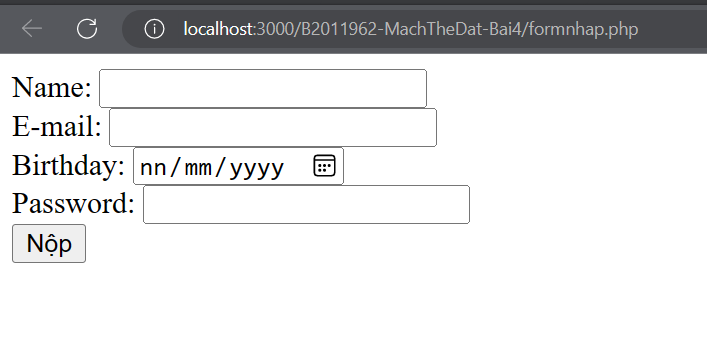
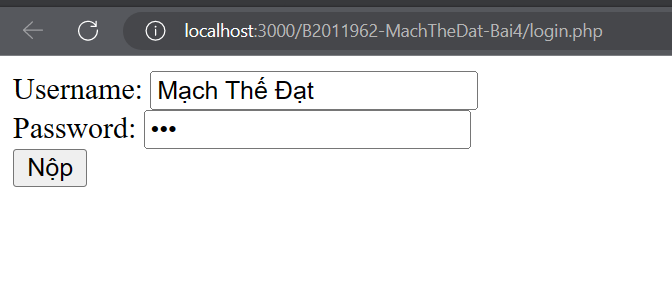
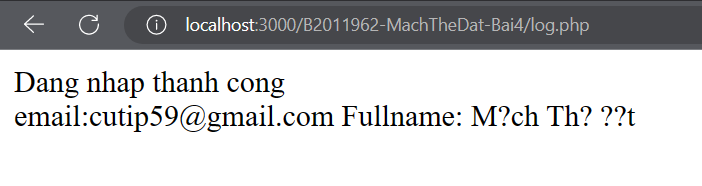
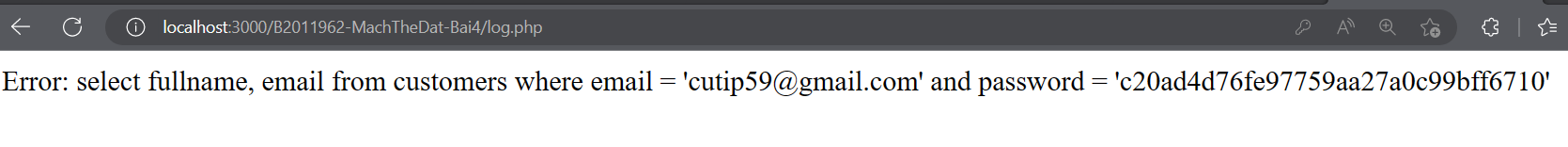
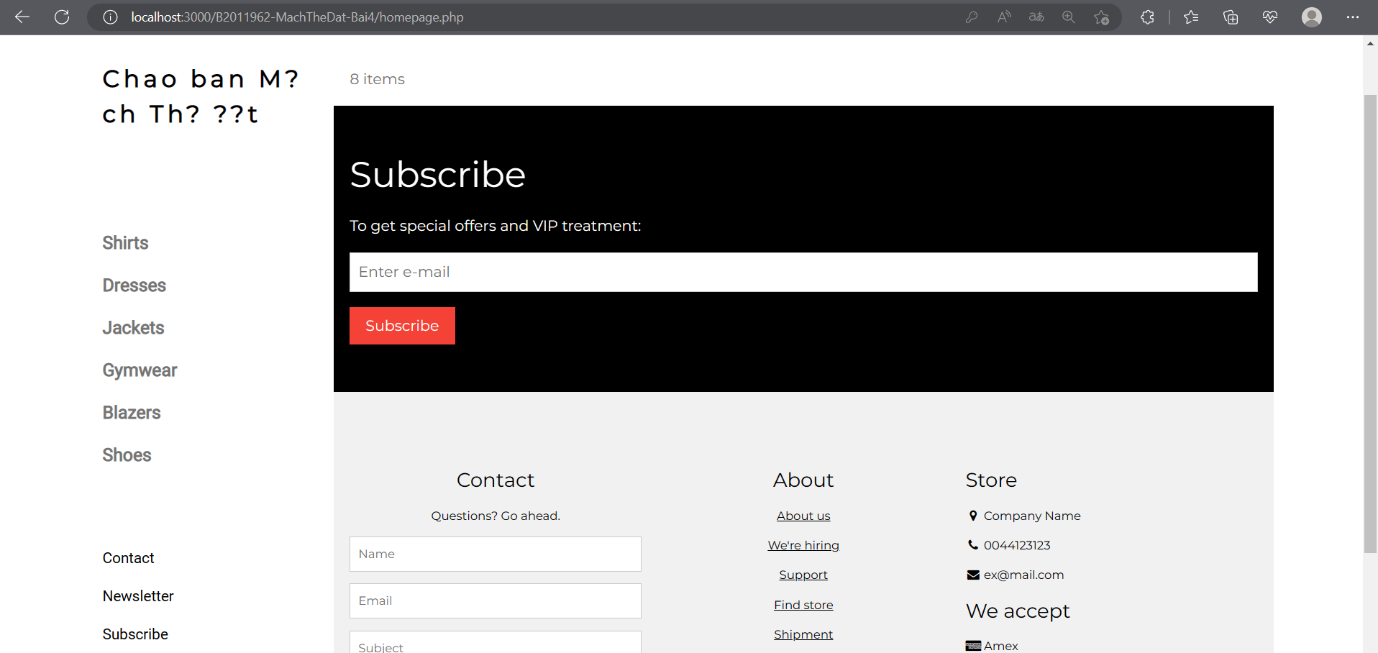
