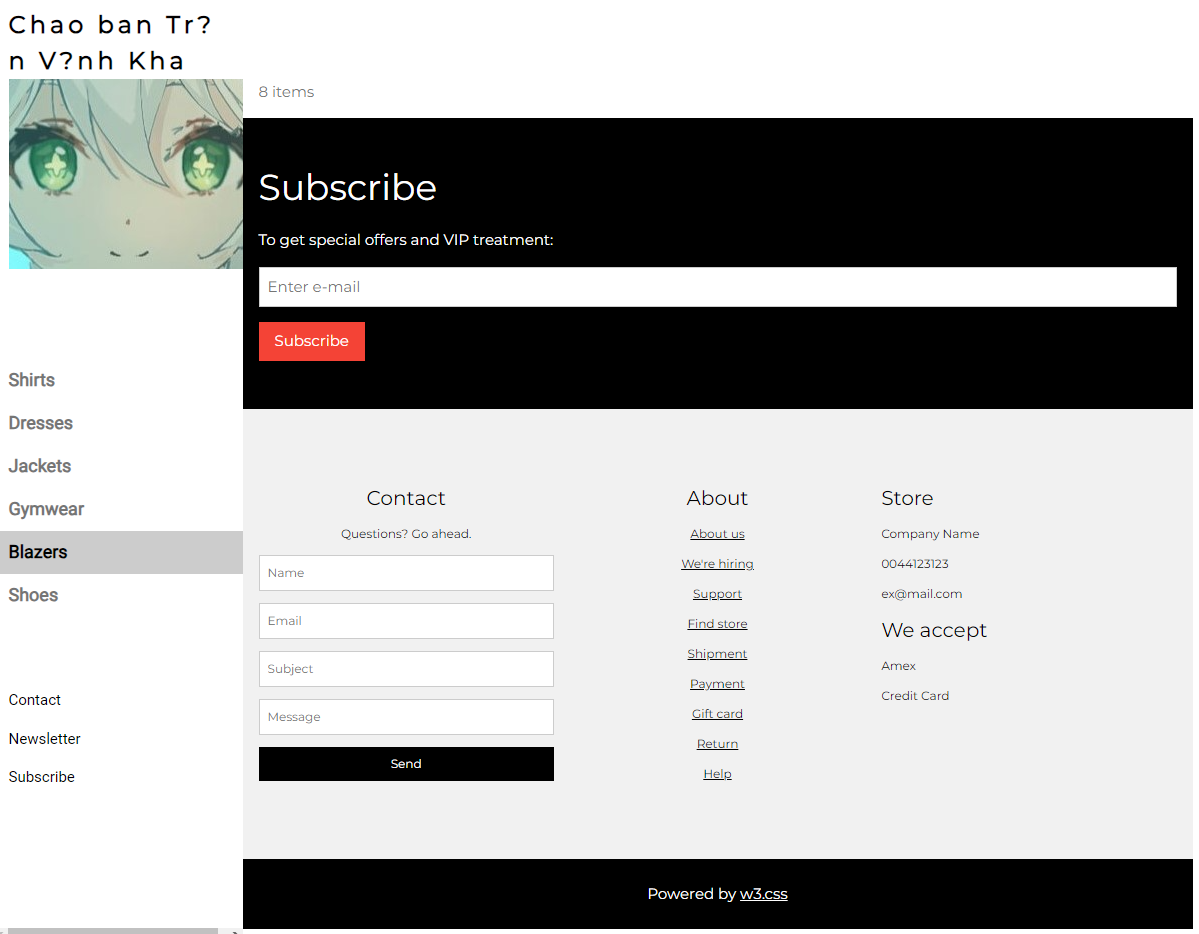
Mật khẩu đã cho không khớp với mã đã băm

## Dùng Cookies lưu lại tên đăng nhập, email và điều hướng

Chỉnh sửa lại log.php với code sau để khi đăng nhập thành công chuyển sang 1 trang chủ, còn nếu không đúng thì trở lại trang đăng nhập

