Bài tập thực hành 03

MSSV: B2203457
Họ và tên sinh viên: LƯU KHẢ NGHỊ

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

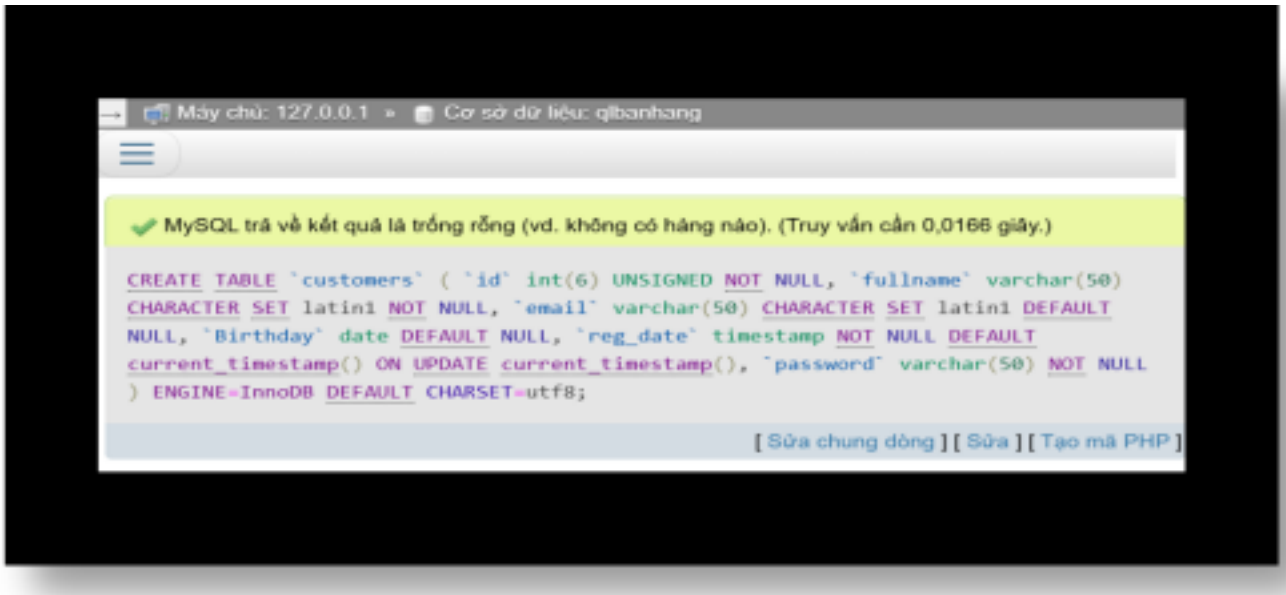
Các bạn đọc và thực hiện viết code như hướng dẫn để làm quen, sau đó đọc trả lời các **Yêu cầu** ở dưới bài hướng dẫn để thực hiện yêu cầu bài thực hành.

Gợi ý Tham khảo:

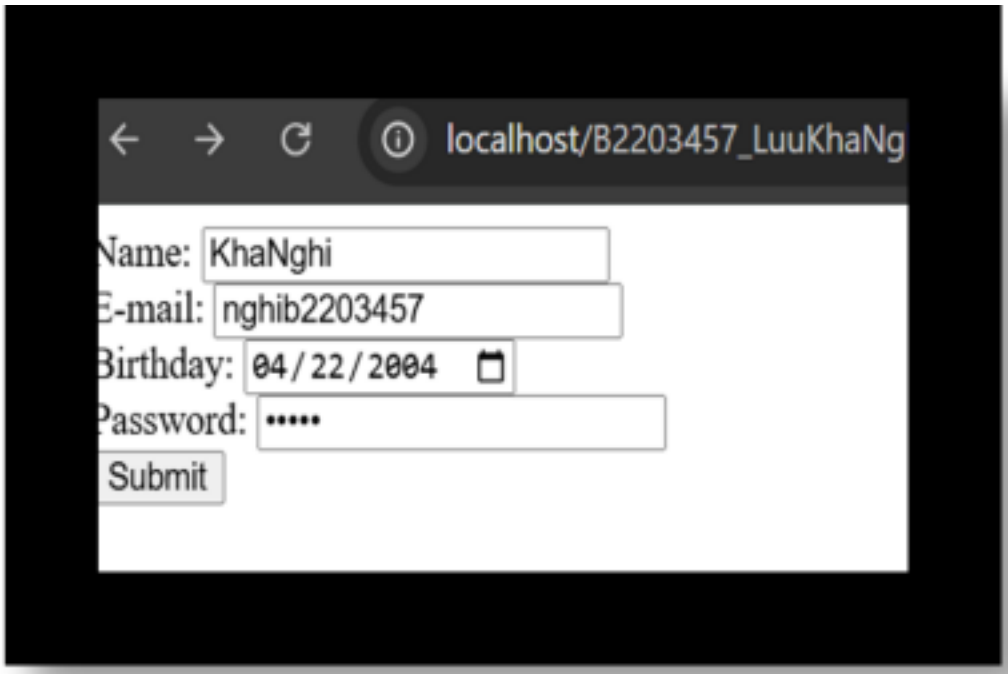
- https://www.w3schools.com/php/
- và các nguồn khác mà sinh viên tìm được

