Pawan Chandra
885887810

Python: Socket Programming

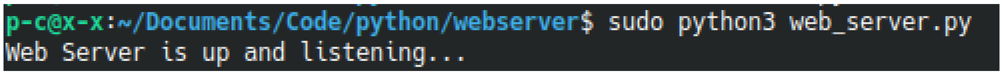
Assignment 1: Web Server

In this assignment, you will develop a simple Web server in Python that is capable of processing only one request. Specifically, your Web server will

1. Create a connection socket when contacted by a client (browser);
2. Receive the HTTP request from this connection;
3. Parse the request to determine the specific file being requested;
4. Get the requested file from the server's file system;
5. Create an HTTP response message consisting of the requested file preceded by header lines; and
6. Send the response over the TCP connection to the requesting browser. If a browser requests a file that is not present in your server, your server should return a "404 Not Found" error message.

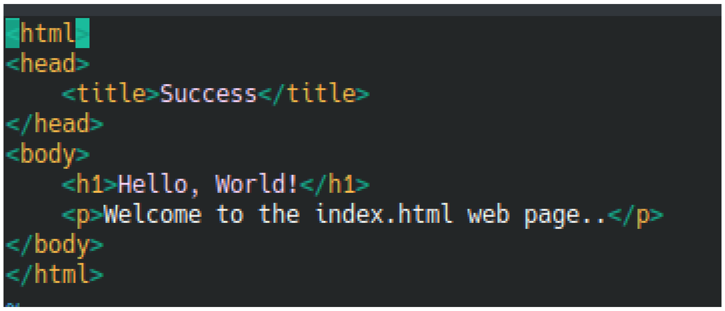
In the Companion Website, we provide the skeleton code for your server. Your job is to complete the code, run your server, and then test your server by sending requests from browsers running on different hosts. If you run your server on a host that already has a Web server running on it, then you should use a different port than port 80 for your Web server.

Part 1:
Here you can see me running the code and the first line after showing "Web Serve is up and listening…"



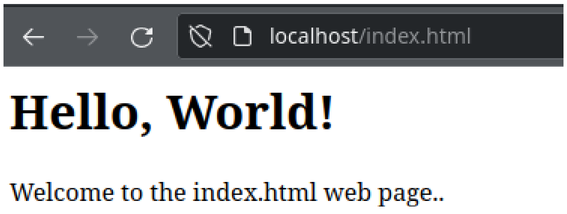
```
p-c@x-x:~/Documents/Code/python/webserver$ sudo python3 web_server.py
Web Server is up and listening...
```

Part 2:
I then create a index.html file using vim in the same directory and go to that file in firefox.
Image of index.html code



```
<html>
<head>
    <title>Success</title>
</head>
<body>
    <h1>Hello, World!</h1>
    <p>Welcome to the index.html web page..</p>
</body>
</html>
```

Image of firefox when accessing index.html

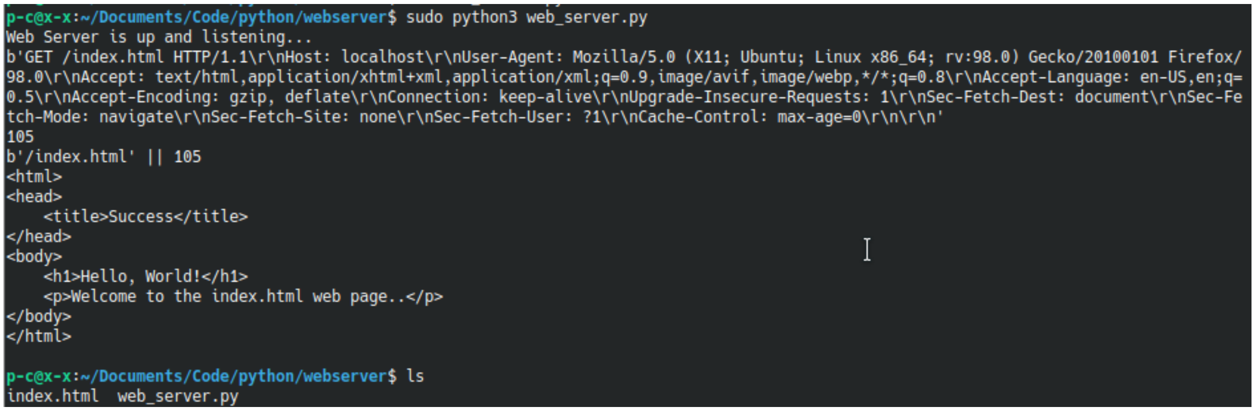


Hello, World!

