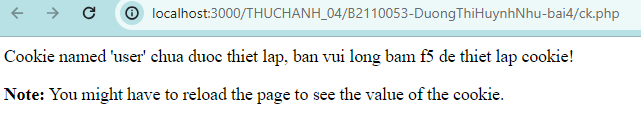
**BÀI THỰC HÀNH 04**

**LẬP TRÌNH PHP NÂNG CAO**

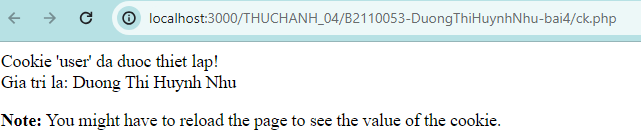
1. ***Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.***

* **Tạo cookie ck.php**

**+** Ban đầu cokie chưa được thiết lập

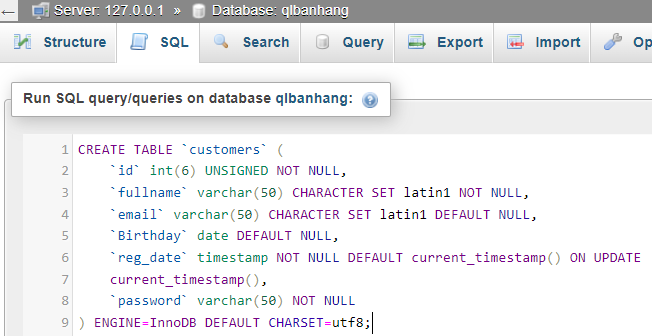


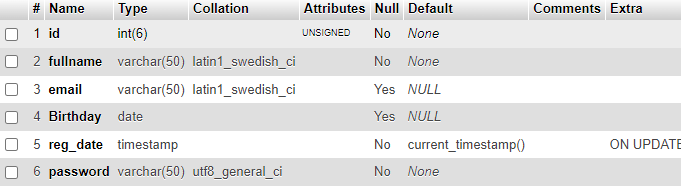
**+** Cookie đã được thiết lập



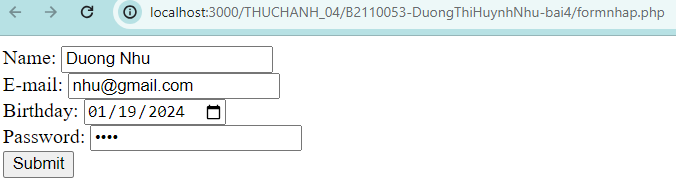
* **Tạo 1 trang đăng ký người dùng với mật khẩu được mã hóa**

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



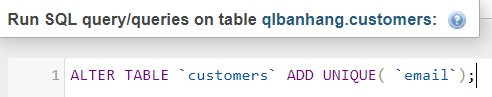


* **Tạo formnhap.php**



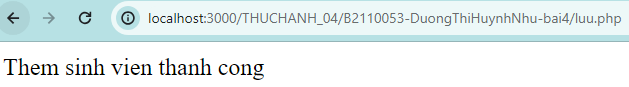
* **Tạo luu.php**

+ Chọn **email** là **unique key**





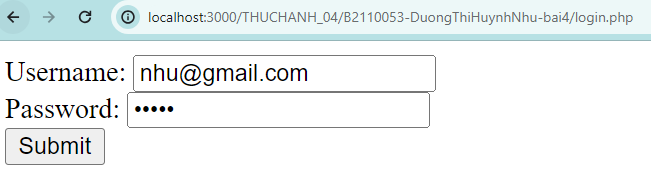
+ Hiển thị khi tạo **luu.php** và nhập thông tin



+ Kết quả lưu thành công

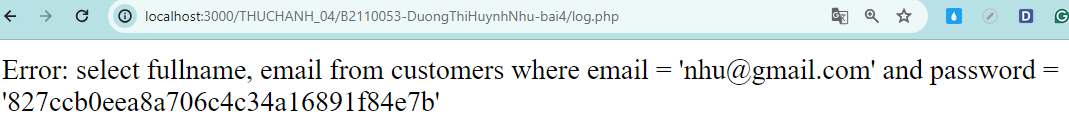


* **Tạo form login.php**

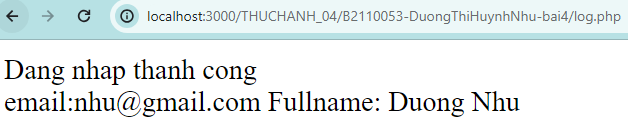


* **Tạo form log.php**

**+** Khi nhập sai:

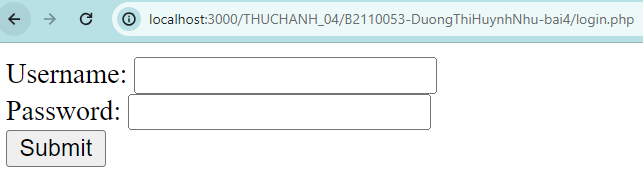


+ Khi nhập đúng:

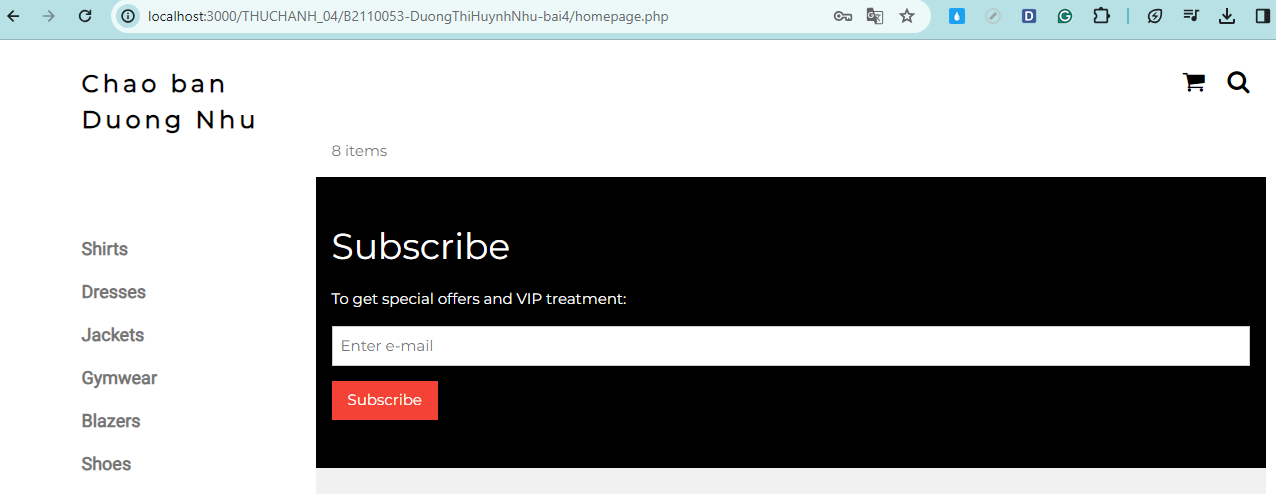


* **Dùng Cookies lưu lại tên đăng nhập, email và điều hướng**

+ Chỉnh sửa lại **log.php** để khi đăng nhập thành công chuyển sang 1 trang chủ, còn nếu không đúng thì trở lại trang đăng nhập (hình bên dưới đăng nhập sai)

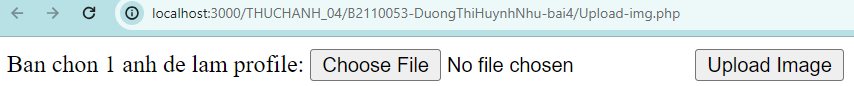


+ Tạo 1 trang chủ để khi đăng nhập thành công chuyển qua trang chủ này **homepage.php** (hình bên dưới đăng nhập thành công)

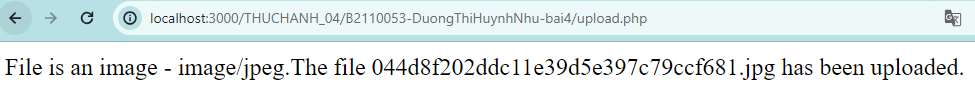


* **Upload tập tin**

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



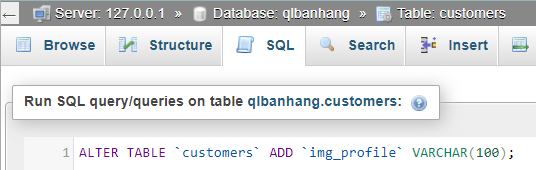
**+** Tập tin **upload.php** xử lý hành động (sau khi upload thành công hiển thị như bên dưới)



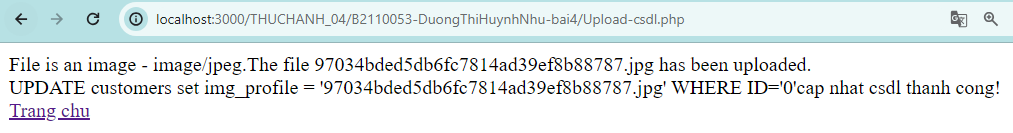
* **Upload ảnh và lưu dữ liệu về ảnh trong CSDL**

+ Trong phần này chúng ta minh họa tải tập tin ảnh làm ảnh profilecủa người dùng +) **Upload-img.php (modified)**

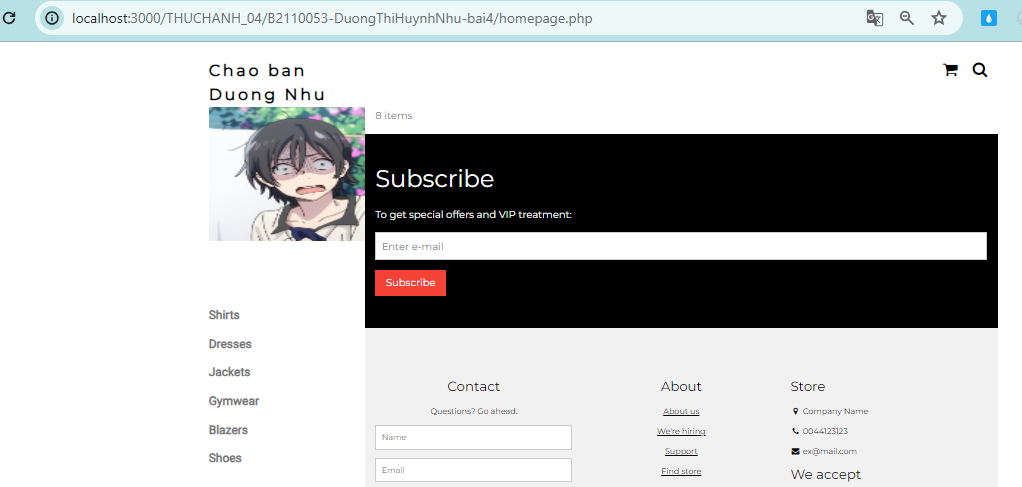
**+)** Thêm cột lưu đường dẫn ảnh



+) **Upload-csdl.php**

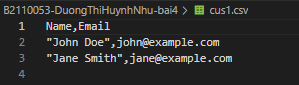
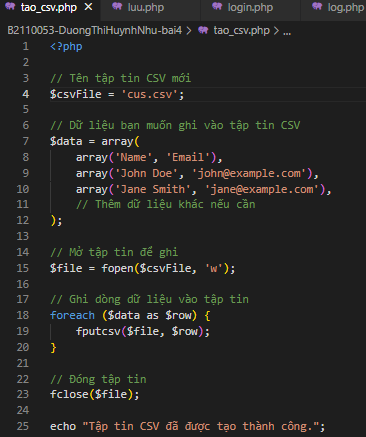