### Log.php (modified)

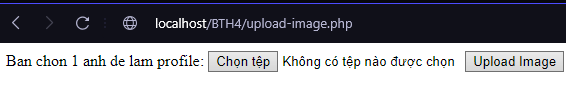
### Homepage.php (modified với ảnh profile)



## Upload tập tin

Upload file, chúng ta tạo form giao diện để upload file ảnh upload-img.php

### Upload-img.php



Tập tin upload.php xử lý hành động

### upload.php

Sau khi chúng ta upload thành công giao diện sau sẽ hiển thị



## Upload ảnh và lưu dữ liệu về ảnh trong CSDL

Trong phần này chúng ta minh họa tải tập tin ảnh làm ảnh profile của người dùng

Thêm cột để lưu đường dẫn ảnh:

[ALTER TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) `customers` ADD `img\_profile` [VARCHAR(](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-types.html)100);

### Upload-csdl.php

Ở đây do biểu diễn từng bước nên đoạn này chúng ta lấy dữ liệu link ảnh profile từ id của người dùng. Trong thực tế, các bạn có thể lưu link ảnh profile vào cookie/session ngay từ khi đăng nhập.

### Upload file csv và đọc dữ liệu vào mảng

Thao tác đọc dữ liệu từ các tập tin là quan trọng, giúp chúng ta backup dữ liệu và phục hồi hay import dữ liệu từ các hệ thống khác. Ví dụ dưới đây giúp minh họa cho các bạn thao tác đọc dữ liệu từ tập tin csv và từ đó đưa vào mảng để phục vụ cho việc xử lý.

#### read-csv.php

<?php

$csv = array();

$name\_file = 'cus.csv';

$lines = file($name\_file, FILE\_IGNORE\_NEW\_LINES);

//dua du lieu tu file csv vao mang: foreach ($lines as $key => $value)

{

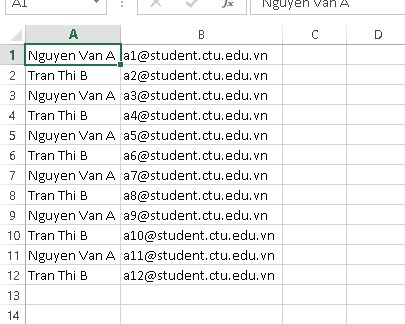
$csv[$key] = str\_getcsv($value);

}

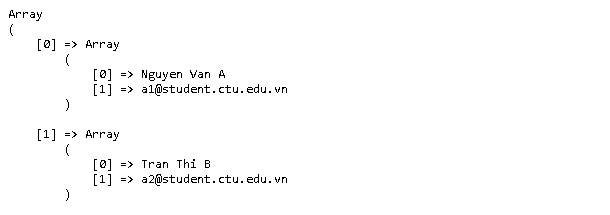
//in mang echo '<pre>'; print\_r($csv); echo '</pre>';

?>

Tạo tập tin cus.csv với dữ liệu như sau

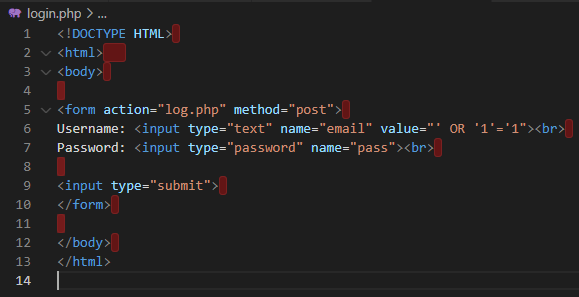


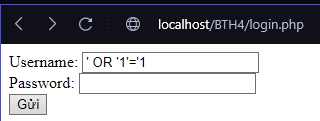
Khi chạy code đọc trên ta sẽ có dữ liệu đã được đưa vào mảng và hiển thị



