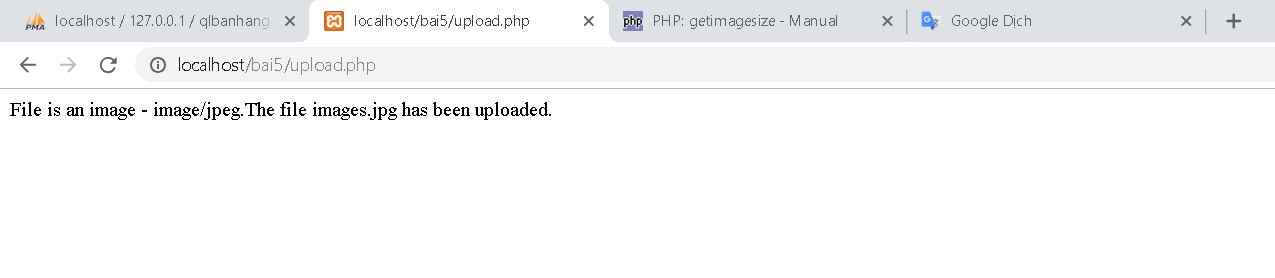
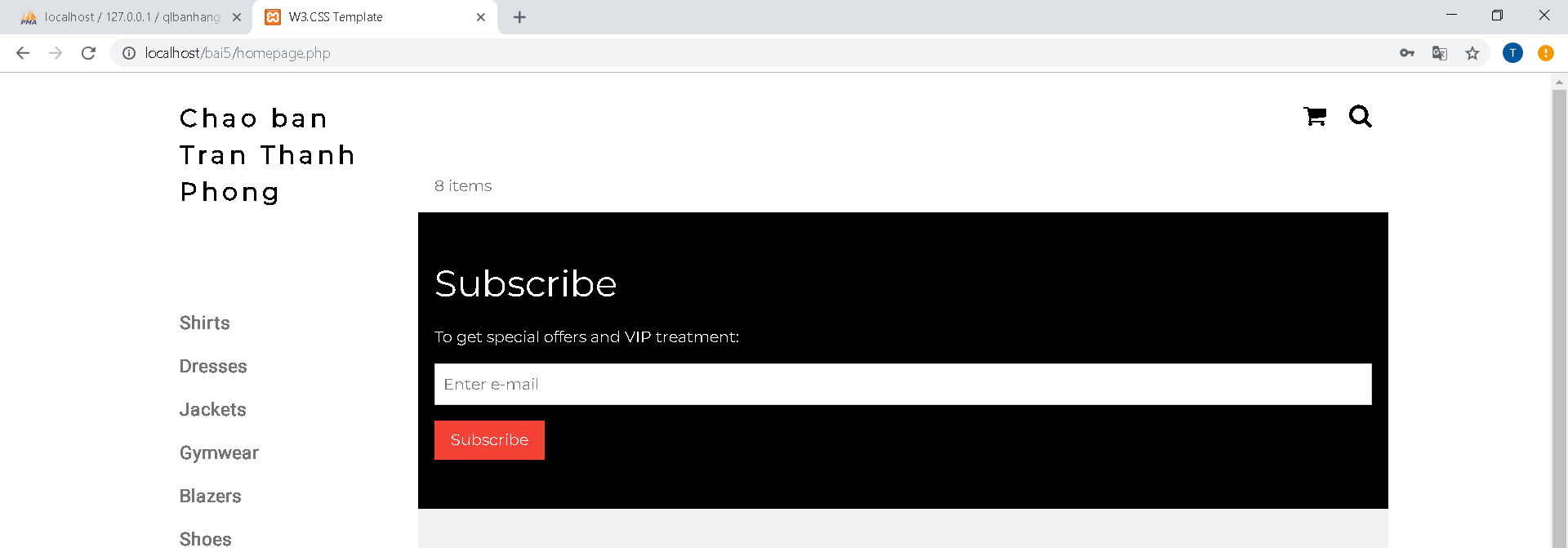