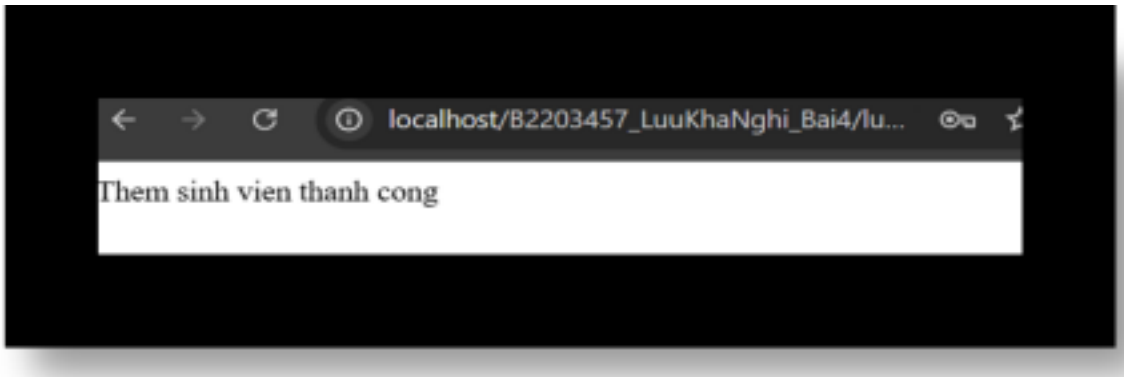
Yêu cầu bài thực hành:

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

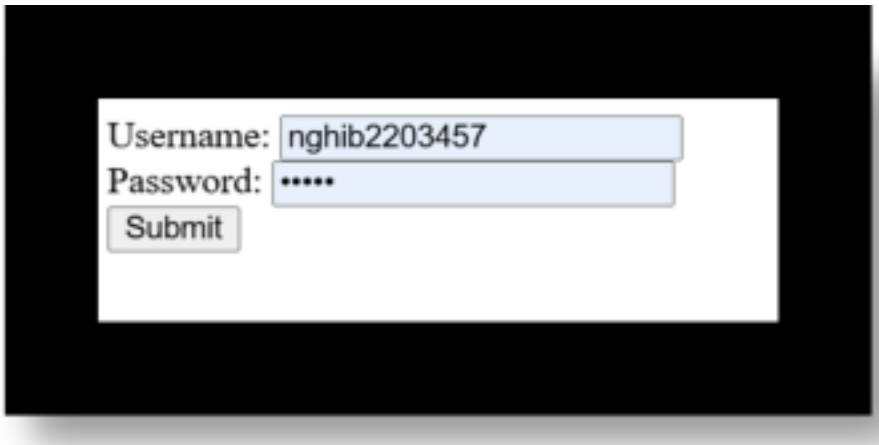


- Tạo 1 trang đăng ký người dùng với mật khẩu được mã hóa:

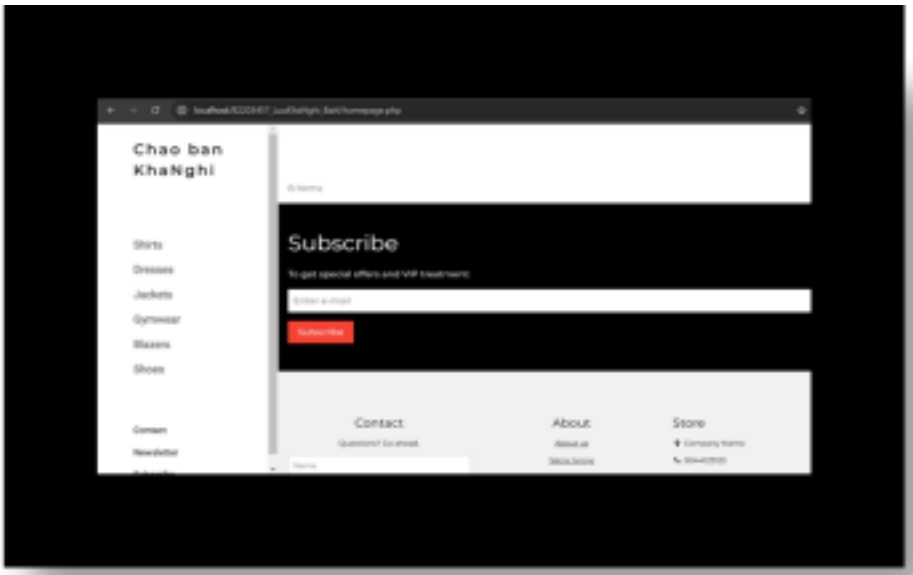




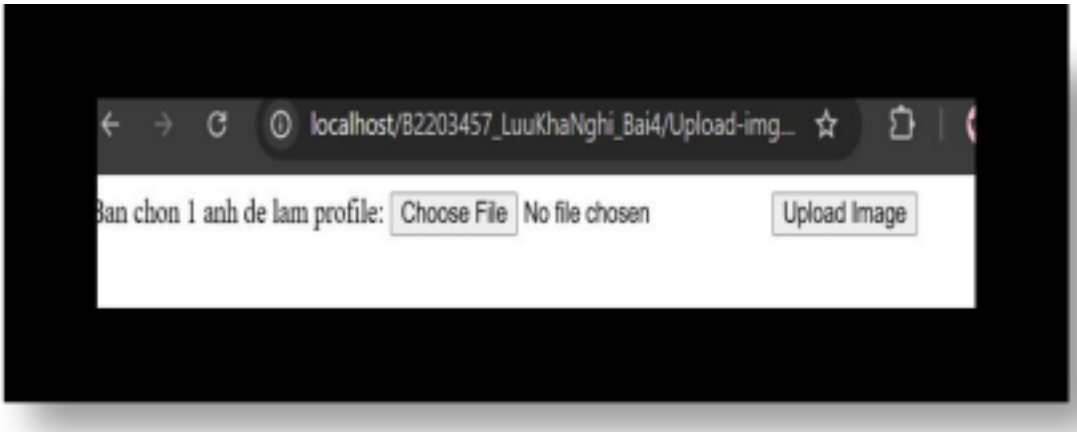
- Tạo form đăng nhập:

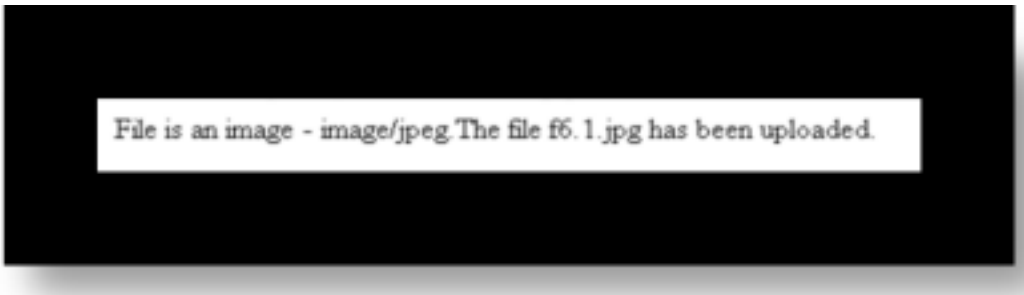


- Dùng Cookies lưu lại tên đăng nhập, email và điều hướng:

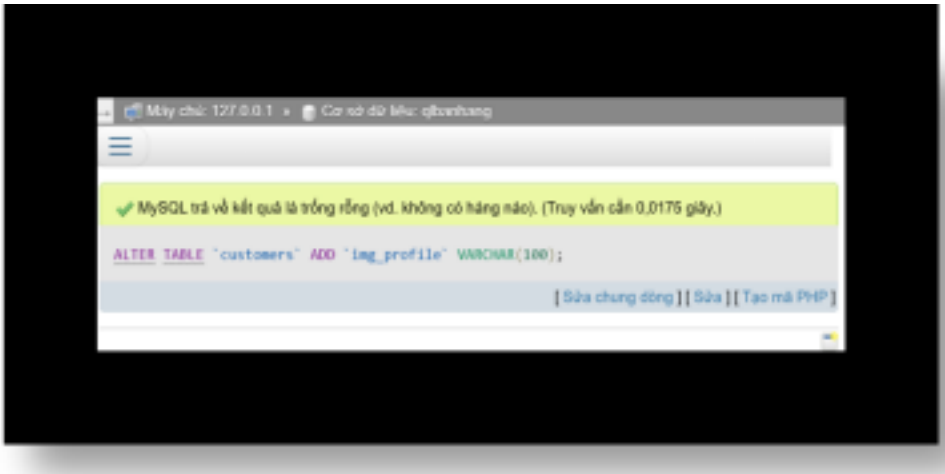


- Upload tập tin:

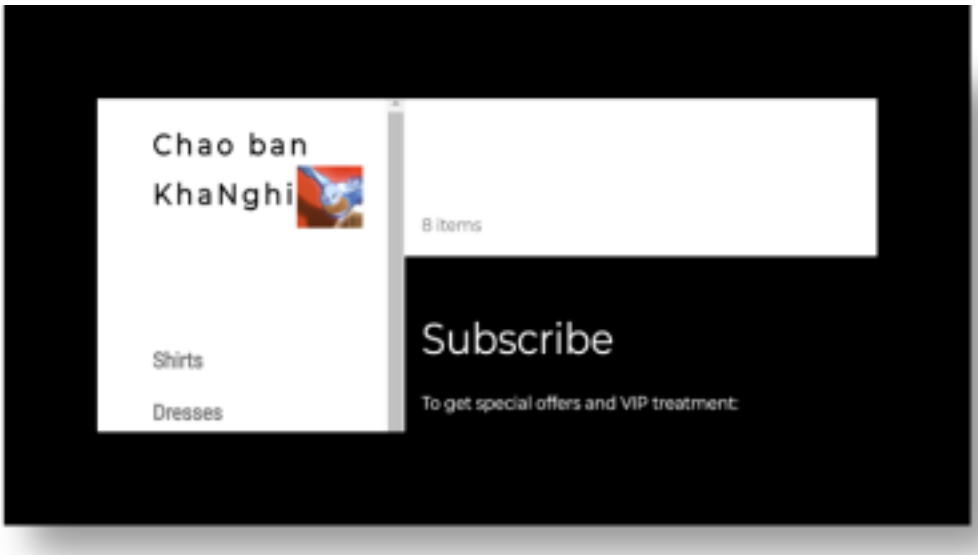




- Upload ảnh và lưu dữ liệu về ảnh trong CSDL

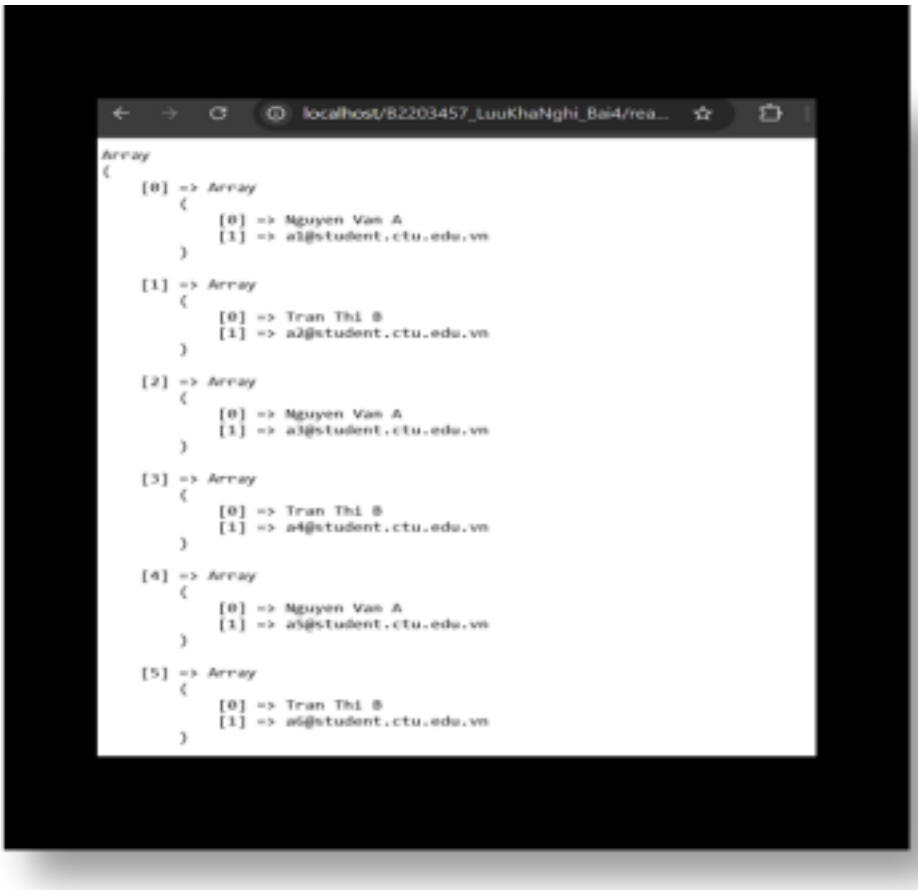


File is an image - image/png.The file tumblr_inline_pg7rfwoGmd1ux8ejd_1280.png has been uploaded.
UPDATE customers set img_profile = 'tumblr_inline_pg7rfwoGmd1ux8ejd_1280.png' WHERE ID='0'cap nhat csdl thanh cong!
Trang chu



id	fullname	email	Birthday	reg_date	password	img_profile
0	KhaNghi	nghib2203457	0000-00-00	2025-02-18 22:21:37	827ccb0eea8a706c4c34a16891f84e7b	Ảnh chụp màn hình 2025-02-18 222118.png

- Upload file csv và đọc dữ liệu vào mảng:



2. ***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]. Ứng dụng kỹ thuật đó vào trang đăng nhập mà bạn đã tạo, chụp lại các kết quả.***

   - SQL Injection (SQLi) là một kỹ thuật tấn công an ninh mà kẻ tấn công có thể khai thác các lỗ hổng trong ứng dụng web thông qua việc chèn mã SQL độc hại vào các truy vấn SQL mà ứng dụng gửi đến cơ sở dữ liệu. Mục tiêu của kỹ thuật này thường là để truy cập, thay đổi hoặc xóa dữ liệu từ cơ sở dữ liệu mà không có quyền hợp lệ. - …

3. ***Dựa vào link[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 đoạn dữ liệu nhỏ mà máy chủ web gửi đến trình duyệt của người dùng và được trình duyệt lưu trữ. Mỗi khi người dùng truy cập lại trang web đó, trình duyệt sẽ gửi cookie tương ứng về máy chủ. Cookie thường được sử dụng để lưu trữ thông tin cần thiết giữa các phiên làm việc của người dùng, chẳng hạn như thông tin đăng nhập, lựa chọn ngôn ngữ, giỏ hàng, và nhiều thông tin khác.

[1] - https://www.w3schools.com/sql/sql_injection.asp

[2] https://www.w3schools.com/php/php_cookies.asp

- Hàm `setcookie()` trong PHP được sử dụng để gửi cookie từ máy chủ đến trình duyệt. Cú pháp cơ bản của hàm này như sau:
setcookie(name, value, expire, path, domain, secure, httponly);
+ name: Tên của cookie. Đây là tham số bắt buộc.
+ value: Giá trị của cookie, có thể là chuỗi hoặc giá trị số. Đây cũng là tham số bắt buộc.
+ expire: Thời gian hết hạn của cookie, được tính bằng giây từ thời điểm UNIX timestamp. Nếu không được thiết lập, cookie sẽ tự động biến mất khi trình duyệt đóng lại.
+ path: Đường dẫn trên máy chủ nơi cookie có sẵn; nếu không được xác định, cookie sẽ chỉ có sẵn trong thư mục chứa trang đã đặt cookie đó.
+ domain: Tên miền mà cookie có thể truy cập. Nếu không được thiết lập, cookie sẽ chỉ có hiệu lực trong miền hiện tại.
+ secure: Tham số này chỉ định rằng cookie chỉ được gửi qua kết nối HTTPS nếu giá trị được đặt là true.
+ httponly: Nếu tham số này được thiết lập, cookie sẽ không thể truy cập thông qua JavaScript, giúp bảo vệ cookie khỏi một số loại tấn công XSS. - Lưu giá trị trong cookie

```
// Lưu cookie với tên "username" và giá trị "JohnDoe", hết hạn sau 1 giờ
setcookie("username", "JohnDoe", time() + 3600, "/");
```

- Lấy giá trị từ cookie

```
if (isset($_COOKIE["username"])) {
 echo "Giá trị của cookie username là: " . $_COOKIE["username"];
} else {
 echo "Cookie chưa được đặt.";
}
```

- **Xóa giá trị trong cookie**

```
// Xóa cookie username
setcookie("username", "", time() - 3600, "/");
```

4. Dựa vào link[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 tạm thời trên máy chủ để giữ trạng thái của người dùng trong suốt phiên làm việc của họ với ứng dụng web. Mỗi khi người dùng truy cập trang web, một session mới sẽ được tạo ra và một ID duy nhất sẽ được gán cho phiên đó. Session thường được sử dụng để lưu trữ thông tin như dữ liệu người

[3] https://www.w3schools.com/php/php_sessions.asp

dùng đã đăng nhập, giỏ hàng, tùy chọn người dùng, và nhiều thông tin khác.

- Những lợi ích của Session:
  + Bảo mật hơn cookie: Dữ liệu của session được lưu trên máy chủ, giúp bảo vệ thông tin nhạy cảm tốt hơn so với cookie, nơi mà dữ liệu được lưu trên máy khách.
  + Đồng bộ dữ liệu: Giúp quản lý và duy trì trạng thái giữa các trang. Dữ liệu được lưu trữ trong session có thể được truy cập từ bất kỳ trang nào trong cùng một ứng dụng.
  + Hết hạn: Session tự động hết hạn sau một khoảng thời gian không hoạt động, giúp bảo vệ thông tin.

- Khởi tạo Session: gọi hàm session_start();

- Lưu giá trị vào Session

```
// Lưu thông tin người dùng vào session
$_SESSION["username"] = "JohnDoe";
$_SESSION["role"] = "admin";
```

- Lấy giá trị từ Session

```
// Kiểm tra và lấy giá trị từ session
if (isset($_SESSION["username"])) {
 echo "Xin chào, " . $_SESSION["username"];
} else {
 echo "Bạn chưa đăng nhập.";
}
```

- Xóa giá trị trong Session

```
// Xóa giá trị username khỏi session
unset($_SESSION["username"]);
```

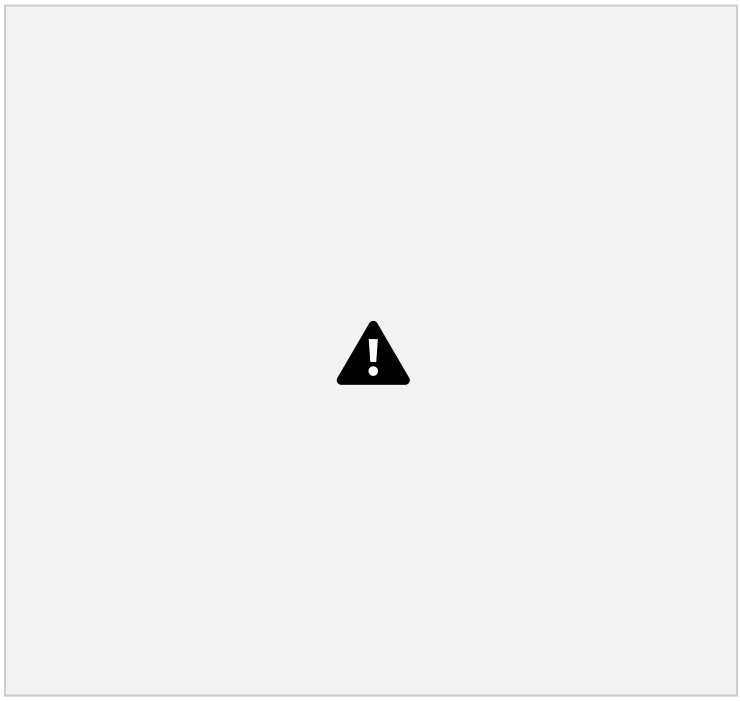
5. Bạn hãy so sánh Cookie và Session.