Welcome to the index.html web page..

Part 3:

Image of what it looks like on the terminal/server when i access index.html from firefox

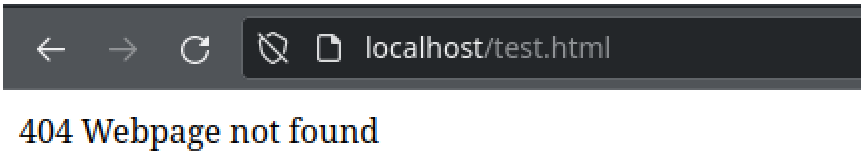


```
p-c@x-x:~/Documents/Code/python/webserver$ sudo python3 web_server.py
Web Server is up and listening...
b'GET /index.html HTTP/1.1\r\nHost: localhost\r\nUser-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:98.0) Gecko/20100101 Firefox/
98.0\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8\r\nAccept-Language: en-US,en;q=
0.5\r\nAccept-Encoding: gzip, deflate\r\nConnection: keep-alive\r\nUpgrade-Insecure-Requests: 1\r\nSec-Fetch-Dest: document\r\nSec-Fe
tch-Mode: navigate\r\nSec-Fetch-Site: none\r\nSec-Fetch-User: ?1\r\nCache-Control: max-age=0\r\n\r\n'
105
b'/index.html' || 105
<html>
<head>
    <title>Success</title>
</head>
<body>
    <h1>Hello, World!</h1>
    <p>Welcome to the index.html web page..</p>
</body>
</html>

p-c@x-x:~/Documents/Code/python/webserver$ ls
index.html  web_server.py
```

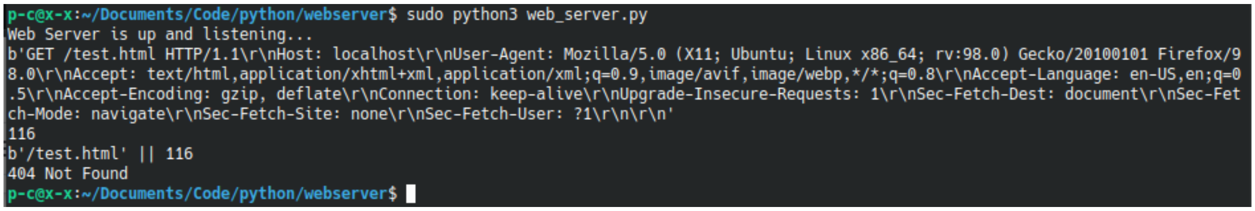
Part 4:

When trying to access files or addresses that do not exist within the directory. The error code 404 will be displayed



