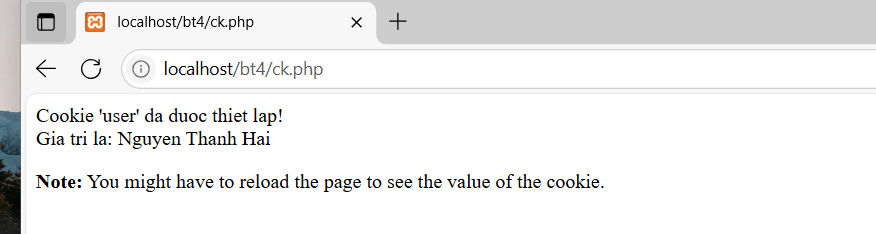
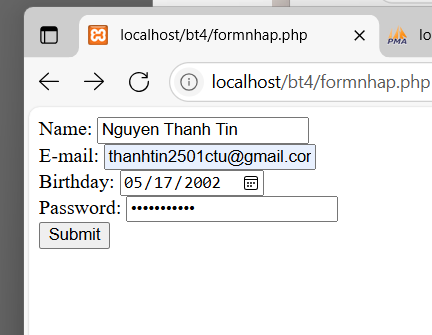
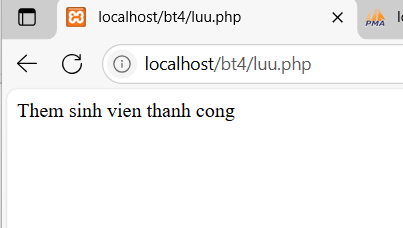
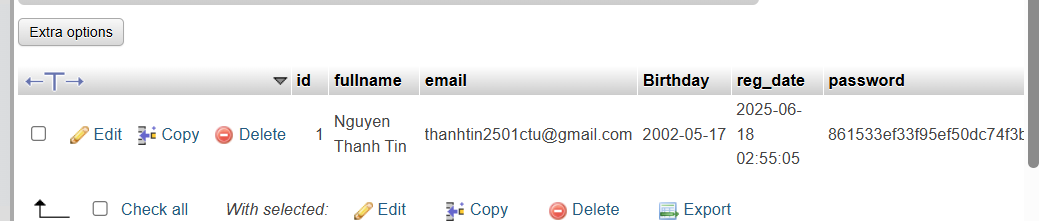
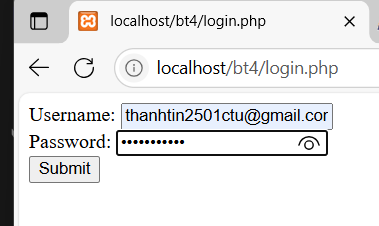
Câu 1:

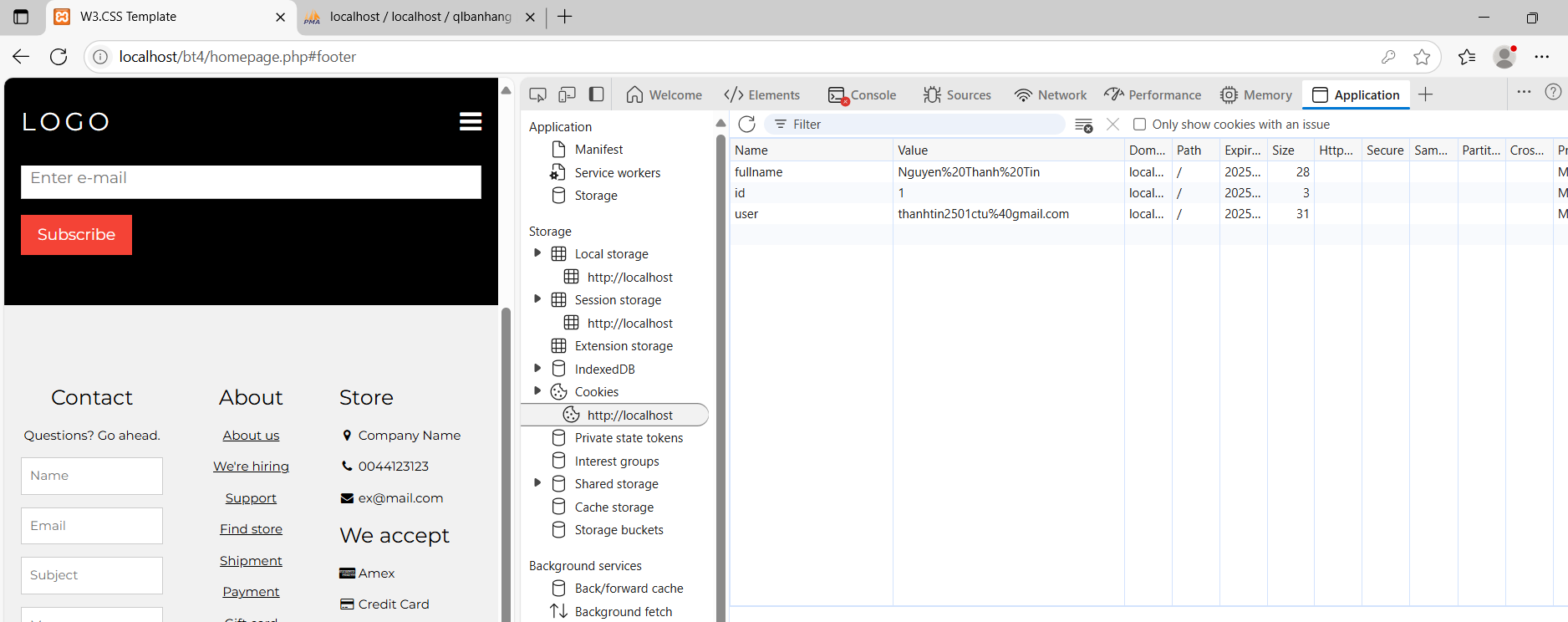
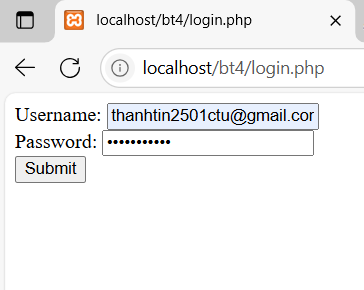


