ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

**TRƯỜNG ĐẠI HỌC KHOA HỌC – TỰ NHIÊN**

**---o0o---**

**ĐỒ ÁN LẬP TRÌNH SOCKET**

Môn học: Mạng máy tinh

Học kỳ II (2019 – 2020)

**XÂY DỰNG WEB SERVER ĐƠN GIẢN**

Sinh viên thực hiện: 18120262 - Trần Nhật Việt

18120508 - Đinh Phong Phú

Lớp: 18TN

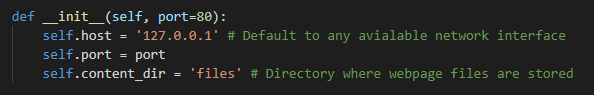
Giảng viên hướng dẫn: Chung Thùy Linh

**Thành phố Hồ Chí Minh, tháng 7 năm 2020**

1. **Cách thực hiện ứng dụng:**

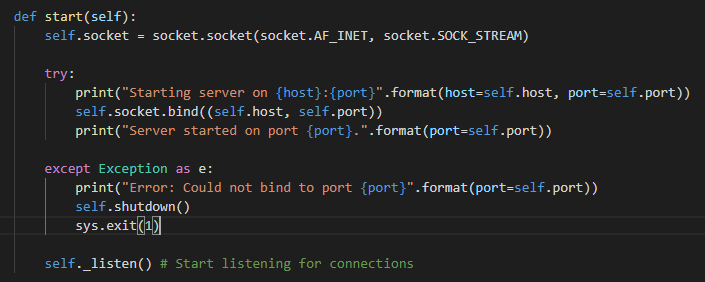
* Sử dụng ngôn ngữ lập trinh python với thư viện chinh là thư viện socket và các thư viện phụ hỗ trợ.
* Class WebServer thể hiện một đối tượng Web Server với các phương thức là:

1. **Phương thức khởi tạo:**



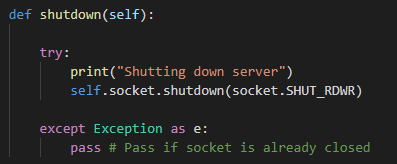
* Khởi tạo một đối tượng Web Server với các thuộc tính cơ bản.
* Host là 127.0.0.1 và port là 80, thư mục chứa các file cần thiết là files.

1. **Phương thức start:**



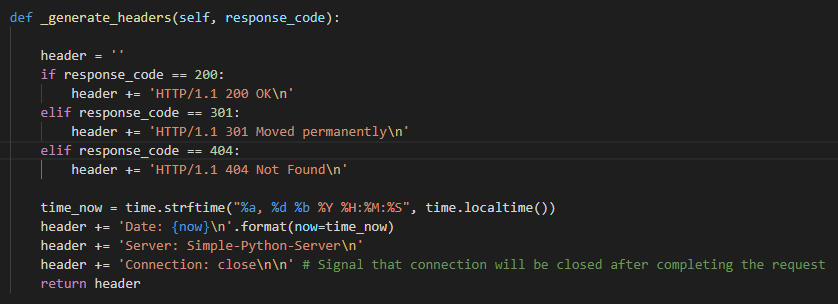
* Phương thức tạo một socket với host và port đã được khởi tạo ở phương thức khởi tạo.
* In thông báo ra màn hình nếu tạo thanh công, nếu không thanh công thì in thông báo và shutdown.

1. **Phương thức shutdown:**



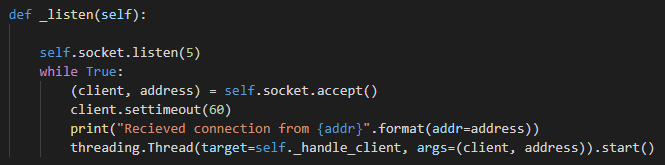
* Phương thức dùng để đóng socket và tắt Server.