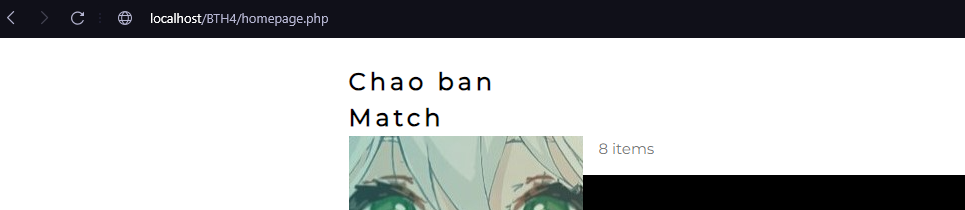
1. Trong vài trường hợp, hacker có thể sử dụng các kỹ thuật SQL Injection để hack hệ thống của bạn. Bạn hãy trình bày SQL Injection là gì và thử các ví dụ trình bày trong [[1]](#footnote-1). Ứng dụng kỹ thuật đó vào trang đăng nhập mà bạn đã tạo, chụp lại các kết quả.

* ***SQL Injection là một kỹ thuật tấn công được sử dụng bởi hacker để xâm nhập vào hệ thống thông qua việc chèn các câu lệnh SQL bất hợp pháp vào các trường dữ liệu nhập của ứng dụng web. Khi ứng dụng web không kiểm tra hoặc xử lý đầu vào của người dùng đúng cách, hacker có thể chèn các đoạn mã SQL độc hại vào các trường dữ liệu đầu vào, từ đó có thể thực hiện các hành động như truy cập, thay đổi hoặc xóa dữ liệu trong cơ sở dữ liệu của ứng dụng.***
* ***Để áp dụng kỹ thuật SQL Injection vào trang đăng nhập, chúng ta có thể thử chèn một đoạn mã SQL độc hại trường đầu vào "email" của biểu mẫu đăng nhập. Đoạn mã này sẽ giả định rằng chúng ta muốn xem thông tin của tất cả các người dùng trong cơ sở dữ liệu.***





Đã vào đc homepage



1. Dựa vào link[[2]](#footnote-2), bạn hãy cho biết Cookie là gì, diễn giải ý nghĩa các tham số trong setcookie(). Cách lưu, lấy, xóa giá trị trong cookie.

***Cookie là một cách để lưu trữ thông tin trên máy tính của người dùng khi họ duyệt qua các trang web. Thông tin này được lưu trữ dưới dạng một cặp tên/giá trị và được gửi từ máy chủ web đến trình duyệt web của người dùng, sau đó được lưu trữ tại đó cho các yêu cầu trang web sau đó***

***setcookie() được sử dụng để tạo cookie. Cú pháp của hàm này là:***

* ***$name: Tên của cookie.***
* ***$value: Giá trị của cookie.***
* ***$expire: Thời gian sống của cookie, tính bằng số giây từ thời điểm hiện tại. Mặc định là 0, nghĩa là cookie sẽ hết hiệu lực khi trình duyệt đóng.***
* ***$path: Đường dẫn trên máy chủ mà cookie có thể truy cập được. Mặc định là '/'.***
* ***$domain: Tên miền của cookie. Mặc định là trống, có nghĩa là cookie chỉ có thể truy cập từ tên miền mà nó được tạo ra.***
* ***$secure: Xác định xem cookie có chỉ được gửi qua kết nối HTTPS hay không. Mặc định là false.***
* ***$httponly: Xác định xem cookie có chỉ được truy cập bằng cách sử dụng HTTP hay không. Mặc định là false***