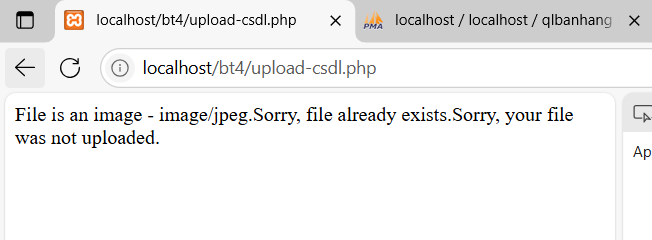
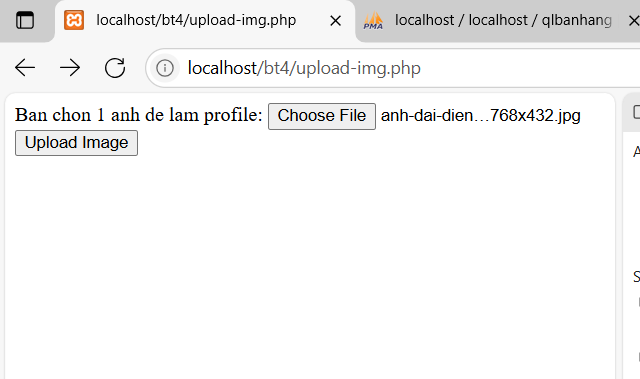