1. **Phương thức \_generate\_headers:**



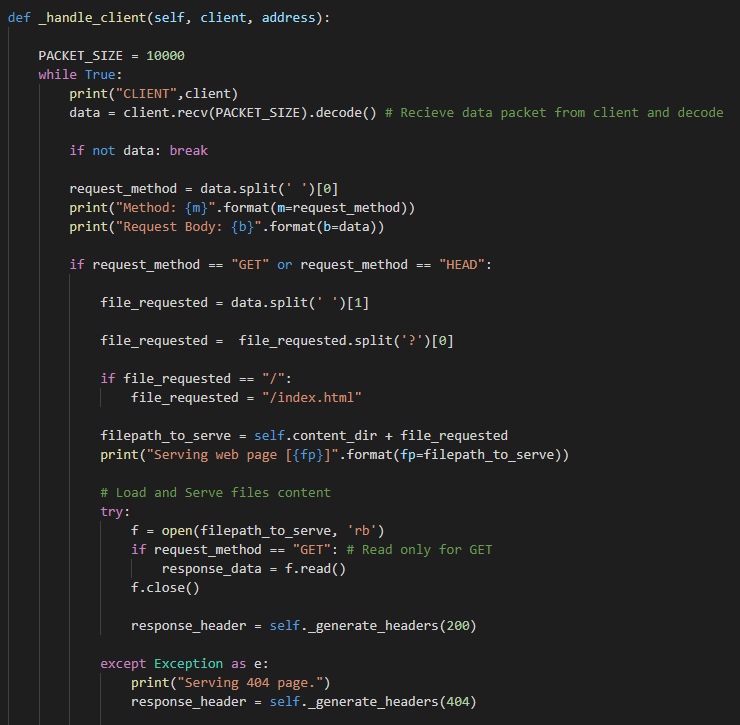
* Phương thức tạo ra header của gói tin gửi từ server đến client với tham số truyền vào là mã phản hồi.
  + Nếu mã là 200 nghĩa là yêu cầu từ client đã được server xử lý thành công.
  + Nếu mã là 301 nghĩa là yêu cầu chuyển trang từ client đã được xử lý.
  + Nếu mã là 404 nghĩa là yêu cầu từ client, server không xử lý được.

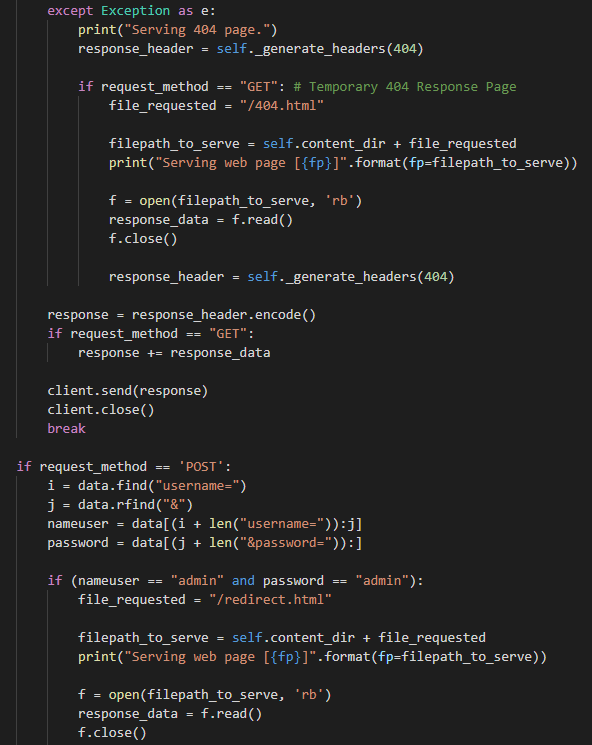
1. **Phương thức \_listen:**

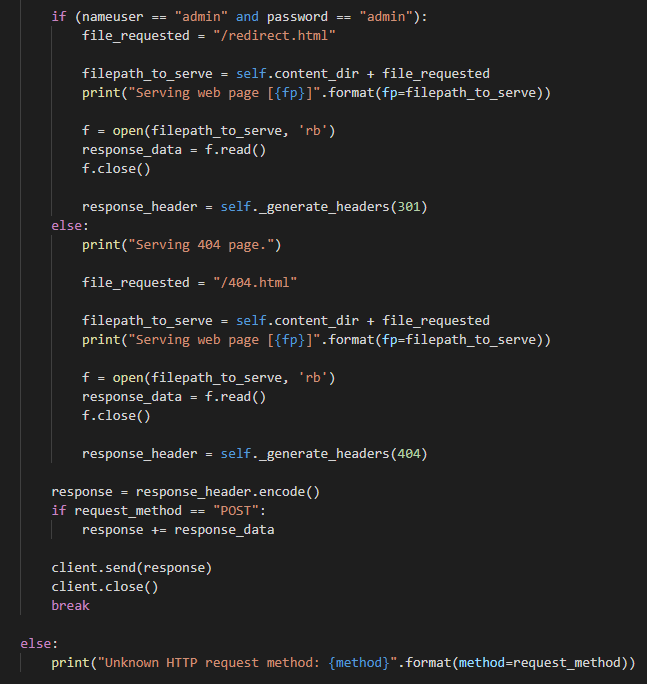


* Phương thức dùng để lắng nghe các kết nối từ client về server.

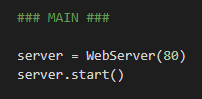
1. **Phương thức \_handle\_client:**







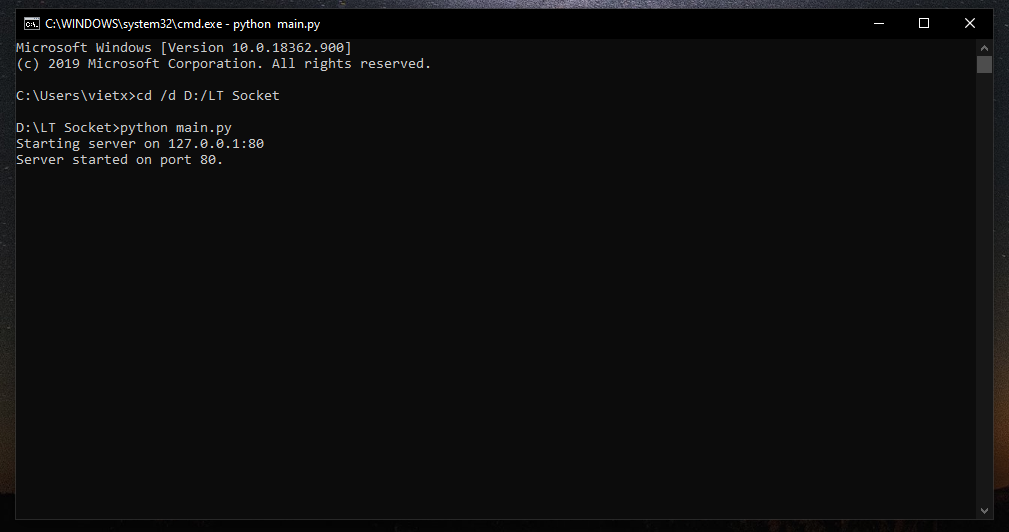
* Phương thức tạo một vòng lặp để nhận các request của client.
* Sau khi nhận được request từ client thì server sẽ xử lý gói tin để xác định phương thức là GET hay POST để xử lý.
* Nếu phương thức là GET nghĩa là Server sẽ phải trả về cho client nội dung file mà client yêu cầu. Nếu đường dẫn client yêu cầu là “/” thì sẽ mặc định chuyển client đến đường dẫn là “/index.html”. Nếu đường dẫn yêu cầu là “/files.html” thì Server sẽ đọc file files.html và hiển thị trang web đó lên cho client. Nếu không đọc được file thì sẽ đọc và hiển thị trang web 404.html.
* Nếu phương thức là POST nghĩa là client đã nhập username và password và muốn login, client gửi POST request lên Server và Server xử lý chuỗi trong request để tách username và password. Nếu username và password thỏa mãn thì sẽ đọc file redirect.html và redirect client đến trang info.html sau khi Server đọc xong file info.html. Nếu username và password không thỏa mãn thì sẽ đọc và hiển thị trang web 404.html.
* Hàm main:



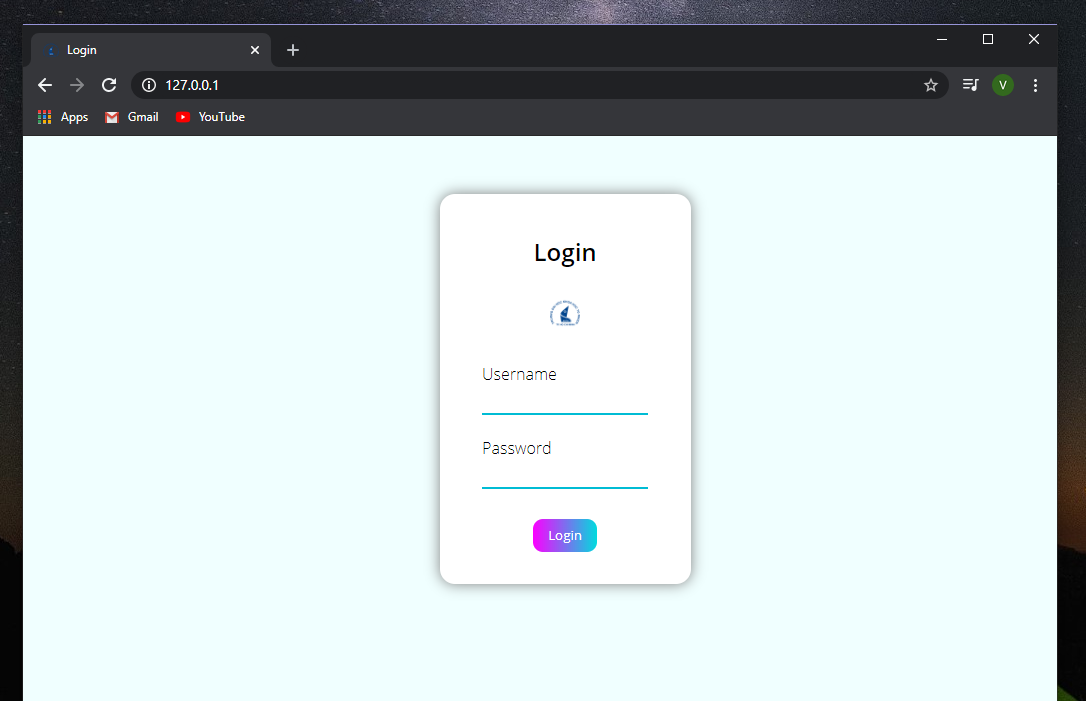
* Khởi tạo Server ở port 80.
* Dùng phương thức start để khởi động Web Server.

1. **Cách test ứng dụng:**

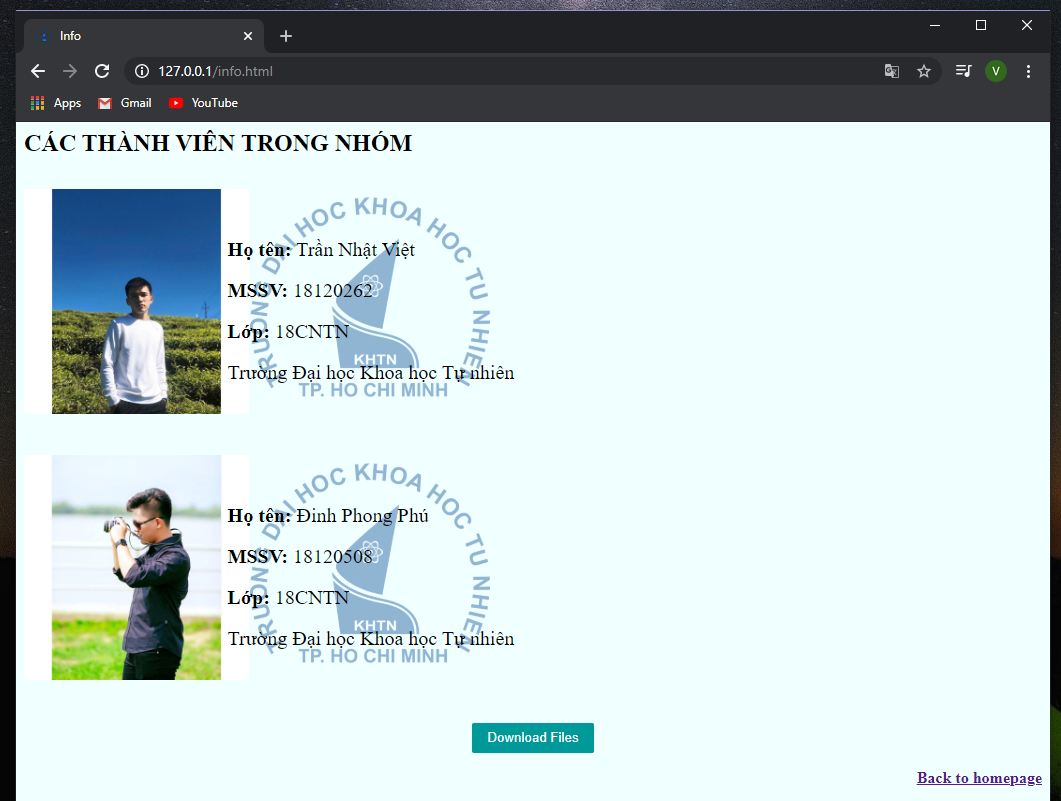
* Ở cmd, chuyển đến thư mục chứa file main.py, chạy file main.py. Sau khi chạy xong thì màn hình cmd sẽ hiện thị đường link để truy cập như hình bên dưới.



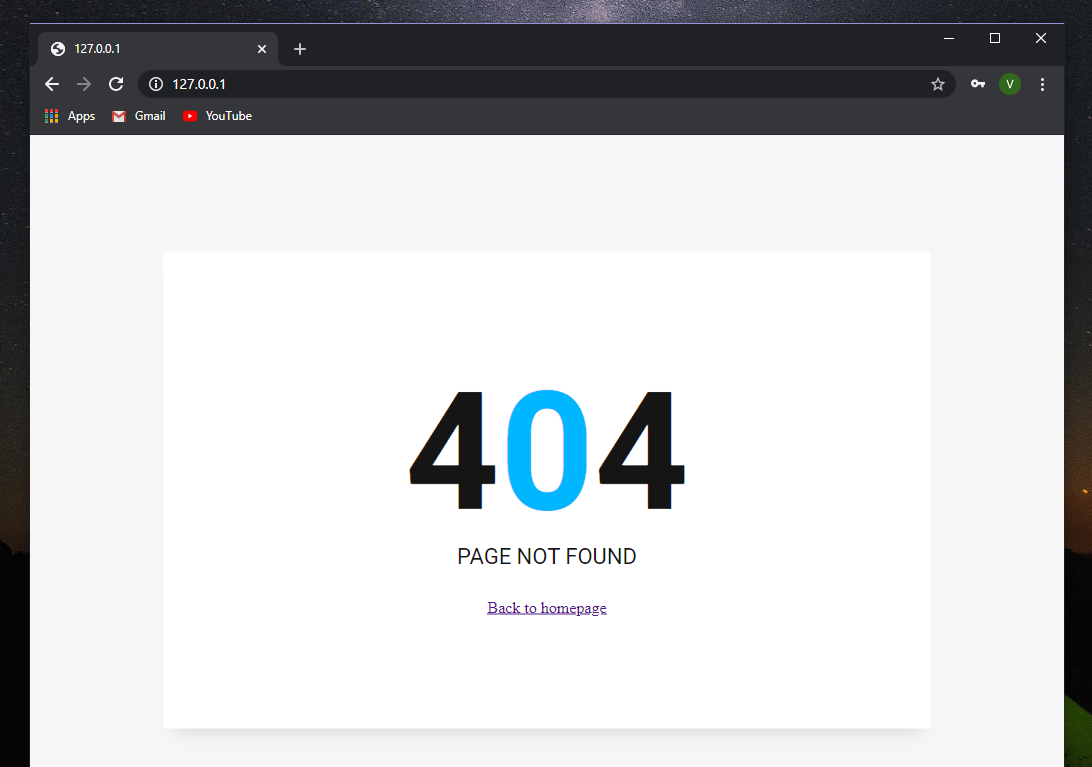
* Mở trinh duyệt web và nhập đường link vào thanh tìm kiếm, trinh duyệt sẽ hiển thị trang index.html như hình bên dưới.



* Nhập username và password vào ô tương ứng, nếu nhập đúng trình duyệt sẽ hiện thị trang info.html như hình bên dưới.



* Nếu nhập username và password sai trình duyệt sẽ hiện thị trang 404.html như hình bên dưới. Nếu muốn trở về trang index.html thì nhấn vào dòng back to homepage.



* Nếu nhập vào thanh tìm kiếm đường link http://127.0.0.1/files.html hoặc nhấn vào ô Download ở trang index.html thì trình duyệt sẽ hiển thị trang files.html như hình bên dưới. Nhấn vào file muốn tải để tải về máy tính.

