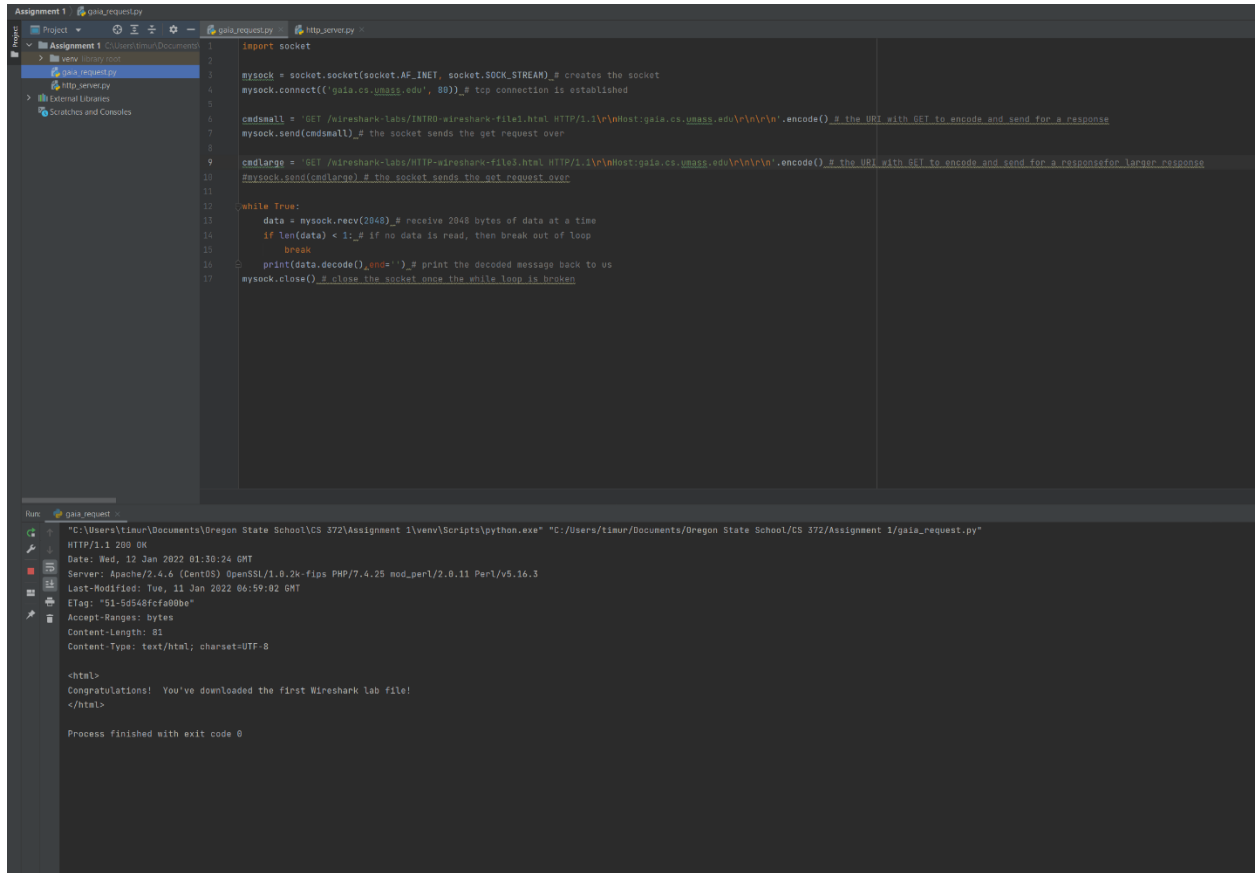


## Programming Assignment 1 Screenshots and Instructions:

### 1) Screenshot of the simple gaia connection



```
Assignment 1  gaia_request.py  http_server.py
Project
  Assignment 1  C:\Users\timur\Documents
    venv\library-root
    gaia_request.py
    http_server.py
  Internet Libraries
  Scritches and Canvases

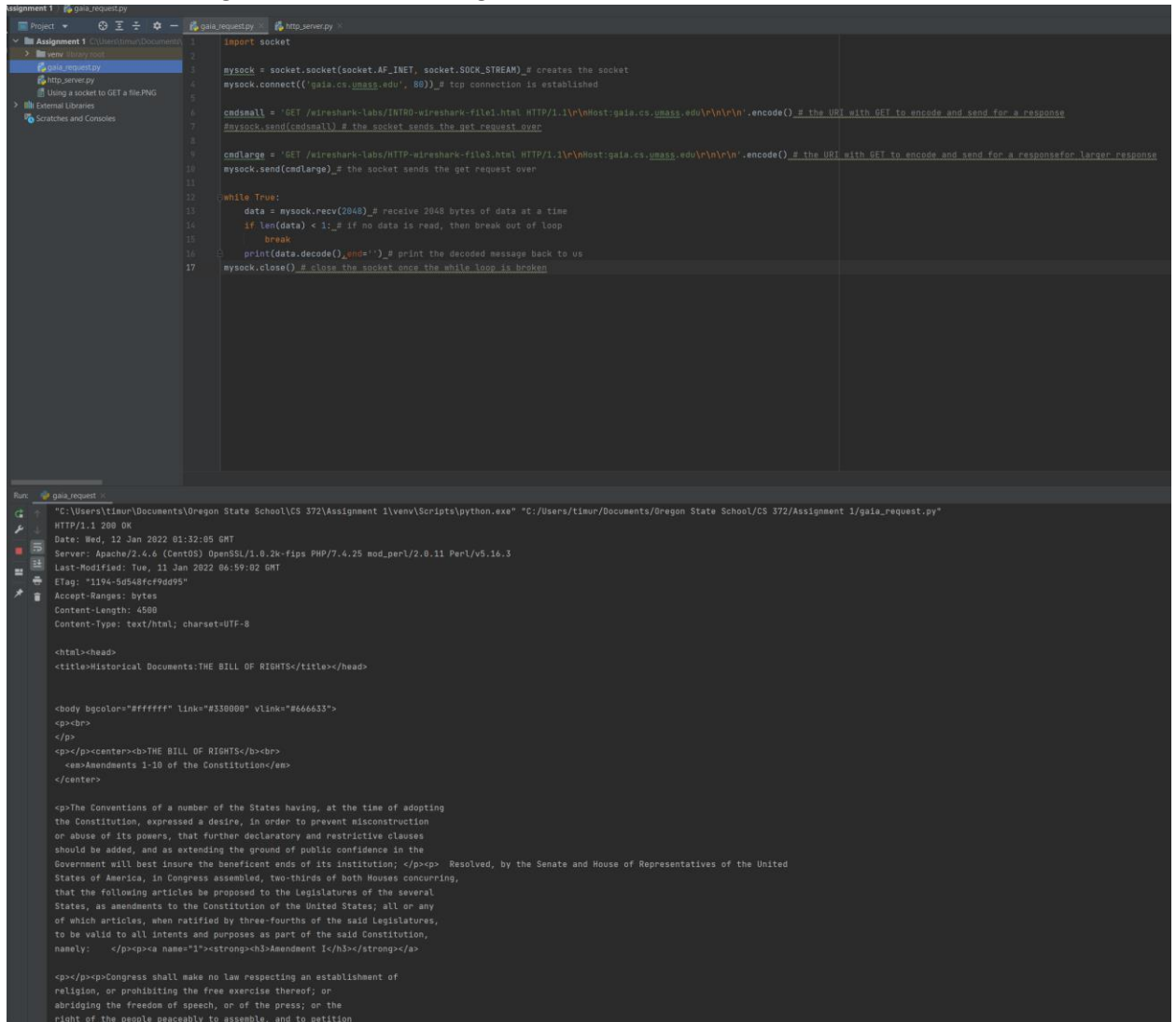
1  import socket
2
3  mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM) # creates the socket
4  mysock.connect(('gaia.cs.umass.edu', 80)) # tcp connection is established
5
6  cmdsmall = 'GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\nHost: gaia.cs.umass.edu\r\n\r\n'.encode() # the HTTP with GET to encode and send for a response
7  mysock.send(cmdsmall) # the socket sends the get request over
8
9  cmdlarge = 'GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\nHost: gaia.cs.umass.edu\r\n\r\n'.encode() # the HTTP with GET to encode and send for a response for larger response
10 mysock.send(cmdlarge) # the socket sends the get request over
11
12 while True:
13     data = mysock.recv(2048) # receive 2048 bytes of data at a time
14     if len(data) < 1: # if no data is read, then break out of loop
15         break
16     print(data.decode('utf-8')) # print the decoded message back to us
17 mysock.close() # close the socket once the while loop is broken

Run  gaia_request.py
"C:\Users\timur\Documents\Oregon State School\CS 372\Assignment 1\venv\Scripts\python.exe" "C:/Users/timur/Documents/Oregon State School/CS 372/Assignment 1/gaia_request.py"
HTTP/1.1 200 OK
Date: Wed, 12 Jan 2022 01:30:24 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.25 mod_perl/2.0.11 Perl/v5.16.3
Last-Modified: Tue, 11 Jan 2022 06:59:02 GMT
ETag: "51-5d548fcfa00be"
Accept-Ranges: bytes
Content-Length: 81
Content-Type: text/html; charset=UTF-8

<html>
  Congratulations! You've downloaded the first Wireshark lab file!
</html>

Process finished with exit code 0
```