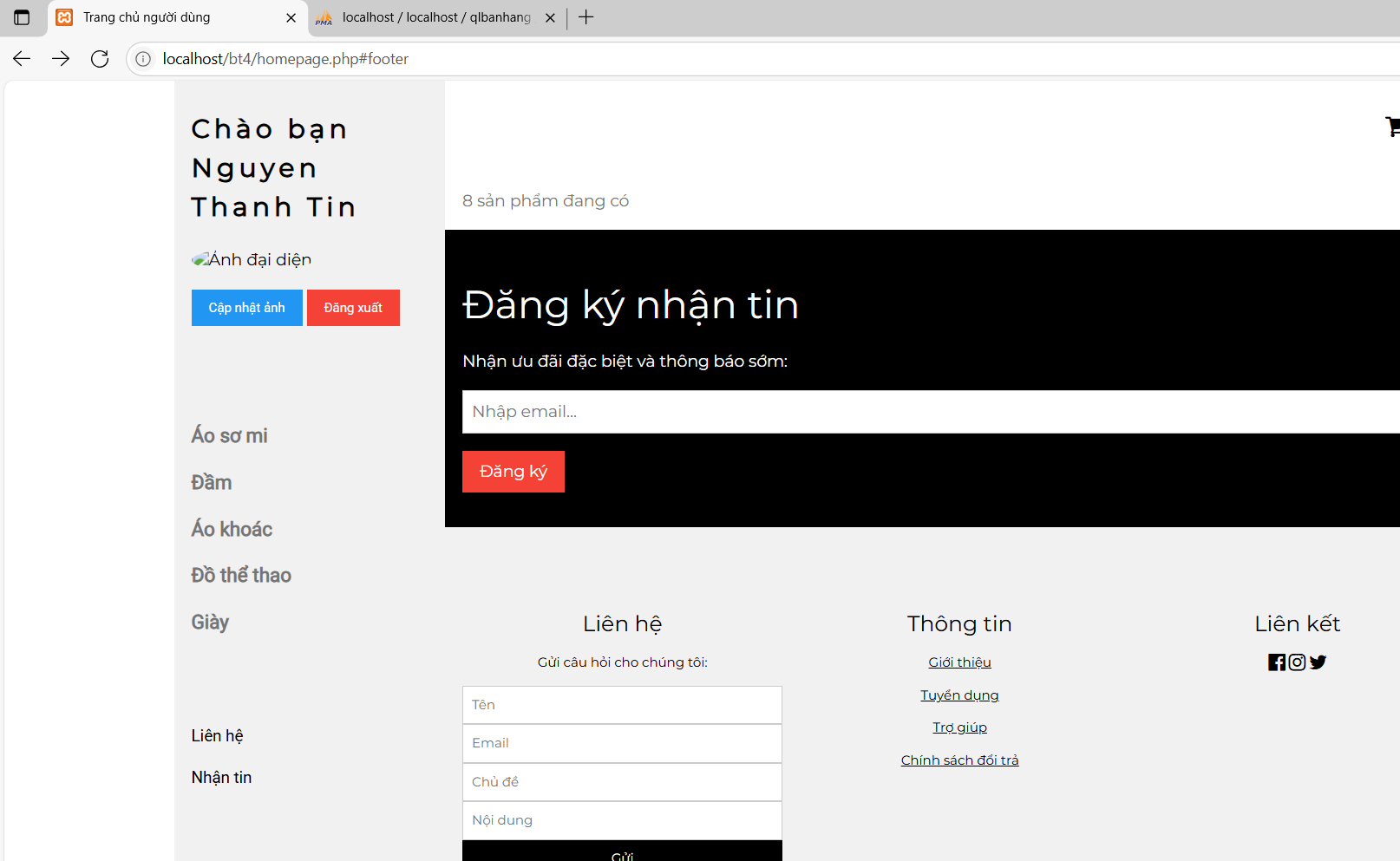


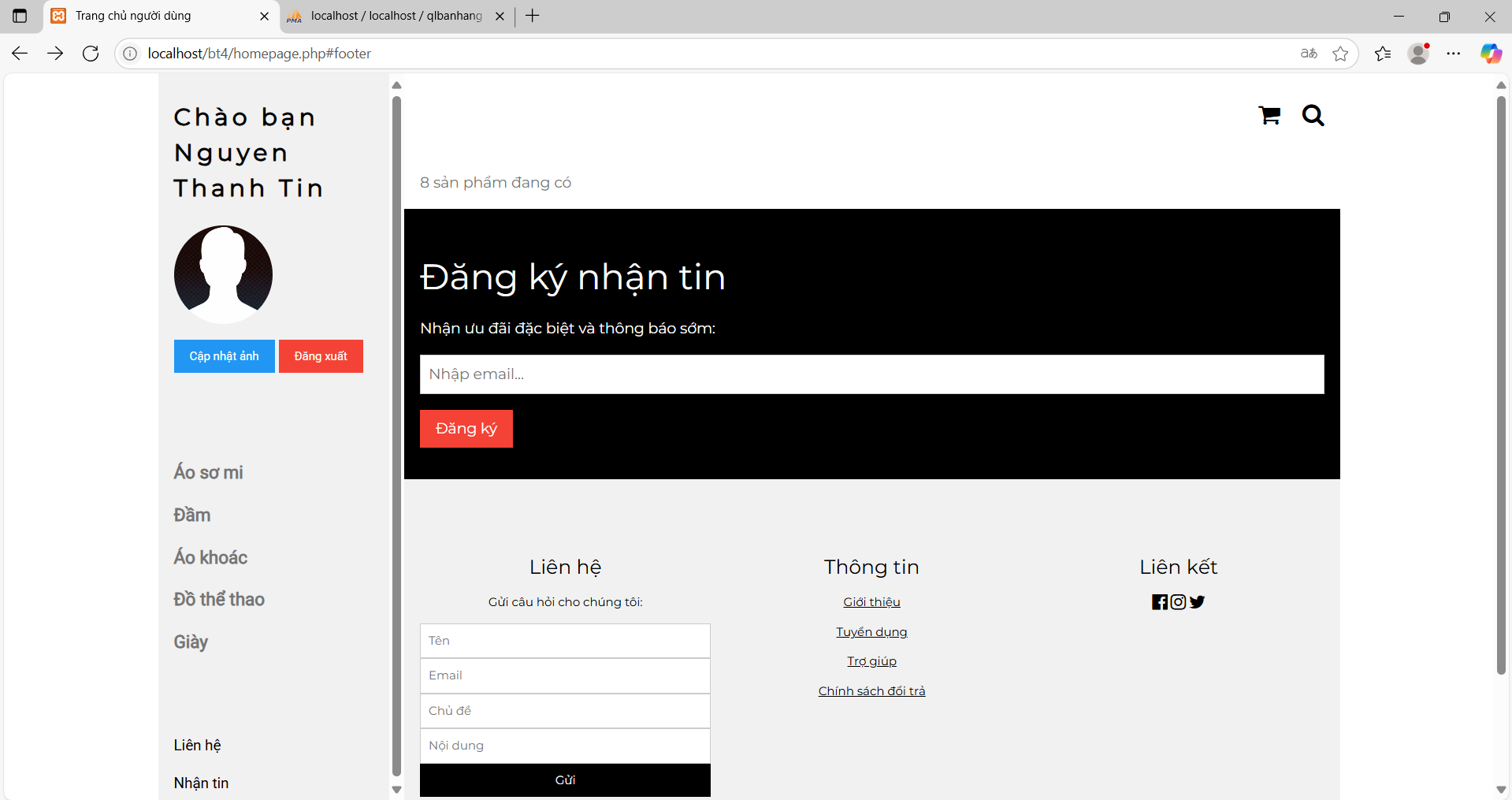


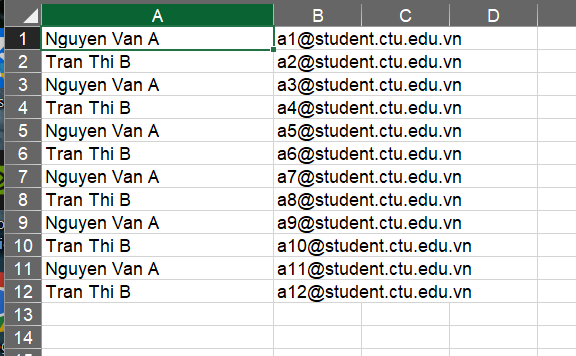


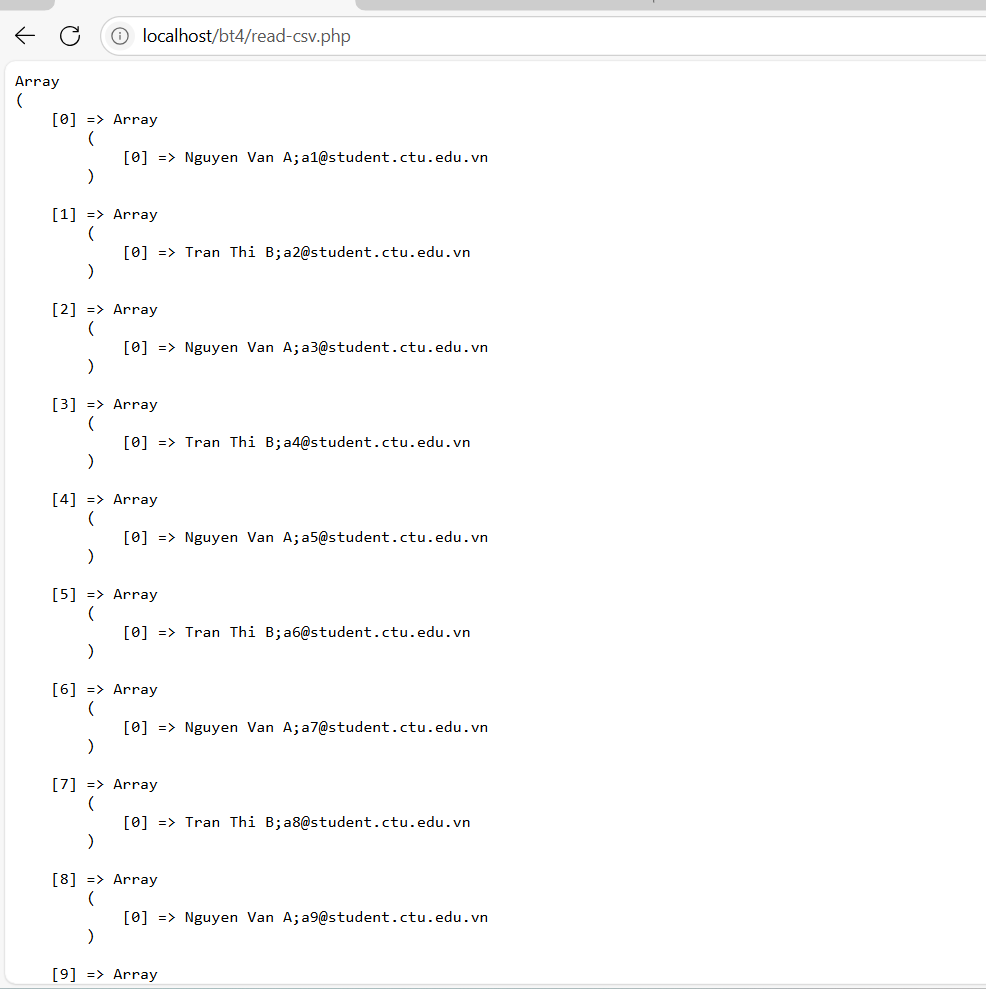












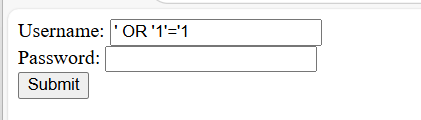
Câu 2:

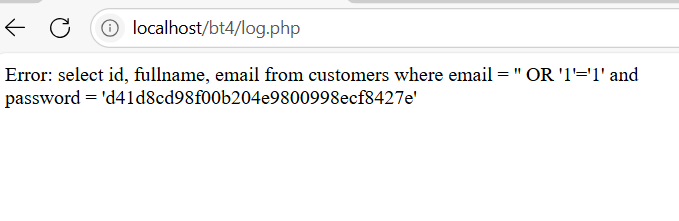
SQL Injection (SQLi) là một kỹ thuật tấn công trong đó kẻ tấn công chèn (inject) các câu lệnh SQL độc hại vào trường đầu vào của người dùng (như tên đăng nhập, mật khẩu) để thao túng cơ sở dữ liệu.

Mục tiêu của SQL Injection có thể là:

* Bypass (vượt qua) đăng nhập.
* Trích xuất thông tin từ database (như mật khẩu, email, thông tin người dùng).
* Chỉnh sửa, xóa hoặc thêm dữ liệu trái phép.
* Thậm chí thực thi lệnh hệ thống nếu server bị cấu hình sai.