Tiêu chí	**Cookie**	**Session**
Định nghĩa	Là một đoạn dữ liệu nhỏ được lưu trữ trên máy khách (trình duyệt).	Là một lưu trữ tạm thời trên máy chủ.
Nơi lưu trữ	Lưu trữ trên máy tính của người dùng (trình duyệt).	Lưu trữ trên máy chủ web.

Kích thước	Giới hạn thường lên đến 4KB cho mỗi cookie.	Không có hạn chế cụ thể về kích thước, phụ thuộc vào bộ nhớ máy chủ.
Thời gian sống	Có thể được cấu hình để hết hạn sau một khoảng thời gian nhất định hoặc cho đến khi trình duyệt đóng lại.	Thường tự động hết hạn sau một khoảng thời gian không hoạt động hoặc khi người dùng đăng xuất.
Sử dụng	Phù hợp cho các thông tin không nhạy cảm, như tùy chọn giao diện người dùng, ngôn ngữ, hoặc lưu trữ thông tin không đăng nhập.	Phù hợp cho thông tin nhạy cảm như dữ liệu người dùng đã xác thực, giỏ hàng, hoặc thông tin cần thiết trong quá trình giao dịch.
Bảo mật	Không an toàn, dễ bị tấn công (như XSS, CSRF) vì dữ liệu lưu trên máy khách.	An toàn hơn, vì dữ liệu không lưu trữ trên máy khách và được bảo vệ trên máy chủ.
Khả năng truy cập	Có thể dễ dàng truy cập qua JavaScript và có thể bị ảnh hưởng bởi các lỗ hổng an ninh.	Chỉ có thể truy cập thông qua server-side code, hạn chế việc can thiệp trái phép từ phía client.
Thao tác	Có thể đọc và ghi giá trị từ trình duyệt.	Thao tác thông qua các hàm như session_start(), $_SESSION, session_destroy().

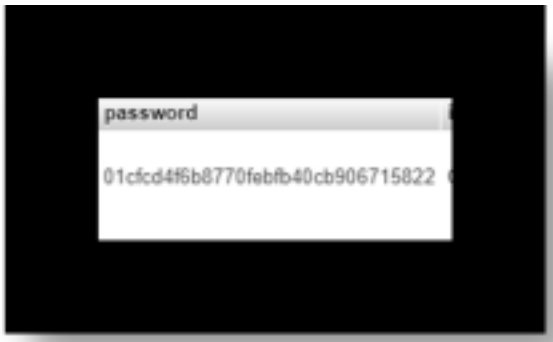
6. ***Chỉnh sửa các đoạn gán, khởi tạo, lấy giá trị cookie trong log.php và homepage.php thay bằng cách dùng Session. Tạo tập tin thoat.php để xóa các giá trị trong session, cookie thực hiện chức năng log out khỏi hệ thống.*** (sửa trong code)
7. ***Bạn hãy tạo form sua_mk.php cho phép người dùng chỉnh 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.



-Trước khi sửa:

-Sau khi sửa:



8. Bạn đọc trong[4] để tìm hiểu và mô tả các cơ chế, các hàm/thủ tục để thực hiện việc upload.

- Việc upload tệp trong PHP liên quan đến nhiều hàm và cơ chế để quản lý việc gửi tệp từ phía client lên server. Dưới đây là mô tả các bước cơ bản và các thành phần quan trọng liên quan đến upload tệp:

- Trước tiên, bạn cần một form HTML với thuộc tính `enctype` để cho phép upload tệp. Trong file `upload.php`, bạn có thể xử lý tệp được gửi từ form. Sau đó sử dụng các hàm và thủ tục liên quan, cuối cùng là kiểm tra lỗi và bảo vệ ứng dụng của bạn khỏi việc upload tệp bừa bãi. - Các biến toàn cục liên quan đến upload:

+ **`$_FILES`**: Biến toàn cục này chứa thông tin về tệp được upload:

  • `$_FILES['file']['name']`: Tên tệp.
  • `$_FILES['file']['type']`: Kiểu tệp.
  • `$_FILES['file']['tmp_name']`: Đường dẫn tệp tạm thời trên server.
  • `$_FILES['file']['error']`: Mã lỗi nếu có vấn đề xảy ra trong quá trình upload. • `$_FILES['file']['size']`: Kích thước tệp.

- Các hàm và thủ tục liên quan

• **`move_uploaded_file()`**: Hàm này được sử dụng để di chuyển tệp từ thư mục tạm thời (địa chỉ trong `$_FILES['file']['tmp_name']`) đến vị trí mong muốn trên server. • **`is_uploaded_file()`**: Kiểm tra xem tệp đã được upload qua HTTP hay chưa. - Kiểm tra lỗi:

  • **UPLOAD_ERR_OK**: Không có lỗi.
  • **UPLOAD_ERR_INI_SIZE**: Tệp vượt quá kích thước cho phép trong `php.ini`. • **UPLOAD_ERR_FORM_SIZE**: Tệp vượt quá kích thước chỉ định trong form. • **UPLOAD_ERR_PARTIAL**: Tệp chỉ được tải lên một phần.
  • **UPLOAD_ERR_NO_FILE**: Không có tệp nào được tải lên.
  • **UPLOAD_ERR_NO_TMP_DIR**: Không có thư mục tạm.
  • **UPLOAD_ERR_CANT_WRITE**: Không thể ghi vào đĩa.
  • **UPLOAD_ERR_EXTENSION**: Một phần mở rộng đã dừng upload.

