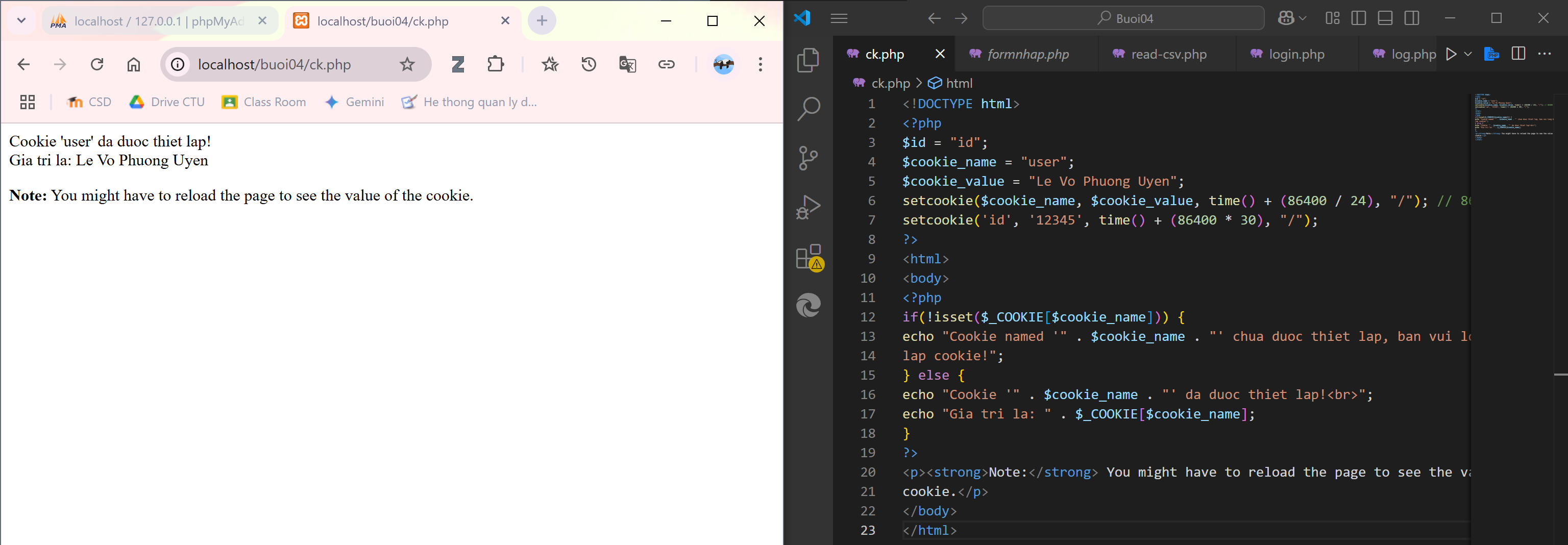
**YÊU CẦU THỰC HÀNH**

**Câu 1:**

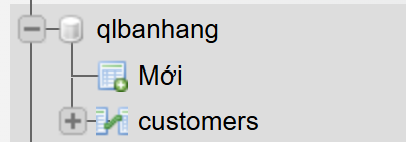
**Tạo Cookies**

Tạo tập tin sử dụng khởi tạo cookies ck.php:

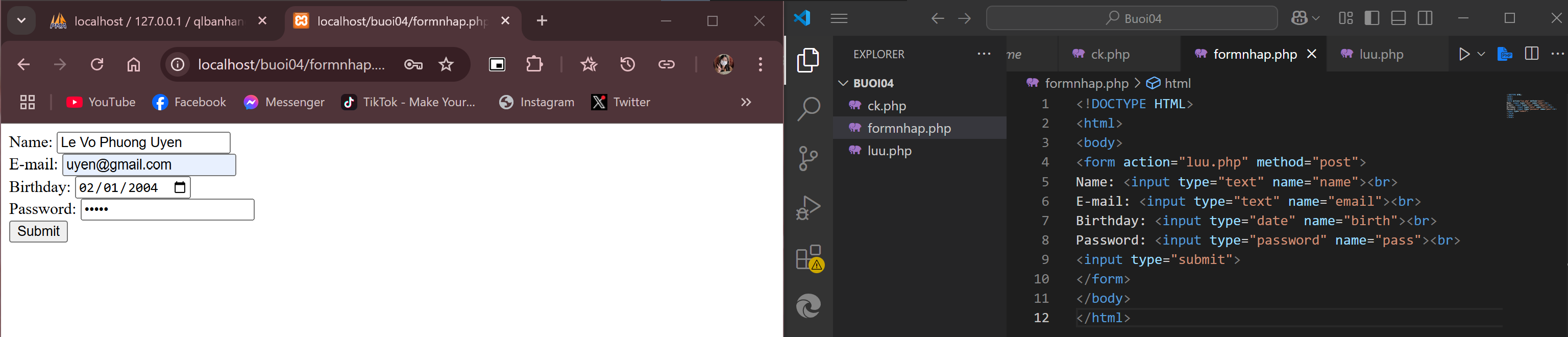


**Tạo 1 trang đăng ký người dùng với mật khẩu được mã hóa**

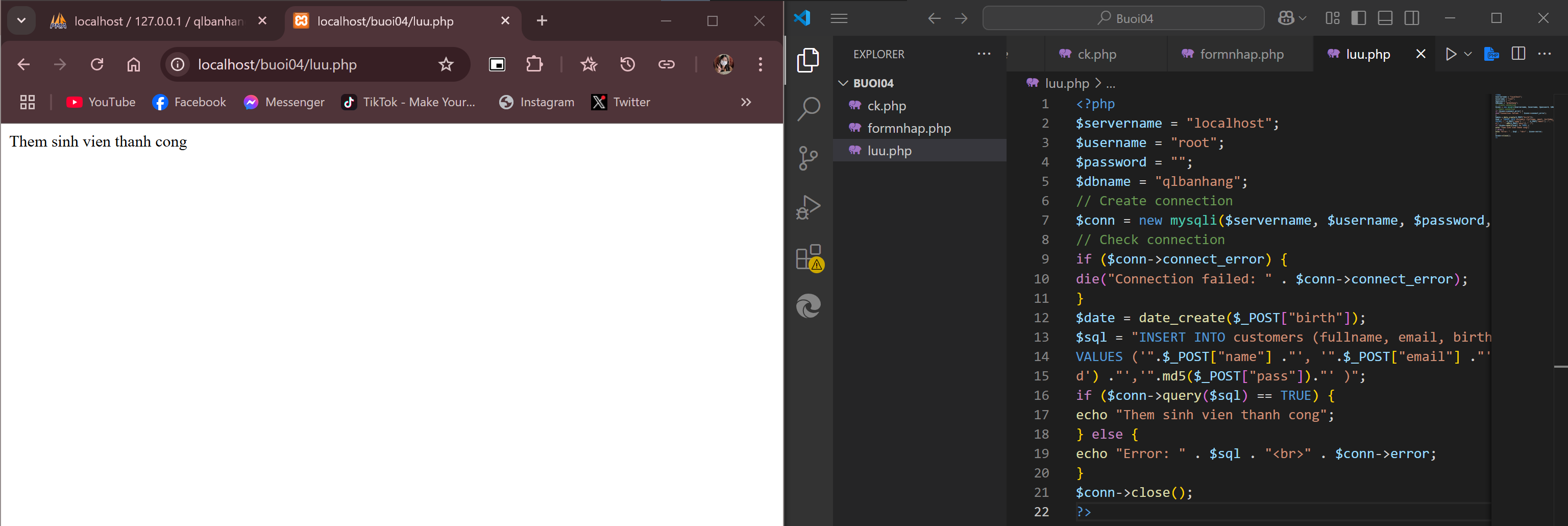
Tạo csdl qlbanhang, bảng customers:



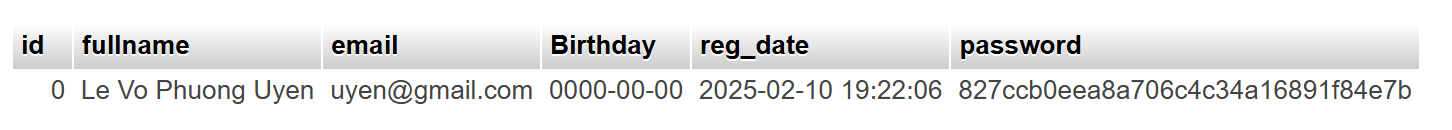
Tạo formnhap.php



Tạo luu.php

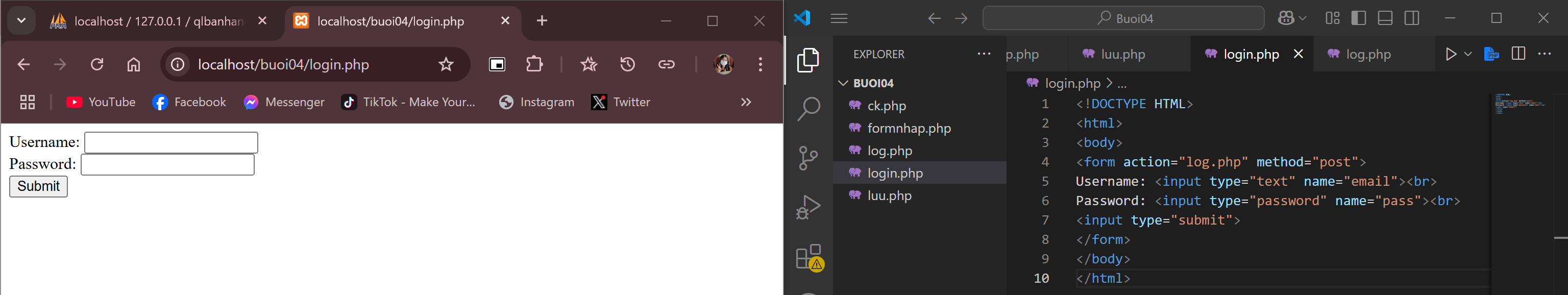


Mật khẩu đã được mã hóa



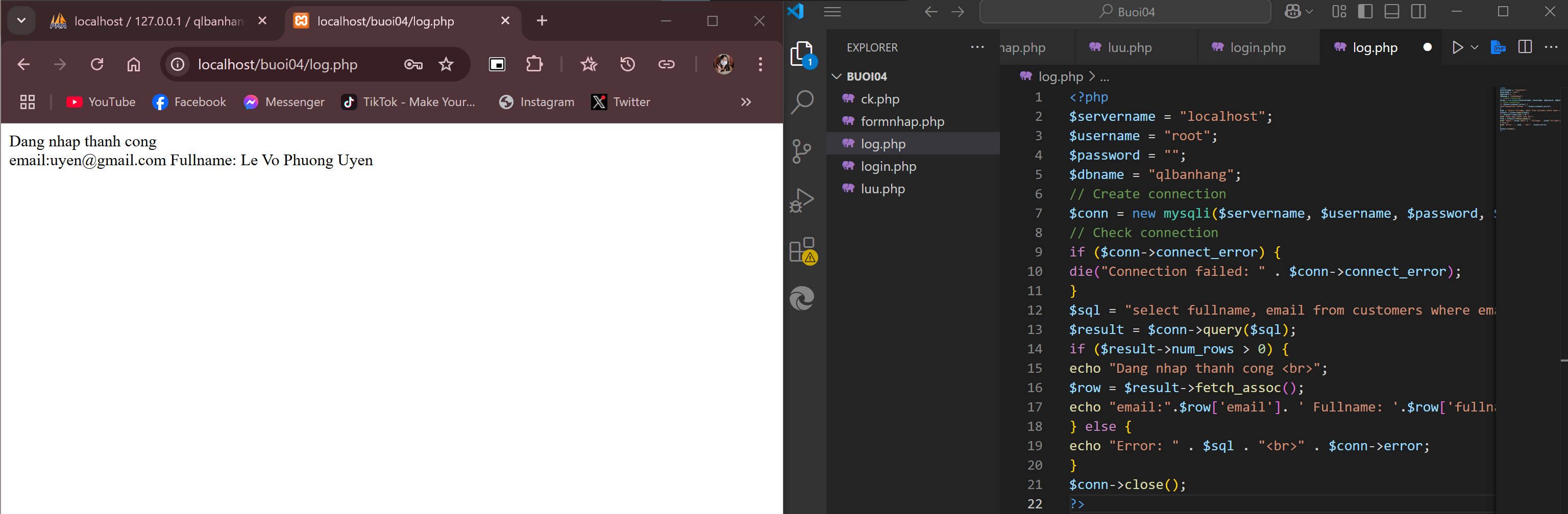
**Tạo form đăng nhập**

Tạo login.php

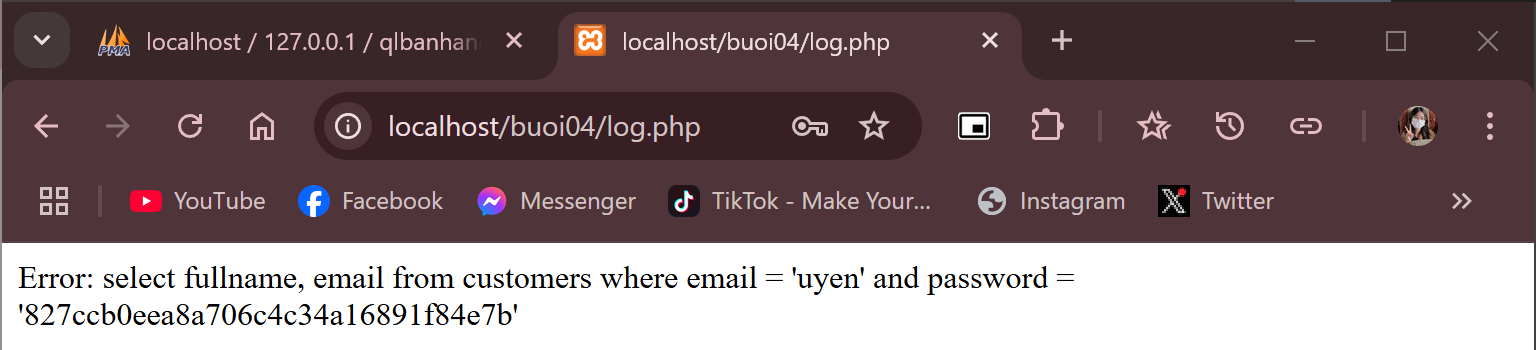


Tạo log.php

* Khi nhập đúng:

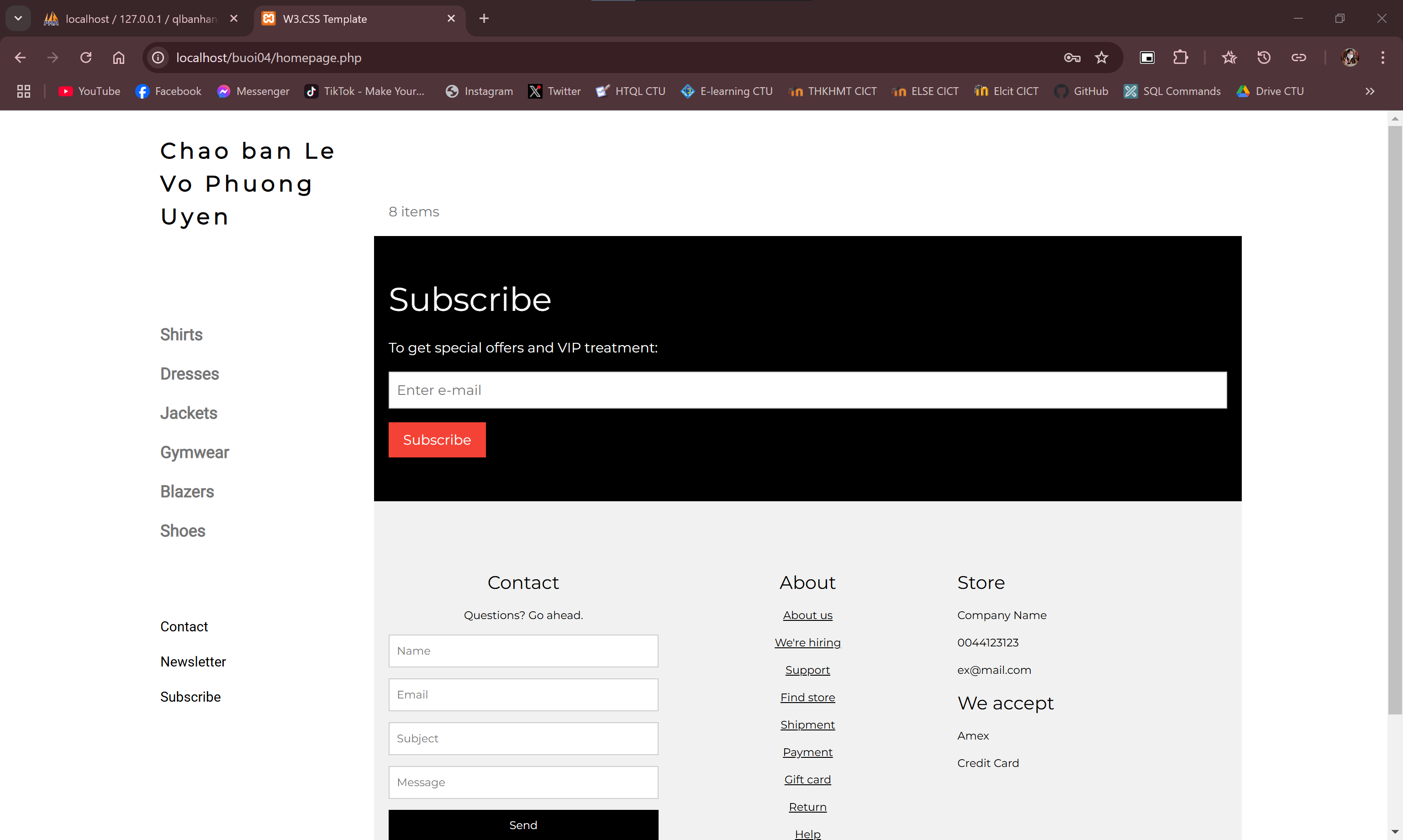


* Khi nhập lỗi:



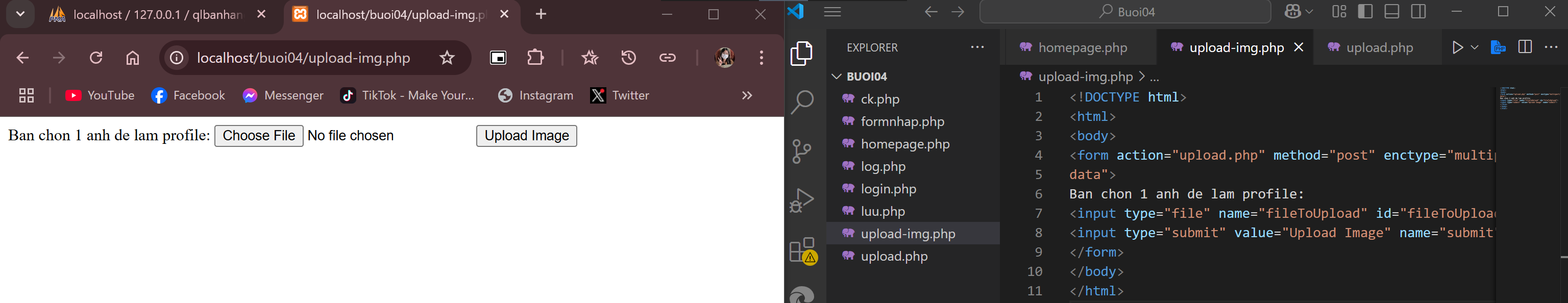
**Dùng Cookies lưu lại tên đăng nhập, email và điều hướng**

Tạo 1 trang chủ để khi đăng nhập thành công sẽ chuyển qua trang chủ này:

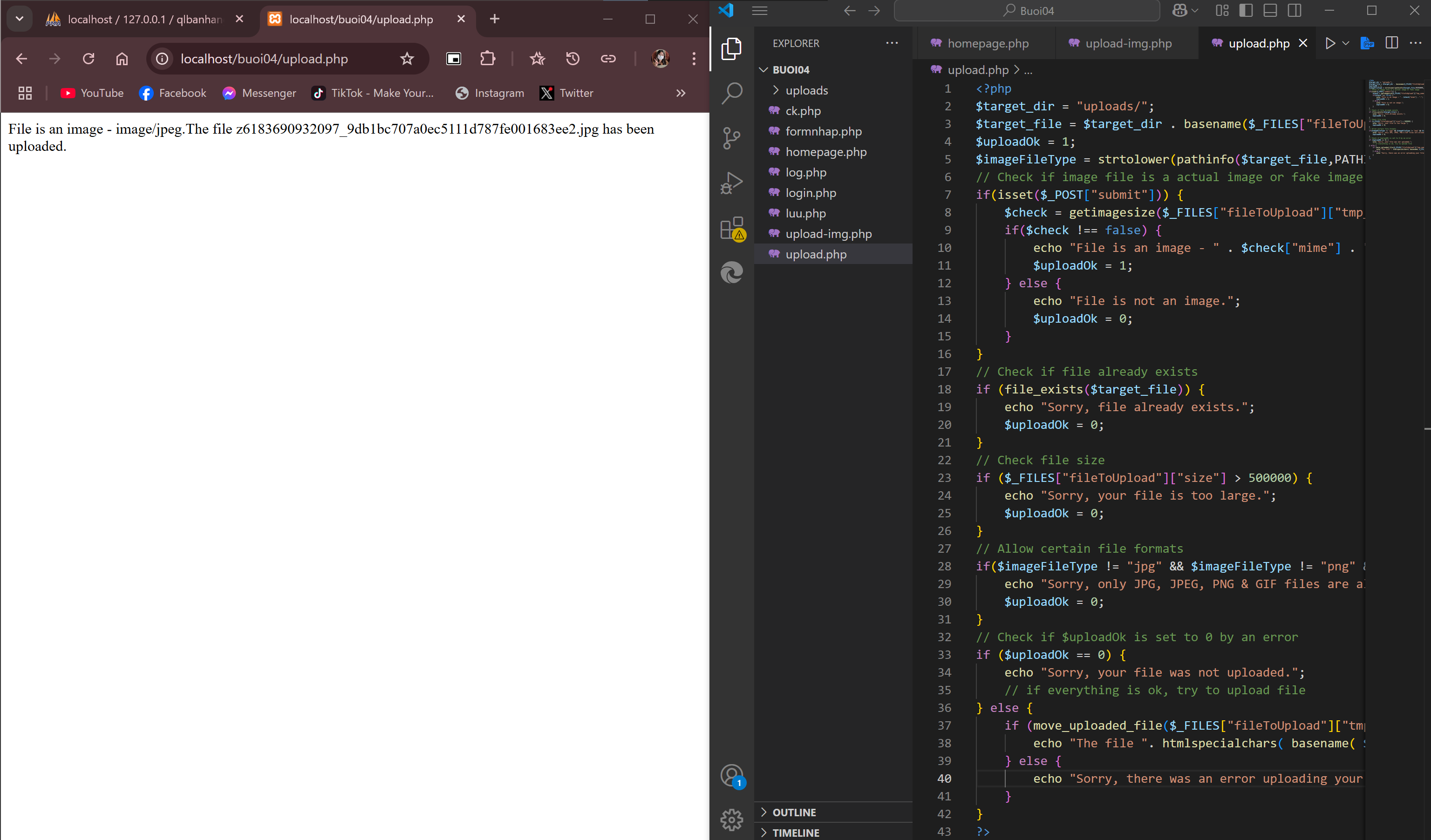


**Upload tập tin**

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



Tạo upload.php để xử lý hành động

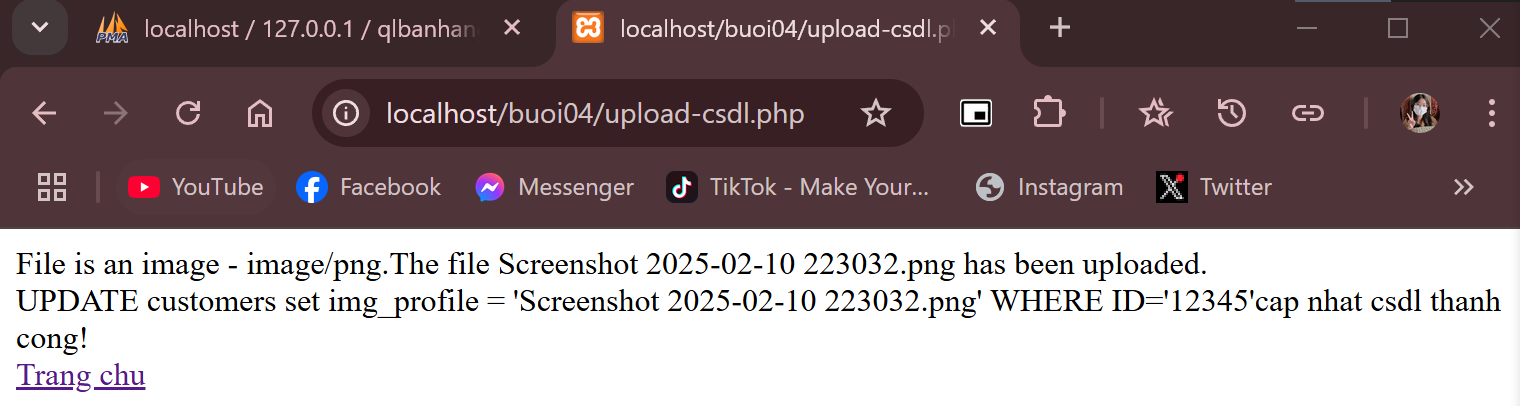


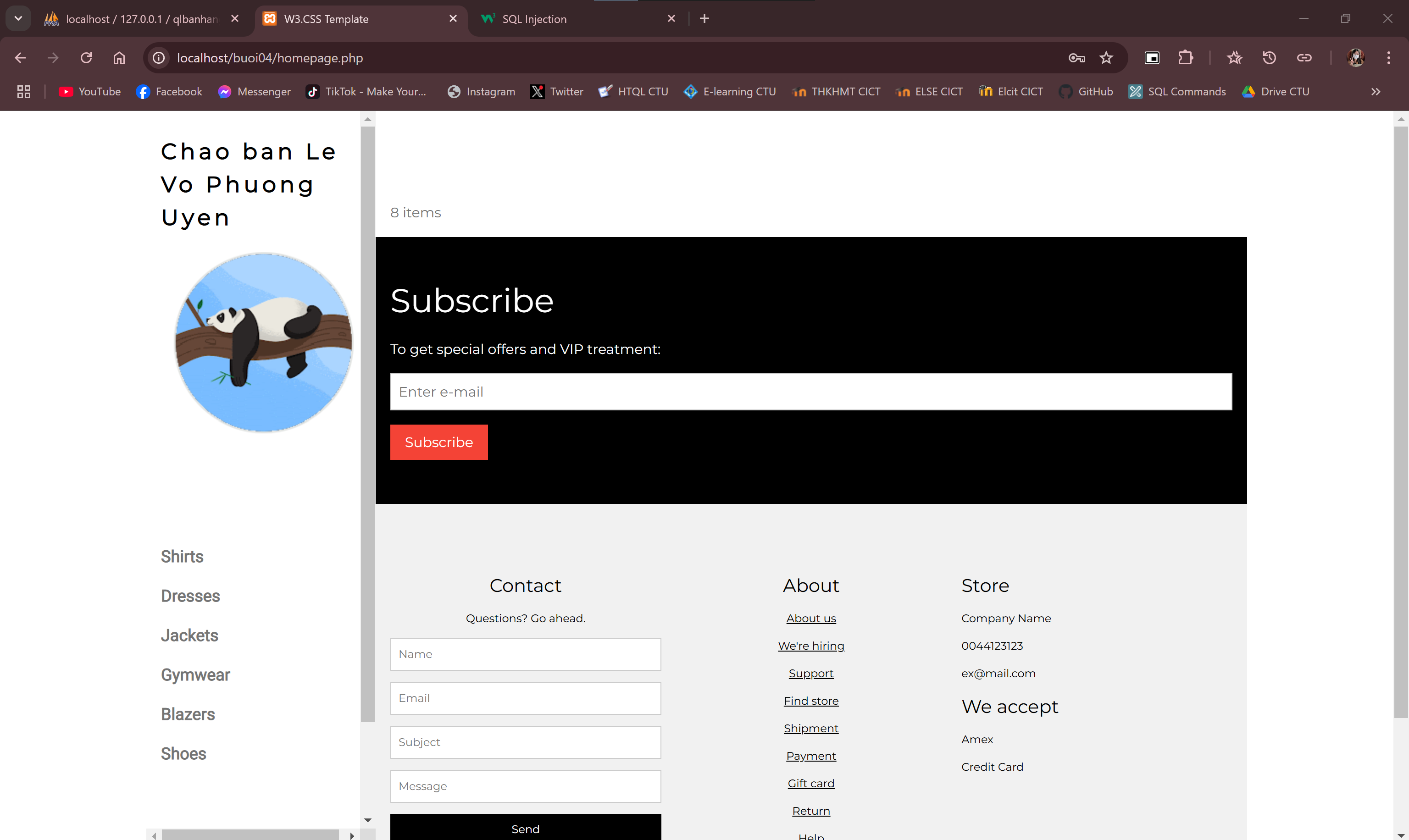
**Upload ảnh và lưu dữ liệu về ảnh trong CSDL**