Câu 3:

Cookie là một tệp nhỏ được lưu trên trình duyệt của người dùng, dùng để ghi nhớ thông tin giữa các lần truy cập website, như:

* Ghi nhớ đăng nhập
* Ghi nhớ giỏ hàng
* Ghi nhớ lựa chọn ngôn ngữ

Diễn giải các tham số trong setcookie(name, value, expire, path, domain, secure, httponly):

* name: Tên cookie.
* value: Giá trị cookie.
* expire: Thời gian hết hạn (timestamp, ví dụ: time() + 3600 là 1 giờ sau).
* path: Đường dẫn URL trong server mà cookie có hiệu lực ("/" là toàn bộ website).
* domain: Miền áp dụng cookie (ví dụ: "example.com").
* secure: true nếu chỉ gửi cookie qua HTTPS.
* httponly: true nếu chỉ cho phép cookie dùng trong HTTP, không cho JavaScript truy cập (chống XSS).

Cách lấy ,lưu ,xóa cookie:

Lưu : setcookie(name, value, expire, path, domain, secure, httponly);

Ví dụ setcookie("username", "NguyenVanA", time() + 3600, "/");

Lấy: $\_COOKIE["username"];

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

Câu 4:

Session được dùng để lưu trữ dữ liệu tạm thời cho từng người dùng khi họ truy cập website. Không giống như Cookie (lưu ở phía client), Session lưu trữ trên server, giúp bảo mật hơn.

Ứng dụng thường thấy: lưu đăng nhập người dùng, giỏ hàng, trạng thái form.

Cách dùng : Bắt buộc phải gọi session\_start(); trước khi dùng session (thường đặt ở đầu file PHP).

Lưu giá trị vào session:

session\_start();

$\_SESSION["username"] = "giatri";

Lấy giá trị vào session:

session\_start();

echo $\_SESSION["username"];

Xóa một biến session:

session\_start();

unset($\_SESSION["username"]);

xóa toàn bộ session:

session\_start();

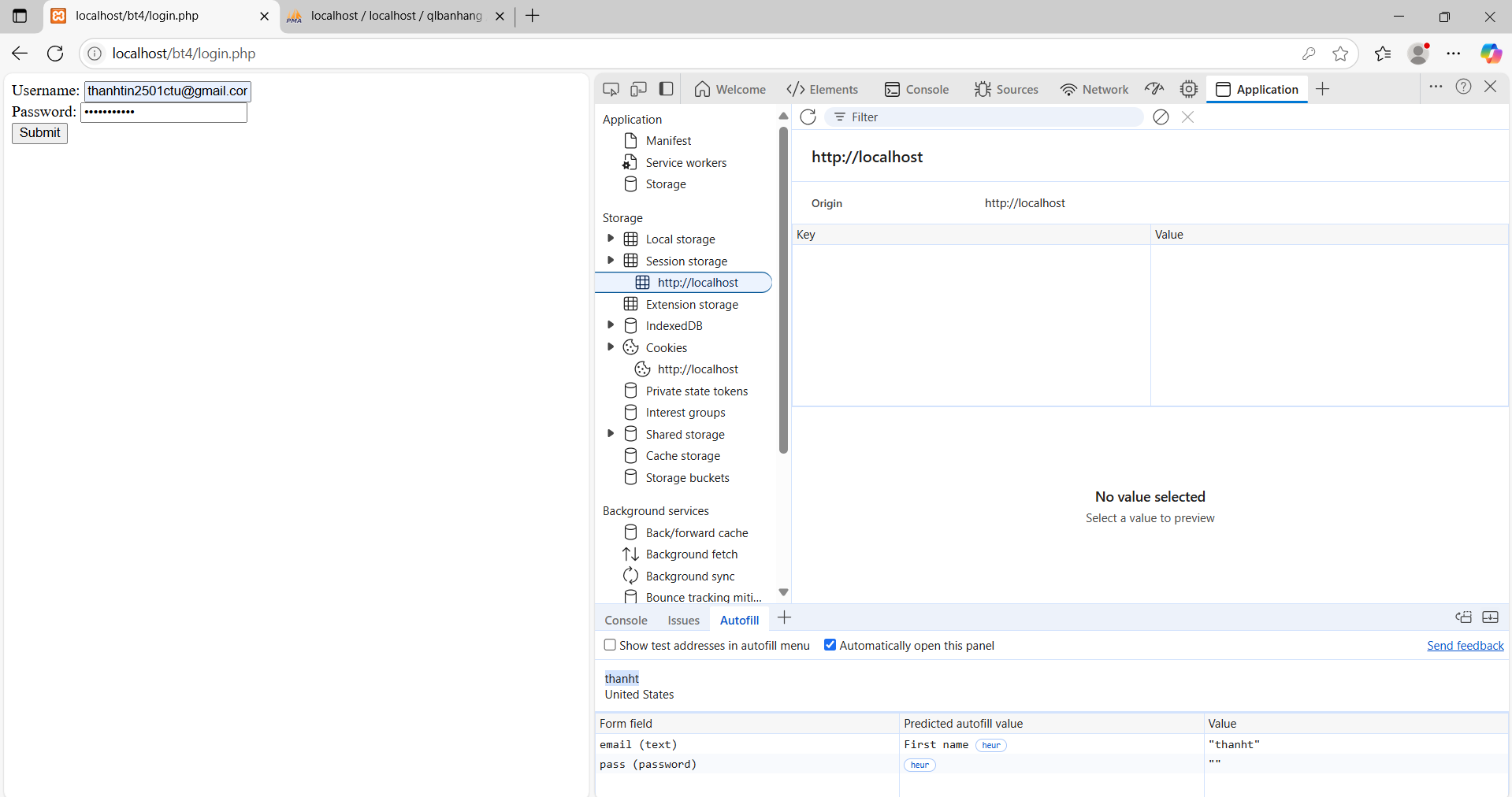
session\_unset(); // Xóa tất cả biến session