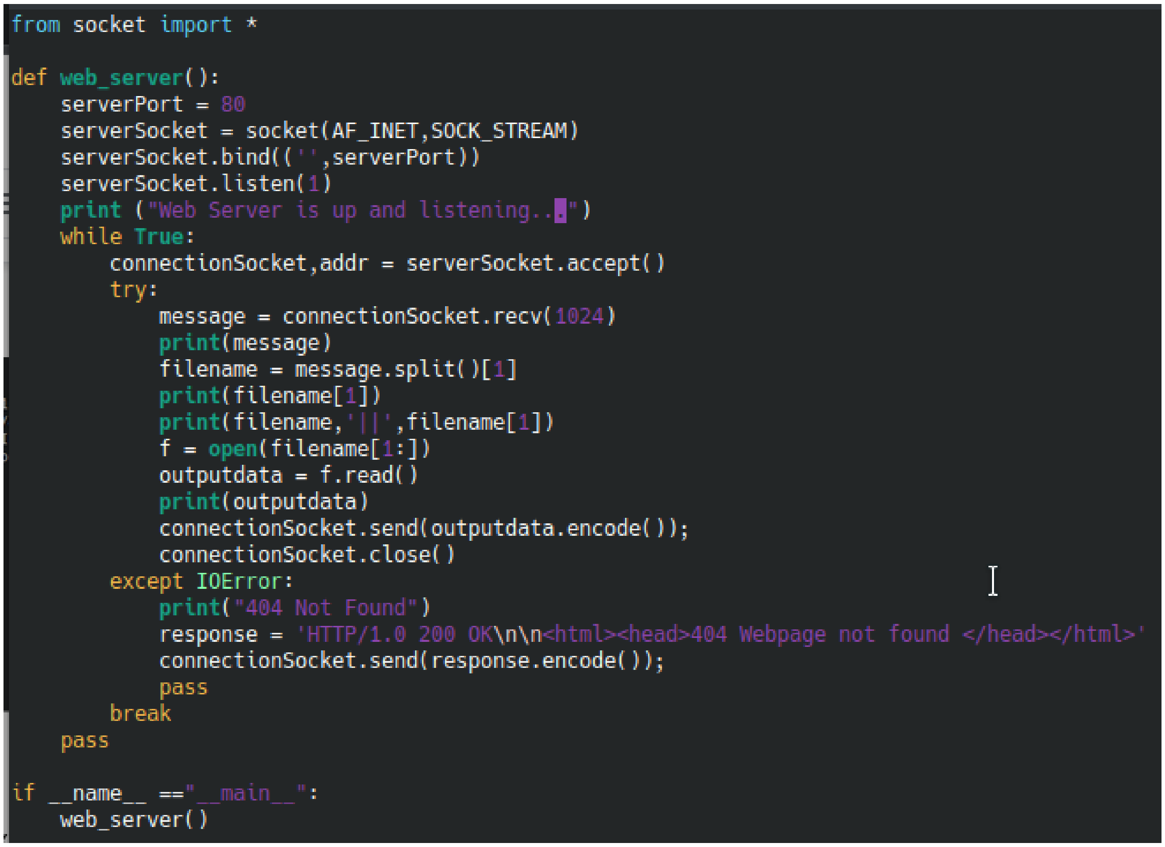
404 Webpage not found

Image of what is displayed on the terminal/server and the actual code



```
p-c@x-x:~/Documents/Code/python/webserver$ sudo python3 web_server.py
Web Server is up and listening...
b'GET /test.html HTTP/1.1\r\nHost: localhost\r\nUser-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:98.0) Gecko/20100101 Firefox/9
8.0\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8\r\nAccept-Language: en-US,en;q=0
.5\r\nAccept-Encoding: gzip, deflate\r\nConnection: keep-alive\r\nUpgrade-Insecure-Requests: 1\r\nSec-Fetch-Dest: document\r\nSec-Fet
ch-Mode: navigate\r\nSec-Fetch-Site: none\r\nSec-Fetch-User: ?1\r\n\r\n'
116
b'/test.html' || 116
404 Not Found
p-c@x-x:~/Documents/Code/python/webserver$ 
```

CODE:



```
from socket import *

def web_server():
    serverPort = 80
    serverSocket = socket(AF_INET,SOCK_STREAM)
    serverSocket.bind(('',serverPort))
    serverSocket.listen(1)
    print ("Web Server is up and listening...")
    while True:
        connectionSocket,addr = serverSocket.accept()
        try:
            message = connectionSocket.recv(1024)
            print(message)
            filename = message.split()[1]
            print(filename[1])
            print(filename,'||',filename[1])
            f = open(filename[1:])
            outputdata = f.read()
            print(outputdata)
            connectionSocket.send(outputdata.encode());
            connectionSocket.close()
        except IOError:
            print("404 Not Found")
            response = 'HTTP/1.0 200 OK\n\n<html><head>404 Webpage not found </head></html>'
            connectionSocket.send(response.encode());
            pass
        break
    pass

if __name__ =="__main__":
    web_server()
```

```
from socket import *

def web_server():
   serverPort = 80
   serverSocket = socket(AF_INET,SOCK_STREAM)
   serverSocket.bind(('',serverPort))
   serverSocket.listen(1)
   print ("Web Server is up and listening...")
   while True:
       connectionSocket,addr = serverSocket.accept()
       try:
           message = connectionSocket.recv(1024)
           print(message)
           filename = message.split()[1]
           print(filename[1])
           print(filename,'||',filename[1])
           f = open(filename[1:])
           outputdata = f.read()
           print(outputdata)
           connectionSocket.send(outputdata.encode());
           connectionSocket.close()
       except IOError:
           print("404 Not Found")
           response = 'HTTP/1.0 200 OK\n\n<html><head>404 Webpage not found </head></html>'
           connectionSocket.send(response.encode());
           pass
       break
   Pass

if __name__ =="__main__":
   web_server()
```