# MÔ TẢ CHỨC NĂNG TRONG ĐỒ ÁN

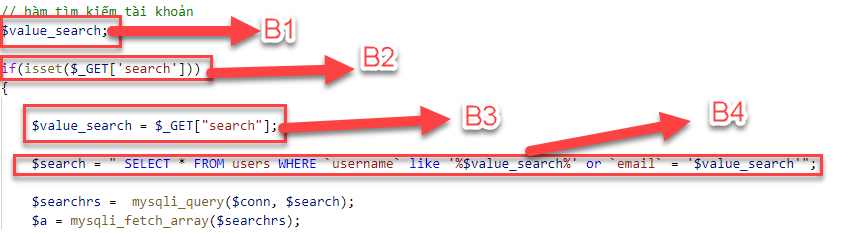
## I. Chức năng tìm kiếm

+ kết nối database



Hình 5 Hàm kết nối với database

Dùng hàm mysqli\_connect() để kết nối vs phpmyadmin



Hình 6 Các bước cho chức năng tìm kiếm

Các bước thực hiện sql tìm cho form list

Bước 1: Tạo 1 biến

Bước 2: Kiểm Tra search có tồn tại hay không nếu có thì vô thực hiện tiếp bước 3

Bước 3: gán cho biến lúc bước 1 tạo bằng GET\_[“search”];

Bước 4: Thực thi câu lệnh sql



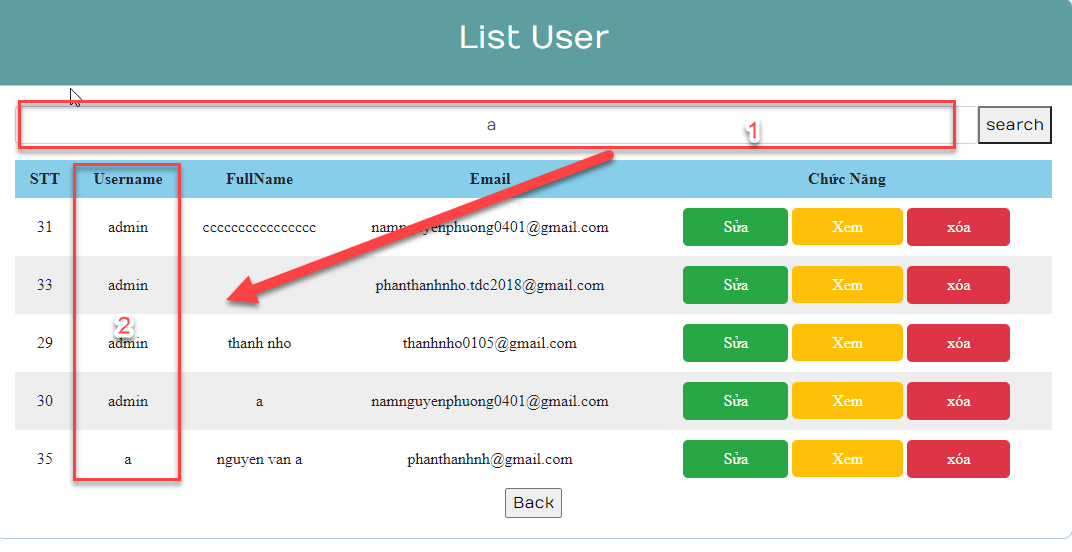
Hình 7 Code các bước cho chức năng tìm kiếm

Bước 5: Kiểm tra biến $a có tồn tại không

Bước 6: Cho vòng lập xuất ra những dữ liệu được tìm được

Bước 7: echo các ten id email của các tài khảo kiếm được ra màn hình list

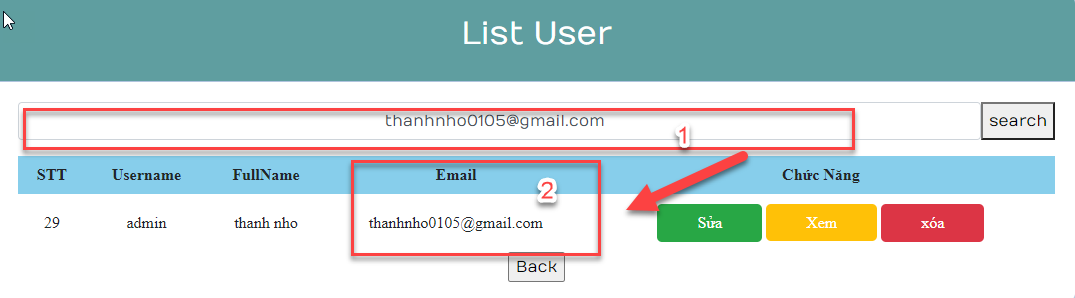
+ Ta có tìm kiếm theo username



Hình 8 Tìm kiếm theo Tên

+ Tìm kiếm theo email

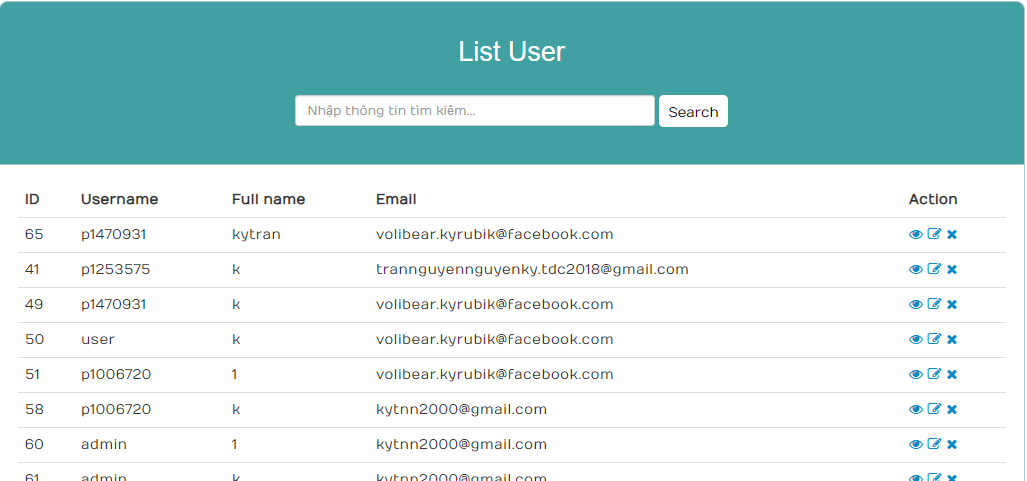
Tìm theo email khi biết rõ email người đó



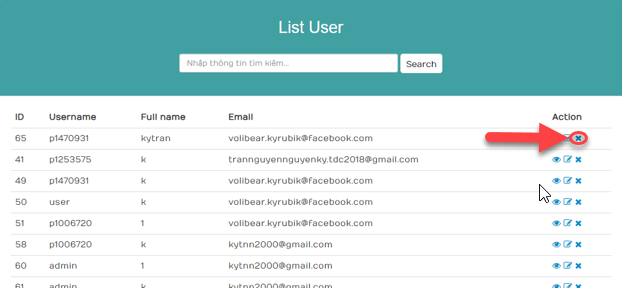
Hình 9 Tìm kiếm theo email

## II. Chức năng xóa

### Nút xóa cơ bản trong một file php:

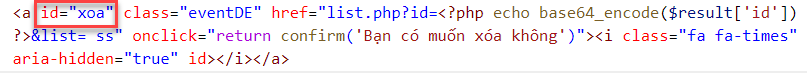


Hình 10 Danh sách ban đầu



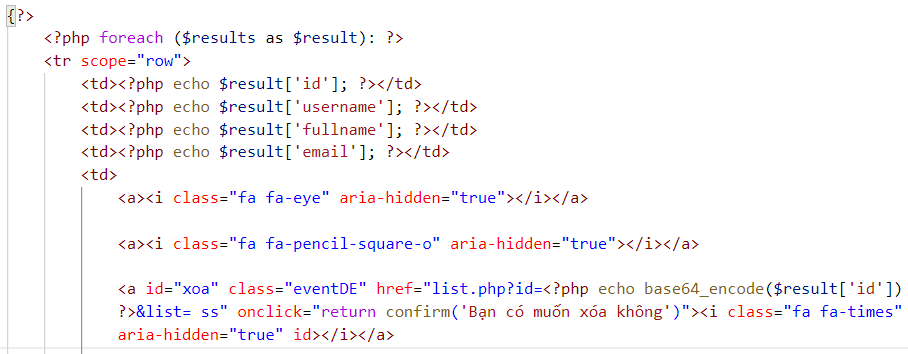
Hình 11 Xác định nút xóa

Bước 1: Xác định nút xóa nằm ở đâu trong source và đặt cho nó là id=”xoa”:



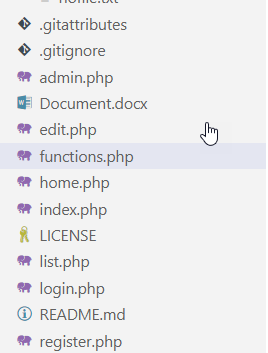
Hình 12 Đặt id cho nút xóa

Bước 2: Chạy vòng foreach để lấy id của giá trị mình cần xóa bằng cách đặt cho nút xóa một cái href như hình



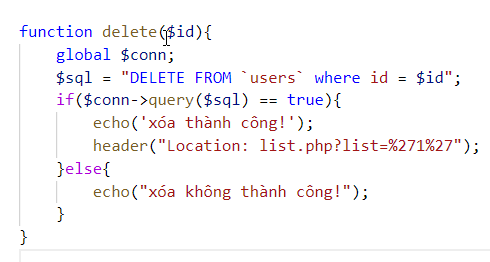
Hình 13 Chạy vòng foreach để lấy giá trị

Bước 3: vào functions.php để viết hàm delete()



Hình 14 Vào hàm functions.php

Bước 4: Viết hàm delete()

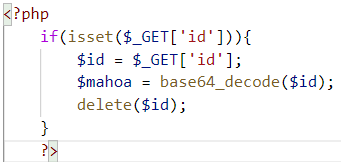


Hình 15 Hàm delete()

Gợi ý : hàm delete() được xóa theo id nếu id của giá trị lấy ra bằng

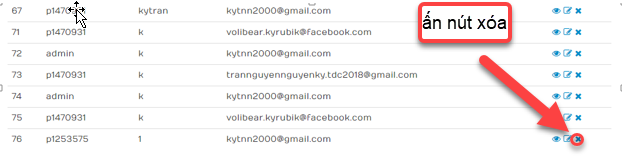
id mà mình xóa thì câu lệnh $sql được thi

Bước 5: Vào trang list.php gọi hàm xóa như hình sau đó nhấm Ctrl + S để lưu lại phần mình vừa làm



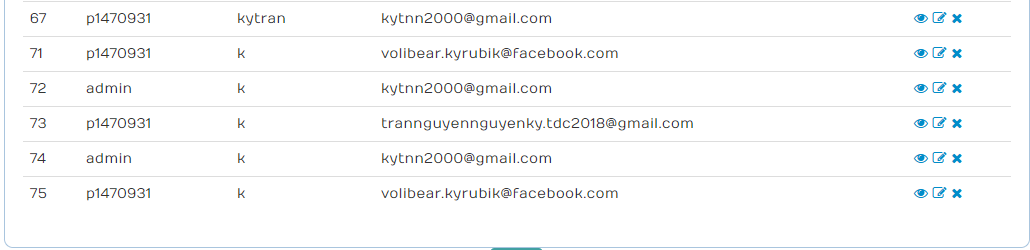
Hình 16 Get id và thực thi hàm xóa

Bước 6: Làm mới lại trang web(ấn F5) và thực thi nút xóa



Hình 17 Thực hiện nút xóa

Sau khi ấn nút xóa



Hình 18 Sau khi thực hiện nút xóa

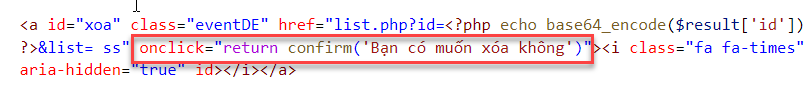
### Code nút xóa có xuất hiện hộp thoại trong một file php:



Hình 19 Hộp thoại nút xóa

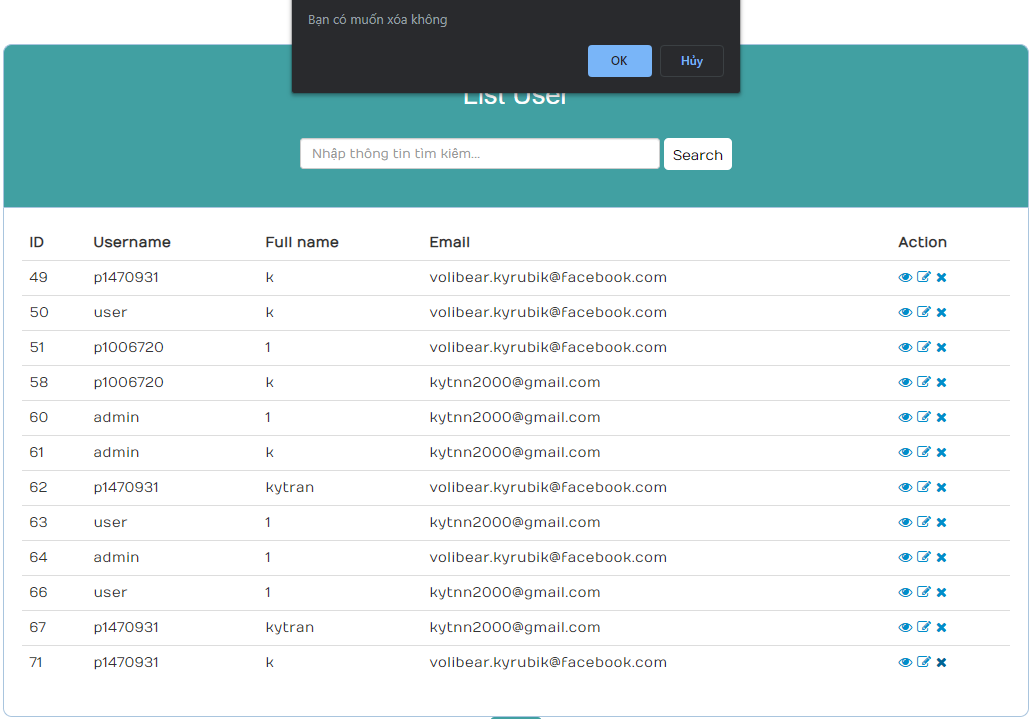
Bước 1: Sử dụng source code xóa ở trên

Thêm vào một sự kiện onclick tại nút xóa như hình



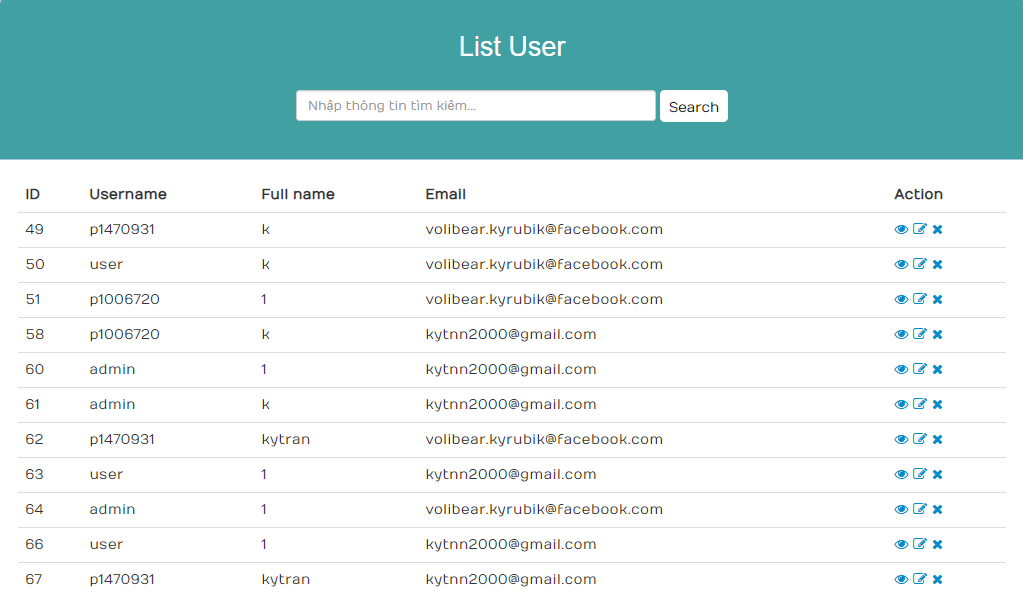
Hình 20 Đặt sự kiện onclick

Bước 2: Vào web ấn F5 và test



Hình 21 F5 và test

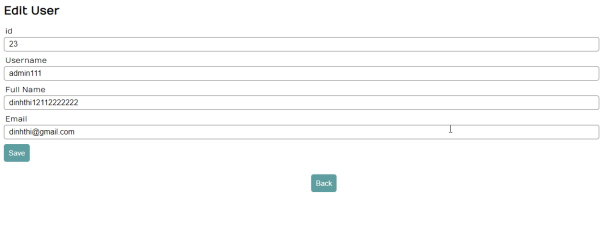
Sau khi ấn ok



Hình 22 Màn hình sau khi ấn ok

## III. Sửa thông tin theo id

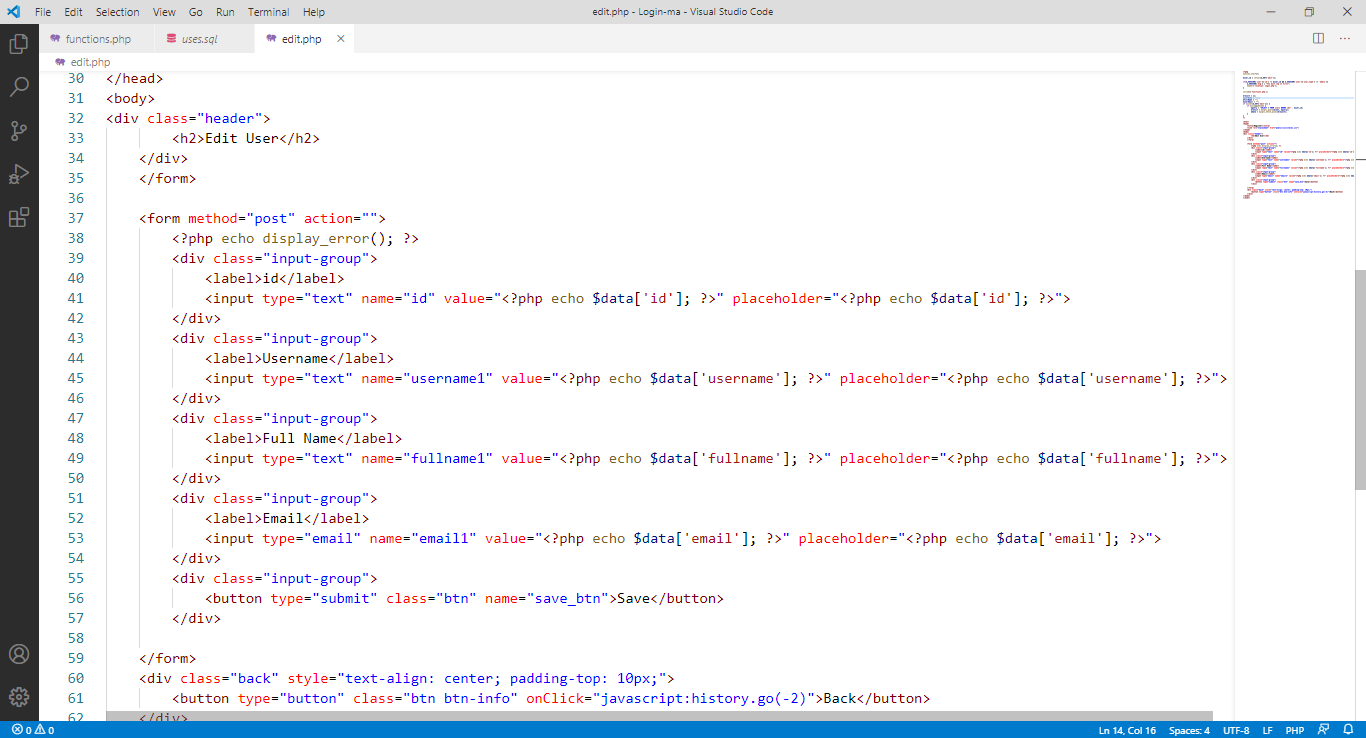
* Sau khi nhấn vào nút “**Sửa**” màn hình sẽ chuyển sang giao diện chỉnh sửa “**Edit user**”, Các form lưu lại cái giá trị ban đầu



Hình 23 Màn hình chỉnh sửa

**Code:**

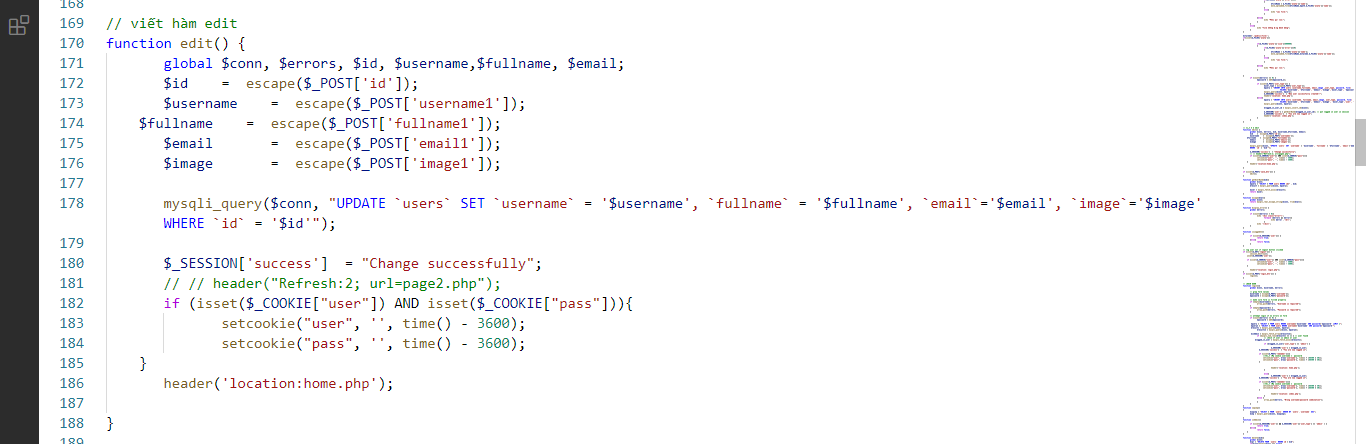
* Trước tiên chúng ta thiết kế giao diện trang edit.php.



Hình 24 Giao diện thiết kế chỉnh sửa trang “Edit.php”

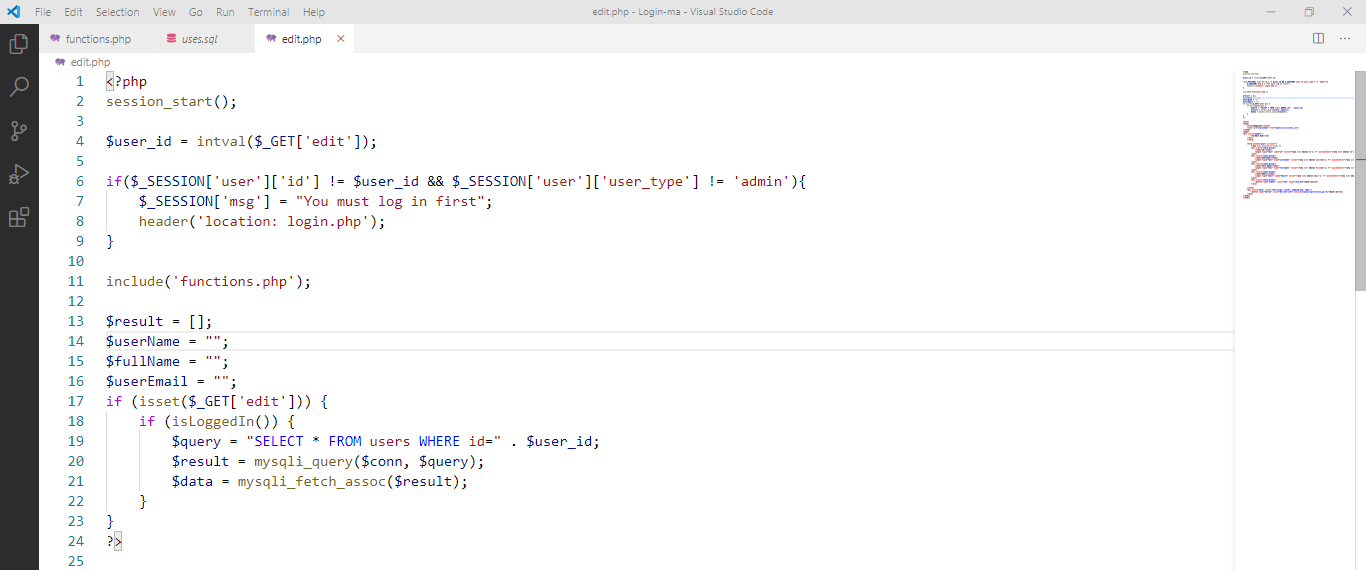
* Sau khi thiết kế giao diện xong ta viết hàm edit trong file **function.php** chỉnh sửa thông tin theo id.

+Tạo thêm



Hình 25 Đoạn code chỉnh sửa user theo id.