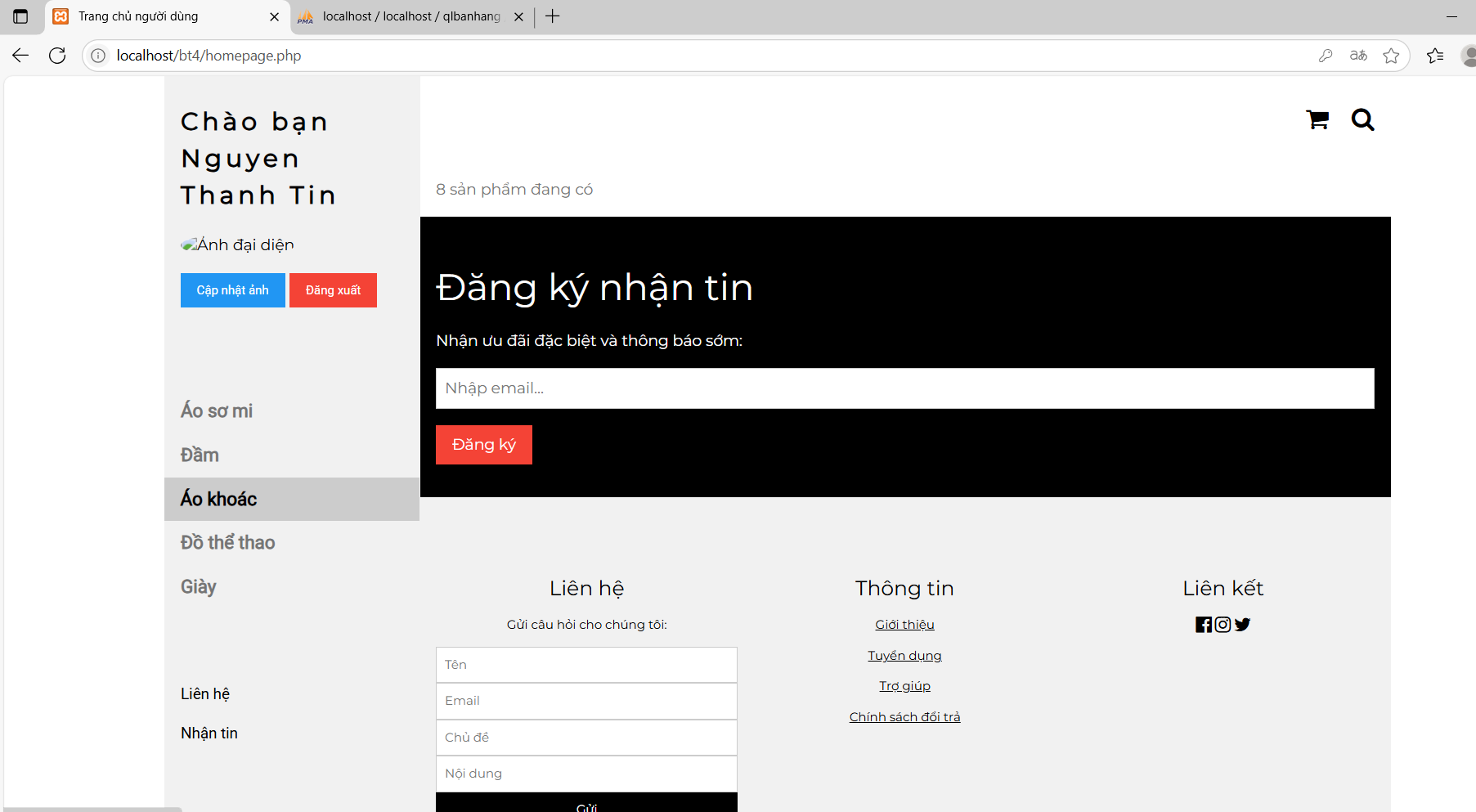
session\_destroy(); // Hủy toàn bộ session

Câu 5:

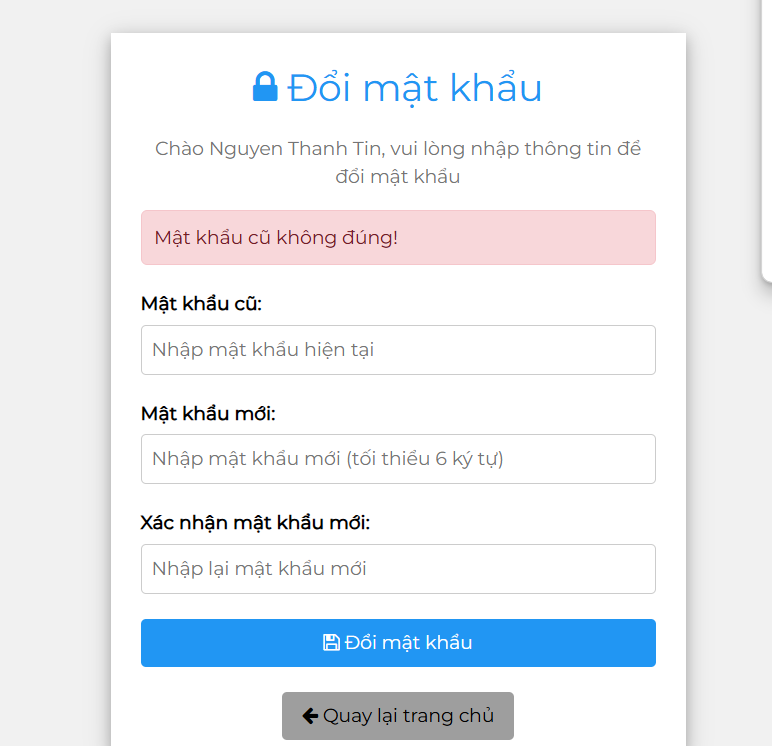
|  |  |
| --- | --- |
| **Tiêu chí** | **So sánh** |
| Lưu trữ | Cookie: Trên máy người dùng (client) Session: Trên máy chủ (server) |
| Kích thước lưu trữ | Cookie: Khoảng 4KB Session: Không giới hạn rõ ràng |
| Bảo mật | Cookie: Kém bảo mật hơn, dễ bị đánh cắp Session: Bảo mật hơn |
| Thời gian sống | Cookie: Có thể thiết lập thời gian hết hạn Session: Tồn tại đến khi đóng trình duyệt hoặc timeout |
| Truy cập | Cookie: Gửi kèm mỗi request Session: Server lưu và xử lý |
| Tốc độ truy xuất | Cookie: Nhanh do lưu trên client Session: Chậm hơn do xử lý server |
| Dễ bị chỉnh sửa | Cookie: Có thể bị chỉnh sửa bởi người dùng Session: Không thể chỉnh sửa bởi người dùng |
| Dùng phổ biến cho | Cookie: Ghi nhớ đăng nhập, theo dõi hành vi Session: Giỏ hàng, lưu trạng thái đăng nhập |

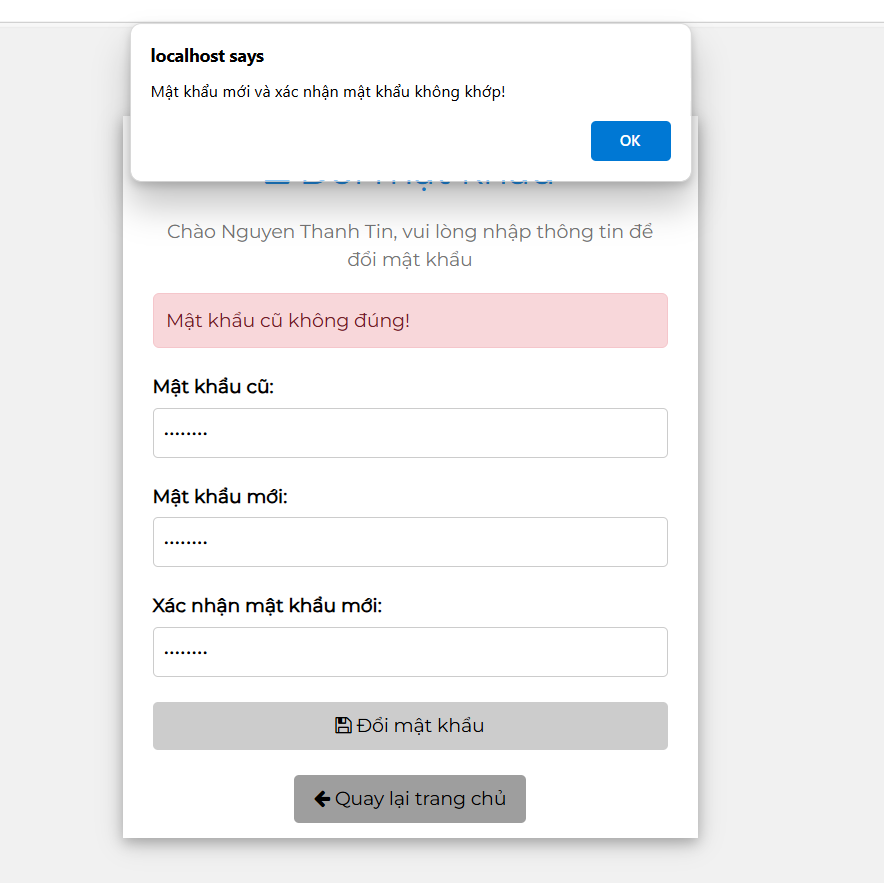
Câu 6:

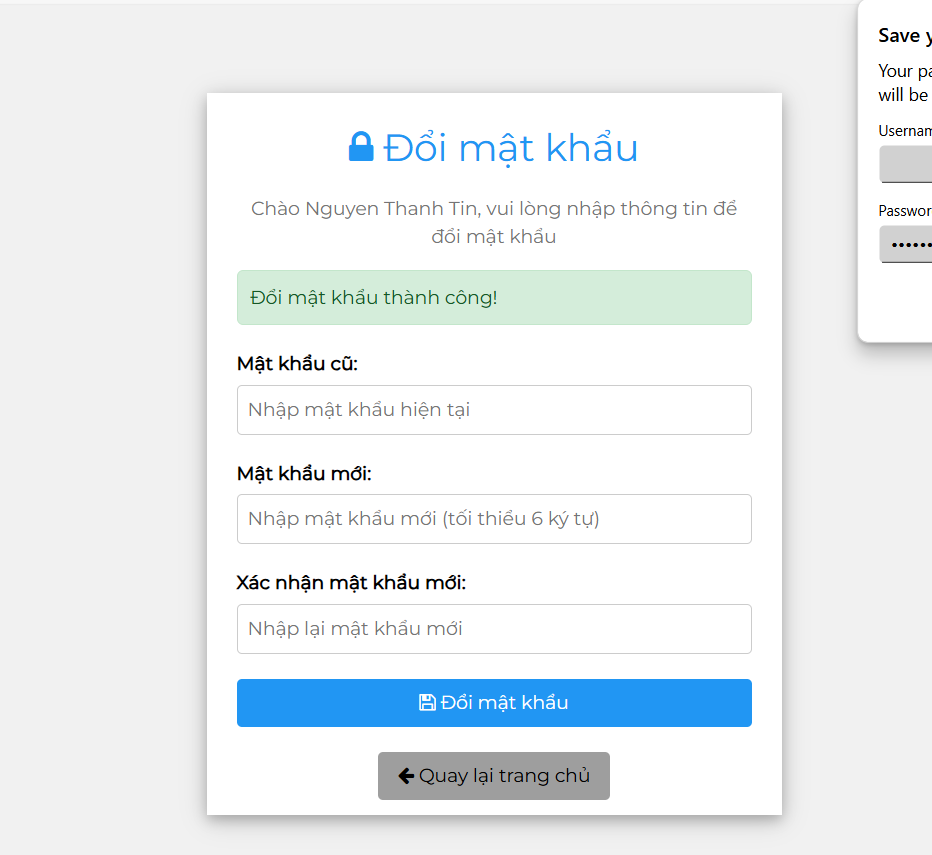




Câu 7:







Câu 8:

Các hàm thủ tục thực hiện việc upload:

**Các hàm chính:**

* $\_FILES["fileToUpload"] - Lấy thông tin file upload
* getimagesize() - Kiểm tra có phải ảnh không
* file\_exists() - Kiểm tra file đã tồn tại chưa
* move\_uploaded\_file() - Upload file lên server
* real\_escape\_string() - Bảo mật SQL

**Quy trình:**

1. **Kiểm tra** file có được chọn không
2. **Validate** kích thước (≤500KB), định dạng (jpg,png,gif), có phải ảnh thực
3. **Upload** file vào thư mục "uploads/"
4. **Lưu** tên file vào database
5. **Thông báo** kết quả

**Biến quan trọng:**

* $uploadOk - Cho phép upload hay không (1/0)
* $target\_file - Đường dẫn file sẽ lưu
* $imageFileType - Định dạng file

**Tóm tắt:** Code kiểm tra file → validate → upload → lưu DB → thông báo

Câu 9:

Chức năng của hàm và các tham số trong hàm :

file($name\_file, FILE\_IGNORE\_NEW\_LINES)

* **Chức năng**: Đọc toàn bộ file thành mảng, mỗi dòng là 1 phần tử
* **Tham số**:
* $name\_file (string): Tên hoặc đường dẫn file cần đọc
* FILE\_IGNORE\_NEW\_LINES (flag): Loại bỏ ký tự xuống dòng (\n, \r) ở cuối mỗi dòng
* **Trả về**: Mảng chứa từng dòng của file

str\_getcsv($value)

* **Chức năng**: Phân tích chuỗi CSV thành mảng các giá trị
* **Tham số**:
* $value (string): Chuỗi CSV cần phân tích (1 dòng)
* **Trả về**: Mảng các giá trị được tách bởi dấu phẩy

print\_r($csv)

* **Chức năng**: In cấu trúc mảng dưới dạng dễ đọc
* **Tham số**:

$csv (array): Mảng cần hiển thị

* **Trả về**: Không có (chỉ in ra màn hình)

foreach ($lines as $key => $value)

* **Chức năng**: Duyệt qua từng phần tử của mảng
* **Tham số**:
  + $lines: Mảng cần duyệt
  + $key: Chỉ số/khóa của phần tử hiện tại
  + $value: Giá trị của phần tử hiện tại

Câu 10:

