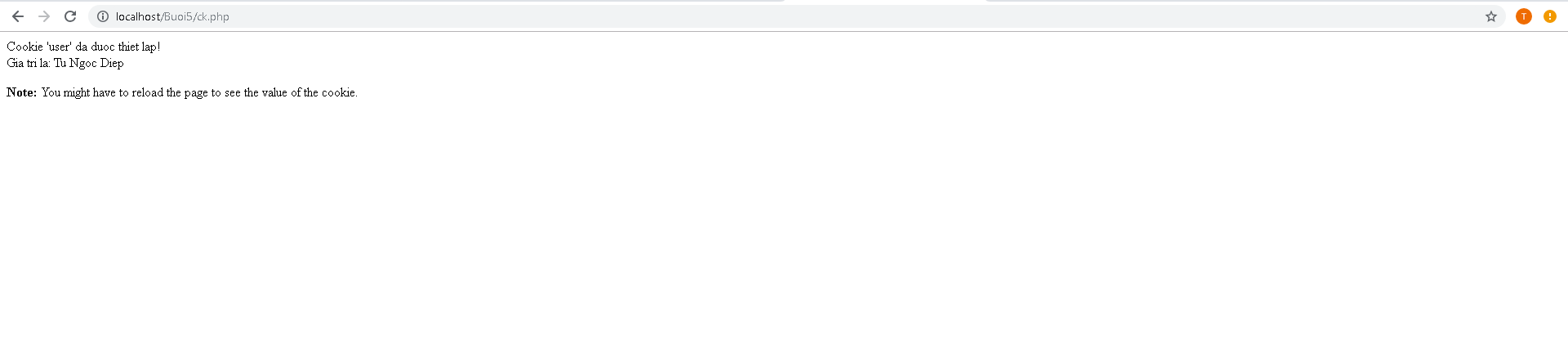
**Tên:** Từ Ngọc Diệp

**MSSV:** B1906307

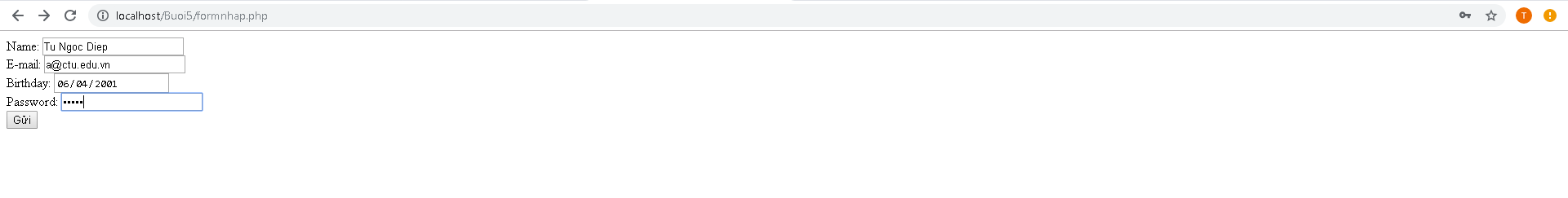
**BUỔI 5**

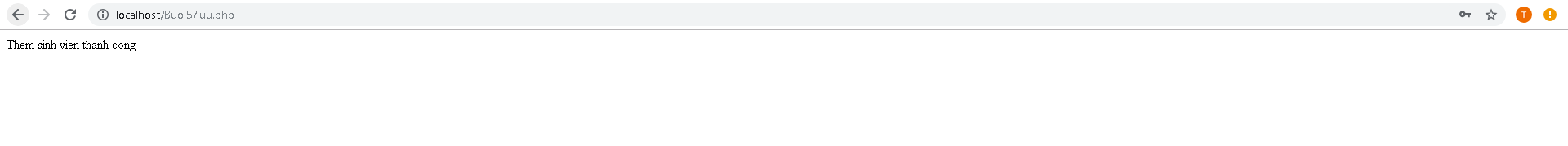
**Câu 1:** Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