* File edit.php. Include class function.php để gọi hàm edit() và lưu vào database

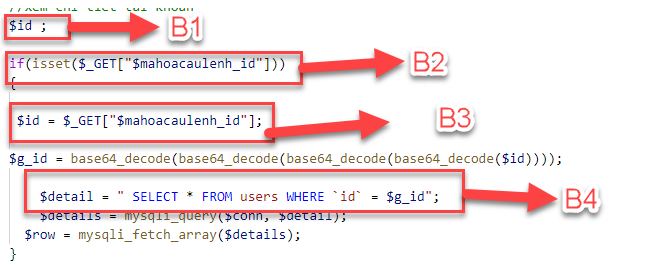


Hình 26 Lưu lại thông tin vào databasse

## IV. Tạo một phần để xem chi tiết người dùng

Khi admin muốn xem thông tin ai đó chi tiết hơn thì sẽ có chức năng xem chi tiết

+ code chức năng xem chi tiết



Hình 27 Code chức năng xem chi tiết user

Bước 1: Tạo 1 biến id

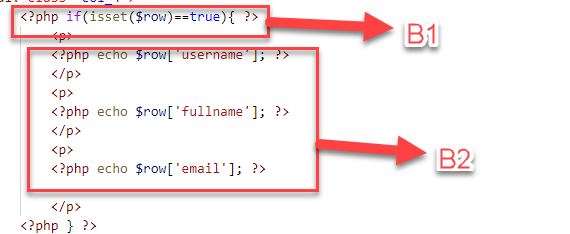
Bước 2: kiểm tra có Rỗng hay không

Nếu không thì thực hiện bước 3

Gấn Id đá được tạo = giá trị get đó

Bước 4: Thực thi câu lệnh sql

Sau khi thực thi thực hiện show thông tin ra màn hình



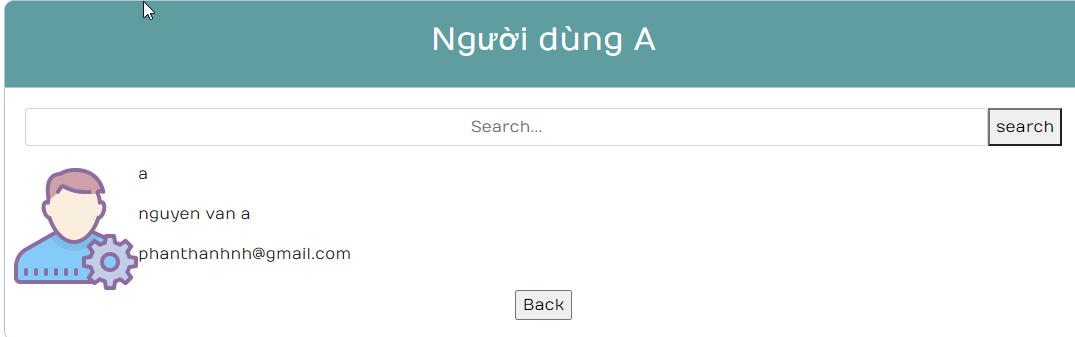
Hình 28 Code bước 1 và bước 2 show thông tin ra màn hình

Code show những thông tin tài khoản

Bước 5: Kiểm tra biến có giá trị không

Nếu có thì thực hiện bước 6

Bước 6: echo các thông tin của tài khoản ng dùng

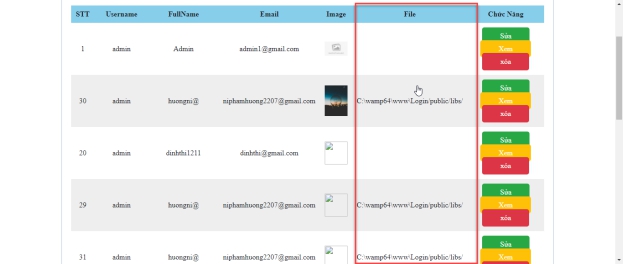


Hình 29 Hiện thị màn hình chi Tiết

Màn hình chi tiêt tài khoản khi admin cần xem

## V. Tạo Upload file

- Tạo form upload file lên



Hình 30 Giao diện chính sau khi upload file thành công

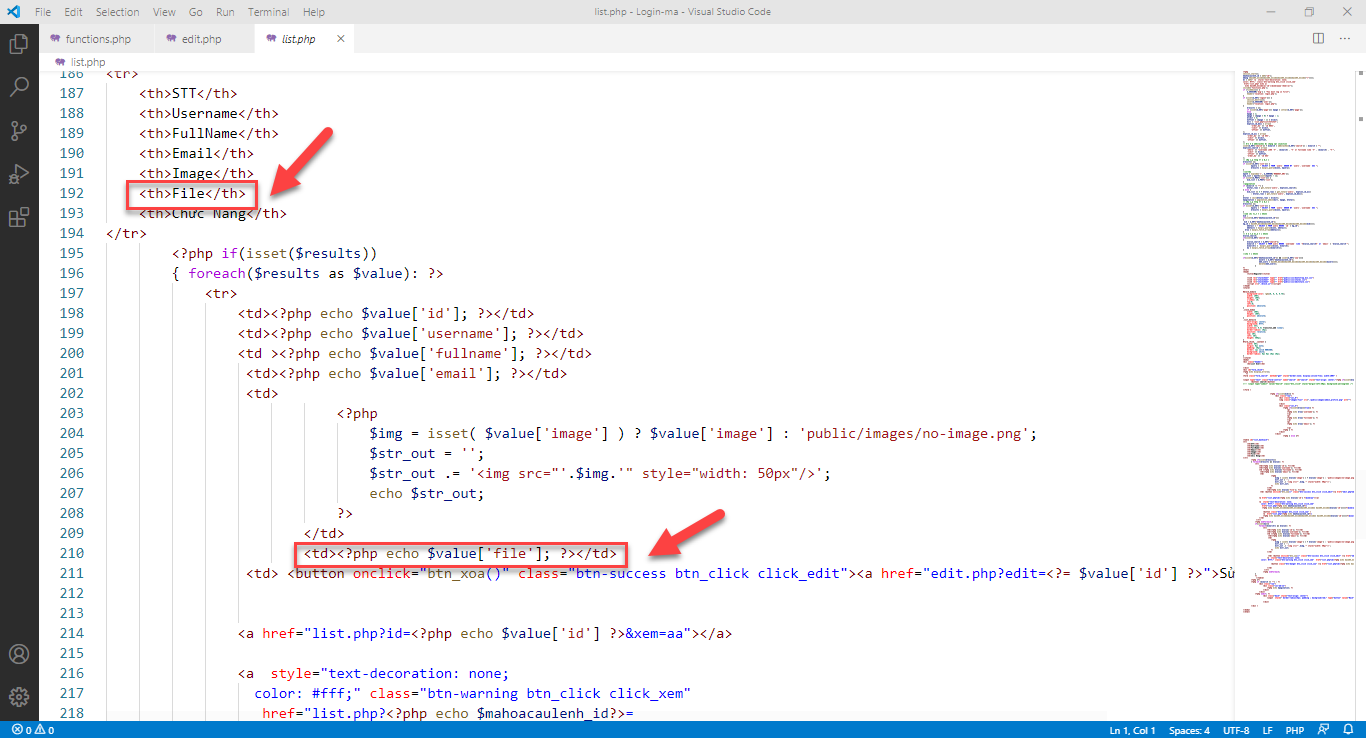
**Code:**

* Thiết kế giao diện trong file “**admin.php**”.



Hình 31 Đoạn code thiết kế giao diện chọn file

* Trong file list.php: Đưa tên file lên giao diện hiển thị user



Hình 32 Tạo thêm cột file và đưa tên file lên giao diện user

* File admin.php. Viết hàm lấy đường đẫn file. Trước tiên ta tạo cái forder trong “**public**” có tên là “**file**”. Rồi ta tạo cái biến để lấy đường dẫn.