**+) Homepage.php (modified với ảnh profile)**

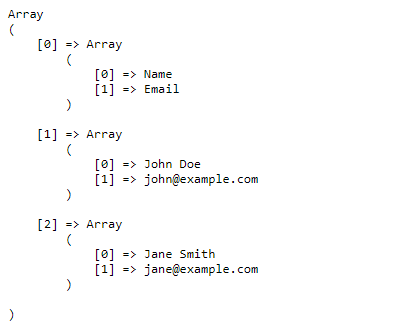


* **Upload file csv và đọc dữ liệu vào mảng**

**+** Tập tin **cus1.csv** có dữ liệu như sau:



+ Kết quả



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 link1. 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 phổ biến trong lĩnh vực bảo mật web, nơi hacker chèn các câu lệnh SQL độc hại vào các trường nhập liệu của ứng dụng web để thực hiện các hoạt động không được phép, như truy cập, thay đổi hoặc xóa dữ liệu trong cơ sở dữ liệu.
* **Ví dụ:** SQL Injection dựa trên **1 = 1** luôn đúng

+) Hacker có thể tạo một bảng cơ sở dữ liệu mới và truy vấn sql như sau:



+) Kết quả trả về dữ liệu của tất cả người dùng gồm email và password (hacker có thể dùng công cụ để mã hóa mật khẩu)



1. ***Dựa vào link2, 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 lưu trữ thông tin nhỏ được gửi từ máy chủ đến trình duyệt người dùng và được lưu trữ trên máy tính của họ. Nó thường được sử dụng để theo dõi thông tin về người dùng, trạng thái đăng nhập, cài đặt trang web, và các thông tin khác liên quan đến trải nghiệm trực tuyến của người dùng.
* **Ý nghĩa các tham số trong setcookie():**

**+**) Cú pháp: **setcookie(name, value, expire, path, domain, secure, httponly);**

1. **name (tên):**

Tên của cookie. (bắt buộc còn lại không bắt buộc)

1. **value (giá trị):**

Giá trị được liên kết với tên cookie.

1. **expire (hết hạn):**

Thời điểm hết hạn của cookie, được xác định dưới dạng timestamp. Nếu không được xác định, cookie sẽ hết hạn khi trình duyệt đóng.

1. **path (đường dẫn):**

Đường dẫn trong trang web mà cookie có thể được truy cập. Nếu được đặt là "/", cookie sẽ có thể được truy cập từ mọi trang trong trang web.

1. **domain (miền):**

Miền mà cookie được liên kết với. Một cookie có thể được chia sẻ giữa các miền con.

1. **secure (bảo mật):**

Nếu được đặt là true, cookie chỉ được gửi qua kết nối bảo mật (HTTPS).

1. **httponly (chỉ dùng trên HTTP):**

Nếu được đặt là true, cookie sẽ chỉ được truy cập thông qua giao thức HTTP và không thể thay đổi bằng cách sử dụng JavaScript.

* **Lưu giá trị trong cookie:**

**setcookie("tên\_cookie", "giá\_trị\_cookie", time() + 3600, "/");**

**VD:** setcookie("user\_id", "123", time() + 3600, "/");

* Tạo một cookie có tên là "user\_id" với giá trị là "123".
* **Lấy giá trị trong cookie:**

**$bien\_cookie = $\_COOKIE["tên\_cookie"];**

**VD:** $user\_id = $\_COOKIE["user\_id"];

* **Xóa giá trị trong cookie:**

**setcookie("tên\_cookie", "", time() - 3600, "/");**

**VD:** setcookie("user\_id", "", time() - 3600, "/");

* Lưu ý: Để xóa cookie, cần đặt thời điểm hết hạn (expire) là một thời điểm trong quá khứ.

1. ***Dựa vào link3, 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, được sử dụng để duy trì trạng thái của người dùng khi họ tương tác với trang web. Nó giúp lưu trữ thông tin trên một thời gian ngắn, thường là trong suốt phiên làm việc của người dùng trên trang web. Session thường được sử dụng để lưu trữ thông tin đăng nhập, giỏ hàng mua sắm, và các thông tin khác liên quan đến trạng thái của người dùng.
* **Lưu giá trị trong Session:**

// Bắt đầu hoặc khôi phục phiên làm việc

**session\_start();**

// Lưu giá trị vào biến session

**$\_SESSION["ten\_bien"] = "gia\_tri";**

**VD:**

session\_start();

$\_SESSION["user\_id"] = 123;

* **Lấy giá trị trong Session:**

// Bắt đầu hoặc khôi phục phiên làm việc

**session\_start();**

// Lấy giá trị từ biến session

**$gia\_tri = $\_SESSION["ten\_bien"];**

**VD:**

session\_start();

$user\_id = $\_SESSION["user\_id"];

* **Xóa giá trị trong Session:**

// Bắt đầu hoặc khôi phục phiên làm việc

**session\_start();**

// Xóa giá trị từ biến session

**unset($\_SESSION["ten\_bien"]);**

**VD:**

session\_start();

unset($\_SESSION["user\_id"]);

* **Lưu ý:**

1. ***Bắt đầu hoặc khôi phục phiên làm việc:***

Trước khi sử dụng hoặc thao tác với Session, bạn cần bắt đầu hoặc khôi phục phiên làm việc bằng hàm **session\_start().**

1. ***Lưu và lấy giá trị:***

Sử dụng mảng **$\_SESSION** để lưu trữ và truy cập giá trị trong Session.