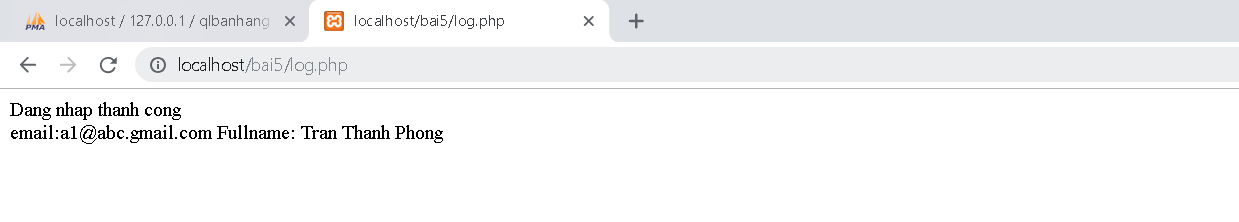
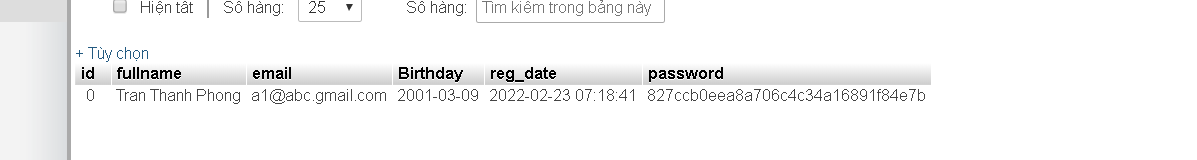
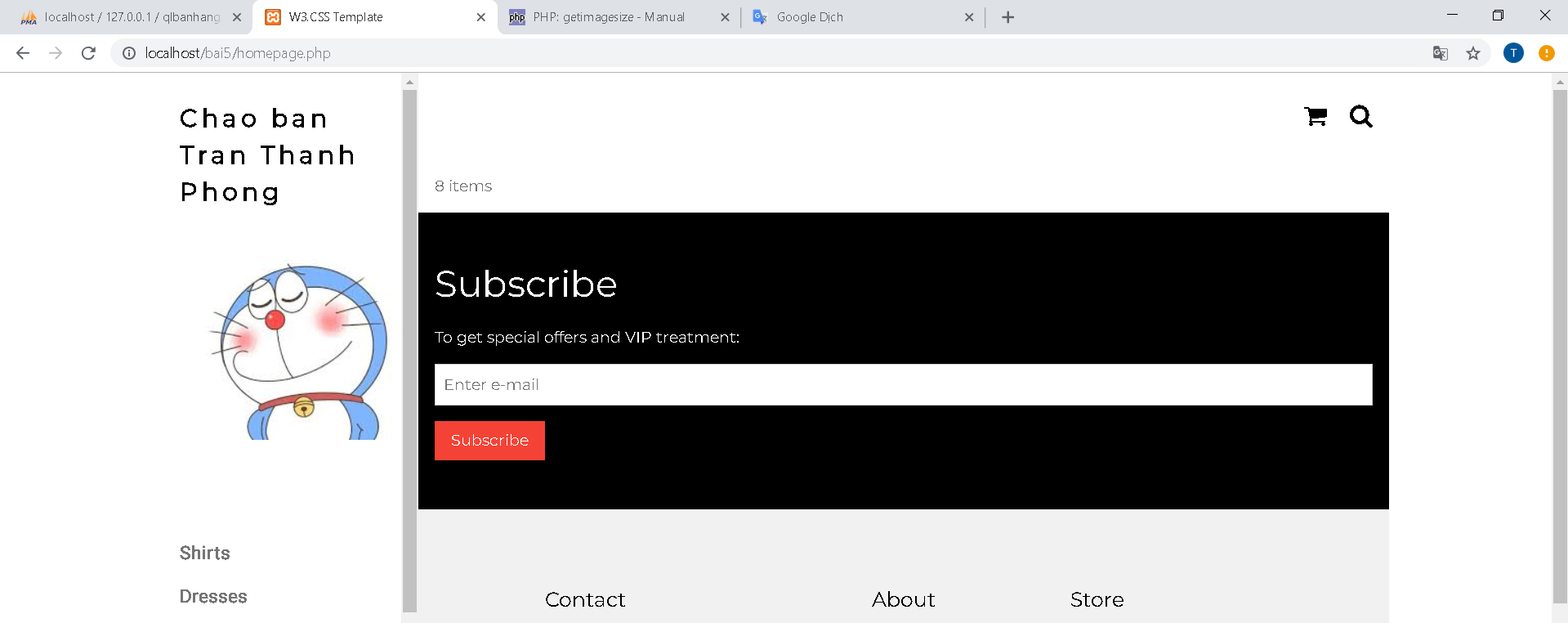
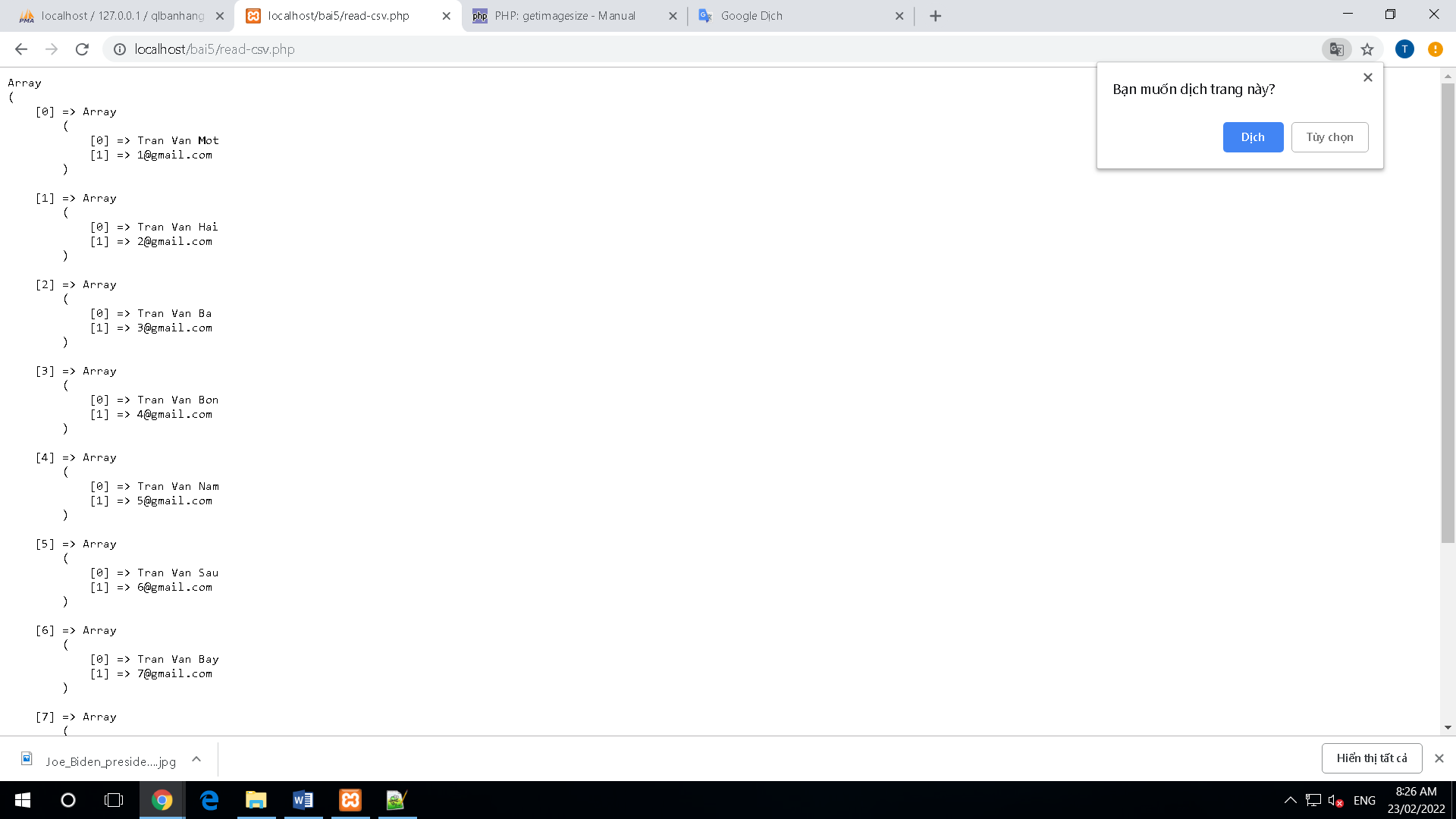
Họ và tên: Trần Thanh Phong – B1906397

***Bài làm***

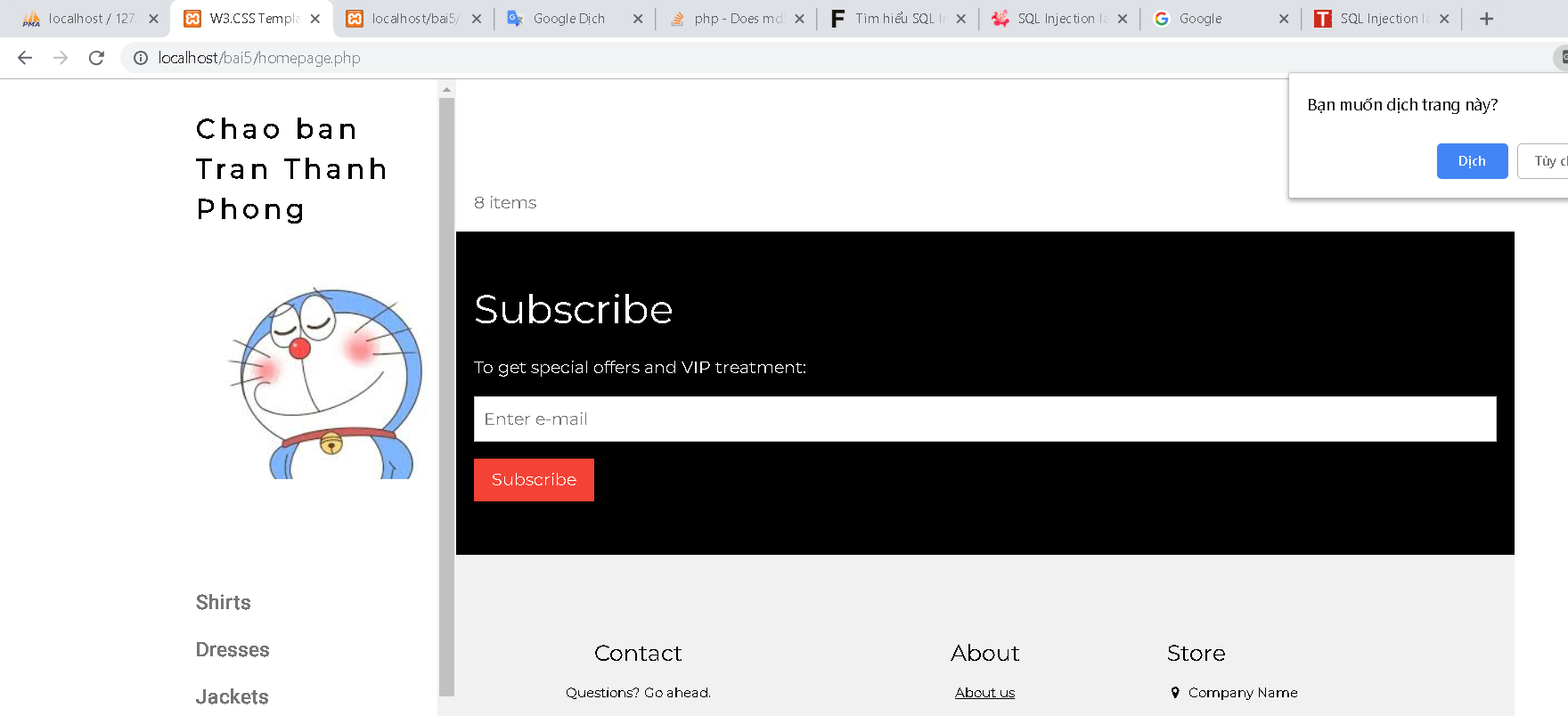
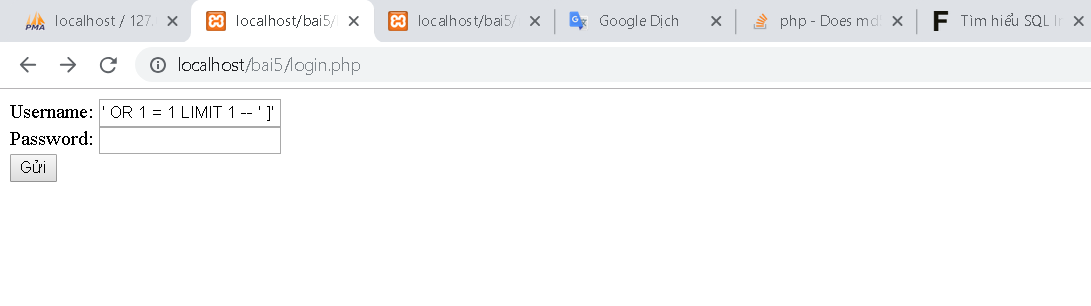
1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.



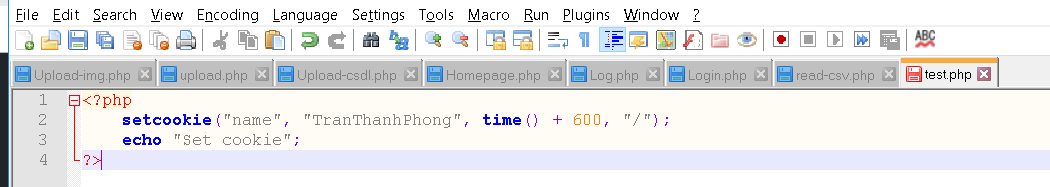
1. Trong vài trường hợp, hacker có thể sử dụng các kỹ thuật SQL Injection để hack hệ thống của bạn. Bạn hãy trình bày SQL Injection là gì và thử các ví dụ trình bày trong . Ứng dụng kỹ thuật đó vào trang đăng nhập mà bạn đã tạo, chụp lại các kết quả.

* **SQL Injection là một kỹ thuật lợi dụng những lỗ hổng về câu truy vấn của các ứng dụng. Được thực hiện bằng cách chèn thêm một đoạn SQL để làm sai lệnh đi câu truy vấn ban đầu, từ đó có thể khai thác dữ liệu từ database. SQL injection có thể cho phép những kẻ tấn công thực hiện các thao tác như một người quản trị web, trên cơ sở dữ liệu của ứng dụng.**
* **Ví dụ:**
  + **Nhập vào ô email với nội dụng ’ OR 1 = 1 LIMIT 1 — ‘ ]**
  + **Nhập vào ô password với nội dụng bất kỳ, sau đó click vào nut Login.**
* **Vì sau kỹ tự “–“, hệ thống sẽ hiểu đó là comment, chương trình sẽ không thực thi những gì phía sau nó. Câu truy vấn có thể rút gọn lại:**
  + **select id, fullname, email from customers where email = ‘’ OR 1 = 1 LIMIT 1**