Hình 33 Đoạn code lấy đường dẫn file

* Ta viết hàm kiểm tra xem form có lấy được file không



Hình 34 Kiểm tra trong form có file



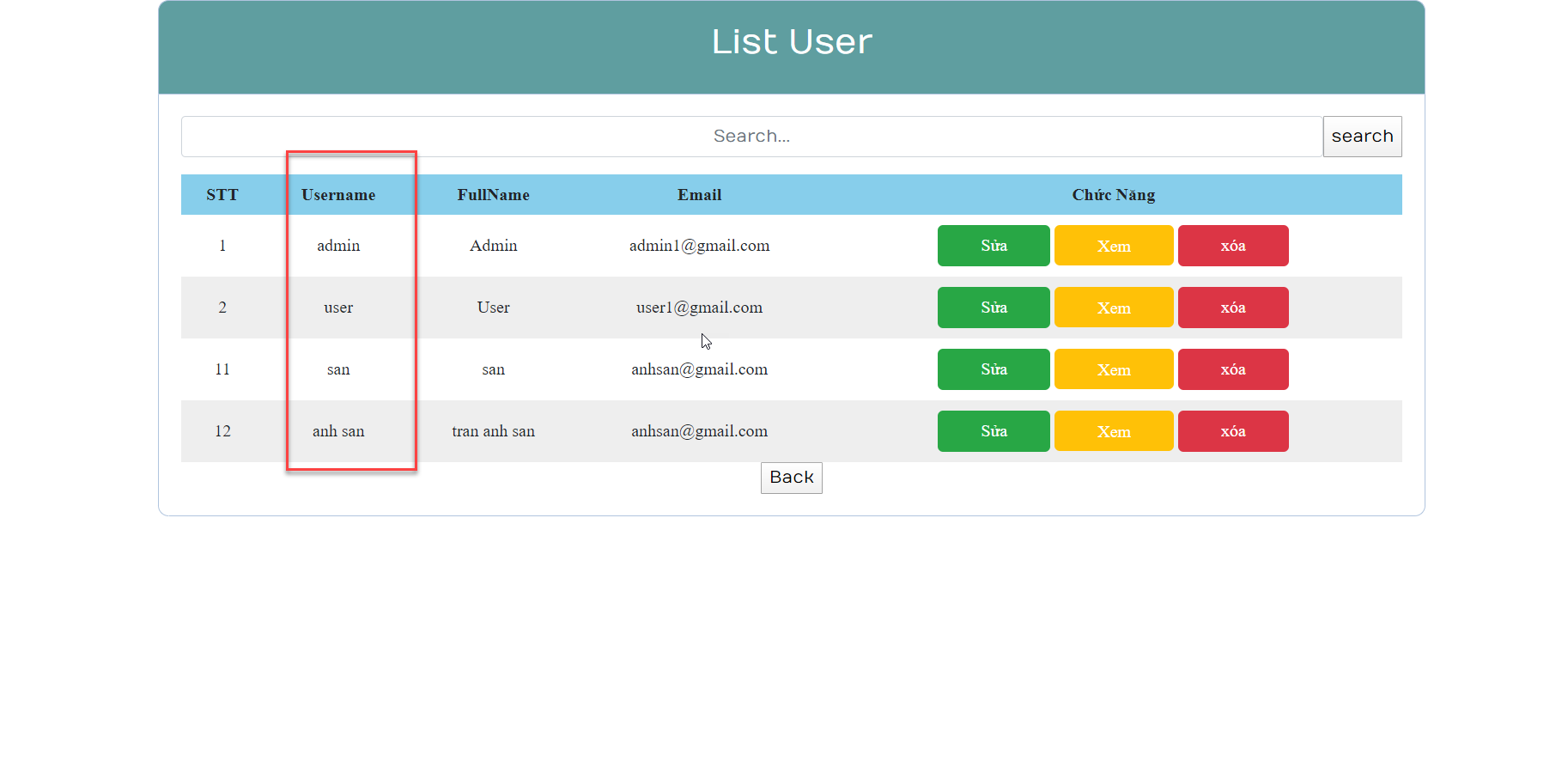
Hình 35 Kiểm tra form có file

* File fucntions.php. Viết hàm register để upload file lên database

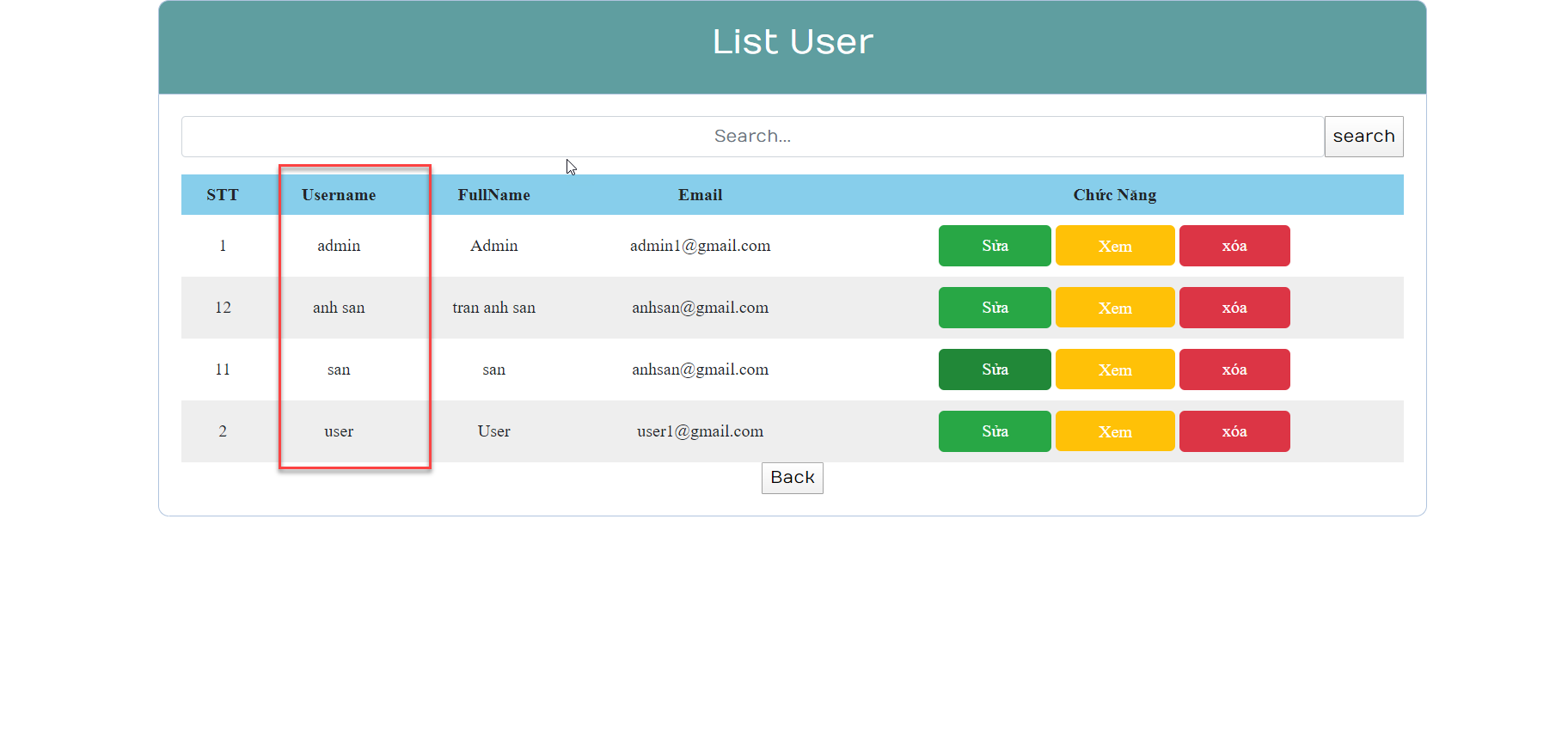


Hình 36 Đoạn code upload file lên database

## VI. Sắp xếp thông tin list user theo thứ tự từ A – Z



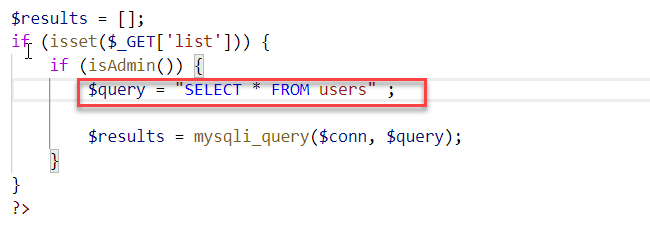
Hình 37 Đây là trang khi chưa sắp xếp



Hình 38 Đây là trang khi đã sắp xếp

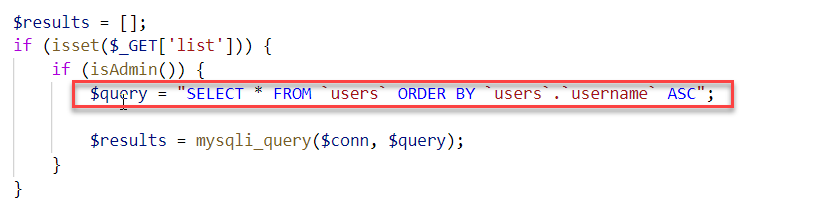
**Cách làm:**

+ Trong trang list.php thay đổi câu truy vấn cũ $query = "SELECT \* FROM users";



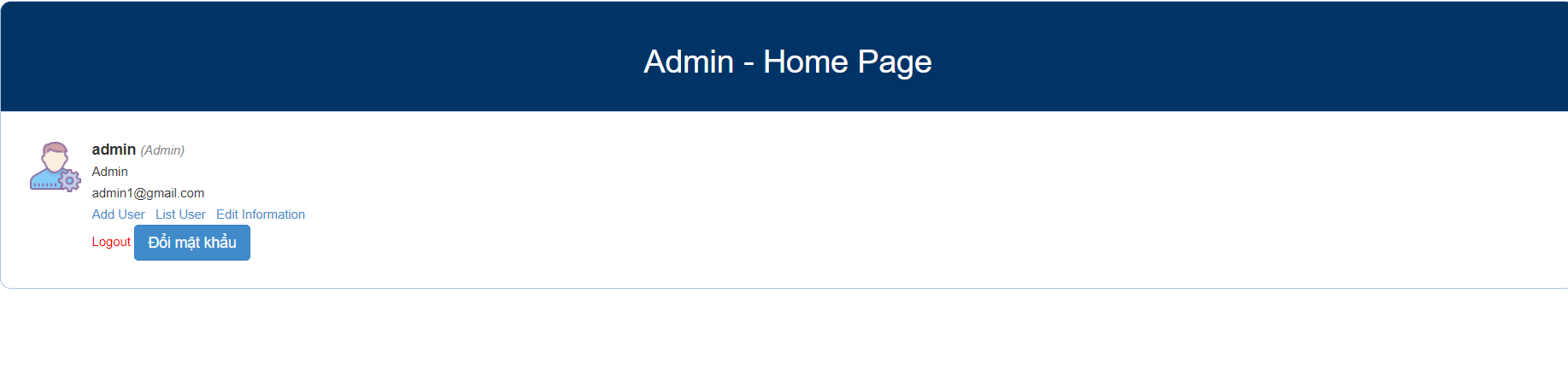
Hình 39 Xác định cấu truy vấn

bằng câu truy vấn mới $query = "SELECT \* FROM `users` ORDER BY `users`.`username` ASC";