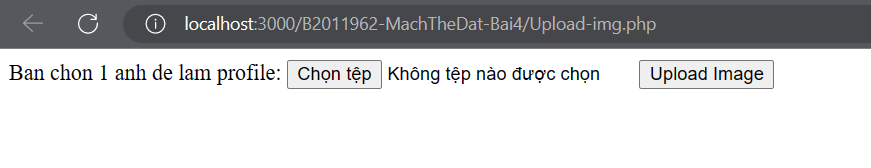
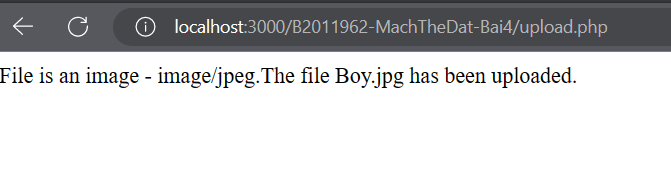
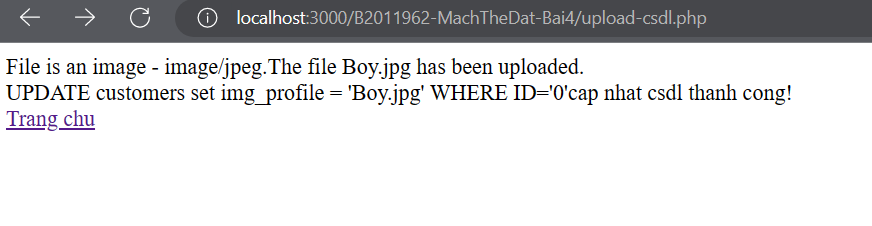
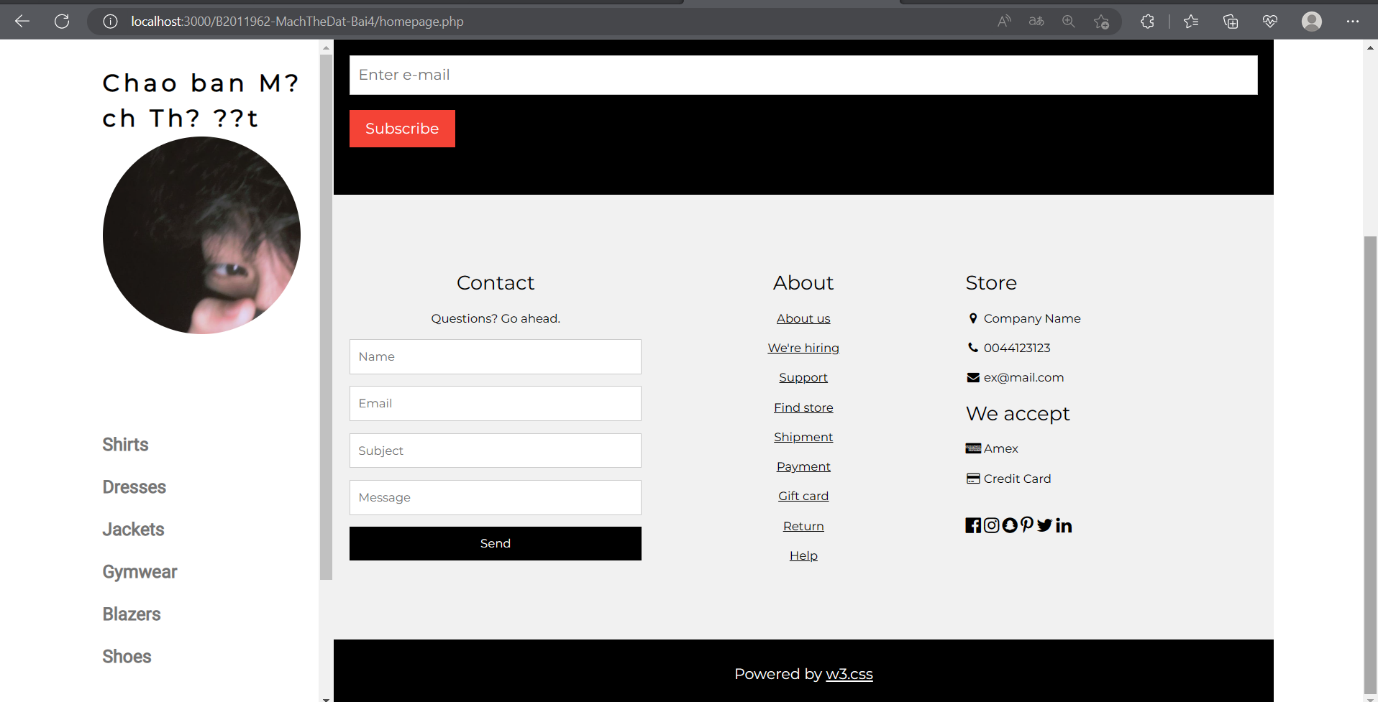
Bài 1 :

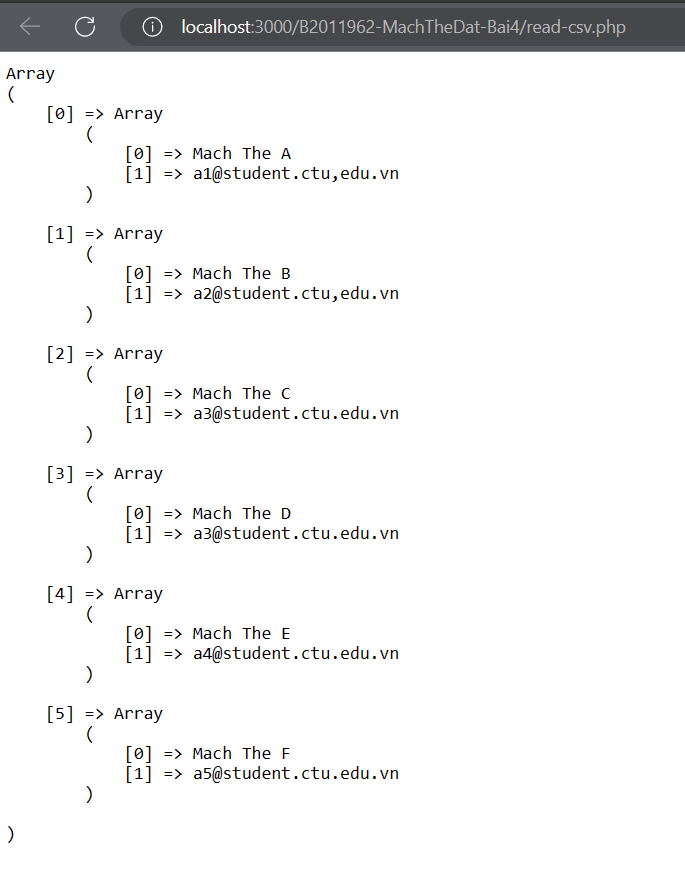












Bài 2 :

SQL Injection là một loại tấn công bảo mật màng máy chủ dữ liệu thông qua

việc chèn các lệnh SQL có ý định xấu vào các truy vấn SQL từ phía người

dùng.

Nếu một trang web không kiểm soát đầy đủ dữ liệu đầu vào từ người dùng, nó

có thể dễ dàng trở thành mục tiêu của tấn công SQL Injection.

Ví dụ: Giả sử một trang web có một form đăng nhập với hai trường là tên đăng

nhập và mật khẩu. Nếu các trường đó được liên kết trực tiếp với các truy vấn

SQL như sau:

SELECT \* FROM users WHERE username=`$username&` AND

password=`$password`

Nếu một người dùng nhập giá trị sau vào trường tên đăng nhập:

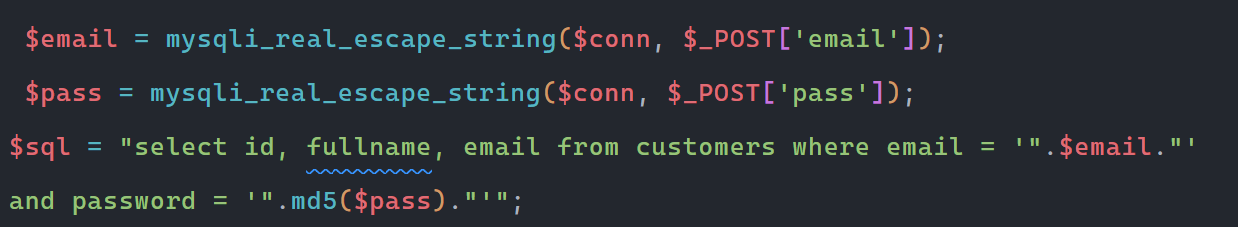
`OR 1=1

Thì truy vấn SQL sẽ trở thành:

SELECT \* FROM users WHERE username=`` OR 1=1 --`; AND password=`$password`

Để tránh SQL Injection, cần kiểm soát chặt chẽ tất cả dữ liệu đầu vào từ

người dùng và sử dụng các phương pháp như escape strings, prepared statements



Bài 3 :

Cookie là một tập hợp các dữ liệu nhỏ được lưu trữ trên máy tính của người

dùng. Khi trình duyệt web gửi một yêu cầu đến một trang web, nếu trang web

đó muốn ghi nhớ thông tin về người dùng, nó sẽ gửi một cookie tới trình

duyệt.

Trình duyệt sẽ lưu lại cookie này và khi gửi các yêu cầu tiếp theo đến

trang web đó, nó sẽ gửi cookie đó với yêu cầu.

Cookie có thể sử dụng để lưu trữ các thông tin như tên đăng nhập, giỏ hàng

mua sắm, lịch sử duyệt trang web, hoặc các thiết lập khác. Chúng ta có thể

cài đặt thời hạn sống của cookie để xác định khi nào nó sẽ bị xóa hoặc hết

hạn.

Hàm set cookie;

setcookie(“Tên trường”,“dữ liệu”, “thời gian tồn tại của cookie”)

Để lưu cookie ta gọi hàm setCookie()

Để lấy cookie ta dùng biến siêu toàn cục $\_COOKIE[“Tên trường”]

Để xóa cookie ta dùng hàm setcookie(“Tên trường”,“dữ liệu”, “thời gian tồn

tại của cookie trừ hơn 1 ngày so với thời gian đã set”)

Bài 4 :

Session là một công nghệ cho phép lưu trữ thông tin trên máy chủ về người

dùng hiện tại. Khi một người dùng truy cập vào một trang web, máy chủ sẽ tạo

ra một session và gắn nó với một ID riêng biệt. Máy chủ sẽ lưu trữ các thông

tin liên quan đến người dùng này trong session đó.

- Cách lưu session :

Gọi phương thức: sesion\_start();

Gọi biến siêu toàn cục:$\_SESSION[“tên session”];

- Cách lấy session :

Gọi lại biến siêu toàn cục $\_SESSION[‘tên session’];

- Cách xóa session :

Ta gọi phương thức session\_unset(); và session\_destroy();

Bài 5 :

Cookie và Session là hai công nghệ khác nhau cho phép lưu trữ thông tin trên

máy chủ và trên máy tính của người dùng. Tuy nhiên, có một số sự khác biệt

quan trọng giữa hai công nghệ này:

Lưu trữ: Cookie được lưu trữ trực tiếp trên máy tính của người dùng, trong

khi Session được lưu trữ trên máy chủ.

Kích thước: Cookie có thể lưu trữ một lượng lớn dữ liệu nhất định, trong khi

Session chỉ có thể lưu trữ một số thông tin nhất định.

An toàn: Cookie có thể bị tấn công bởi các hacker, trong khi Session được

mã hóa và giữ an toàn hơn.

Hạn chế: Cookie có thể bị xóa hoặc hết hạn sau một khoảng thời gian nhất

định, trong khi Session sẽ bị xóa khi người dùng đăng xuất hoặc đóng trình

duyệt.

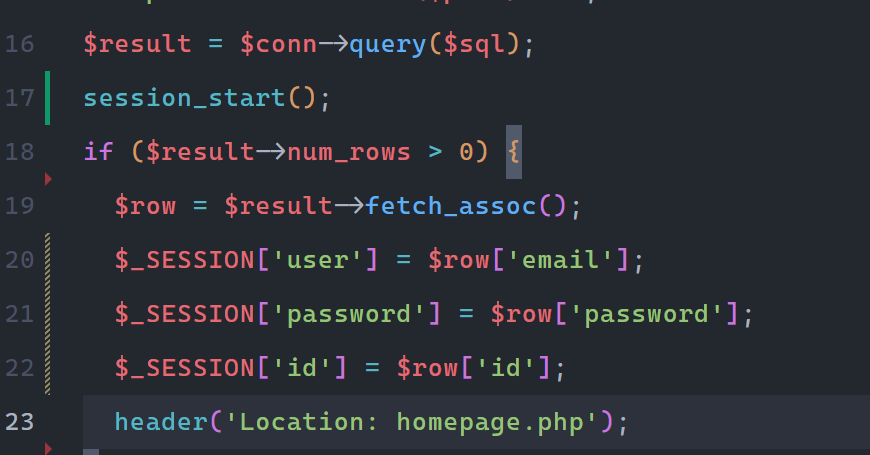
Sử dụng: Cookie được sử dụng để lưu trữ các thông tin như tên đăng nhập, giỏ

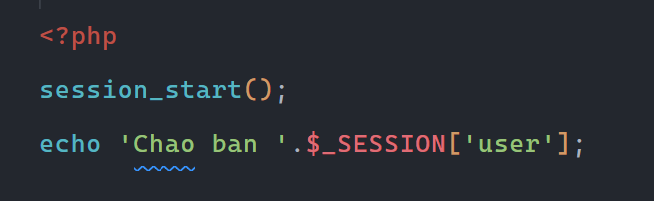
hàng mua sắm, hoặc bất kỳ thông tin nào mà trang web muốn ghi nhớ về người

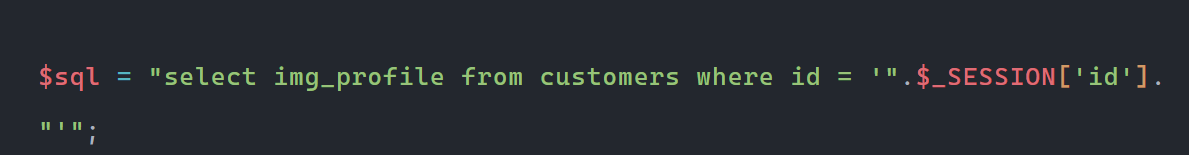
dùng, trong khi Session được sử dụng để lưu trữ thông tin trạng thái của