Tạo Cookies



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

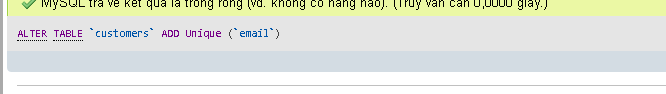




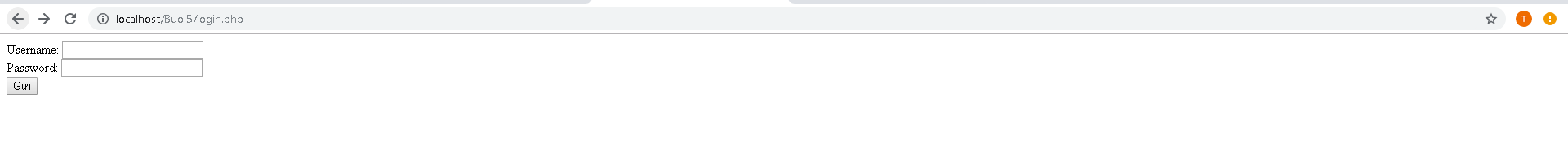
Sau khi chúng ta nhập xong mật khẩu với 12345 nhưng khi quan sát trong CSDL ta sẽ thấy mật khẩu đã được mã hóa ra chuỗi 827ccb0eea8a706c4c34a16891f84e7b:



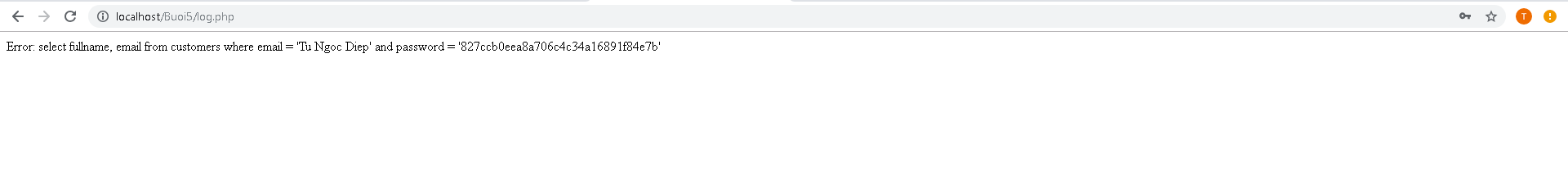
Trong trường hợp chúng ta muốn dùng email là username để đăng nhập ta nên đảm bảo 2 student không có email trùng nhau. Để đảm bảo điều này ta có thể chọn email là unique key (vì ta đã chọn id la primary key, và 1 bảng không thể có 2 khóa chính)