***Để lưu trữ giá trị trong cookie: setcookie("username", "John", time() + 3600, "/");***

***Để lấy giá trị của một cookie: echo $\_COOKIE["username"];***

***Để xóa giá trị của một cookie:******setcookie("username", "", time() - 3600, "/");***

***// hoặc***

***unset($\_COOKIE["username"]);***

1. Dựa vào link[[3]](#footnote-3), bạn hãy cho biết Session dùng để làm gì. Cách lưu, lấy, xóa giá trị trong Session.

***Session là một cơ chế lưu trữ thông tin trên máy chủ cho mỗi phiên làm việc của người dùng trên trang web. Session cho phép lưu trữ các biến dữ liệu trên máy chủ và gắn chúng với mỗi phiên làm việc của người dùng. Thông tin được lưu trữ trong session thường là các biến phiên như thông tin đăng nhập, giỏ hàng của người dùng, hoặc bất kỳ thông tin nào khác cần được duy trì trong suốt phiên làm việc của họ trên trang web***.

Lưu giá trị vào session:

// Bắt đầu session

session\_start();

// Lưu giá trị vào session

$\_SESSION['username'] = 'John';

Lấy giá trị từ session:

// Bắt đầu session

session\_start();

// Lấy giá trị từ session

$username = $\_SESSION['username'];

echo $username; // In ra: John

Xóa giá trị từ session:

// Bắt đầu session

session\_start();

// Xóa một biến từ session

unset($\_SESSION['username']);

// Hoặc xóa toàn bộ session

session\_destroy();

1. Bạn hãy so sánh Cookie và Session.

***Lưu trữ dữ liệu:***

Cookie: Dữ liệu được lưu trữ trên máy tính của người dùng trong một tệp cookie, được gửi từ máy chủ đến trình duyệt và được gắn liền với mỗi yêu cầu của trang web.

Session: Dữ liệu được lưu trữ trên máy chủ, thông qua một phiên làm việc duy nhất được tạo ra cho mỗi người dùng. Thông tin session được duy trì trên máy chủ và không được lưu trữ trên máy tính của người dùng.

***Thời gian sống:***

Cookie: Có thể thiết lập thời gian sống cho cookie. Cookie có thể tồn tại trong một khoảng thời gian cụ thể hoặc có thể chỉ tồn tại trong một phiên làm việc của trình duyệt.

Session: Thời gian sống của session thường được kiểm soát bởi cấu hình máy chủ. Session thường tồn tại cho đến khi người dùng đóng trình duyệt hoặc hết thời gian không hoạt động.

***An toàn***:

Cookie: Có thể bị sửa đổi hoặc đánh cắp bởi hacker nếu không được cấu hình đúng cách. Thông tin trong cookie có thể được xem trực tiếp bởi người dùng.

Session: Thông tin session được lưu trữ trên máy chủ và không bị nguy cơ bị đánh cắp trực tiếp từ phía người dùng. Tuy nhiên, session có thể bị tấn công bằng cách sử dụng các kỹ thuật như session hijacking hoặc session fixation.

***Sử dụng:***

Cookie: Thích hợp cho việc lưu trữ thông tin nhỏ và ít nhạy cảm như thông tin đăng nhập, thiết lập ngôn ngữ hoặc tham số tuỳ chỉnh.

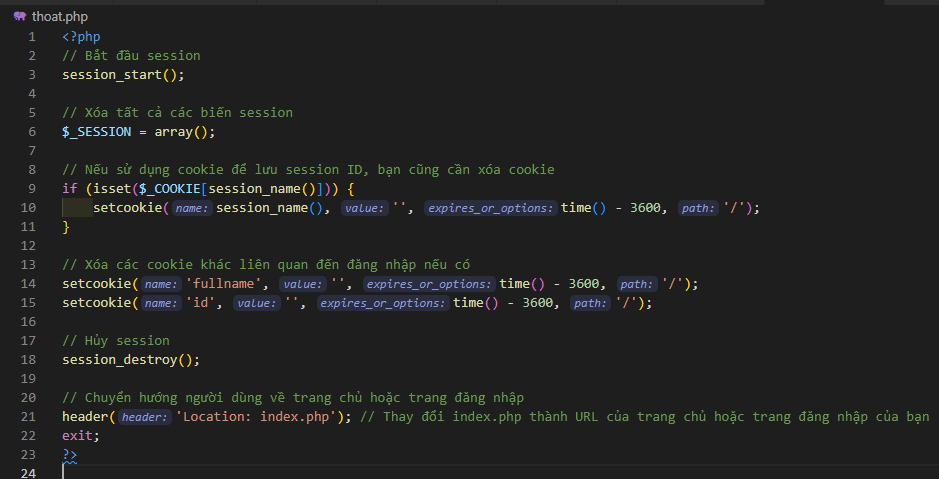
Session: Thích hợp cho việc lưu trữ thông tin lớn hơn và nhạy cảm như thông tin phiên làm việc của người dùng, giỏ hàng của người dùng, hoặc thông tin phiên làm việc khác.

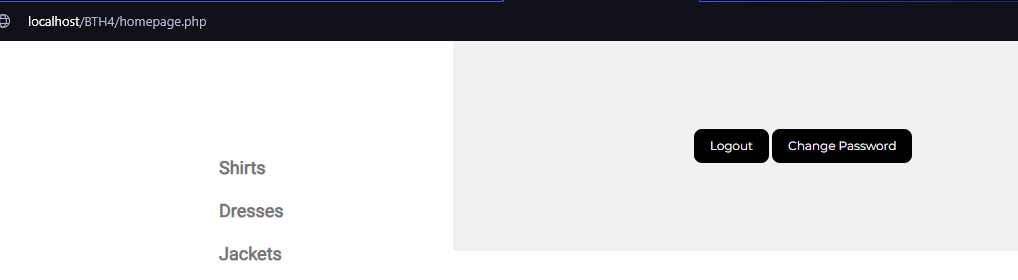
***Hiệu suất:***

Cookie: Do dữ liệu được lưu trữ trên máy tính của người dùng, việc sử dụng cookie có thể giúp giảm tải cho máy chủ.

Session: Do dữ liệu được lưu trữ trên máy chủ, việc sử dụng session có thể tăng tải cho máy chủ nếu có nhiều phiên làm việc được tạo ra.

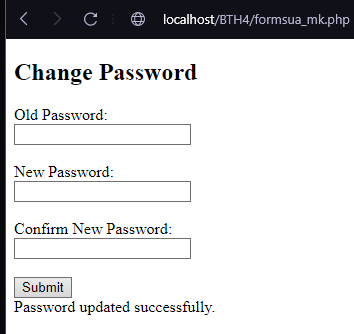
1. Chỉnh sửa các đoạn gán, khởi tạo, lấy giá trị cookie trong log.php và homepage.php thay bằng cách dùng Session. Tạo tập tin thoat.php để xóa các giá trị trong session, cookie thực hiện chức năng log out khỏi hệ thống.





Logout sẽ xóa sạch

1. Bạn hãy tạo form sua\_mk.php cho phép người dùng chỉnh sửa mật khẩu sau khi đăng nhập. Yêu cầu gồm 3 input: ô để nhập mật khẩu cũ, 1 ô để nhập mật khẩu mới, 1 ô cho phép nhập mật lại mật khẩu mới. Khi nhập xong cần đảm bảo: mật khẩu cũ là khớp với CSDL đang lưu, 2 ô nhập mật khẩu mới phải khớp với nhau và không giống với mật khẩu cũ, nếu đáp ứng điều kiện thì tiến hành băm mật khẩu với md5 và lưu mật khẩu mới vào CSDL.



1. Bạn đọc trong[[4]](#footnote-4) để tìm hiểu và mô tả các cơ chế, các hàm/thủ tục để thực hiện việc upload.

PHP cung cấp các cơ chế và hàm/thủ tục để thực hiện việc upload file từ máy tính của người dùng lên máy chủ

Cơ chế upload file trong PHP:

Form HTML: Để cho phép người dùng tải lên file, chúng ta cần tạo một form HTML với thuộc tính enctype="multipart/form-data".

PHP Script: Sau khi người dùng gửi form, chúng ta sử dụng PHP để xử lý file đã được tải lên và lưu trữ nó ở một vị trí cụ thể trên máy chủ.

Các hàm/thủ tục trong PHP để thực hiện việc upload:

move\_uploaded\_file(): Hàm này được sử dụng để di chuyển file đã tải lên từ vị trí tạm thời đến vị trí cố định trên máy chủ. move\_uploaded\_file($file\_tmp, $destination)

$\_FILES superglobal: PHP sử dụng $\_FILES superglobal để truy cập thông tin về file đã tải lên từ form.

is\_uploaded\_file(): Hàm này kiểm tra xem file có phải là một file đã được tải lên hay không.is\_uploaded\_file($filename)

move\_uploaded\_file(): Hàm này được sử dụng để di chuyển file đã tải lên từ vị trí tạm thời đến vị trí cố định trên máy chủ.

is\_uploaded\_file(): Hàm này kiểm tra xem file có phải là một file đã được tải lên hay không.

filesize(): Hàm này trả về kích thước của file (trong byte).

1. Bạn hãy đọc[[5]](#footnote-5), mô tả chức năng của hàm này và các tham số trong hàm (tham khảo thêm từ read-csv.php).

* Hàm file() trong PHP được sử dụng để đọc toàn bộ nội dung của một file và trả về một mảng các dòng trong file đó. Mỗi phần tử trong mảng tương ứng với một dòng trong file.

file(string $filename, int $flags = 0, resource $context = ?): array|false

Tham số:

$filename: Đường dẫn tới file mà bạn muốn đọc.

$flags (Tùy chọn): Cờ điều khiển việc đọc file. Có thể là một hoặc kết hợp của các hằng số sau:

FILE\_USE\_INCLUDE\_PATH: Tìm kiếm file trong include\_path.

FILE\_IGNORE\_NEW\_LINES: Loại bỏ ký tự xuống dòng (newline) khỏi các dòng trong mảng trả về.

FILE\_SKIP\_EMPTY\_LINES: Loại bỏ các dòng trống từ kết quả trả về.

FILE\_APPEND: Nếu file không tồn tại, tạo mới và ghi nối vào cuối file.

$context (Tùy chọn): Một nguồn tài nguyên được tạo bằng stream\_context\_create().

Giá trị trả về:

Hàm file() trả về một mảng chứa các dòng trong file. Nếu không thể đọc file hoặc file không tồn tại, hàm sẽ trả về false.

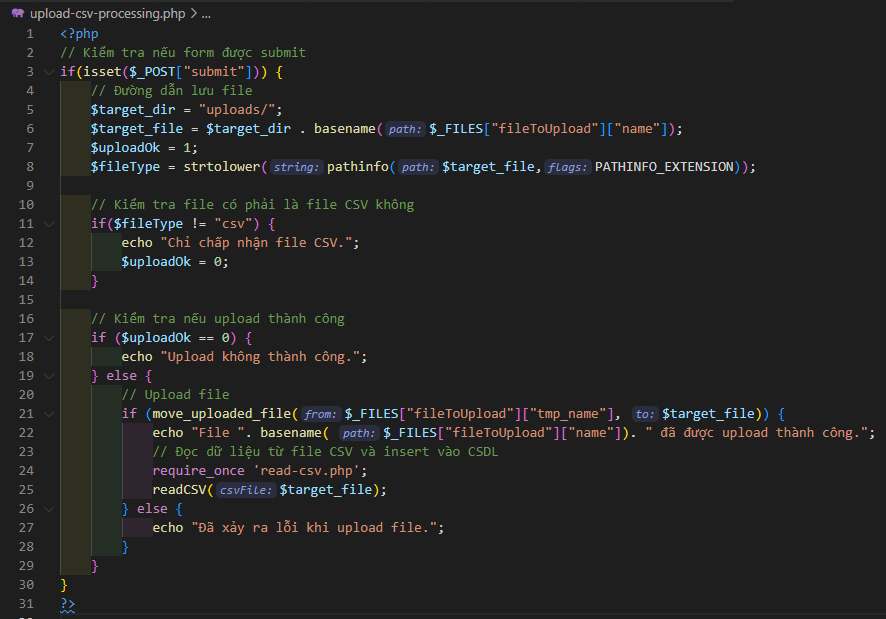
1. Bạn hãy tạo 1 tập tin csv ít nhất 10 dòng dữ liệu với các cột dữ liệu như bảng customer trong csdl qlbanhang. Tạo tập tin upload-csv.php với giao diện cho phép upload các tập

tin csv, và tập tin upload-csv-processing để xử lý nút xử lý sự kiện upload file csv và đưa dữ liệu vào bảng customers trong CSDL. Gợi ý:

○ Xem cấu trúc của bảng customer, mở excel và nhập liệu lưu lại với định dạng CSV.

○ Tham khảo upload-img.php để thiết kế giao diện upload file

○ Tham khảo upload-csdl.php để thiết kế action xử lý việc upload file, chú ý chỉnh sửa loại tập tin chấp nhận các file csv. Tham khảo: 6,[[6]](#footnote-6). Ở đoạn sau khi upload thành công, bạn lấy tên file vừa upload đưa vào hàm đọc tập tin. Bạn tham khảo read-csv.php chỉnh sửa đọc dữ liệu từ csv đưa vào mảng để lần lượt thực hiện insert từng dòng dữ liệu trong csv vào CSDL.





**Chú ý:**

* Các bạn nộp file word: Quy tắc đặt tên file: **<mssv>-<hoten>-<bai><stt\_bai thực hành>.docx** nộp lên Classroom (VD: **B123456-NguyenVanA-bai1.docx**), kèm với các file khác được yêu cầu như phần câu hỏi đã nêu. **Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file zip**. File zip đặt tên như file word.
* Mỗi câu các bạn trả lời bằng hình hoặc dạng text tùy vào yêu cầu của câu hỏi và **TRẢ LỜI THEO ĐÚNG THỨ TỰ CÂU HỎI**. Nếu câu nào không trả lời được các bạn cứ để số thứ tự câu hỏi và bỏ trống phần trả lời.
* Các câu trả lời có tham khảo trên Internet phải trích dẫn link/nguồn.
* *Vi phạm 1 trong các điều sau đây bài thực hành sẽ bị 0 điểm:*

○Đặt tên KHÔNG ĐÚNG quy tắc được yêu cầu.

○ Bài không đủ các thành phần (word, code+data (nếu có),...) đã được yêu cầu.

○ Bài không thực hiện đúng yêu cầu “**Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file .zip”**

○ Bị phát hiện copy, sao chép từ các bạn khác

○ Phần trả lời không ghi rõ trả lời cho câu nào

○ Thứ tự câu trả lời không đúng thứ tự câu hỏi

6 [https://www.php.net/manual/en/features.file-upload.post-method.php,](https://www.php.net/manual/en/features.file-upload.post-method.php)

1. <https://www.w3schools.com/sql/sql_injection.asp> [↑](#footnote-ref-1)
2. <https://www.w3schools.com/php/php_cookies.asp> [↑](#footnote-ref-2)
3. <https://www.w3schools.com/php/php_sessions.asp> [↑](#footnote-ref-3)
4. <https://www.w3schools.com/php/php_file_upload.asp> [↑](#footnote-ref-4)
5. <https://www.w3schools.com/php/func_filesystem_file.asp> [↑](#footnote-ref-5)
6. https://stackoverflow.com/questions/6654351/check-file-uploaded-is-in-csv-format [↑](#footnote-ref-6)