Hình 40 Thay đổi câu truy vấn mới

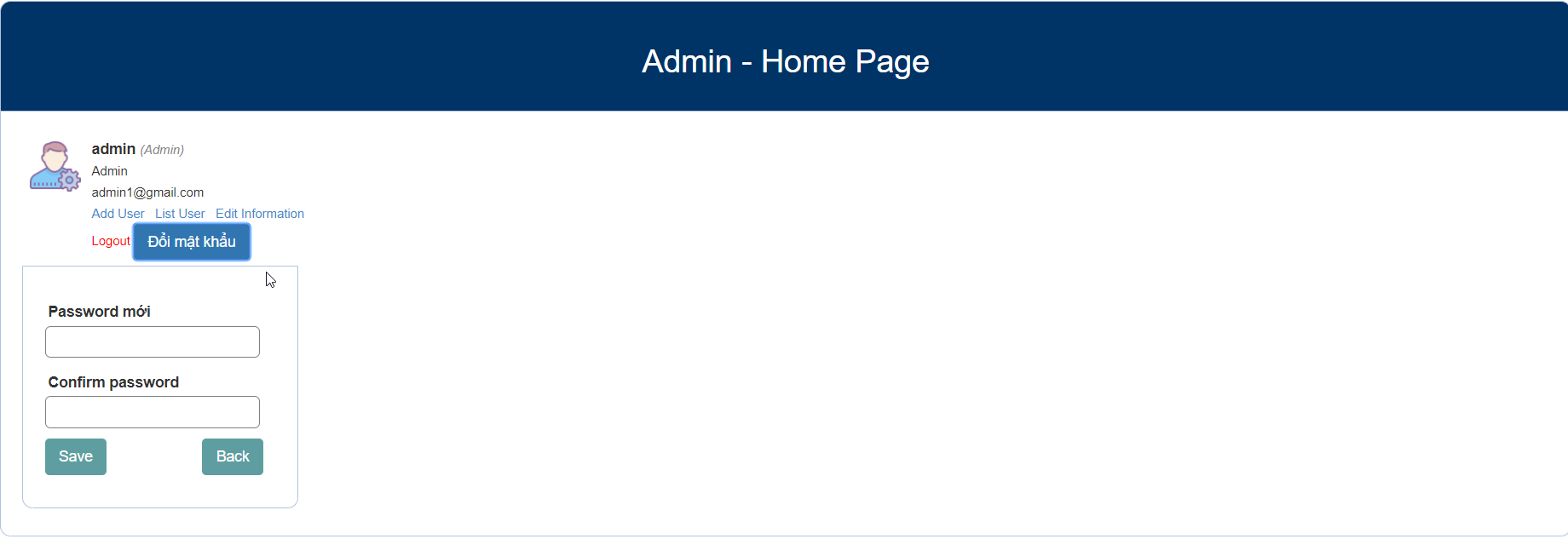
## VII. Thay đổi mật khẩu cho admin và user



Hình 41 Trang home

*5.Giao diện trang chính khi admin được nhập thành công*

- Sau khi người dùng hoặc quản trị viên đăng nhập thành công, nếu muốn đổi mật khẩu người dùng có thể click vào nút  tại trang chính sau đó sẽ xuất hiện Modal đổi mật khẩu.



Hình 42 Giao diện modal khi ấn đổi password

- Tiếp đó người dùng nhập mật khẩu mới và xác nhận lại mật khẩu.

- Cuối cùng nhấn nút  để lưu mật khẩu mới.

**Cách làm:**

+ Bước 1: Tạo Modal thay đổi password trong trang home.php và index.php.



Hình 43 Code html modal đổi password

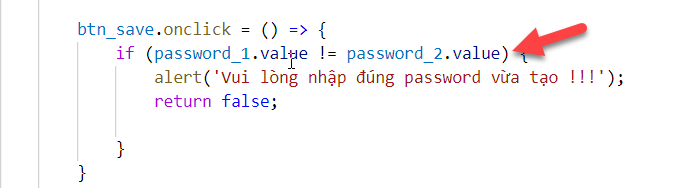
+ Bước 2: Viết code javascript kiểm tra password nhập lại có trùng với password mới vừa tạo nếu không trùng sẽ thông báo cho người dùng.

- Lấy trường password vừa nhập bằng javascript .



Hình 44 Lấy trường password

- Kiểm tra password có nhập trùng với nhau hay không. Nếu không trung thông báo cho người dùng biết.



Hình 45 Code javascript kiểm tra password

+ Bước 3: Viết code php update password mới cho user và admin.

- Viết code php lấy giá trị password mới.



Hình 46 Code php lấy giá trị password mới

- Tiếp theo mã hóa password theo chuẩn MD5 trước khi lưu password vào database.



Hình 47 Code mã hóa password

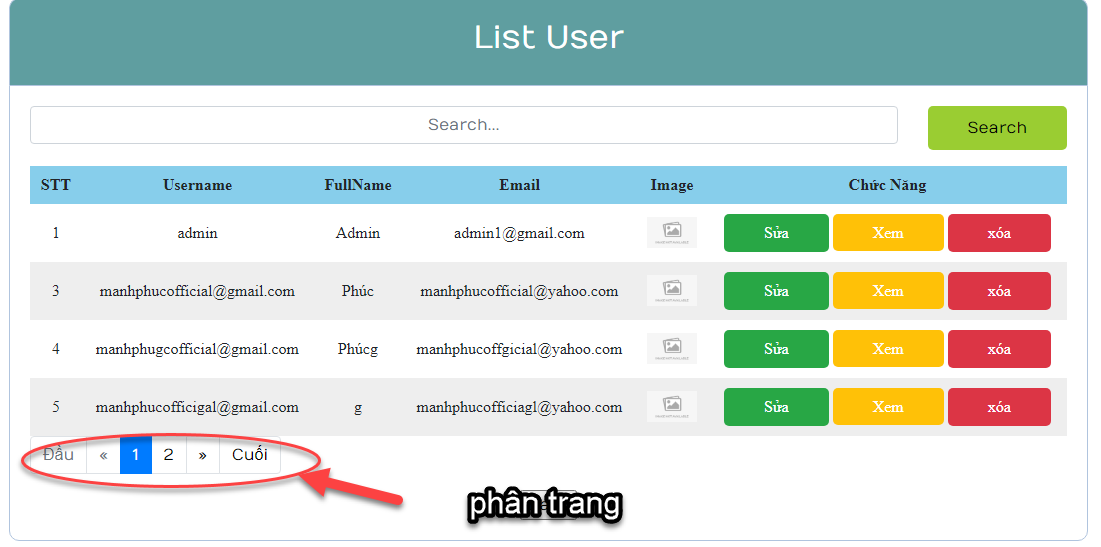
- Cuối cùng là viết câu lệnh update password đã mã hóa vào database.



Hình 48 Câu lệnh update password dã mã hóa

## VIII. Phân Trang

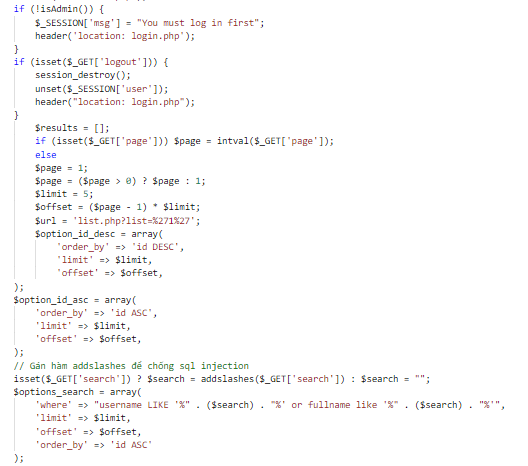
Màn hình hiển thị:



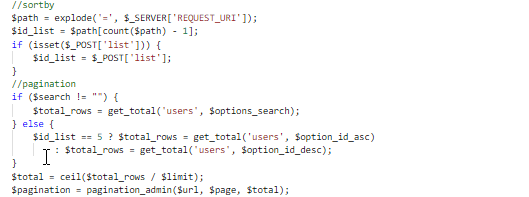
Hình 49 Màn hình phân trang



Hình 50 Code của chức năng và sử lý trong functions.php



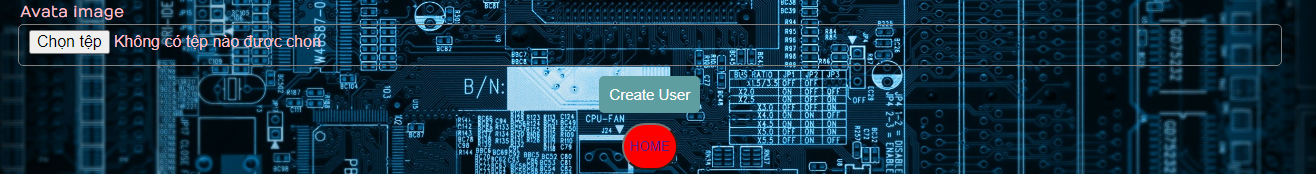
Hình 51 Code sử lý trong list.php



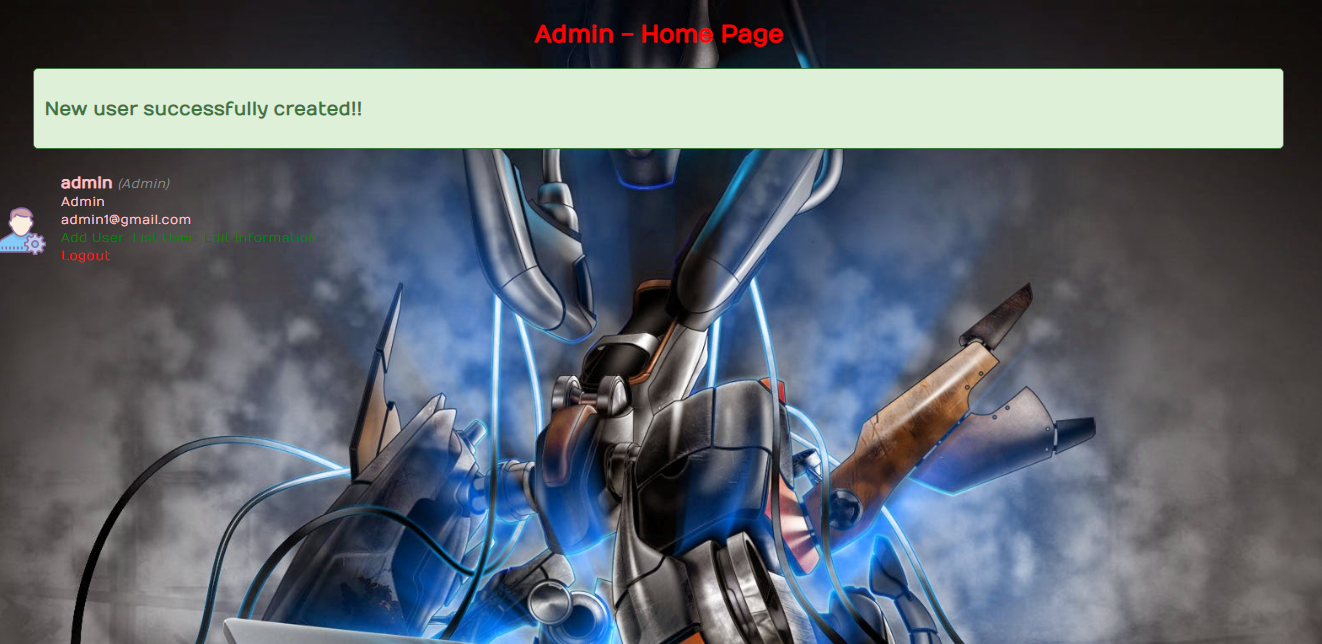
Hình 52 Code sử lý trong list.php

## IX. Upload hình ảnh

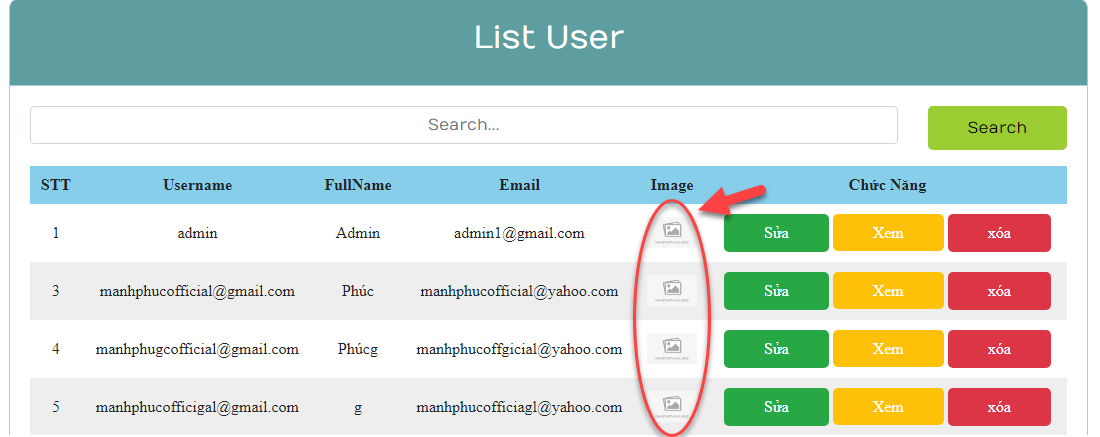
Ở giao diện màn hình:



Hình 53 Màn hình upload ảnh



Hình 54 Màn hình thêm thông tin thành công



Hình 55 Màn hình thêm ảnh thành công

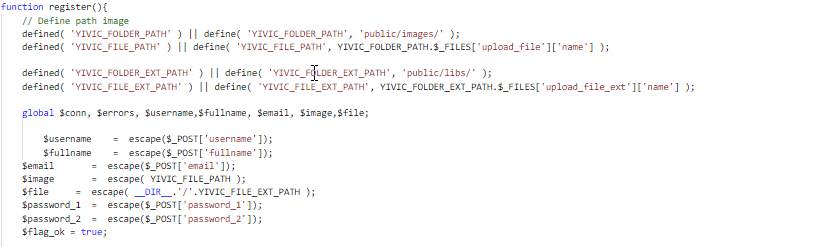
Chú ý nếu ảnh đã vướt quá kích thước hoặc file đó k chưa ảnh thì nó sẽ hiện thị như sau:



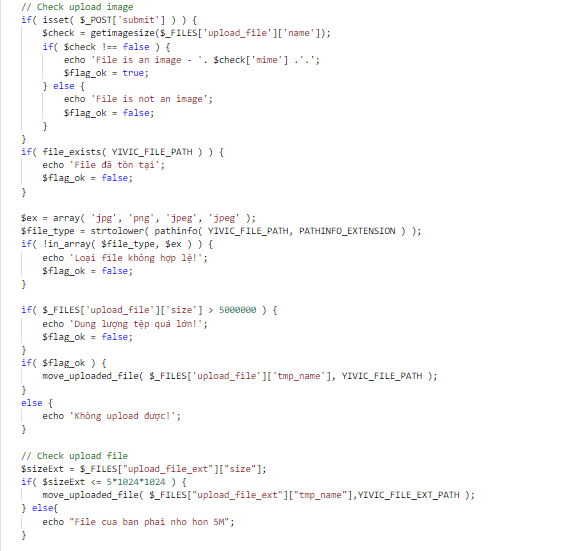
Hình 56 Mô tả hình ảnh



Hình 57 Code hiện thị giao diện upload images



Hình 58 Code sử lý upload images



Hình 59 Chức năng sử lý hình ảnh trang đăng ký user

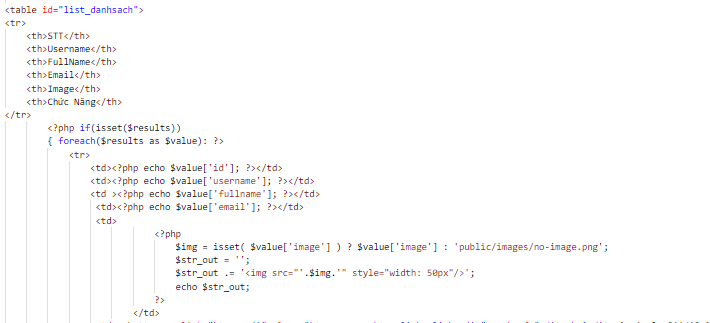
B1: Kiểm tra nút file

B2: Kiếm tra đuôi hình

B3: Kiếm tra k có lỗi cho chạy

B4: Dùng hàm move upload file

B5: Gán tên file



Hình 60 Code Hiện thị user



Hình 61 Code Hiện thị user



Hình 62 Code Hiện thị user

# CODE CSS THEO LESS/SASS

* 1. Less sass

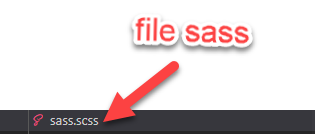
Trong quá trình code giao diện đã dùng sass để thiết kế giao diện

Sass rất thông dụng hiện nay, nó giúp cho chúng ta rất nhiều như những quản lý code css dễ dàng và rành mạch hơn , và rất dễ bảo trì hơn sau khi chúng ta viết sass thì có thẻ biên dịch bằng ngôn ngữ css… hiện nay rất có nhiều ứng dụng hộ trở biên dịch qua lại

