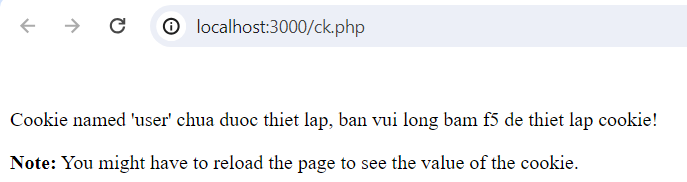
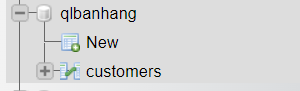
**Câu 1:**

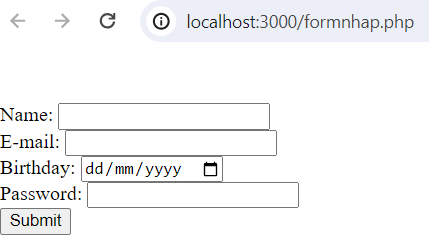
* Chạy tập tin ck.php:

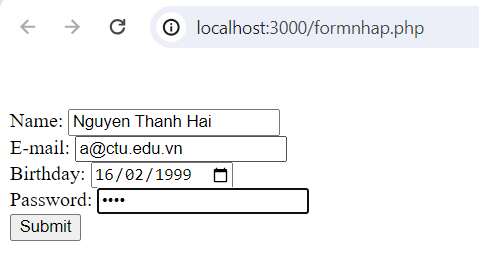


* Tạo csdl qlbanhang và bảng CUSTOMERS:

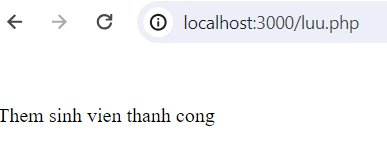


* Tạo formnhap.php:

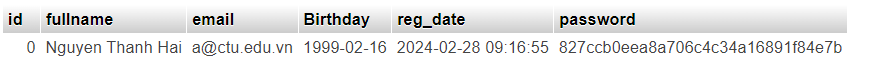




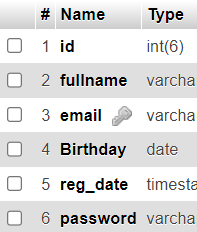
* Tạo luu.php có mã hóa mật khẩu với giải thuật md5 để bảo mật mật khẩu:



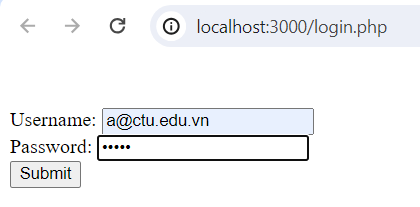
* Sau khi chúng ta nhập xong mật khẩu với 12345 nhưng khi quan sát trong CSDL ta sẽ thấy mật khẩu đã được mã hóa ra chuỗi **827ccb0eea8a706c4c34a16891f84e7b**:



* Chọn email là unique key:

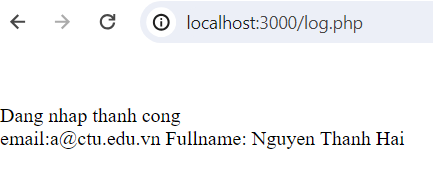


* Tạo tập tin login.php:

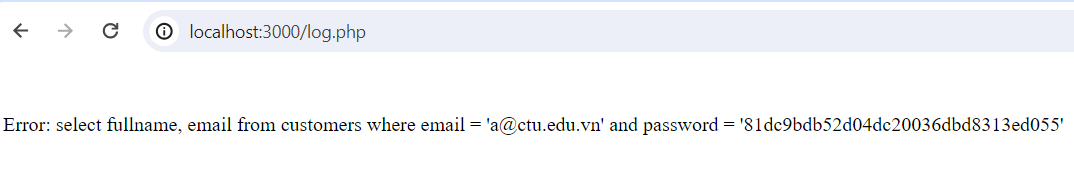


* Tạo tập tin log.php:

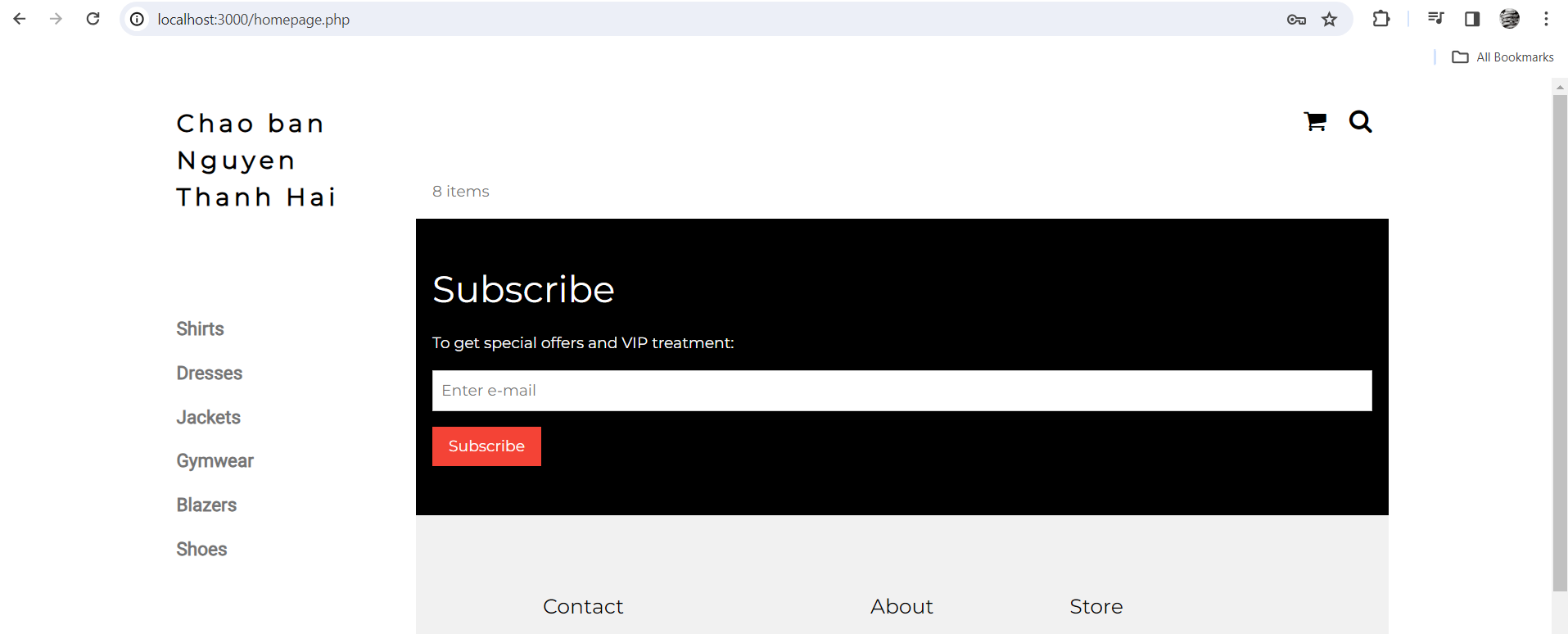
Khi ta nhập đúng:



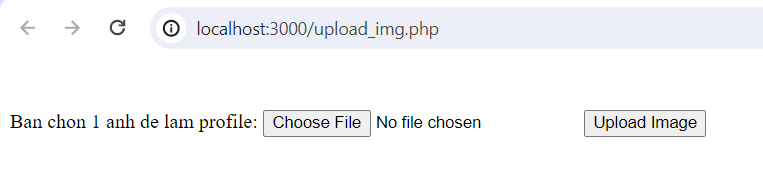
Khi ta nhập sai:



* Tạo tập tin homepage.php:

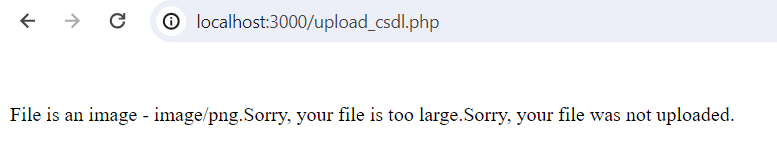


* Tạo tập tin upload\_img.php:

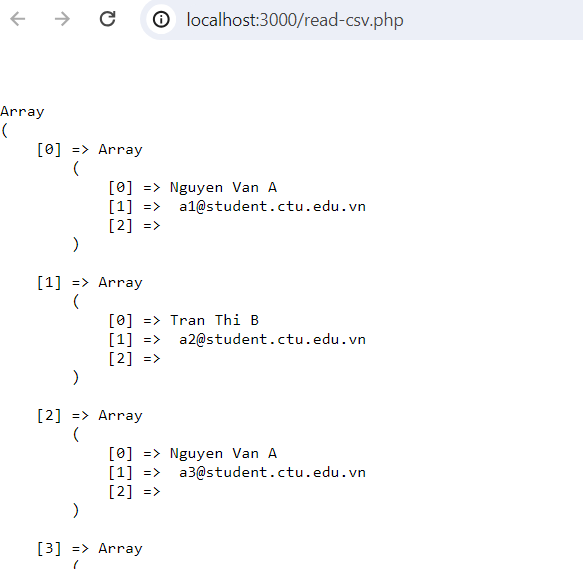


* Tạo tập tin upload.php:

Khi không tải ảnh lên được:

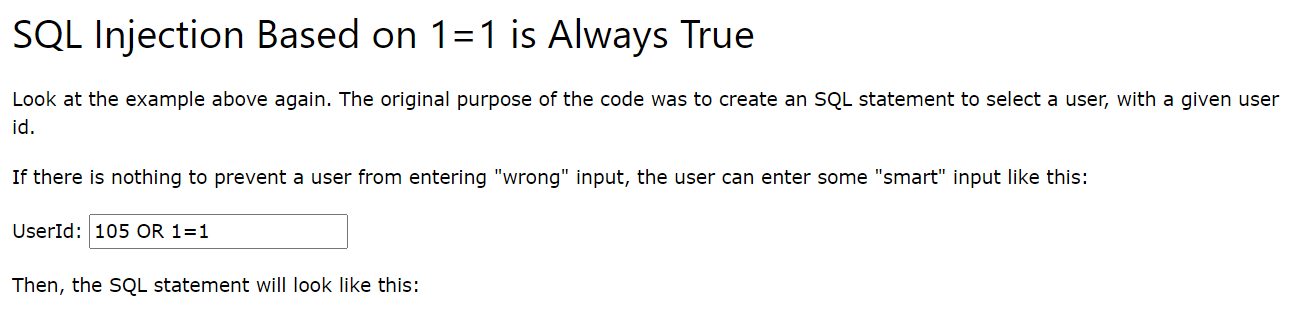


* Upload tập tin cus.csv:



**Câu 2:**

* SQL Injection là một kỹ thuật mà hacker lợi dụng lỗ hổng trong dữ liệu đầu vào của các ứng dụng web và sử dụng các câu lệnh SQL bất hợp pháp để đánh cắp dữ liệu user hay cập nhật csdl.



* Khi hacker lợi dụng lỗ hổng và nhập một chuỗi SQL như OR ‘1=1’, Vì '1'='1' là luôn luôn đúng trong SQL, điều này sẽ khiến cho truy vấn trả về tất cả các dòng từ bảng người dùng, và do đó, người dùng sẽ được cho phép đăng nhập mà không cần biết tên người dùng hoặc mật khẩu chính xác.

**Câu 3:**

* Cookies là tập hợp các tệp thông tin chứa dữ liệu được lưu trữ trên máy tính của người dùng do chính họ tạo ra. Chúng có thể được dùng để lưu trữ thông tin hay ghi nhớ thông tin đăng nhập, tùy chọn hay các hoạt động duyệt web của người dùng.
* **setcookie**(*name, value, expire, path, domain, secure, httponly*);

+ $name: tên cookie

+ $value: giá trị tương ứng với $name

+ $expire: thời gian cookie bị xóa

+ $path: đường dẫn web sử dụng

+ $domain: tên miền website

+ $secure: có sử dụng https hay không

* Lưu cookie: setcookie()
* Lấy cookie: $name = $\_COOKIE[‘user’];
* Xóa cookie: đặt thời gian quá hạn: setcookie(“user”,””, time() - 3600);

**Câu 4:**

* Sessions là dữ liệu được lưu trữ trên máy chủ.
* Lưu sessions: session\_start()
* Lấy sessions: $name = $\_SESSION[‘user’];
* Xóa sessions: dùng *session\_unset();* để xóa hết biến, còn *sessions\_destroy();* để xóa hết sessions.

**Câu 5:**

|  |  |  |
| --- | --- | --- |
|  | Cookies | Sessions |
| Địa chỉ lưu | Lưu ở client | Lưu ở server |
| Thời gian | Set được thời gian | Khi trình duyệt đóng |
| Bảo mật | Dễ sửa | Khó sửa |
| Khởi tạo | Không | session\_start(); |

**Câu 6:**

* Sửa tệp log.php:

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlbanhang";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$sql = "SELECT id, fullname, email FROM customers WHERE email = '"

. $\_POST["email"] . "' AND password = '" . md5($\_POST["pass"]) . "'";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

$row = $result->fetch\_assoc();

// Gán giá tri vào session

session\_start();

$\_SESSION['user'] = $row['email'];

$\_SESSION['fullname'] = $row['fullname'];

$\_SESSION['id'] = $row['id'];

header('Location: homepage.php');

} else {

echo "Error: " . $sql . "<br>" . $conn->error;

// Tro ve trang dang nhap sau 3 giay

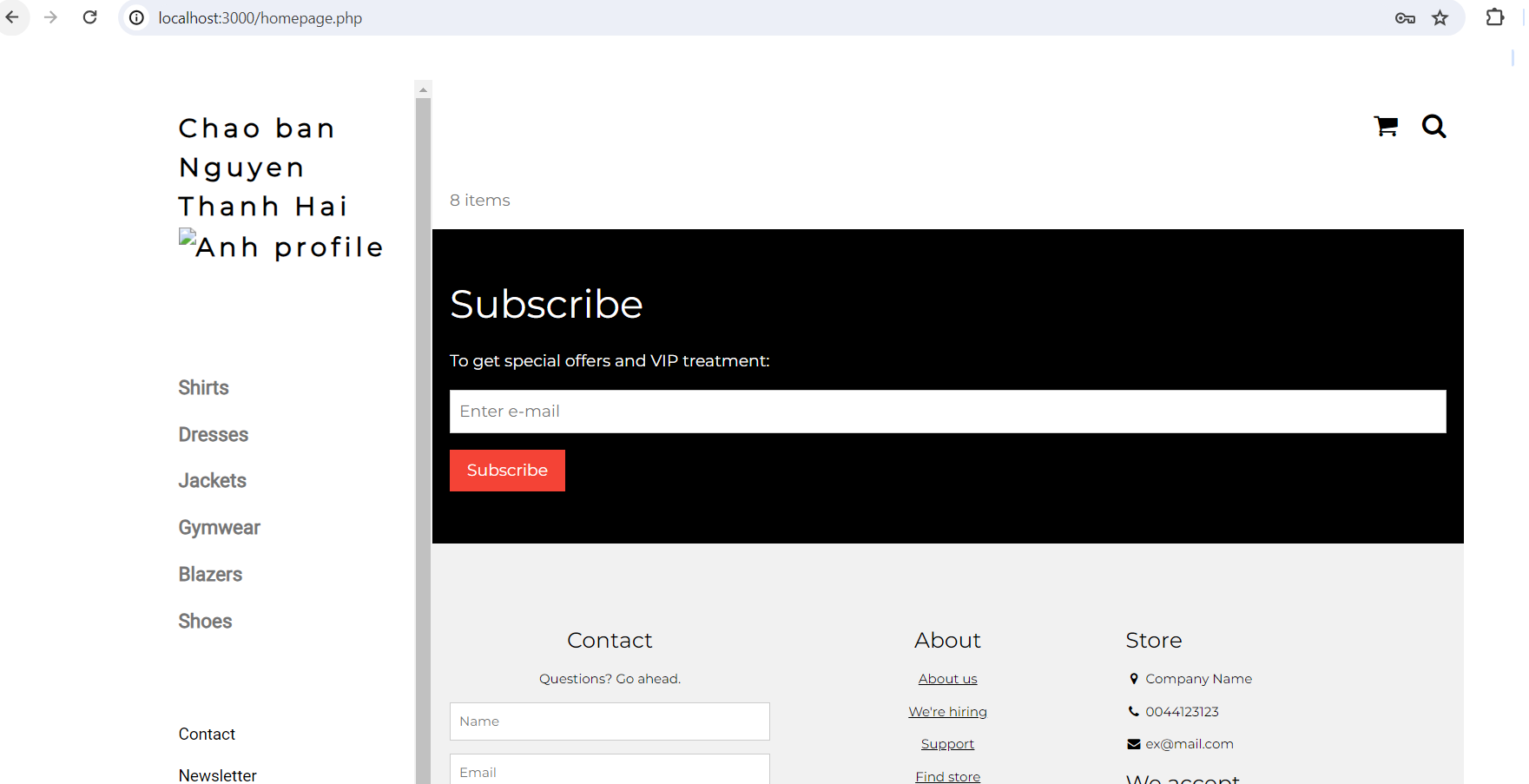
header('Refresh: 3;url=login.php');

}

$conn->close();

?>

* Sửa tệp homepage.php:



* Thêm tệp thoat.php:

<?php

session\_start();

// Xoa cac gia tri trong sesion

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

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

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

// Xoa session

session\_destroy();

//Chuyen huong ve trang dang nhap

header('Location: login.php');

?>

**Câu 7:**

**Câu 8:**

* Sử dụng thẻ ‘input’, method là ‘POST’, chú ý sử dụng thuộc tính enctype = "multipart/form-data” vì nó chỉ định loại nội dung nào sẽ sử dụng khi gửi biểu mẫu. Nếu không thực hiện những yêu cầu trên thì file upload sẽ không hoạt động. Sau đó sử dụng thẻ ‘type = “file”’ để người dùng có thể chọn tệp tin.
* Sử dụng các hàm và thủ tục để xử lý việc tải lên như hàm ‘move\_upload\_file()’ để di chuyển tệp tin được tải lên vào vị trí lưu trữ trên máy chủ và biến siêu toàn cục ‘$\_FILE’ để truy cập thông tin về tệp tin đã tải lên.
* Kiểm tra mã lỗi liên quan qua ‘$\_FILE[‘file’][‘error’]’.

**Câu 9:**

* Hàm file() dùng để đọc một file và xuất ra dưới dạng 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)*.

Với:

+ filename: tên file, cần có đường dẫn tới file.

+ flag: là các cờ tùy chọn (không bắt buộc) với các options: FILE\_USE\_INCLUDE\_PATH, FILE\_IGNORE\_NEW\_LINES, FILE\_SKIP\_EMPTY\_LINES.

+ context: ngữ cảnh của việc xử lý tập tin(không bắt buộc), có thể bỏ qua bằng cách sử dụng NULL.

**Câu 10:**

**TÀI LIỆU THAM KHẢO**

**1 https://www.w3schools.com/sql/sql\_injection.asp**

**2 https://www.w3schools.com/php/php\_cookies.asp**

**3 https://www.w3schools.com/php/php\_sessions.asp**

**4 https://www.w3schools.com/php/php\_file\_upload.asp**

**5** [**https://www.w3schools.com/php/func\_filesystem\_file.asp**](https://www.w3schools.com/php/func_filesystem_file.asp)