Tạo form đăng nhập

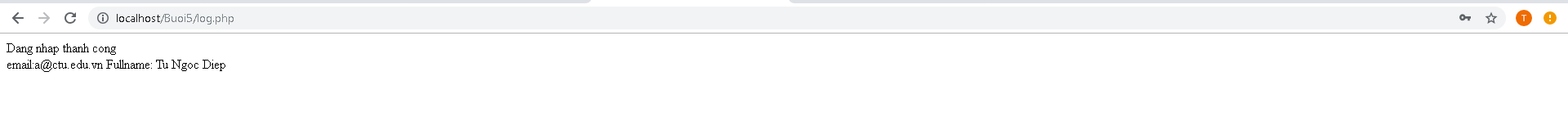


Khi chúng ta nhập sai, nhìn vào thông báo lỗi ta thấy

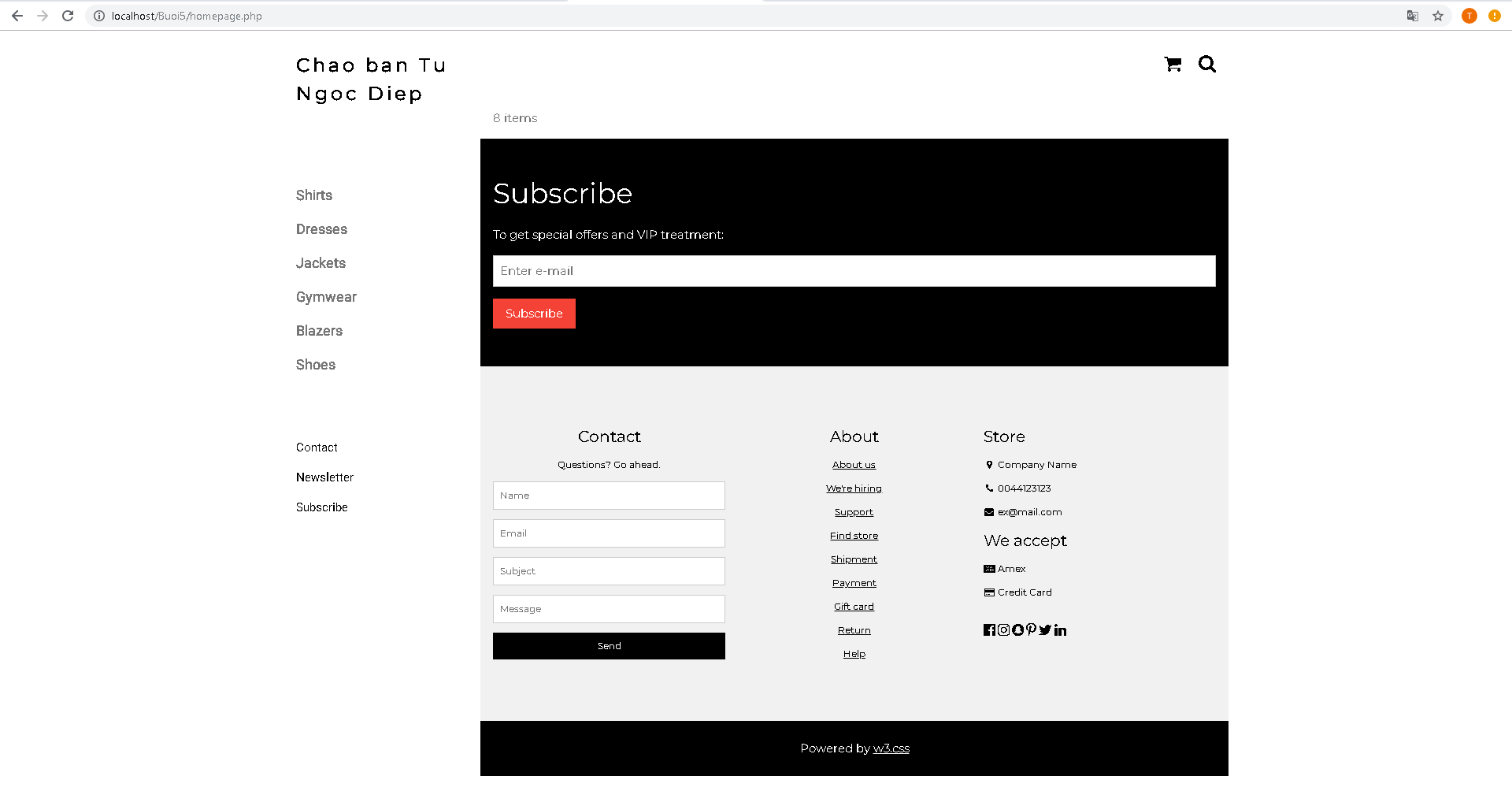


Mật khẩu đã cho không khớp với mã đã băm

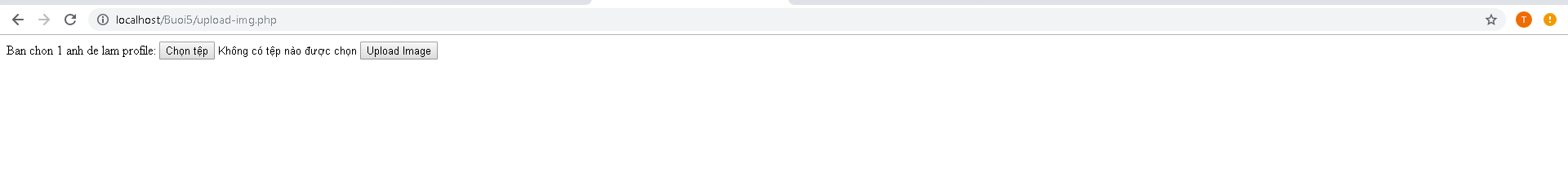
Khi ta nhập đúng

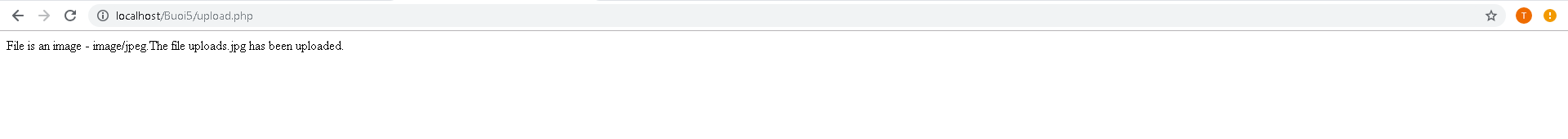


Dùng Cookies lưu lại tên đăng nhập, email và điều hướng

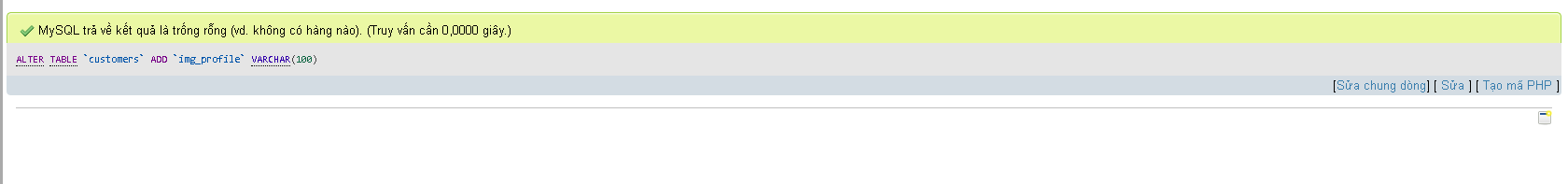


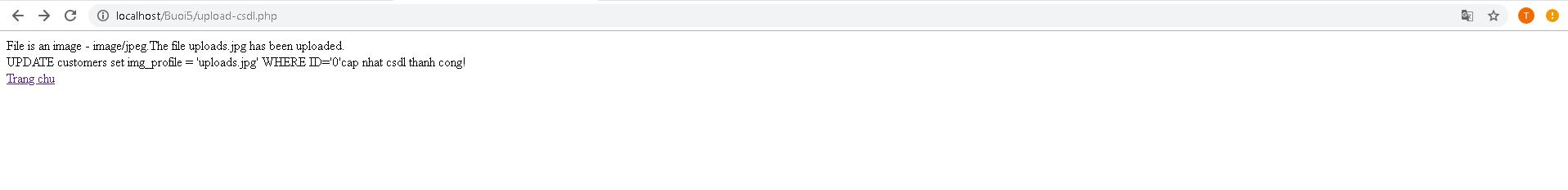
## Upload tập tin

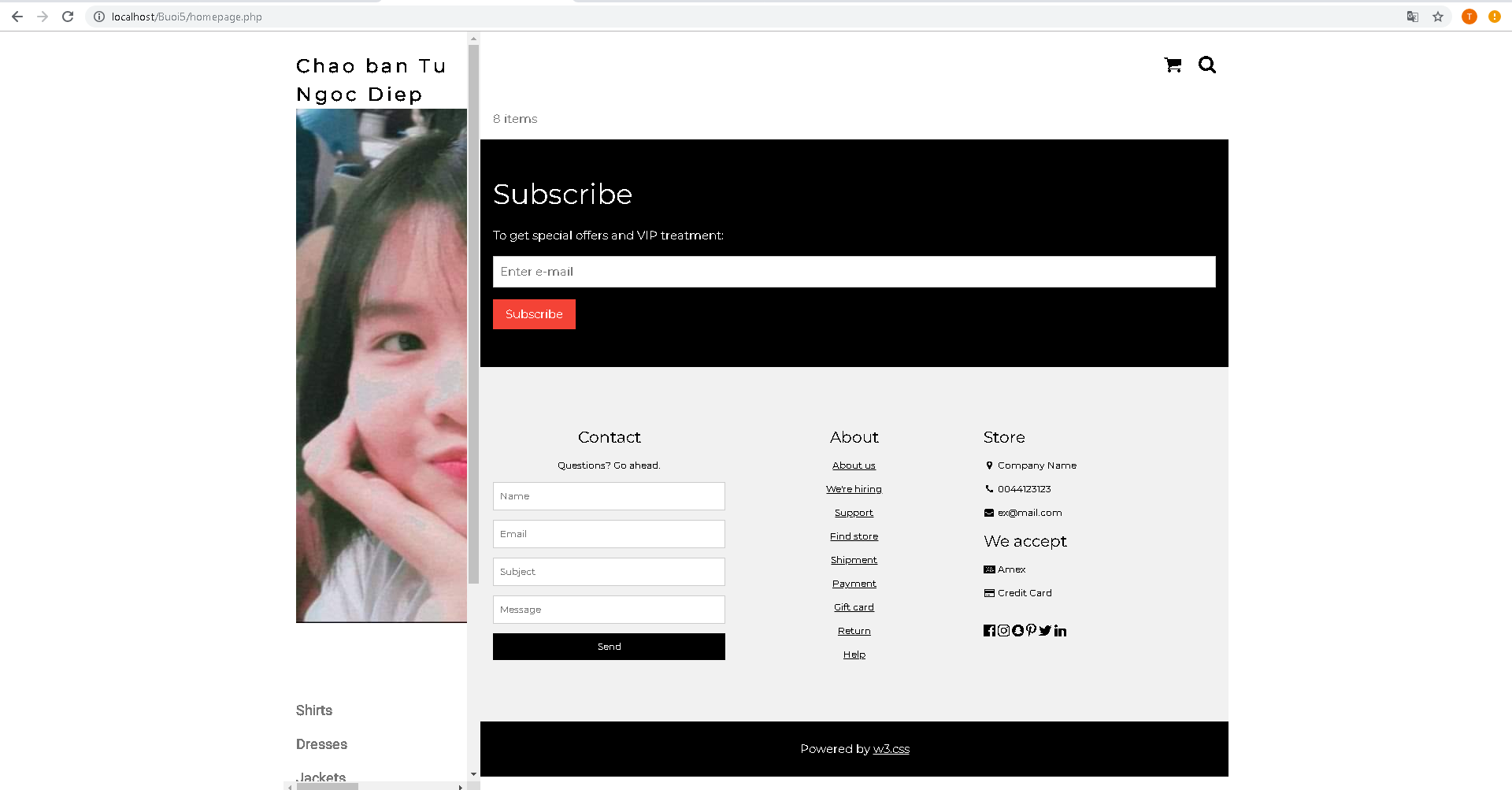