Ví dụ như: ứng dụng koala

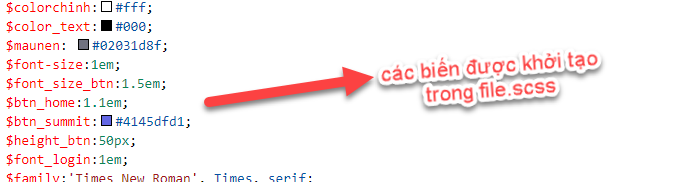
## 1.các bước tạo sass và buil sass thành file css

+ đầu tiền chúng ta tạo file name.scss



(tạo file.scss)

Khi chúng ta tạo được file scss thì chúng ta tiến hành code theo sass tuy nhiên sass với css cũng không khác quá nhiều chỉ khác cách lồng vào nhau



(tạo các biến cho các giá trị cần dùng)

Các biến được tạo trong sass nhằm mục đích dễ bảo trì và dễ quãn lý

Khi chúng ta cần gọi tên biến

Ví dụ:



(cách sử dụng biến đã đặt lấy giá trị)

Đây là chỉnh font-chữ và lấy ra biến được lưu font chữ được chỉnh sữa để thông dụng cho nhiều form

+ cấu trúc viết scss

Cấu trúc viết cho form login khi bằng viết bằng sass khi chưa được biên dịch



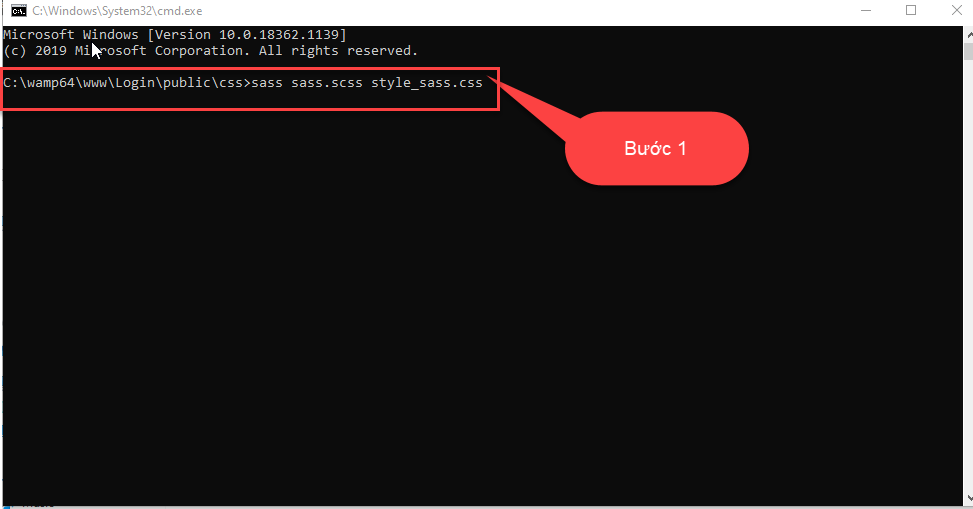
(Cấu trúc viết sass lồng nhau từ lớp cha xuống lớp con 1)



(Cấu trúc viết sass lồng nhau từ lớp cha xuống lớp con 2)

Chúng ta sẽ lồng nhau từ lớp cha theo rành mạch không được viết tùm lum sass giúp chúng ta trình bày code 1 cách rành mạch để khi bảo trị sẽ dễ dàng hơn rất nhiều.

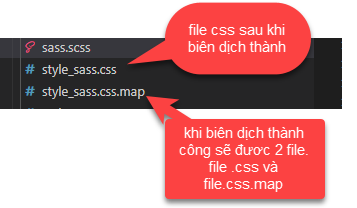
**+** cách biên dịch sass sang css



(câu lệnh build file.scss)

Bước 1: nhập câu lệnh

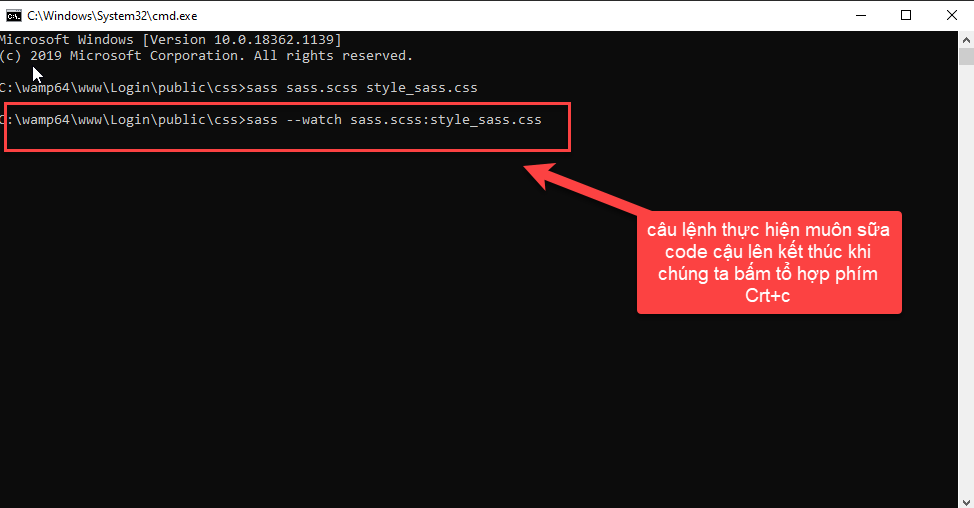
Sass tên file.scss tên khi đổi ra.css tên tự đặt



(sau khi build thành công)

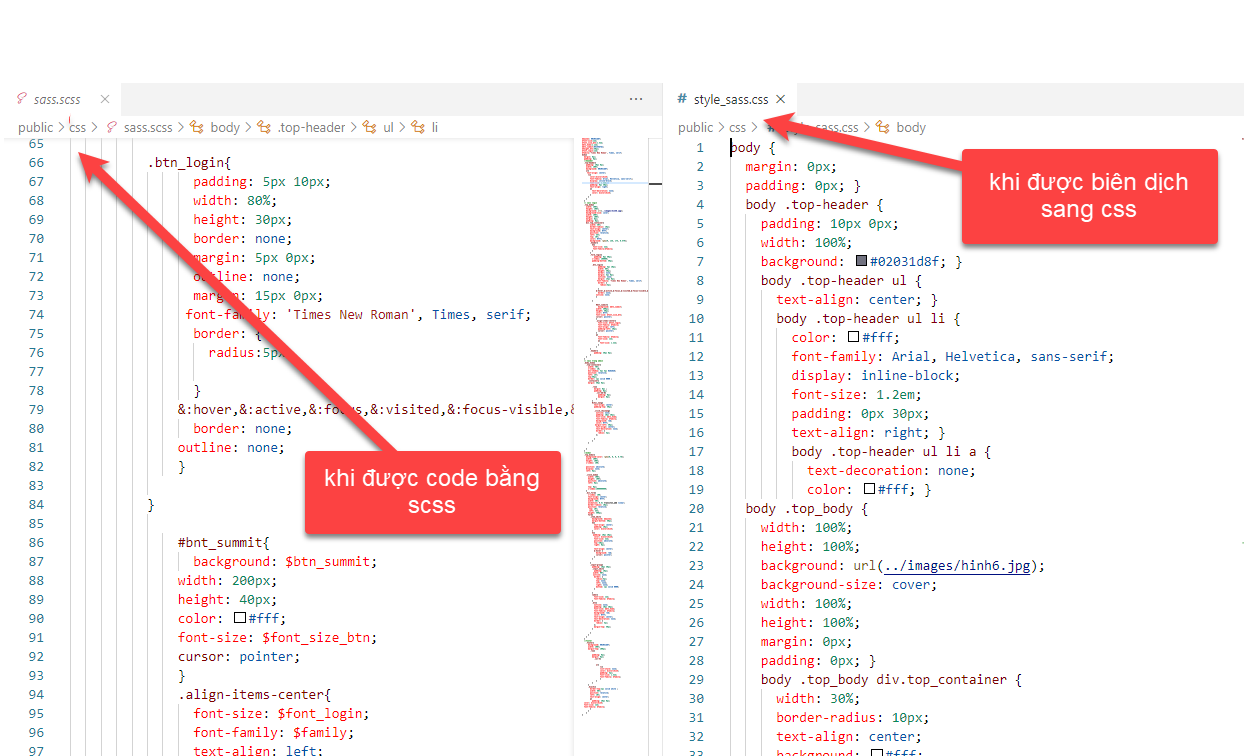
Sau khi chúng ta chạy thành công câu lênh sẽ sinh ra 2 file css và flie.css.map

+ sửa code của sass và tự thay đổi trong css



(câu lệnh thực thi sửa file.scss)

Kết quả khi được biên dịch



(kết qua khi biên dịch thành file.css)