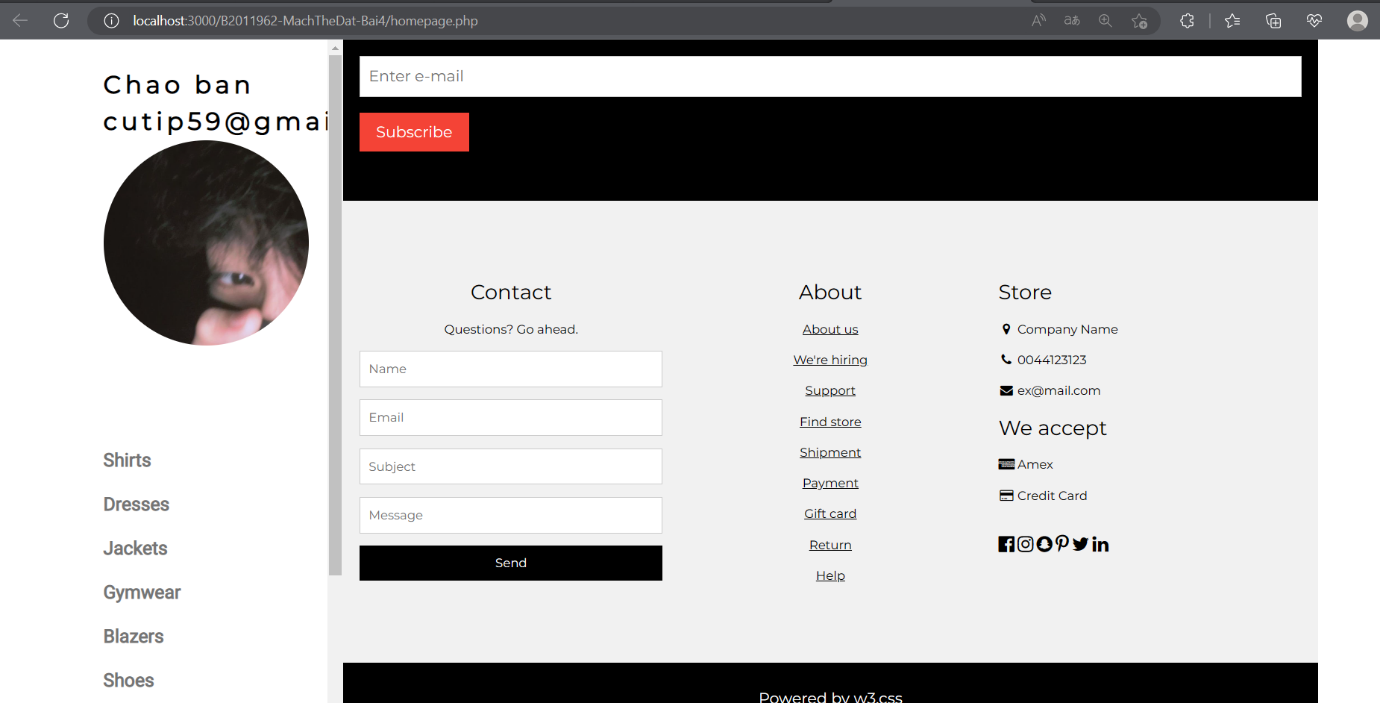
người dùng

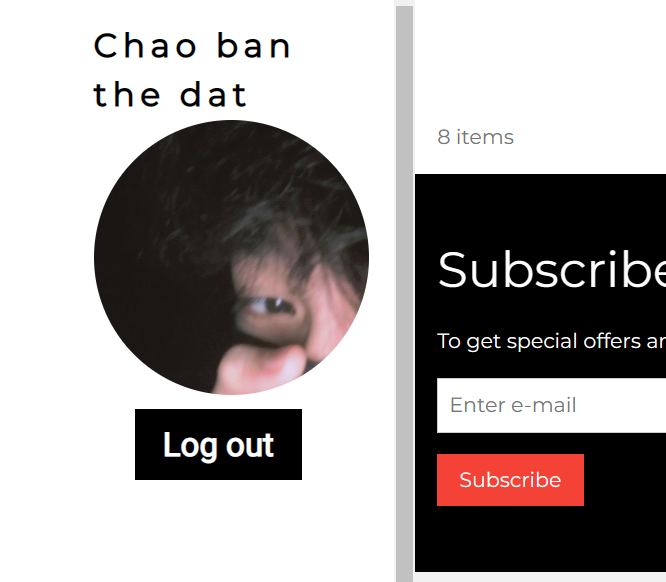
Bài 6 :