Sửa file upload-img.php

Tạo upload-csdl.php

Sửa file homepage.php

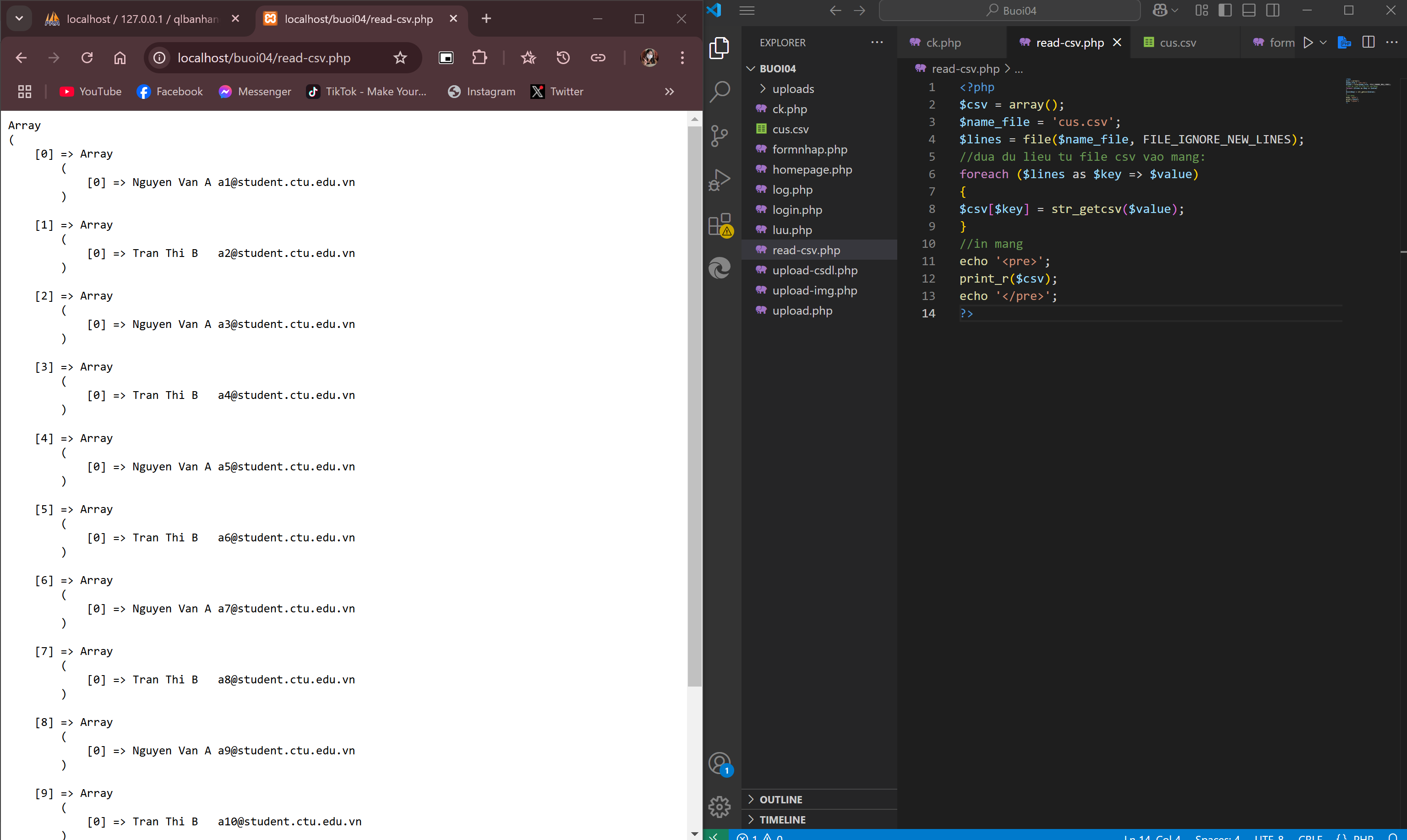




**Upload file csv và đọc dữ liệu vào mảng**

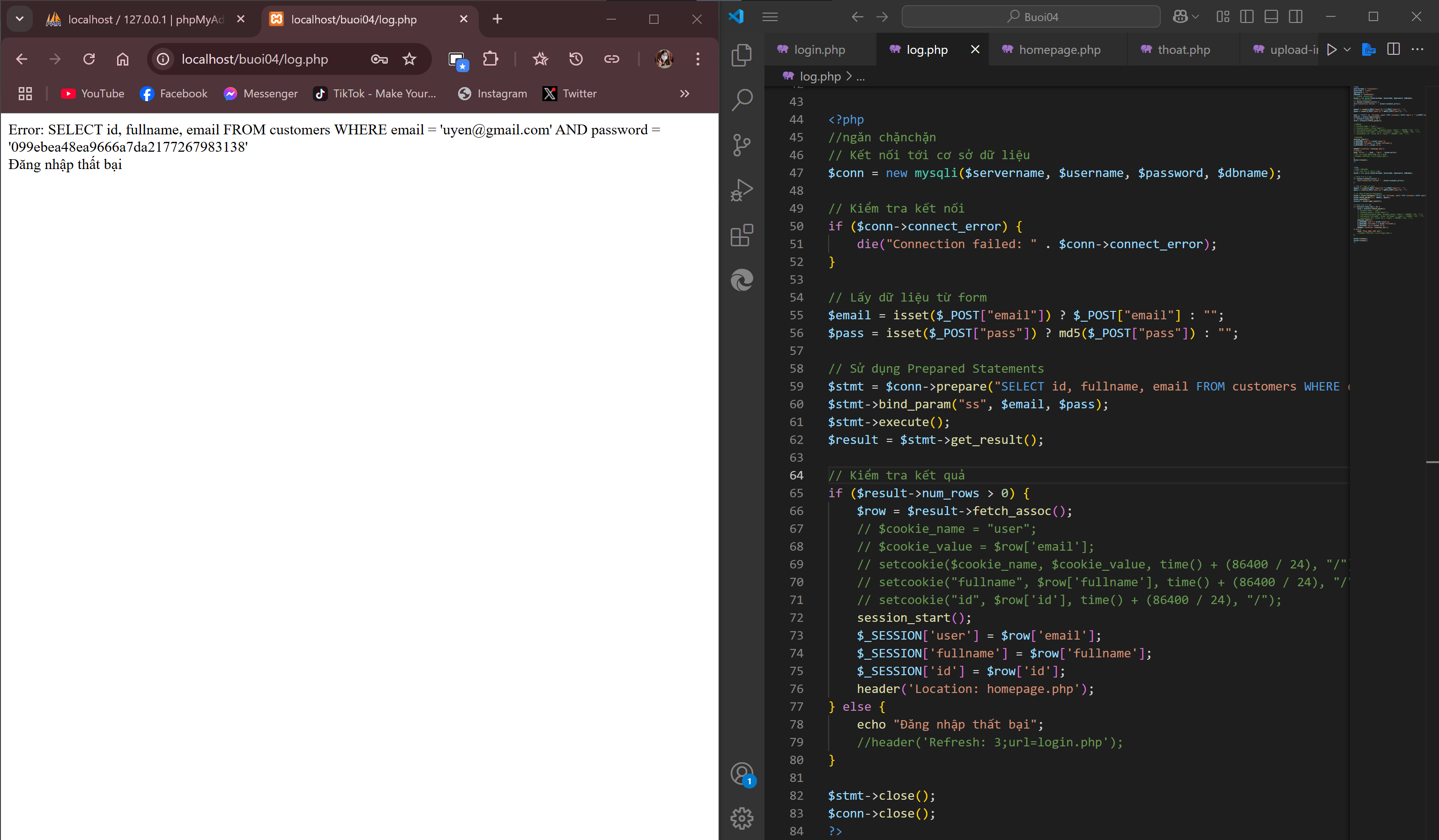
Tạo read-csv.php

Đọc file cus.csv



**Câu 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: là một trong những kỹ thuật hack phổ biến. Hoạt động bằng cách khai thác các lỗ hổng bảo mật, sử dụng các mã độc hại vào trong các câu lệnh SQL nhằm lấy cắp, thay đổi dữ liệu hay điều khiển hệ thống.
* Ứng dụng vào trang đăng nhập:



**Câu 3: 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 tệp nhỏ mà máy chủ nhúng vào máy tính của người dùng. Mỗi khi máy tính yêu cầu một trang trình duyệt, nó sẽ gửi cookie. Với php có thể tạo và truy xuất các giá trị cookie.
* Ý nghĩa các tham số trong setcookie()

+ name: tên của cookie.

+ value: giá trị của cookie.

+ expire: thời điểm cookie hết hạn. Nếu tham số này bị bỏ qua hoặc đặt thành 0, cookie sẽ hết hạn khi trình duyệt đóng.

+ path: đường dẫn máy chủ của cookie. Nếu đặt thành “/” cookie sẽ khả dụng trong toàn bộ miền. Nếu đặt thành “/php/” cookie chỉ khả dụng trong thư mục php và thư mục con của php.

+ domain: tên miền của cookie. Để cookie khả dụng trên tất cả các miền phụ của example.com, nên đặt tên miền thành “example.com”. Đặt [www.example.com](http://www.example.com) sẽ chỉ khả dụng trong miền phụ www.

+ secure: chỉ định cookie có nên chỉ được truyền qua kết nối HTTPS an toàn hay không. TRUE cho biết cookie sẽ chỉ được thiết lập nếu có kết nối an toàn..

+ httponly: nếu đặt thành TRUE, cookie sẽ chỉ có thể truy cập được thông qua giao thức HTTP. Điều này có thể giúp giảm tình trạng đánh cắp danh tính thông qua XSS.

* Cách lưu cookie: setcookie("myCookie", $value, time() + 3600, "/");
* Cách lấy cookie:

if(!isset($\_COOKIE[$cookie\_name])) {

echo "Cookie named '" . $cookie\_name . "' is not set!";

} else {

echo "Cookie '" . $cookie\_name . "' is set!<br>";

echo "Value is: " . $\_COOKIE[$cookie\_name];

}

* Cách xóa cookie: setcookie("user", "", time() - 3600);

**Câu 4: Dựa vào link3, bạn hãy cho biết Session dùng để làm gì. Cách lưu, lấy, xóa giá trị trong Session.**

* Session là một cách lưu trữ thông tin trên biến để sử dụng trên nhiều trang. Không giống như cookie, thông tin không được lưu trữ trên máy tính của người dùng
* Cách lưu giá trị trong Session: $\_SESSION["favcolor"] = "green";
* Cách lấy giá trị: echo "Favorite color is " . $\_SESSION["favcolor"];
* Cách xóa giá trị: session\_destroy();

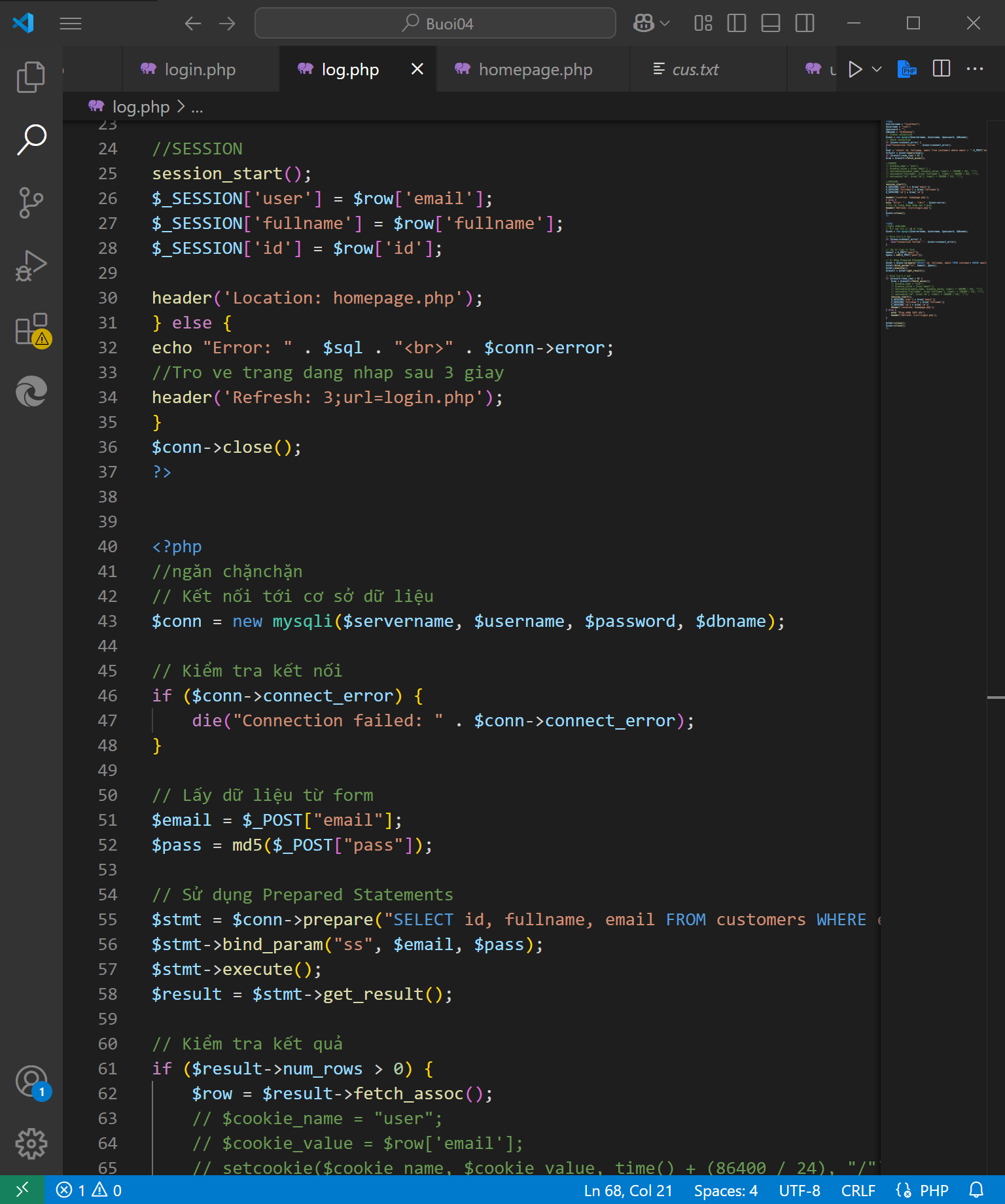
**Câu 5: Bạn hãy so sánh Cookie và Session.**

* Giống nhau: đều dùng để lưu trữ thông tin người dùng
* Khác nhau:

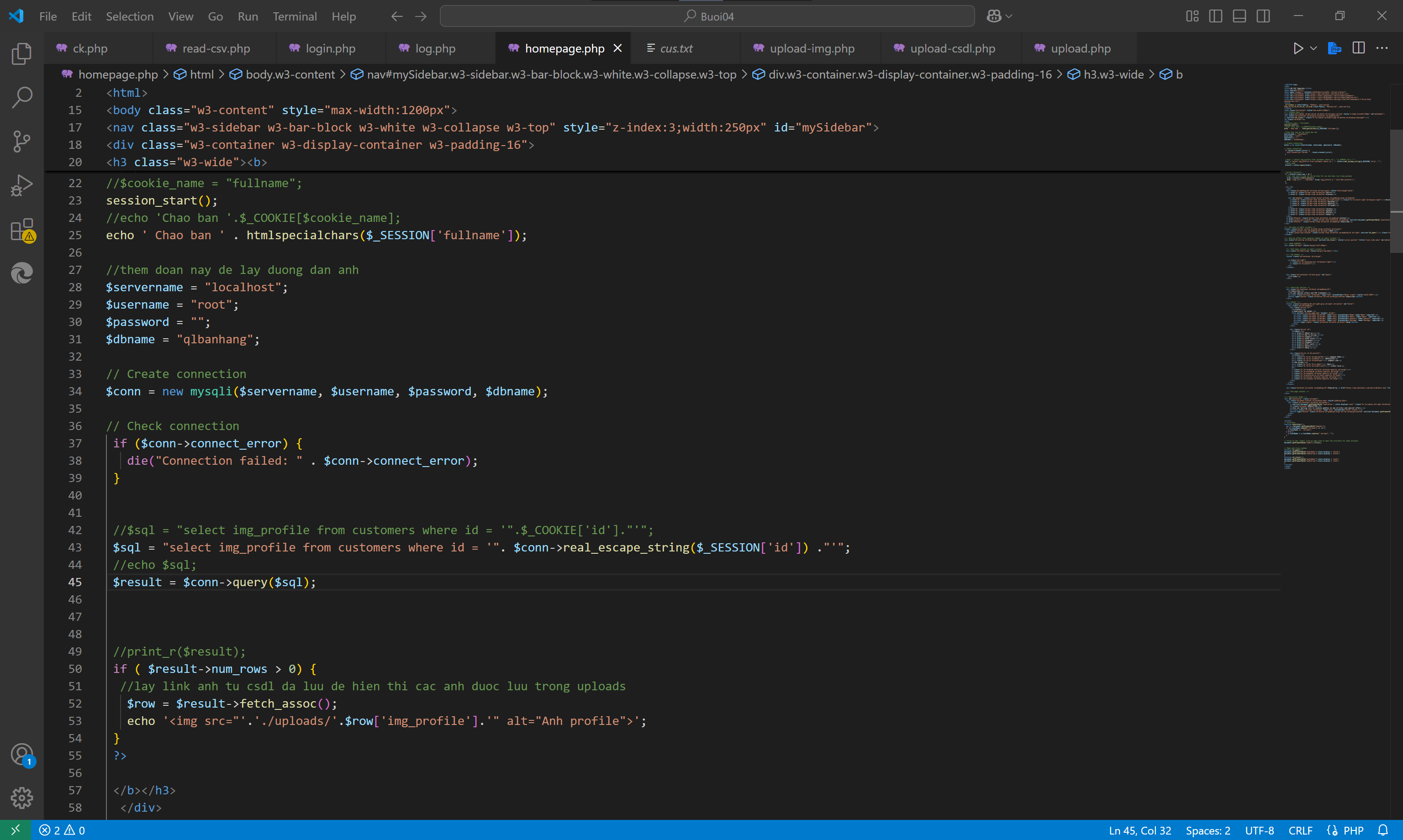
|  |  |
| --- | --- |
| Cookie | Session |
| Lưu trữ trên trình duyệt của người dùng | Lưu trữ trên máy chủ |
| Dung lượng lưu trữ nhỏ | Dung lượng lưu trữ lớn |
| Hết hạn sau một khoảng thời gian | Tồn tại trong suốt phiên làm việc |
| Dễ bị đánh cắp | An toàn hơn |
| Lưu trữ thông tin không nhạy cảm | Lưu trữ thông tin riêng tư |

**Câu 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.**

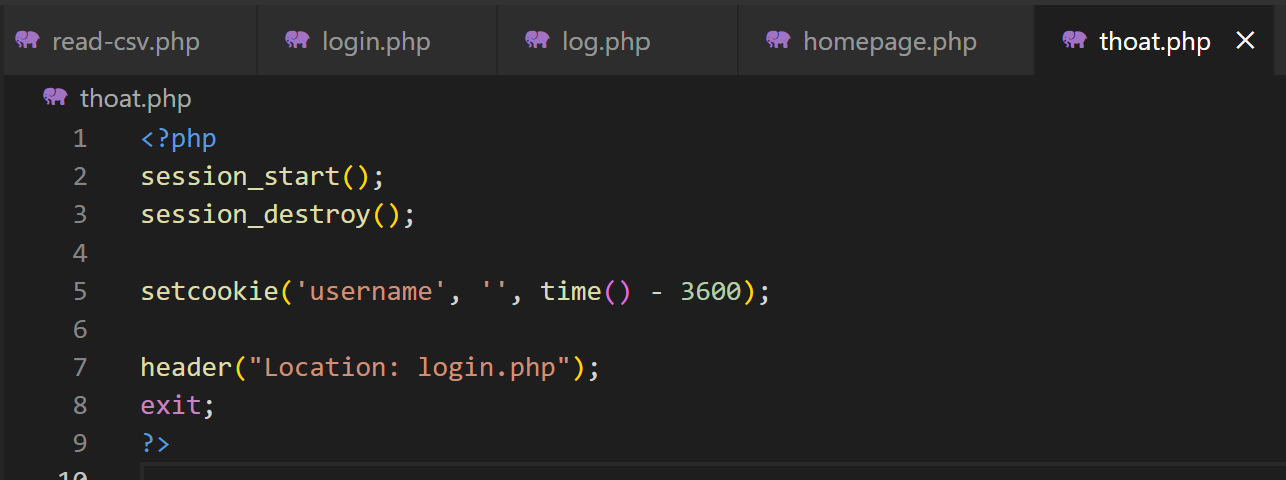
* Chỉnh sửa các đoạn gán, khởi tạo, lấy giá trị cookie trong log.php:

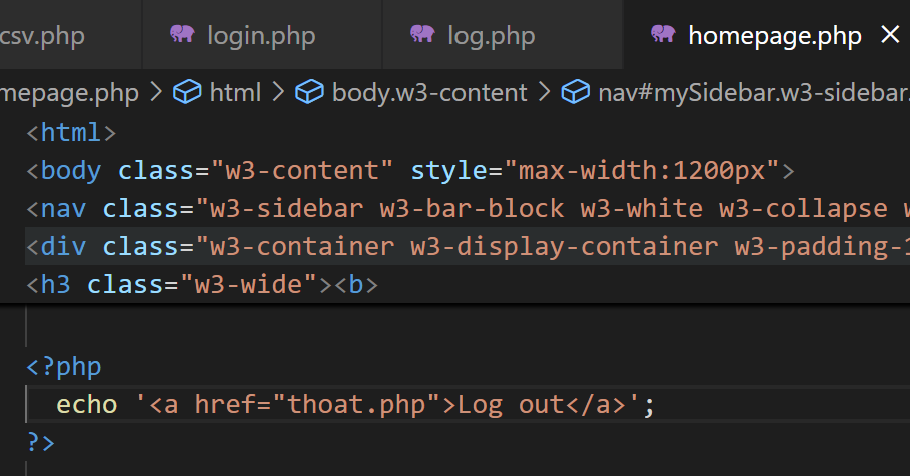


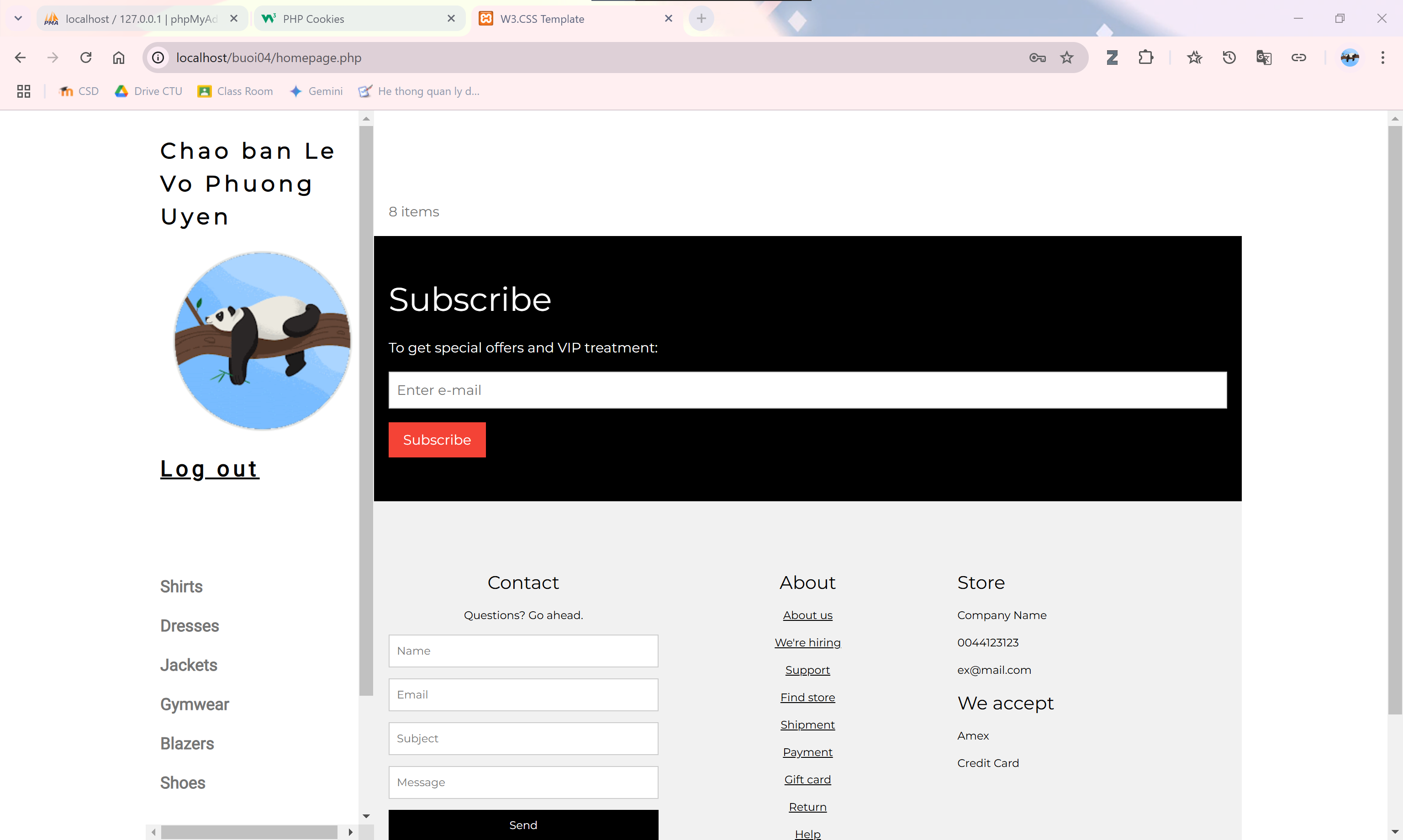
* Chỉnh sửa các đoạn gán, khởi tạo, lấy giá trị cookie trong homepage.php:



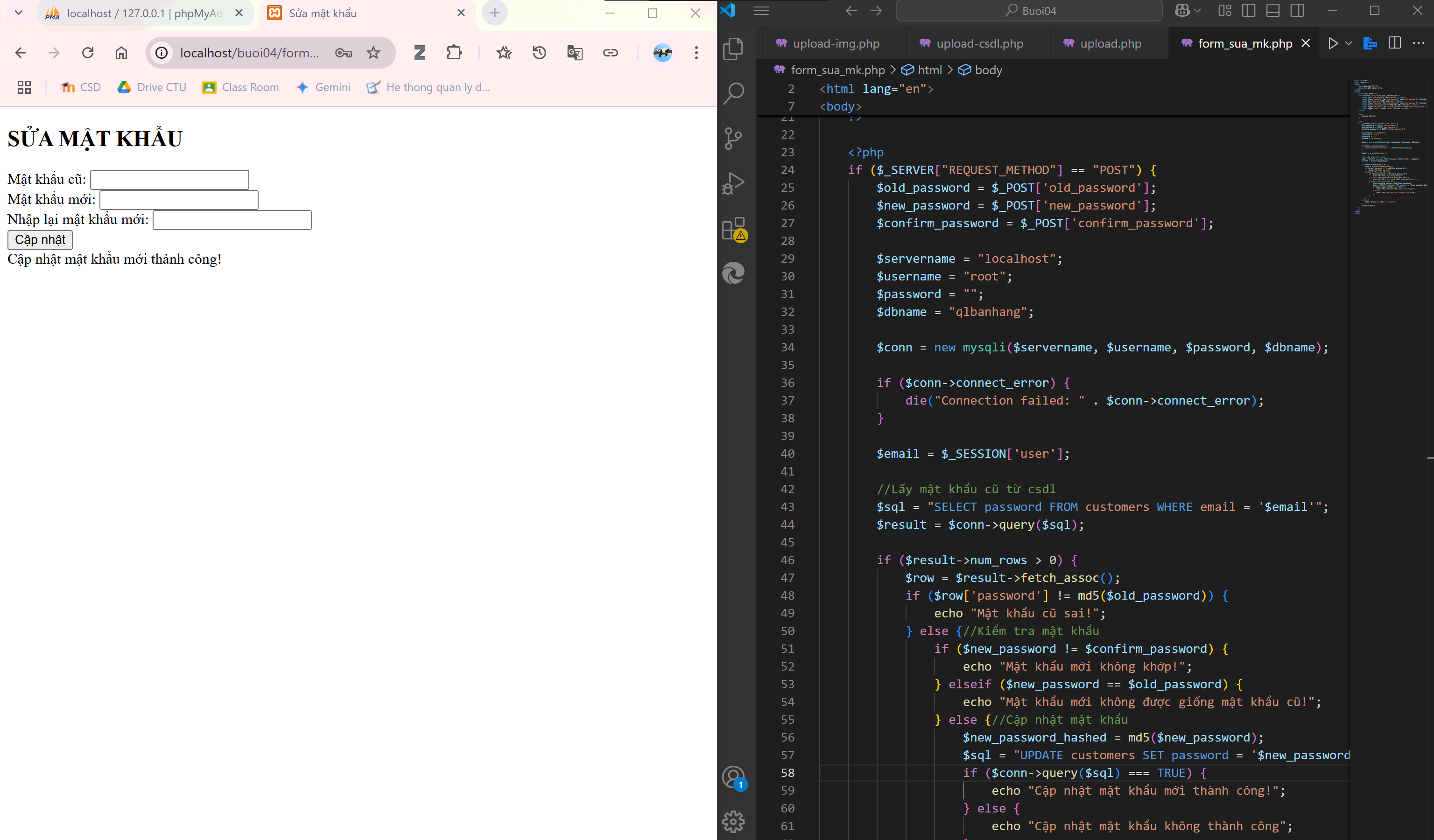
* Tạo tập tin thoat.php:







**Câu 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.**

****

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

* Các cơ chế:

+ Đảm bảo cấu hình file\_uploads trong file php.ini đã được bật

+ Tạo form HTML cho phép người dùng chọn tệp để upload

<!DOCTYPE html>

<html>

<body>

<form action="upload.php" method="post" enctype="multipart/form-data">

Select image to upload:

<input type="file" name="fileToUpload" id="fileToUpload">

<input type="submit" value="Upload Image" name="submit">

</form>

</body>

</html>

+ Tạo file upload.php để tải tệp lên

<?php

$target\_dir = "uploads/";

$target\_file = $target\_dir . basename($\_FILES["fileToUpload"]["name"]);

$uploadOk = 1;

$imageFileType = strtolower(pathinfo($target\_file,PATHINFO\_EXTENSION));

// Check if image file is a actual image or fake image

if(isset($\_POST["submit"])) {

$check = getimagesize($\_FILES["fileToUpload"]["tmp\_name"]);

if($check !== false) {

echo "File is an image - " . $check["mime"] . ".";

$uploadOk = 1;

} else {

echo "File is not an image.";

$uploadOk = 0;

}

}

?>

+ Kiểm tra xem tệp muốn upload đã tồn tại hay chưa

if (file\_exists($target\_file)) {

echo "Sorry, file already exists.";

$uploadOk = 0;

}

+ Giới hạn kích thước tệp

if ($\_FILES["fileToUpload"]["size"] > 500000) {

echo "Sorry, your file is too large.";

$uploadOk = 0;

}

+ Giới hạn loại tệp

if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType != "jpeg"

&& $imageFileType != "gif" ) {

echo "Sorry, only JPG, JPEG, PNG & GIF files are allowed.";

$uploadOk = 0;

}

+Hoàn tất quá trình upload tệp

* Các hàm/thủ tục:

+ basename($path): trả về tên file từ đường dẫn

+ pathinfo($path, $options): lấy thông tin về đường dẫn file

+ getimagesize($\_FILES[“fileToUpload”][“tmp\_name”]): kiểm tra file có phải hình ảnh không và trả về kích thước của file

+ file\_exists($target\_file): kiểm tra file đã tồn tại chưa

+ move\_uploaded\_file($\_FILES[“fileToUpload”][“tmp\_name”], $target\_file): di chuyển file từ thư mục tạm thời đến thư mục đích

**Câu 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(): đọc nội dung một file vào mảng. Mỗi dòng từ file là một phần tử của mảng.
* Các tham số:

+ filename: chỉ định đường dẫn đến tệp để đọc

+ flag: có thể là một hằng hoặc nhiều hằng số sau:

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

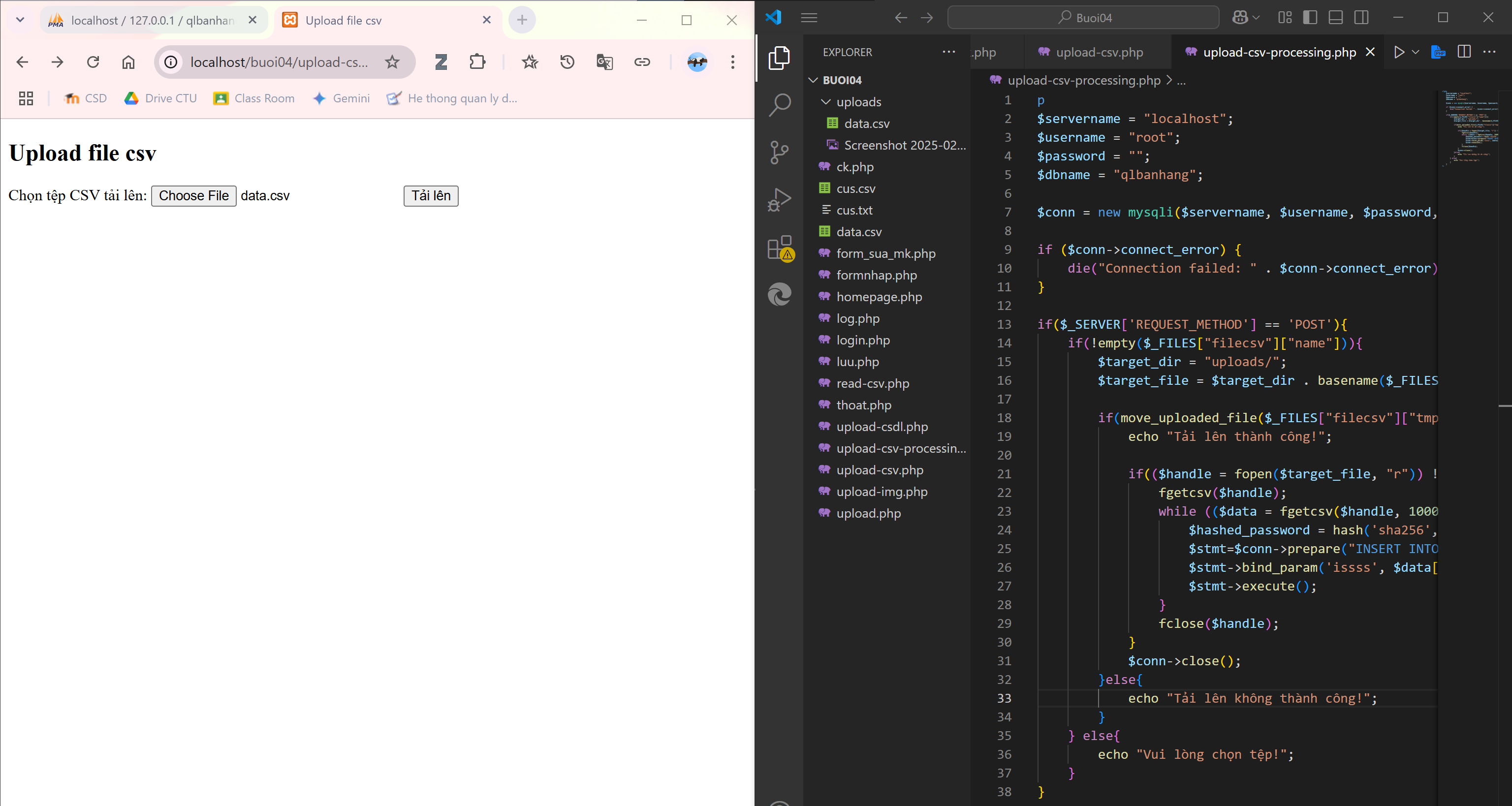
+ context: tập hợp các tùy chọn để điều chỉnh hành vi của luồng dữ liệu. Có thể bỏ qua bằng cách sử dụng NULL.

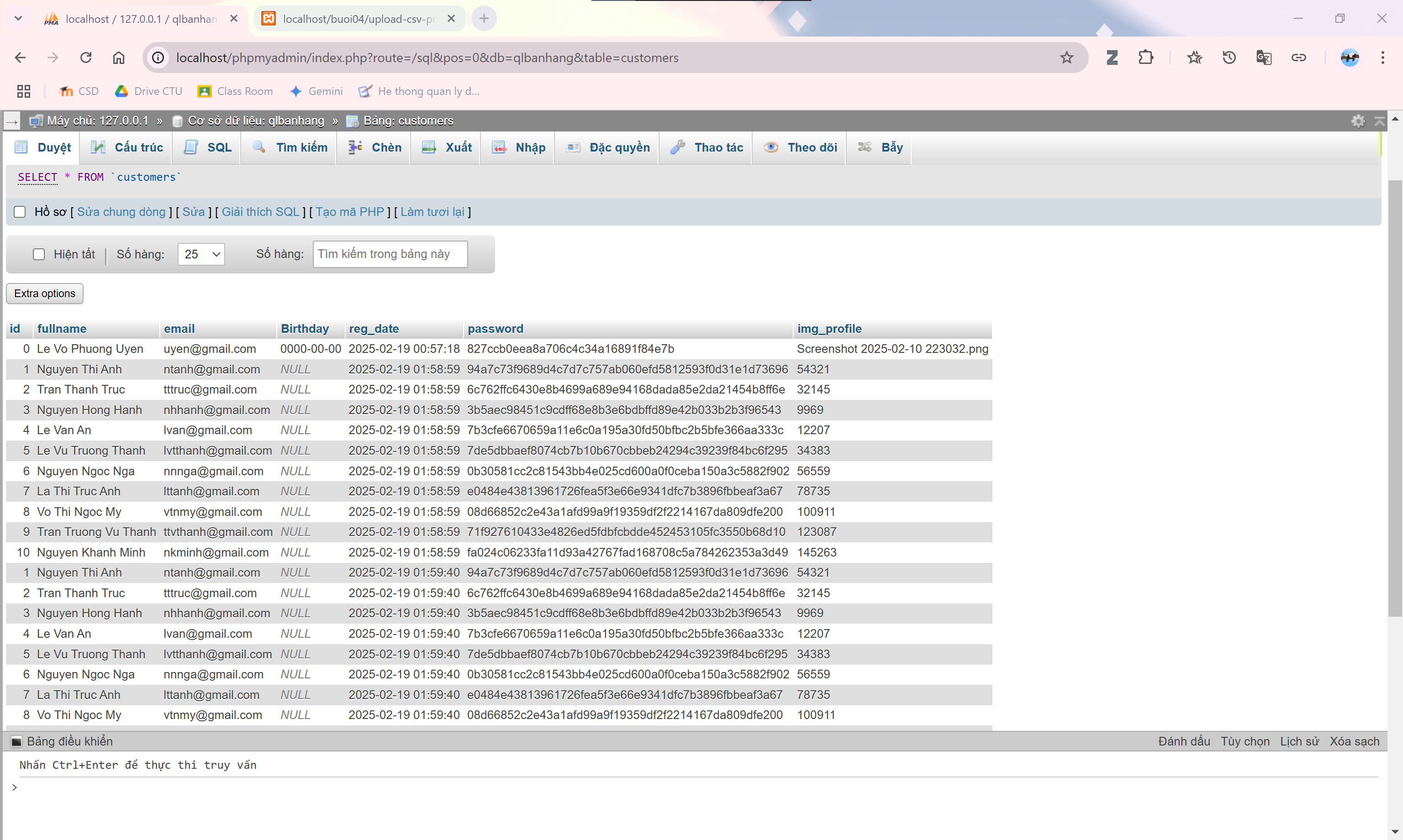
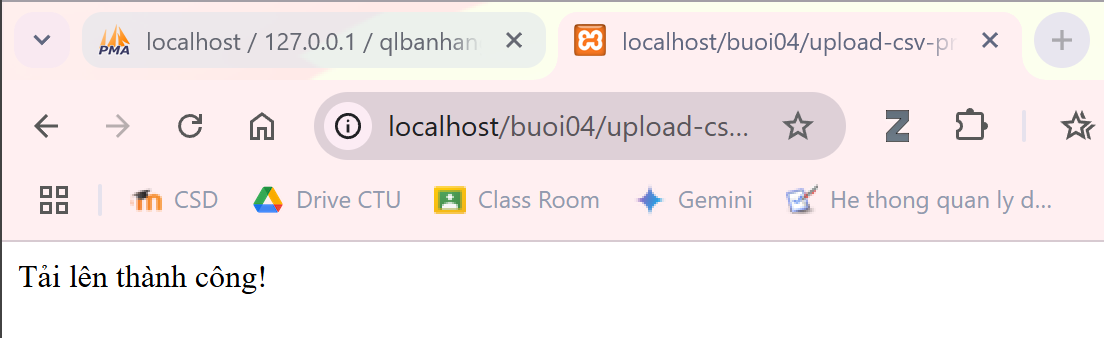
**Câu 10: Bạn hãy tạo 1 tập tin csv ít nhất 10 dòng dữ liệu với các cột dữ liệu như bảng customer trong csdl qlbanhang. Tạo tập tin upload-csv.php với giao diện cho phép upload các tập tin csv, và tập tin upload-csv-processing để xử lý nút xử lý sự kiện upload file csv và đưa dữ liệu vào bảng customers 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.**

****

****