## Upload ảnh và lưu dữ liệu về ảnh trong CSDL

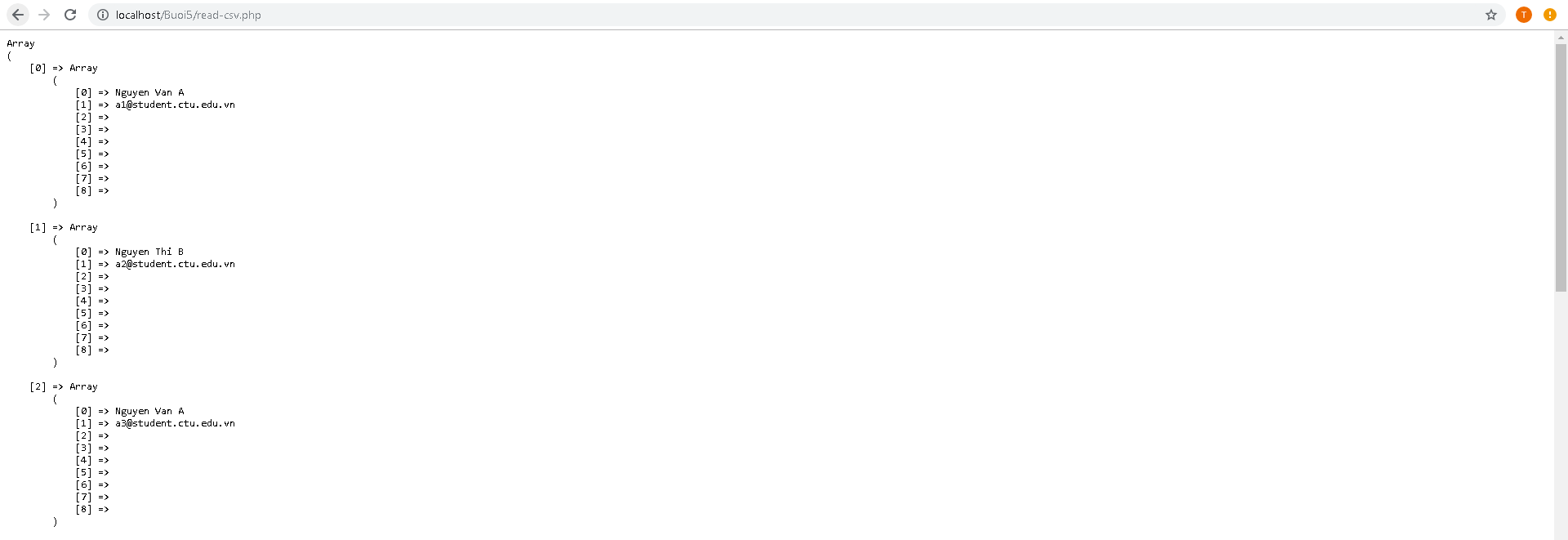






Ở đây do biểu diễn từng bước nên đoạn này chúng ta lấy dữ liệu link ảnh profile từ id của người dùng. Trong thực tế, các bạn có thể lưu link ảnh profile vào cookie/session ngay từ khi đăng nhập.

Upload file csv và đọc dữ liệu vào mảng



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

* SQL Injection là một trong những kiểu hack web bằng cách inject các mã SQL query/command vào input trước khi chuyển cho ứng dụng web xử lí, bạn có thể login mà không cần username và password, remote execution (thực thi từ xa), dump data và lấy root của SQL server. Công cụ dùng để tấn công là một trình duyệt web bất kì, chẳng hạn như Internet Explorer, Netscape, Lynx, ...
* Ví dụ:

txtUserId = getRequestString("UserId");

txtSQL = "SELECT \* FROM Users WHERE UserId = " + txtUserId;

uName = getRequestString("username");

uPass = getRequestString("userpassword");

sql = 'SELECT \* FROM Users WHERE Name ="' + uName + '" AND Pass ="' + uPass + '"'

SELECT \* FROM Users WHERE UserId = 105 OR 1=1;

SELECT UserId, Name, Password FROM Users WHERE UserId = 105 or 1=1;

Link tham khảo: <https://quantrimang.com/tan-cong-kieu-sql-injection-va-cac-phong-chong-trong-asp-net-34905>

**Câu 3**: Dựa vào link, bạn hãy cho biết Cookies là gì, diễn giải ý nghĩa các tham số trong setcookie(). Cách lưu, lấy, xóa giá trị trong cookie.