1. ***Xóa giá trị:***

Sử dụng hàm **unset()** để xóa giá trị khỏi Session.

1. ***Kết thúc phiên làm việc:***

Khi phiên làm việc kết thúc (thường là khi người dùng đăng xuất hoặc đóng trình duyệt), Session sẽ tự động kết thúc và thông tin được xóa.

1. ***Bạn hãy so sánh Cookie và Session.***
2. **Địa điểm lưu trữ:**

* ***Cookie:*** Lưu trữ thông tin trên máy tính của người dùng.
* ***Session:*** Lưu trữ thông tin trên máy chủ.

1. **Thời gian tồn tại:**

* ***Cookie:*** Thời gian tồn tại được quyết định bởi thời điểm hết hạn mà bạn thiết lập.
* ***Session:*** Thời gian tồn tại thường là trong suốt phiên làm việc và sẽ bị xóa khi người dùng đóng trình duyệt.

1. **An toàn:**

* ***Cookie:*** Dữ liệu cookie được lưu trữ trên máy tính của người dùng và có thể bị sửa đổi hoặc thay đổi bởi người dùng.
* ***Session:*** Dữ liệu session được lưu trữ trên máy chủ, giúp giảm rủi ro bị thay đổi từ phía người dùng.

1. **Sự bảo mật:**

* ***Cookie:*** Có thể được sử dụng với tùy chọn Secure để chỉ được truyền qua kết nối an toàn (HTTPS). Có thể sử dụng HttpOnly để ngăn chặn truy cập từ JavaScript.
* ***Session:*** Dữ liệu session không được truyền trực tiếp đến máy tính của người dùng, giảm nguy cơ bị đánh cắp thông tin.

1. **Quyền sở hữu:**

* ***Cookie:*** Thuộc về máy tính của người dùng và có thể được chia sẻ giữa các trang web hoặc các miền con của cùng một trang web.
* ***Session:*** Thuộc về máy chủ và không thể chia sẻ giữa các trang web khác nhau.

1. **Khả năng lưu trữ dữ liệu:**

* ***Cookie:*** Thường được sử dụng để lưu trữ những thông tin nhỏ như thông tin đăng nhập, giỏ hàng, cài đặt cá nhân.
* ***Session:*** Thích hợp để lưu trữ các thông tin lớn và nhạy cảm về bảo mật.

1. **Yêu cầu hỗ trợ trên máy chủ:**

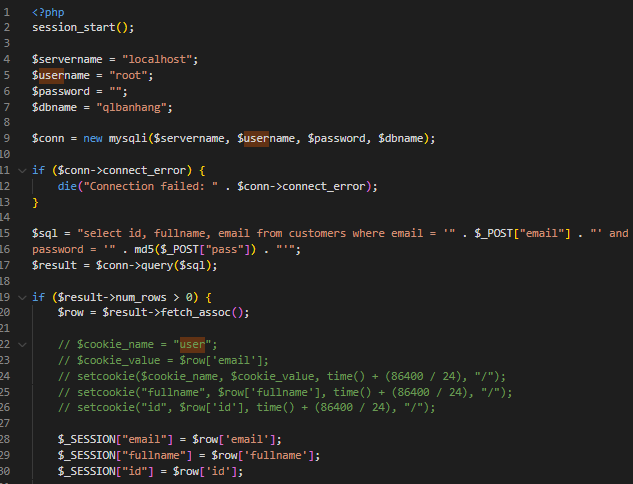
* ***Cookie:*** Không yêu cầu hỗ trợ từ phía máy chủ.
* ***Session:*** Yêu cầu hỗ trợ từ máy chủ để lưu trữ dữ liệu session.

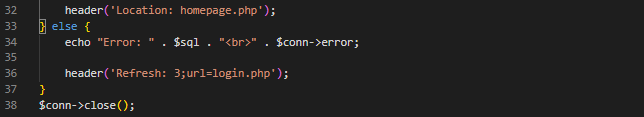
1. **Hiệu suất:**

* ***Cookie:*** Dữ liệu được lưu trữ trên máy tính của người dùng, giúp giảm gánh nặng cho máy chủ.
* ***Session:*** Dữ liệu được lưu trữ trên máy chủ, có thể tạo ra áp lực đối với hiệu suất máy chủ khi số lượng người dùng tăng cao.

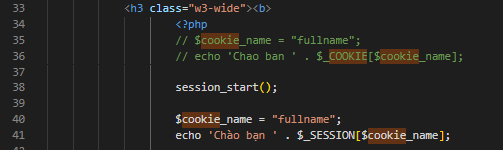
1. ***Chỉnh sửa các đoạn gán, khởi tạo, lấy giá trị cookie trong log.php và homepage.php thay bằng cách dùng Session. Tạo tập tin thoat.php để xóa các giá trị trong session, cookie thực hiện chức năng log out khỏi hệ thống.***

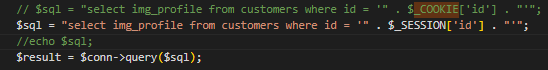
* **Thay đổi bằng cách dùng Session trong log.php**