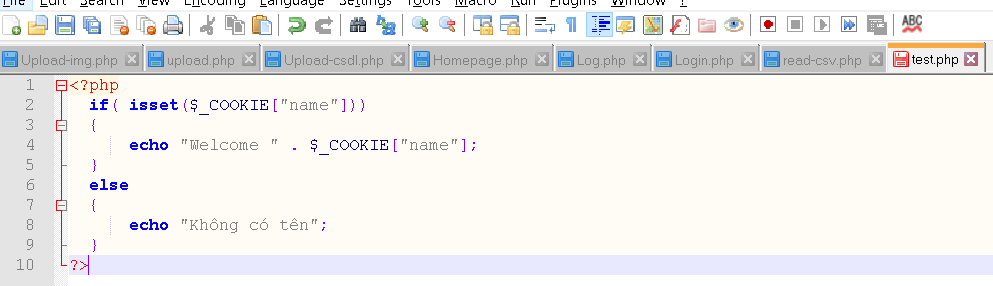
1. Dựa vào link, bạn hãy cho biết Cookies là gì, diễn giải ý nghĩa các tham số trong setcookie(). Cách lưu, lấy, xóa giá trị trong cookie.

* **Cookie là một đoạn văn bản ghi thông tin được tạo ra và lưu trên trình duyệt của máy người dùng. Cookie thường được tạo ra khi người dùng truy cập một website, cookie sẽ ghi nhớ những thông tin như tên đăng nhập, mật khẩu, các tuỳ chọn do người dùng lựa chọn đi kèm. Các thông tin này được lưu trong máy tính để nhận biết người dùng khi truy cập vào một trang web.**
* **Khi người dùng truy cập đến một trang web có sử dụng cookie, web server của trang đó sẽ tự động gửi cookie đến máy tính của người dùng. Những cookie này tự động được tổ chức trong hệ thống máy tính. Khi truy cập đến các trang web sử dụng được cookie đã lưu, những cookie này tự động gửi thông tin của người dùng về cho chủ của nó (người tạo ra cookie). Tuy nhiên những thông tin do cookie ghi nhận không được tiết lộ rộng rãi, chỉ có website chứa cookie mới có thể xem được những thông tin này. Cookie được xem là một thành phần không thể thiếu được với những website có khối lượng dữ liệu lớn, có số lượng người dùng đông, và có những chức năng đi kèm với thành viên đăng ký. Phần lớn các website này là các website thương mại điện tử.**
* **Function setcookie($name, $value, $expire = 0, $path="", $domain = "", $security = false, $httponly = false);**
  + **Giá trị name là tham số bắt buộc có, còn lại các tham số có thể có hoạc không.**
    - **$name là tên cookie, tên này được sử dụng để truy cập vào biến $\_COOKIE để lấy thông tin**
    - **$value giá trị gán cho cookie**
    - **$expire Thời điểm hết hạn (hủy) của cookie, thời gian dạng Unix timestamp, có thể dùng hàm time() để lấy Unix timestamp. Sau thời điểm này Cookie bị hủy lưu tại trình duyệt. Nếu thiết lập là 0 thì cookie hủy khi đóng trình duyệt**
    - **$path đường dẫn mà cookie có hiệu lực. Sử dụng / cho biết cookie có hiệu lực ở cấp độ domain - mọi URL thuộc domain, nếu thiết lập là /foo/ thì Cookie chỉ có hiệu lực với các URL dạng /foo/ và các con của nó như /foo/abc/... Nếu để giá trị mặc định "" thì có hiệu lực theo đường dẫn hiện tại**
    - **$domain tên miền**
    - **$security = true nếu sử dụng HTTPS để truyền dữ liệu (mặc định false)**
    - **$httponly mặc định là false, nếu true thì chỉ cho phép cookie với giao thức HTTP**
* **Lưu cookie:**