* Cookie thường được sử dụng để xác định người dùng. Cookie là một tệp nhỏ mà máy chủ nhúng vào máy tính của người dùng. Mỗi khi cùng một máy tính yêu cầu một trang có trình duyệt, nó cũng sẽ gửi cookie. Với PHP, bạn có thể vừa tạo và lấy các giá trị cookie.
* Ý nghĩa các tham số trong setcookie(): setcookie(name, value, expire, path, domain, secure, httponly);
  + Name: Chuỗi mang tên của cookie
  + Value: Chuỗi mang giá trị của cookie tương ứng với tên đã cho
  + Expire: Thời gian hết hạn của cookie
  + Path: Đường dẫn của cookie (đến 1 thư mục nào đó trên máy chủ. Tham số này cho biết cookie sẽ chỉ được truyền đi nếu như trang web mà trình duyệt yêu cầu nằm trên thư mục đó, thay vì lúc nào cũng phải gửi đi tới bất kỳ khu vực nào).
  + Domain: cookie này sẽ được gửi tới domain nào
  + Secure: Chế độ bảo mật.
* Cách lưu, lấy, xóa giá trị trong cookie:

Xóa:

<?php  
// set the expiration date to one hour ago  
setcookie("user", "", time() - 3600);  
?>  
<html>  
<body>  
  
<?php  
echo "Cookie 'user' is deleted.";  
?>  
  
</body>  
</html>

**Lấy:**

<?php  
$cookie\_name = "user";  
$cookie\_value = "John Doe";  
setcookie($cookie\_name, $cookie\_value, time() + (86400 \* 30), "/"); // 86400 = 1 day  
?>  
<html>  
<body>  
  
<?php  
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];  
}  
?>  
  
</body>  
</html>

Link tham khảo: <https://www.w3schools.com/php/php_cookies.asp>

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

* Session dùng để lưu lại dữ liệu của người dùng sử dụng website. ... Máy chủ sau khi xác thực được thông tin bạn cung cấp là đúng nó sẽ sinh ra một tập tin chứa dữ liệu cần lưu trữ của người dùng.
* Cách lưu, lấy, xóa giá trị trong Session

Xóa:

<?php  
session\_start();  
?>  
<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
session\_unset();

session\_destroy();   
?>  
  
</body>  
</html>

Lấy:

<?php  
session\_start();  
?>  
<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
print\_r($\_SESSION);  
?>  
  
