

Socket Programming 1 Screenshots

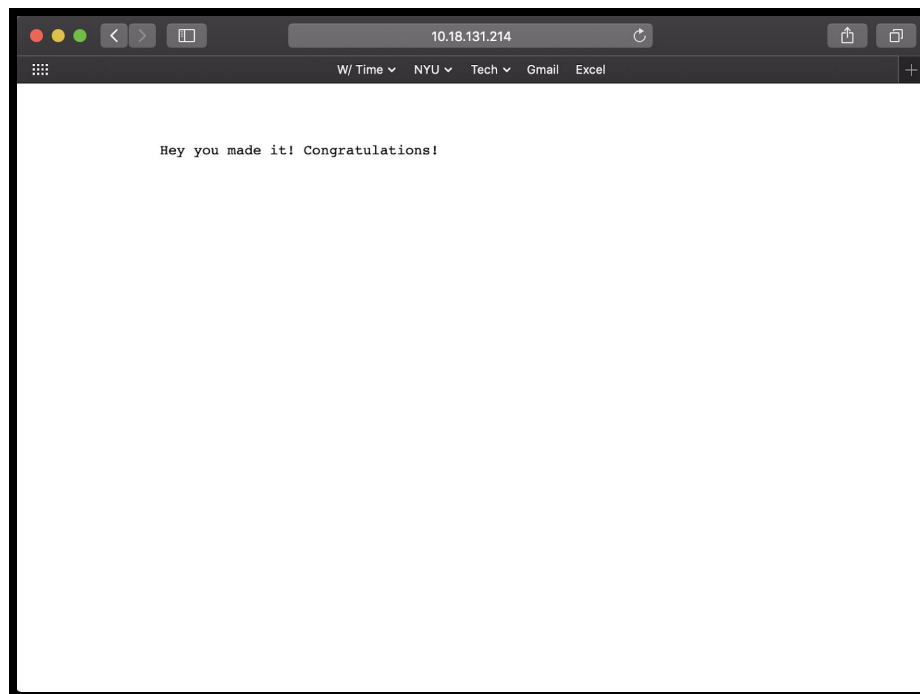
Gina Joerger

09/28/2019

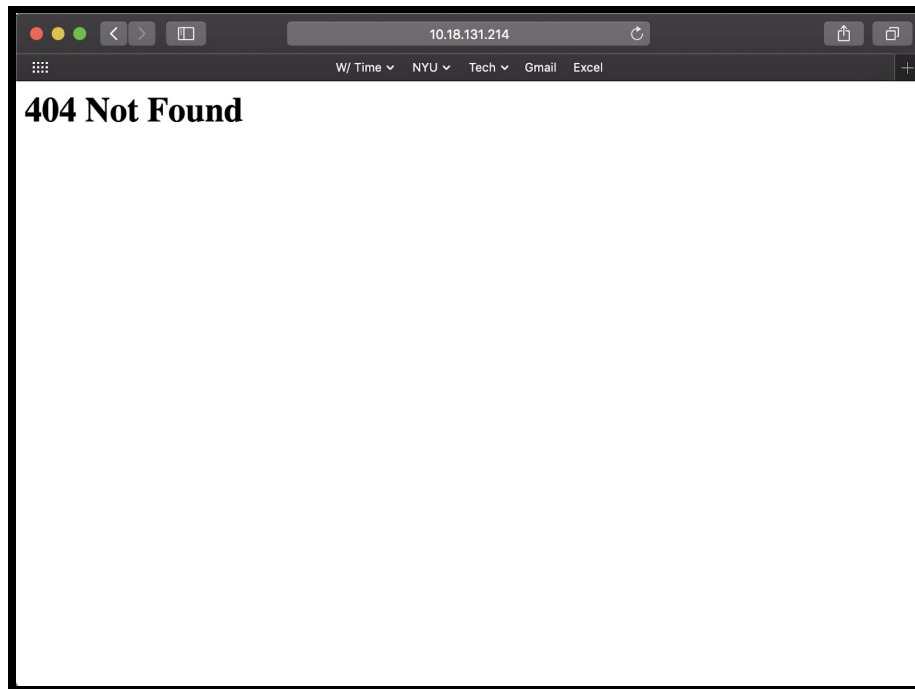
IP Configuration Screenshot:

```
[10-18-131-214:~ ginajoerger$ ifconfig |grep inet
inet 127.0.0.1 netmask 0xff000000
inet6 ::1 prefixlen 128
inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
inet6 fe80::aede:48ff:fe00:1122%en5 prefixlen 64 scopeid 0x7
inet6 fe80::44e:b53:eed:1076%en0 prefixlen 64 secured scopeid 0x8
inet 10.18.131.214 netmask 0xffff0000 broadcast 10.18.255.255
inet6 fe80::e889:78ff:fe39:aedb%awdl0 prefixlen 64 scopeid 0xa
inet6 fe80::e03f:8ed2:84f8:49c7%utun0 prefixlen 64 scopeid 0x10
inet6 fe80::e11e:76:ff6a:9625%utun1 prefixlen 64 scopeid 0x11
```

File Found Successfully:



File Not Found Successfully:



Server Running Successfully:

A screenshot of the Spyder Python IDE interface. The main editor window displays a Python script named "Socket Programming 1.py". The script is a simple HTTP server that listens on port 8081 and serves files from the directory "/Users/ginajoerger/Desktop/Computer Networking/Socket Programming 1/". The script includes comments and error handling for file not found. The console window on the right shows the output of the script, indicating that the server is running and ready to serve. The status bar at the bottom shows "Permissions: RW", "End-of-lines: LF", "Encoding: UTF-8", "Line: 44", "Column: 71", and "Memory: 56 %".

```
1#!/usr/bin/env python3
2# -*- coding: utf-8 -*-
3
4Created on Fri Sep 27 18:50:04 2019
5
6@author: ginajoerger
7
8
9#Import socket module
10from socket import *
11import sys #In order to terminate the program
12
13#Prepare a server socket
14serverPort = 8081 #declared server port
15serverSocket = socket(AF_INET, SOCK_STREAM)
16serverSocket.bind(('', serverPort)) #binds port to address
17serverSocket.listen(1) #listen to one connection at a time
18
19while True:
20    #Establish the connection
21    print('Ready to serve...')
22    connectionSocket, addr = serverSocket.accept() #accept and establish new connection
23    try:
24        message = connectionSocket.recv(1024)
25        filename = message.split()[1]
26        f = open(filename[1:])
27        outputdata = f.read()
28
29        connectionSocket.send(b'HTTP/1.1 200 OK\r\n\r\n') #Send one HTTP header line into socket
30
31        #Send the content of the requested file to the client
32        for i in range(0, len(outputdata)):
33            connectionSocket.send(outputdata[i].encode('utf-8'))
34        connectionSocket.send(b'\r\n')
35        connectionSocket.close()
36    except IOError:
37        #Send response message for file not found
38        connectionSocket.send(b'HTTP/1.1 404 Not Found\r\n\r\n')
39        connectionSocket.send(b'<html><head></head><body><h1>404 Not Found</h1></body></html>\r\n')
40        #Close client socket
41        connectionSocket.close()
42
43serverSocket.close()
44sys.exit() #Terminate the program after sending the corresponding
```