



Computer Networking Assignment 2

Lab 2: Web Server Lab

In this lab, you will learn the basics of socket programming for TCP connections in Python: how to create a socket, bind it to a specific address and port, and send and receive a HTTP packet. You will also learn some basics of HTTP header format.

You will develop a web server that handles one HTTP request at a time. Your web server should accept and parse the HTTP request, get the requested file from the server's file system, create an HTTP response message consisting of the requested file preceded by header lines, and send the response directly to the client. If the requested file is not present in the server, the server should send an HTTP "404 Not Found" message back to the client.

Code

Below you will find the skeleton code for the Web server. You are to complete the skeleton code. The places where you need to fill in code are marked with **#Fill in start** and **#Fill in end**. Each place may require one or more lines of code.

Running the Server

Put an HTML file (e.g., helloworld.html) in the same directory that the server is in. Run the server program. Determine the IP address of the host that is running the server (e.g., 127.0.0.1). Open a browser and provide the corresponding URL. For example: <http://127.0.0.1:13331/helloworld.html>

'helloworld.html' is the name of the file you placed in the server directory. Note also the use of the port number after the colon. GradeScope will test your code using port 13331. In the above example, we have used the port number 13331. The browser should then display the contents of helloworld.html. If you omit ":13331", the browser will assume port 80 and you will get the web page from the server only if your server is listening at port 80. Next, try to get a file that is not present at the server. You should get a "404 Not Found" message.

What to Hand in

Submit the code to GradeScope (include helloworld.html if it exists) using your GitHub repository. Program file name must be solution.py.



Notes:

- There are clients (browsers) that will not present HTML content unless encoded HTTP headers are submitted with the message from the web server.
- HTTP status codes “200 OK” and “404 Not Found” are required to be part of the Web Server in order to receive full credit on this assignment.

The .py file is available to download using the link below

<https://drive.google.com/file/d/1xlahicmgwQ4nTxSMO2a2tyfFoCohsJT1/view?usp=sharing>

Skeleton Python Code for the Web Server

```
# import socket module
from socket import *
# In order to terminate the program
import sys

def webServer(port=13331):
    serverSocket = socket(AF_INET, SOCK_STREAM)
    #Prepare a server socket
    serverSocket.bind(("", port))
    #Fill in start
    #Fill in end
    while True:
        #Establish the connection
        #print('Ready to serve...')
        connectionSocket, addr = #Fill in start #Fill in end
        try:
            #Fill in start
            message = #Fill in start #Fill in end
            filename = message.split()[1]
            f = open(filename[1:])
            outputdata = #Fill in start #Fill in end
            #Send one HTTP header line into socket.
            #Fill in start
            #Fill in end
            #Send the content of the requested file to the client
            for i in range(0, len(outputdata)):
                connectionSocket.send(outputdata[i].encode())
            connectionSocket.send("\r\n".encode())
            connectionSocket.close()
        except IOError:
            # Send response message for file not found (404)
            #Fill in start
            #Fill in end
```



```
        #Close client socket
        #Fill in start
        #Fill in end
    except (ConnectionResetError, BrokenPipeError):
        pass
serverSocket.close()
sys.exit() # Terminate the program after sending the corresponding
data
if __name__ == "__main__":
    webServer(13331)
```



FAQ

Q: I am getting the following error in gradescope:

`"cp: cannot stat '/autograder/submission/solution.py': No such file or directory"`

A: If you are submitting a python solution, all python submissions must have the filename titled "solution.py" (minus the quotation marks). Make sure your file meets this naming requirement.

Recommended Textbook Reference

- Chapter 2: 2.7 Socket Programming: Creating Network Applications

Most Common issues

1. Print statements are not commented out. If uncommented, they may cause errors in your gradescope submission. Print statements should only be used for local troubleshooting.
2. Improper encoding
 - a. See or <https://pythontic.com/modules/socket/send>
3. Not uploading reference helloworld.html file or building in helloworld into the code submission (either option may suffice)
4. Inappropriate socket management (i.e. not opening or closing the socket appropriately)
5. Syntactical errors