</body>  
</html>

Link tham khảo: <https://www.w3schools.com/php/php_sessions.asp>

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

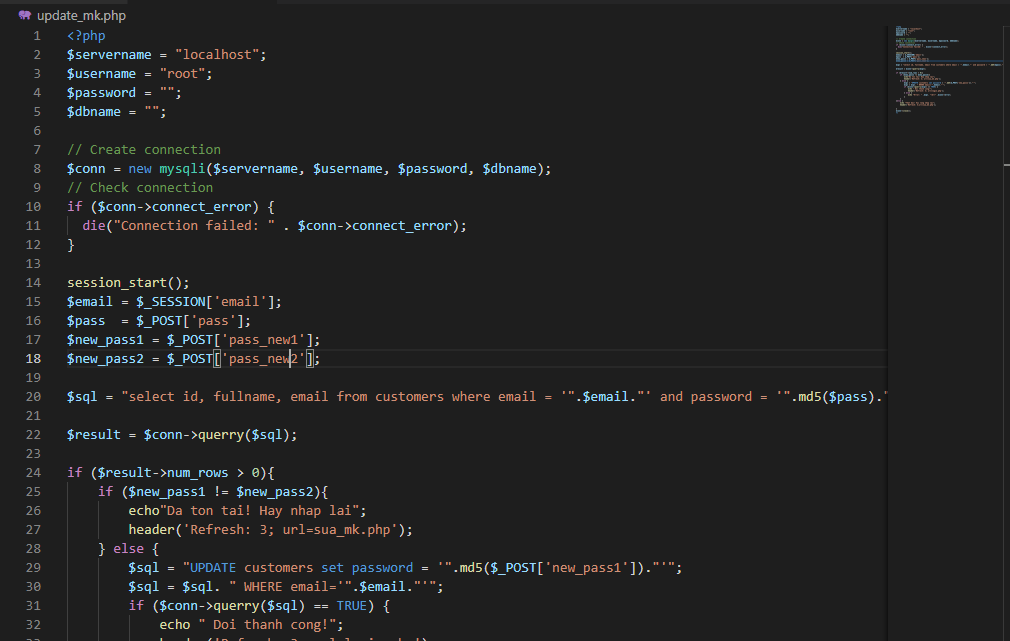
| **Cookie** | **Session session là gì session là gì** |
| --- | --- |
| **Cookie** được lưu trữ trên trình duyệt của người dùng. | **Session** không được lưu trữ trên trình duyệt. |
| Dữ liệu **cookie** được lưu trữ ở phía client. | Dữ liệu **session** được lưu trữ ở phía server. |
| Dữ liệu **cookie** dễ dàng sửa đổi hoặc đánh cắp khi chúng được lưu trữ ở phía client. | Dữ liệu **session** không dễ dàng sửa đổi vì chúng được lưu trữ ở phía máy chủ. |
| Dữ liệu **cookie** có sẵn trong trình duyệt đến khi expired. | Sau khi đóng trình duyệt sẽ hết phiên làm việc (session) |

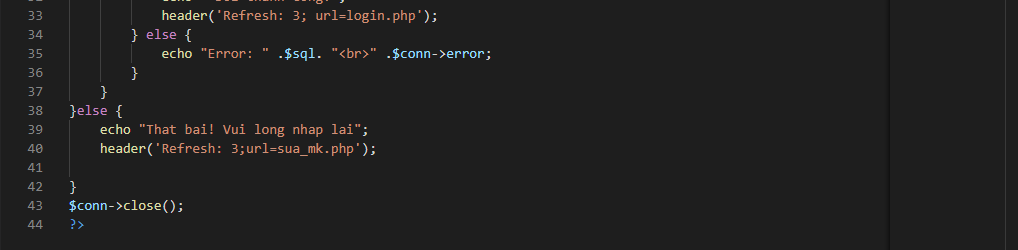
Link tham khảo: <https://topdev.vn/blog/session-la-gi-cookie-la-gi/>

**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.

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

$target\_dir = "upload/";

Chỉ định thư mục nơi tệp sec được đặt

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

Chỉ định đường dẫn của tệp sẽ được tải lên

$uploadOk = 1;

Chưa được sử dụng

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

Giữa phần mở rộng của tệp với tệp

if (file\_exists($target\_file)) {

  echo "Sorry, file already exists.";

  $uploadOk = 0;

}

Kiểm tra file đã tồn tại hay chưa

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

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

  $uploadOk = 0;

}

Giới hạn ích thước tệp

**Câu 9:** Bạn hãy đọc, mô tả chức năng của hàm này và các tham số trong hàm (tham khảo thêm từ read-csv.php).

file(filename, flag, context)

Trong đó:

filename: yêu cầu, chỉ định đường dẫn đến tệp để đọc

flag: Không bắt buộc. Có thể là một hoặc nhiều hằng số sau:

FILE\_USE\_INCLUDE\_PATH - Tìm kiếm tệp trong đường dẫn bao gồm (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: Không bắt buộc. Chỉ định ngữ cảnh của xử lý tệp. Ngữ cảnh là một tập hợp các tùy chọn có thể sửa đổi hành vi của một luồng. Có thể được bỏ qua bằng cách sử dụng NULL.

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

**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