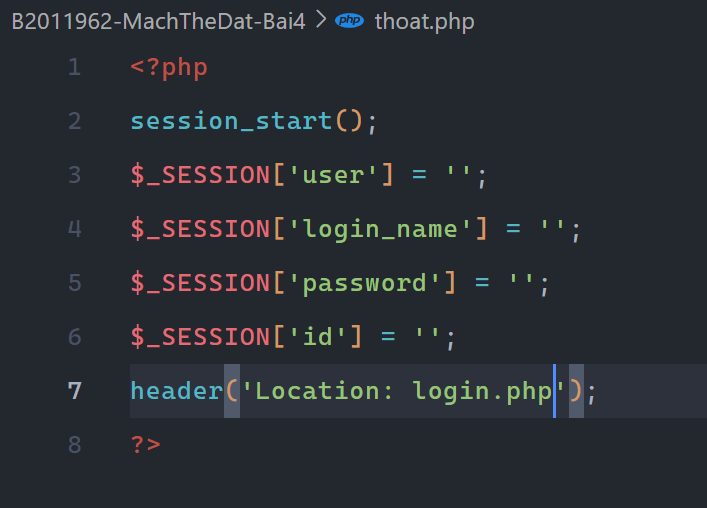








button log out để thoát và chuyển qua thoat.php



Câu 7 :

Câu 8:

Trang web https://www.w3schools.com/php/php\_file\_upload.asp cung cấp một

hướng dẫn chi tiết về cách sử dụng PHP để upload tệp tin từ máy khách lên

máy chủ, Ta cần tạo một form HTML để gửi tệp tin đến máy chủ và sử dụng PHP

để xử lý tệp tin được gửi lên.

Các bước chính để thực hiện việc upload tệp tin như sau:

**1.**Tạo một form HTML với thuộc tính enctype= “multipart/form-data”

và một trường input kiểu file.

**2.**Kiểm tra xem tệp tin được gửi lên có hợp lệ hay không bằng cách sử dụng

$\_FILES[“file”][ “error”].

**3.**Sử dụng hàm move\_uploaded\_file() để di chuyển tệp tin từ thư mục tạm thời

lên thư mục mong muốn trên máy chủ.

**4.**Kiểm tra xem tệp tin được upload có phải là định dạng hình ảnh hợp lệ bằng

cách sử dụng getimagesize().

**5.**Sử dụng các hàm như filesize() để lấy thông tin về kích thước của tệp tin.

Câu 9:

Trang web https://www.w3schools.com/php/func\_filesystem\_file.asp giới thiệu

về hàm file() trong PHP. Hàm này được sử dụng để đọc toàn bộ nội dung của

một tệp tin và trả về nội dung đó dưới dạng mảng.

Tham số của hàm file():

file\_name: Là đường dẫn đầy đủ của tệp tin mà bạn muốn đọc.

use\_include\_path (tùy chọn): Cho phép bạn cố gắng tìm tệp tin trong các

đường dẫn bổ sung.

context (tùy chọn): Là một stream context để sử dụng khi đọc tệp tin.

Hàm file() trả về một mảng chứa các dòng nội dung của tệp tin. Ví dụ, file

read-csv.php sử dụng hàm file() để đọc nội dung tệp tin CSV và trả về dữ

liệu dưới dạng mảng.

Trước khi sử dụng hàm file(), ta nên kiểm tra xem tệp tin có tồn tại và có

thể đọc được hay không bằng cách sử dụng hàm is\_readable().