**CPSC 471 – Programming Assignment 1**

Python Code for Web Server

from socket import \*  
import sys  
  
  
def tcp\_server\_socket\_prog():  
 server\_socket = socket(AF\_INET, SOCK\_STREAM)  
 # Prepare a server socket  
 server\_port = 80  
 server\_socket.bind(('127.0.0.1', server\_port))  
 server\_socket.listen()  
  
 while True:  
 # Establish the connection  
 print('Ready to serve...')  
 connection\_socket, addr = server\_socket.accept()  
 try:  
 message = connection\_socket.recv(1024)  
 filename = message.split()[1]  
 f = open(filename[1:])  
 output\_data = f.read()  
 # Send one HTTP header line into socket  
 connection\_socket.send(b'200 OK')  
 # Send the content of the requested file to the client  
 for i in range(0, len(output\_data)):  
 connection\_socket.send(output\_data[i].encode())  
 connection\_socket.send("\r\n".encode())  
 connection\_socket.close()  
 sys.exit()  
  
 except IOError:  
 # Send response message for file not found  
 connection\_socket.send(b'404 File not found')  
 connection\_socket.close()  
 # Close client socket  
 server\_socket.close()  
 sys.exit()  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 tcp\_server\_socket\_prog()

Client Browser

Text

Description automatically generated

404 Error

Graphical user interface, text

Description automatically generated

Python Code for Client

import sys  
from socket import \*  
  
  
def tcp\_client\_socket\_prog(server\_host, server\_port, filename):  
 client\_socket = socket(AF\_INET, SOCK\_STREAM)  
 client\_socket.connect((server\_host, server\_port))  
 REQUEST = "GET /" + filename + " HTTP/1.1\r\nHost: " + server\_host + "\r\nConnection: keep-alive\r\nCache-Control: max-age=0\r\nUpgrade-Insecure-Requests: 1\r\nUser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.5060.114 Safari/537.36\r\nAccept-Language: en-US\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\nSec-GPC: 1\r\nSec-Fetch-Site: none\r\nSec-Fetch-Mode: navigate\r\nSec-Fetch-User: ?1\r\nSec-Fetch-Dest: document\r\nAccept-Encoding: gzip, deflate, br\r\n\r\n"  
 client\_socket.sendall(REQUEST.encode())  
 server\_response = client\_socket.recv(1024)  
 print("From Server: ", server\_response)  
 while server\_response:  
 print(server\_response.decode())  
 server\_response = client\_socket.recv(1024 \* 10\*\*6)  
 client\_socket.close()  
  
arguments = sys.argv  
tcp\_client\_socket\_prog(arguments[1], int(arguments[2]), arguments[3])

Client Side Success Message

Text

Description automatically generated

Client Side 404 Not Found Message

Text

Description automatically generated