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).

- [4] https://www.w3schools.com/php/php_file_upload.asp

[5] https://www.w3schools.com/php/func_filesystem_file.asp

- Hàm `file()` trong PHP được sử dụng để đọc nội dung của một tệp và trả về nội dung đó dưới dạng một mảng. Mỗi phần tử trong mảng tương ứng với một dòng trong tệp, bao gồm cả ký tự xuống dòng.
- Chức năng của hàm `file()`

  - **Đọc tệp**: Hàm này cho phép bạn đọc toàn bộ nội dung của một tệp và lưu từng dòng vào các phần tử của một mảng.
  - **Trả về mảng**: Kết quả trả về là một mảng, trong đó mỗi phần tử là một dòng của tệp. Nếu có lỗi xảy ra trong quá trình đọc tệp, hàm sẽ trả về `FALSE`.

- Cú pháp: file(string $filename, int $flags = 0, resource $context = null): array|false - Các tham số của hàm `file()`

  `$filename` (Bắt buộc):

  - Đây là đường dẫn đến tệp mà bạn muốn đọc.

  `$flags` (Tùy chọn):

  - Có thể là một hoặc nhiều hằng số sau:
    - **`FILE_USE_INCLUDE_PATH`**: Tìm kiếm tệp trong `include_path` được định nghĩa trong `php.ini`.
    - **`FILE_IGNORE_NEW_LINES`**: Bỏ qua ký tự xuống dòng ở cuối mỗi phần tử của mảng.
    - **`FILE_SKIP_EMPTY_LINES`**: Bỏ qua các dòng trống trong tệp.

  `$context` (Tùy chọn):

  - Đây là ngữ cảnh của handle tệp. Ngữ cảnh là một tập hợp các tùy chọn có thể thay đổi hành vi của một luồng. Nếu không cần thiết, bạn có thể bỏ qua tham số này bằng 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

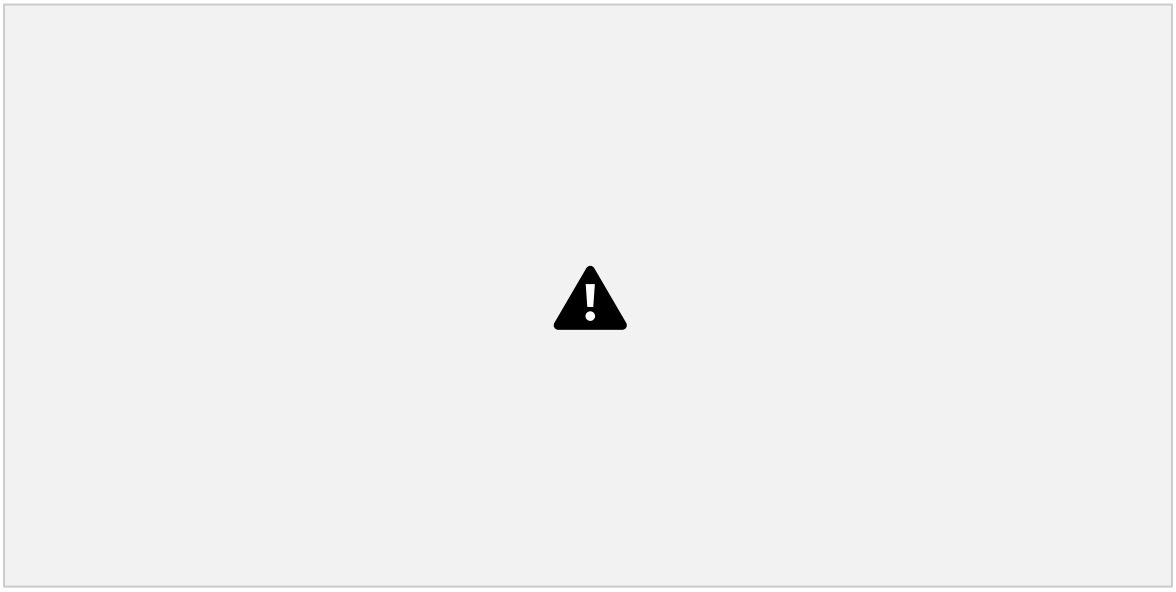
sự kiện upload file csv và đưa dữ liệu vào bảng customers trong CSDL. Gợi ý:

- ○ 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

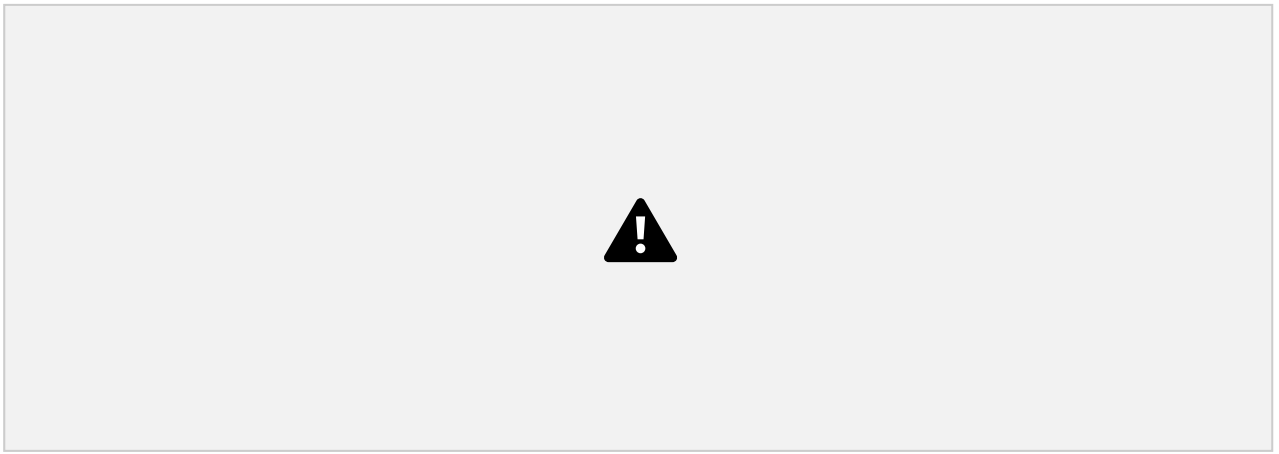
- [6] https://www.php.net/manual/en/features.file-upload.post-method.php,

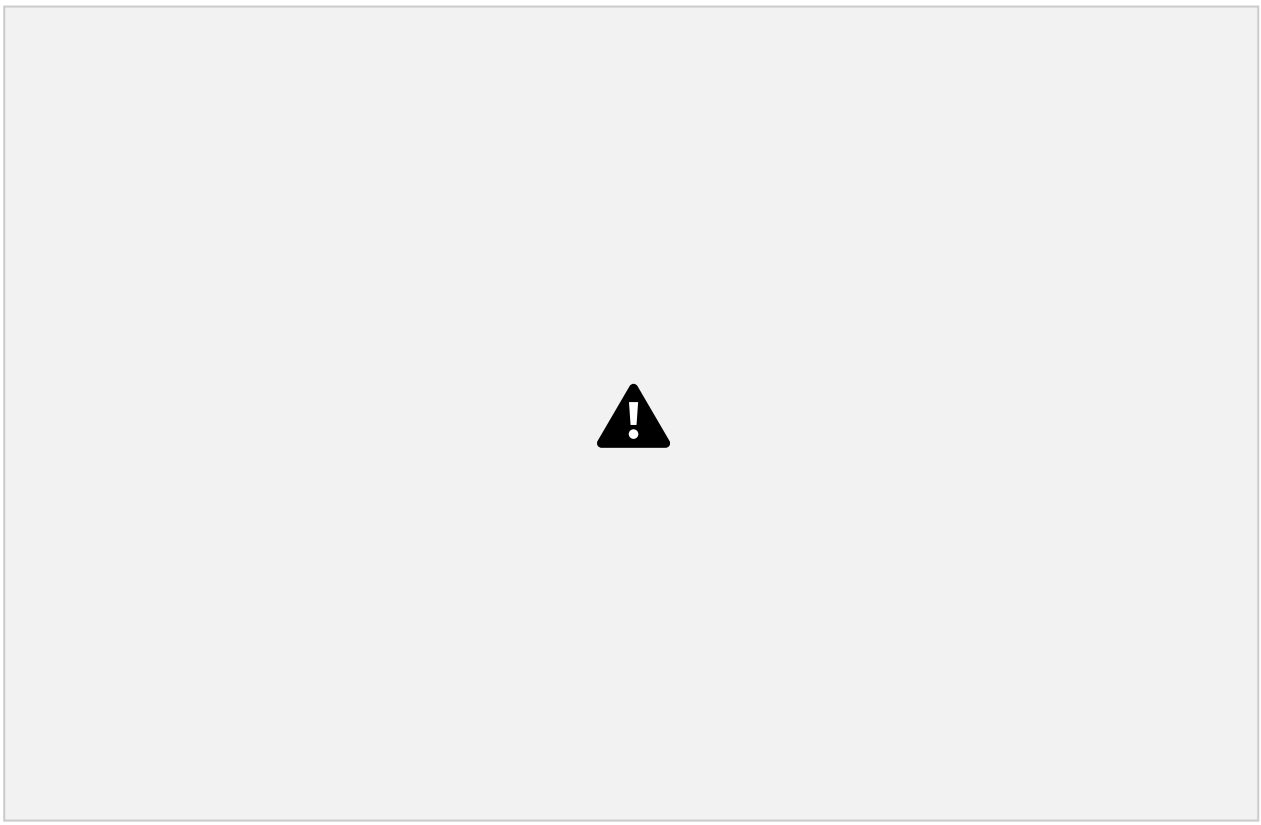
[7] https://stackoverflow.com/questions/6654351/check-file-uploaded-is-in-csv-format

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