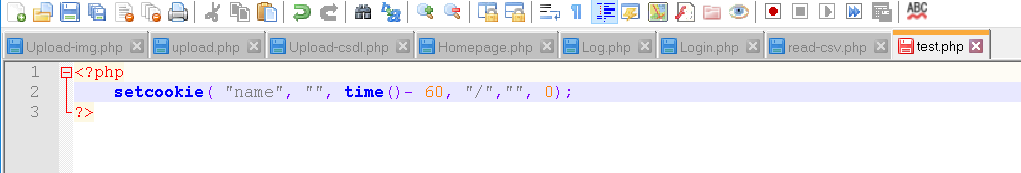
**Đoạn mã trên thiết lập lưu Cookie với tên name, giá trị là TranThanhPhong, hết hạn sau 10 phút (tức 600 giây)**

* **Lấy cookie:**



**Bạn dùng biến $\_COOKIE là một mảng để truy cập thông tin cookie do trình duyệt gửi lên.**

* **Xóa cookie:**



**Sử dụng chính hàm setcookie với thủ thuật đặt thời gian đã hết hạn.**

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

* **Session là một khái niệm phổ biến được dùng trong lập trình các website có kết nối với cơ sở dữ liệu database. Đặc biệt các chức năng như đăng nhập, đăng xuất người dùng sẽ khó có thể thực hiện được nếu không sử dụng session.**
* **Session đơn giản là 1 cách để chúng ta lưu lại dữ liệu của người dùng sử dụng website. Giá trị của session được lưu trong một tập tin trên máy chủ. Ví dụ khi bạn đăng nhập vào một trang web và đăng nhập với tài khoản đã đăng ký trước đó. Máy chủ sau khi xác thực được thông tin bạn cung cấp là đúng nó sẽ sinh ra một tập tin (hay chính là session của trình duyệt của bạn) chứa dữ liệu cần lưu trữ của người dùng.**
* **Bạn có thể tuỳ ý quyết định xem nên lưu trữ những thông tin nào vào Session. Nhưng thông thường chúng ta chỉ nên lưu những thông tin tạm thời trong session ví dụ như số lượng sản phẩm người dùng đã thêm vào giỏ nhưng chưa mua, hay những nhật xét đang được viết và lưu dưới dạng nháp nhưng chưa gửi đi. Những dữ liệu sử dụng lâu dài như nội dung nhận xét đã được gửi đi hay số sản phẩm đã được mua thì nên được thực hiện ở máy chủ chứa cơ sở dữ liệu.**
* **Lưu session:**

**$\_SESSION['name'] = $\_POST['username'];**

* **Lấy session:**

**$TENBIEN = $\_SESSION['name'];**

* **Xóa session:**
  + **unset($\_SESSION['name']); -- Để hủy một biến nào đó lưu trữ trong Session**
  + **session\_destroy(); -- Để hủy toàn bộ Session**

1. Bạn hãy so sánh Cookie và Session.

* **So sánh:**
  + **Cookie:**
    - **Cookie được lưu trữ trên trình duyệt của người dùng.**
    - **Dữ liệu cookie được lưu trữ ở phía máy khách.**
    - **Dữ liệu cookie dễ dàng sửa đổi khi chúng được lưu trữ ở phía khách hàng.**
    - **Dữ liệu cookie có sẵn trong trình duyệt của chúng ta đến khi hết hạn.**
  + **Session:**
    - **Session không được lưu trữ trong trình duyệt của người dùng.**
    - **Dữ liệu session được lưu trữ ở phía máy chủ.**
    - **Dữ liệu session không dễ dàng sửa đổi vì chúng được lưu trữ ở phía máy chủ.**
    - **Dữ liệu session có sẵn cho trình duyệt chạy. Sau khi đóng trình duyệt sẽ mất thông tin session.**

1. 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.
2. 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.
3. Bạn đọc trong để tìm hiểu và mô tả các cơ chế, các hàm/thủ tục để thực hiện việc upload.
4. Bạn hãy đọc, mô tả chức năng của hàm này và các tham số trong hàm (tham khảo thêm từ read-csv.php).
5. 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