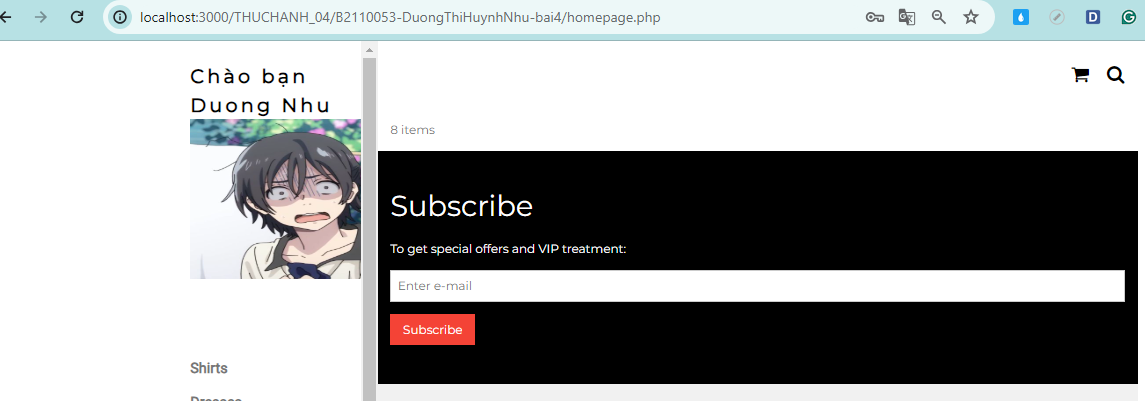


* **Thay đổi bằng cách dùng Session trong homepage.php**



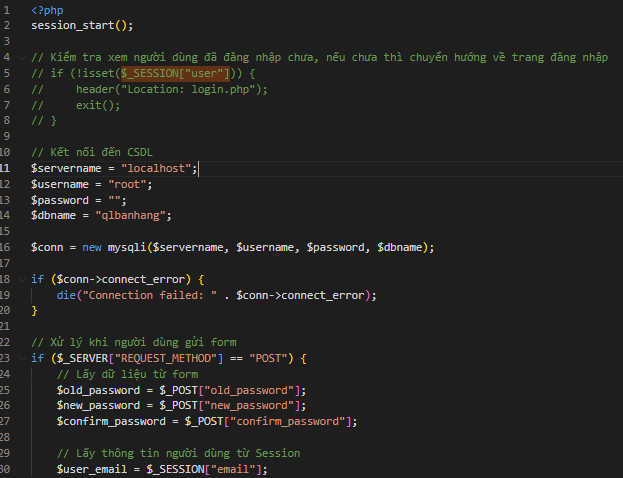


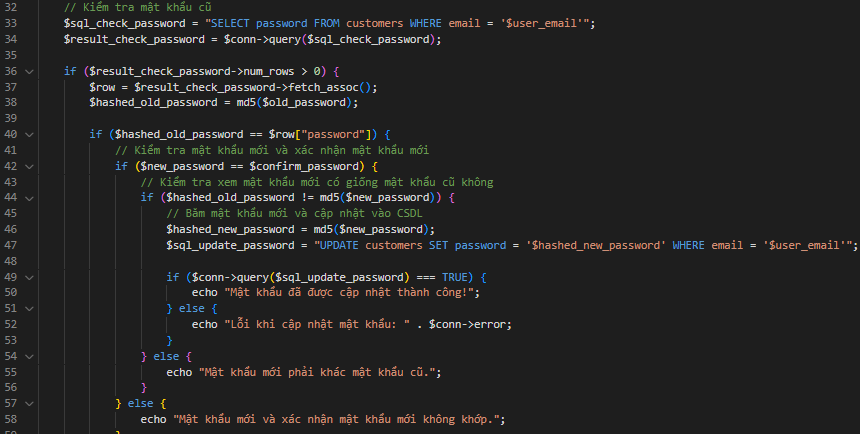
* **Đăng nhập thành công sẽ chuyển sang homepage.php**

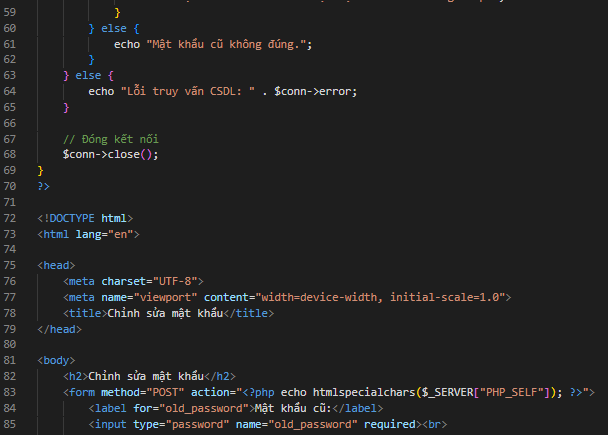


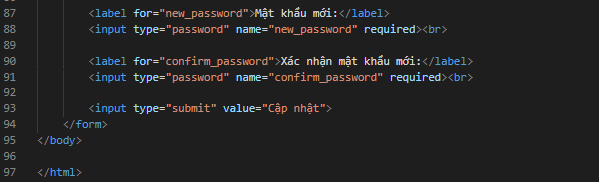
1. ***Bạn hãy tạo form sua\_mk.php cho phép người dùng chỉnh sửa mật khẩu sau khi đăng nhập. Yêu cầu gồm 3 input: ô để nhập mật khẩu cũ, 1 ô để nhập mật khẩu mới, 1 ô cho phép nhập mật lại mật khẩu mới. Khi nhập xong cần đảm bảo: mật khẩu cũ là khớp với CSDL đang lưu, 2 ô nhập mật khẩu mới phải khớp 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.***

* **Form sua\_mk.php**

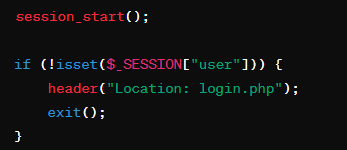








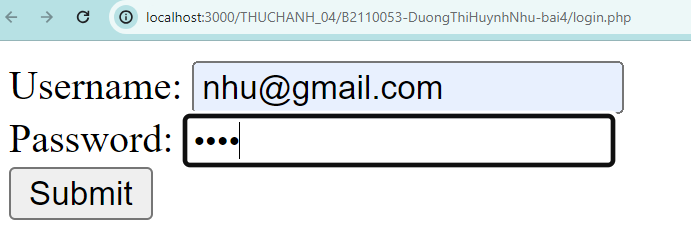
* **Thêm kiểm tra Session ở trang homepage.php**



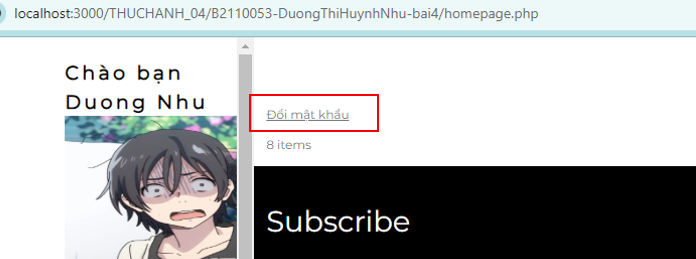
* **Thêm đường link đến trang sua\_mk.php trong homepage.php**



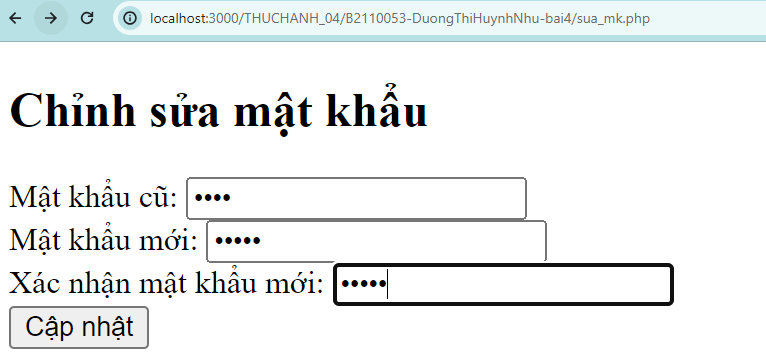
* **Tiến hành đăng nhập**



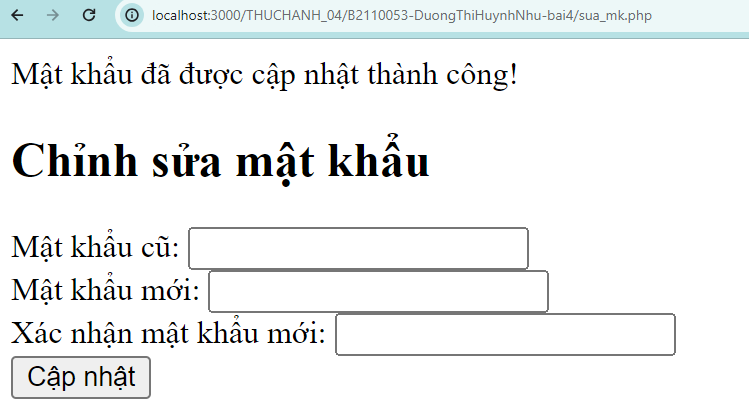
* **Đăng nhập thành công sẽ chuyển tới trang chủ**



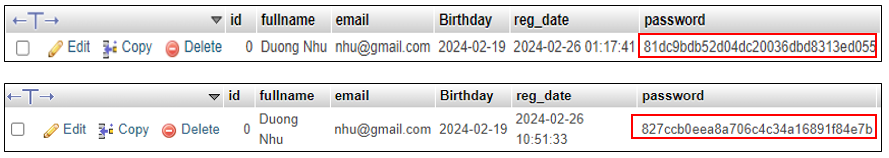
* **Nhấn vào đổi mật khẩu để chuyển hướng sang trang đổi mật khẩu**



* **Thông báo cập nhật mật khẩu thành công**



* **Mật khẩu đã được thay đổi (so sánh trước và sau khi đổi)**



1. ***Bạn đọc trong4 để tìm hiểu và mô tả các cơ chế, các hàm/thủ tục để thực hiện việc upload.***

* **Định cấu hình tệp "php.ini":**

+ Trước tiên, đảm bảo rằng PHP được cấu hình để cho phép tải tệp lên.

+ Trong tệp **"php.ini"**, tìm kiếm lệnh **file\_uploads** và đặt thành **On**



* **Tạo biểu mẫu HTML**

**+** Tạo một biểu mẫu HTML cho phép người dùng chọn tệp hình ảnh họ muốn tải lên



**+** Một số quy tắc cần tuân theo đối với biểu mẫu HTML ở trên:

+) Đảm bảo rằng biểu mẫu sử dụng **method = "post"**

+) Biểu mẫu cũng cần thuộc tính sau: **enctype="multipart/form-data"**. Nó chỉ định loại nội dung nào sẽ sử dụng khi gửi biểu mẫu

* Nếu không có các yêu cầu trên thì việc tải file lên sẽ **không thành công**.

+ Những điều khác cần chú ý:

+) Thuộc tính **type="file"** của thẻ **<input>** hiển thị trường đầu vào dưới dạng điều khiển chọn tệp, với nút **"Submit"** bên cạnh điều khiển đầu vào

+ Biểu mẫu ở trên gửi dữ liệu đến một tệp có tên **"upload.php**", tệp này sẽ tạo tiếp theo.

* **Tạo tập lệnh PHP tệp tải lên**

+ Tệp **"upload.php"** chứa mã để tải tệp lên

+ Tập lệnh PHP giải thích:

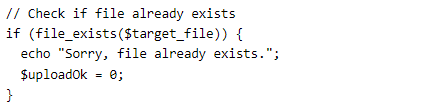
* **$target\_dir = "uploads/"** - chỉ định thư mục nơi tệp sẽ được đặt
* **$target\_file** chỉ định đường dẫn của tệp sẽ được tải lên
* **$uploadOk=1** chưa được sử dụng (sẽ được sử dụng sau)
* **$imageFileType** chứa phần mở rộng tệp của tệp (viết thường)
* Tiếp theo kiểm tra file ảnh là ảnh thật hay ảnh giả

**+ Lưu ý:** Cần tạo một thư mục mới có tên **"uploads"** trong thư mục chứa tệp **"upload.php".** Các tập tin được tải lên sẽ được lưu ở đó.



* **Kiểm tra xem tệp đã tồn tại chưa**

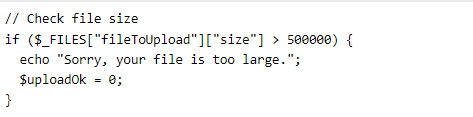
+ Có thể thêm một số **hạn chế**. Đầu tiên, sẽ kiểm tra xem tệp đã **tồn tại** trong thư mục **"uploads"** hay chưa. Nếu đúng như vậy, một **thông báo lỗi sẽ hiển thị** và **$uploadOk** được đặt thành **0**



* **Giới hạn kích thước tệp**

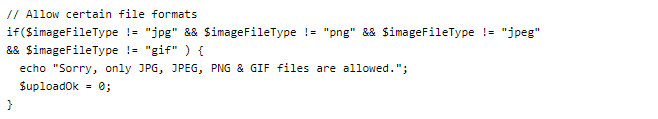
+ **Trường** **nhập tệp** trong biểu mẫu HTML ở trên có tên là **"fileToUpload".**

+ Bây giờ, muốn **kiểm tra kích thước** của tập tin. Nếu tệp **lớn hơn 500KB**, thông báo lỗi sẽ hiển thị và **$uploadOk** được đặt thành **0**

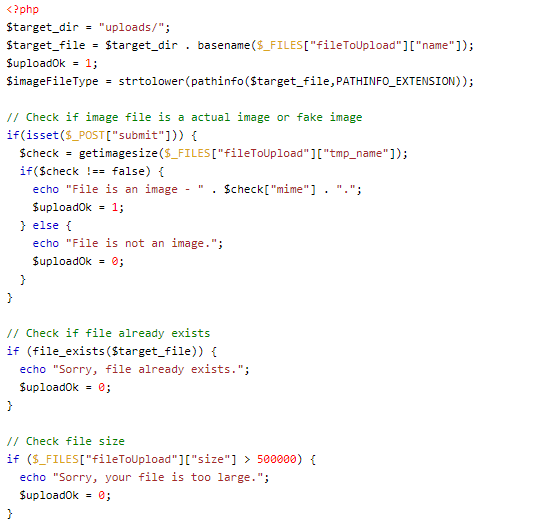


* **Giới hạn loại tệp**

+ Mã bên dưới chỉ cho phép người dùng tải lên các tệp **JPG, JPEG, PNG và GIF**. Tất cả các loại tệp khác đều đưa ra **thông báo lỗi** trước khi đặt **$uploadOk** thành **0**



* **Tập tin upload.php hoàn chỉnh**





***9. Bạn hãy đọc 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:** file() đọc một tập tin vào một mảng. Mỗi phần tử mảng chứa một dòng từ tệp, với ký tự dòng mới vẫn được đính kèm.

+ **Cú pháp:** **file(filename, flag, context)**

**+ Trong đó:**

**filename:** Chỉ đường dẫn tới tệp cần đọc (bắt buộc)

**flag:** Có thể là một hoặc nhiều hằng số sau (không bắt buộc)

* **FILE\_USE\_INCLUDE\_PATH -** Tìm kiếm tệp trong include\_path (trong php.ini)
* **FILE\_IGNORE\_NEW\_LINES -** Bỏ qua dòng mới ở cuối mỗi mảng
* **FILE\_SKIP\_EMPTY\_LINES -** Bỏ qua các dòng trống trong tệp

**context:** Chỉ xử lý ngữ cảnh của tập tin. Ngữ cảnh là một tập hợp các tùy chọn có thể sửa đổi hành vi của luồng. Có thể bỏ qua cách sử dụng NULL.

***10. 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***

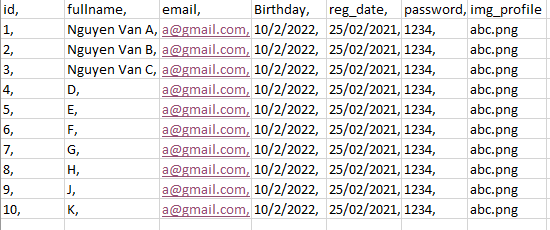
***sửa loại tập tin chấp nhận các file csv. Tham khảo: 6,7. Ở đoạn sau khi upload***

***thành công, bạn lấy tên file vừa upload đưa vào hàm đọc tập tin. Bạn tham khảo***

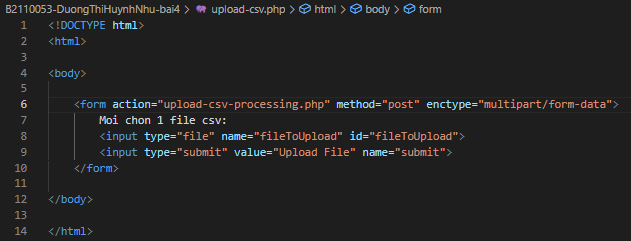
***read-csv.php chỉnh sửa đọc dữ liệu từ csv đưa vào mảng để lần lượt thực hiện***

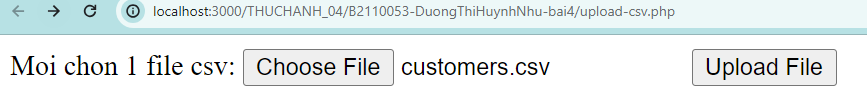
***insert từng dòng dữ liệu trong csv vào CSDL.***

* **Tạo file dữ liệu customers.csv**

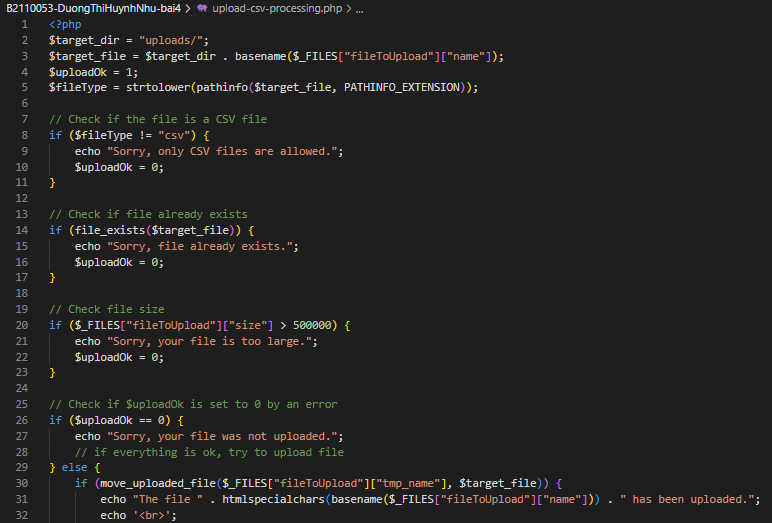


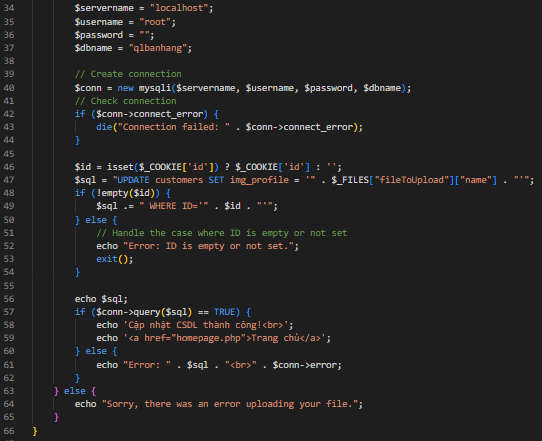
* **Tập tin upload-csv.php**

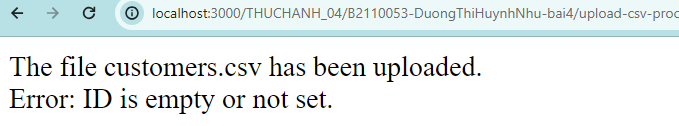




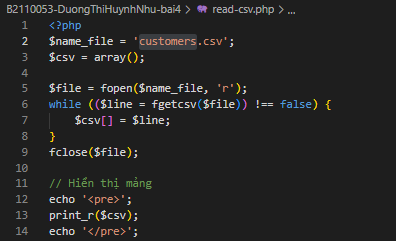
* **Tạo upload-csv-processing**

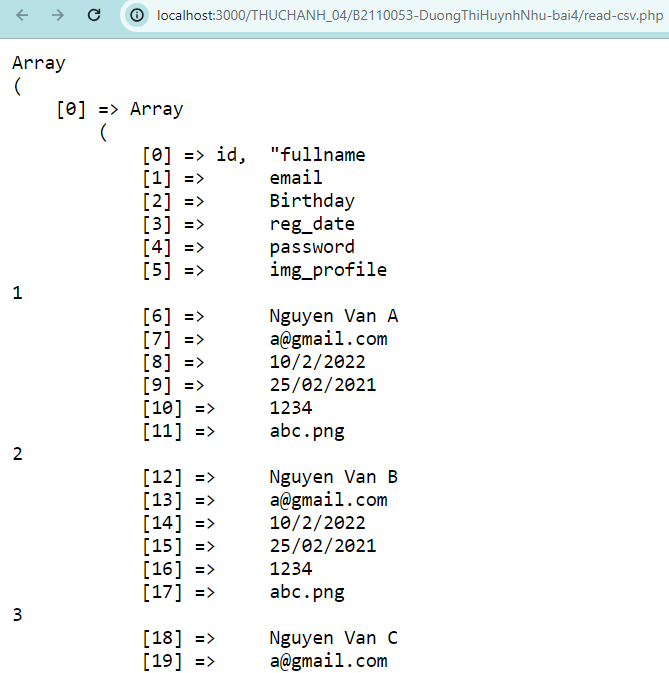


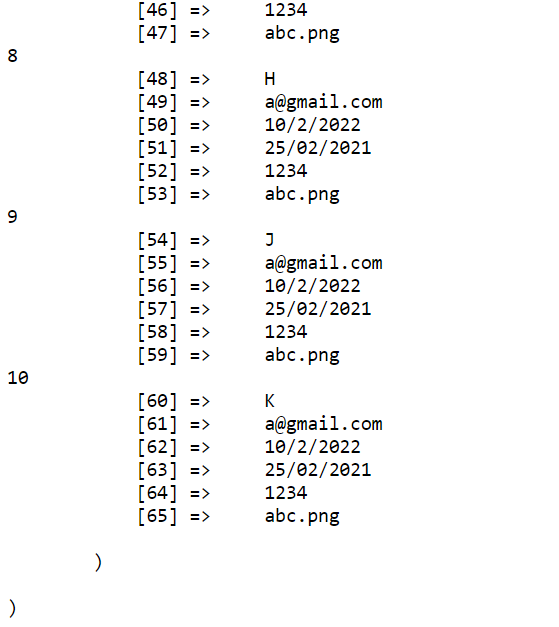




* **Sửa read-csv.php**







**CÁC TÀI LIỆU THAM KHẢO**

**[1]** [**https://chat.openai.com/**](https://chat.openai.com/)

**[2]** [**https://www.w3schools.com/php/**](https://www.w3schools.com/php/)