## 2) Screenshots of the gaia connection for longer data



The screenshot shows a Python script named `gaia_request.py` and its output in a terminal window. The script is designed to connect to a web server (gaia.cs.umass.edu) and retrieve data using a socket connection. It handles both small and large data requests by sending different commands.

```
1 import socket
2
3 mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM) # creates the socket
4 mysock.connect(('gaia.cs.umass.edu', 80)) # tcp connection is established
5
6 cmdsmall = 'GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\nHost: gaia.cs.umass.edu\r\n\r\n'.encode() # the URI with GET to encode and send for a response
7 mysock.send(cmdsmall) # the socket sends the get request over
8
9 cmdlarge = 'GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1\r\nHost: gaia.cs.umass.edu\r\n\r\n'.encode() # the URI with GET to encode and send for a response for larger response
10 mysock.send(cmdlarge) # the socket sends the get request over
11
12 while True:
13     data = mysock.recv(2048) # receive 2048 bytes of data at a time
14     if len(data) < 1: # if no data is read, then break out of loop
15         break
16     print(data.decode('utf-8')) # print the decoded message back to us
17 mysock.close() # close the socket once the while loop is broken
```

The terminal output shows the execution of the script, displaying the HTTP response from the server. The response includes headers and HTML content, indicating a successful connection and data retrieval.

```
Run: "C:\Users\timur\Documents\Oregon State School\CS 372\Assignment 1\venv\scripts\python.exe" "C:\Users\timur\Documents\Oregon State School\CS 372\Assignment 1\gaia_request.py"
HTTP/1.1 200 OK
Date: Wed, 12 Jan 2022 01:32:05 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.25 mod_perl/2.0.11 Perl/v5.16.3
Last-Modified: Tue, 11 Jan 2022 06:59:02 GMT
ETag: "1194-5d548fc9d995"
Accept-Ranges: bytes
Content-Length: 4590
Content-Type: text/html; charset=UTF-8

<html><head>
<title>Historical Documents:THE BILL OF RIGHTS</title></head>

<body bgcolor="#ffffff" link="#330000" vlink="#666633">
<p><br>
</p>
<p><center><b>THE BILL OF RIGHTS</b><br>
<em>Amendments 1-10 of the Constitution</em>
</center>

<p>The Conventions of a number of the States having, at the time of adopting
the Constitution, expressed a desire, in order to prevent misconstruction
or abuse of its powers, that further declaratory and restrictive clauses
should be added, and as extending the ground of public confidence in the
Government will best insure the beneficent ends of its institution; </p><p> Resolved, by the Senate and House of Representatives of the United
States of America, in Congress assembled, two-thirds of both Houses concurring,
that the following articles be proposed to the Legislatures of the several
States, as amendments to the Constitution of the United States; all or any
of which articles, when ratified by three-fourths of the said Legislatures,
to be valid to all intents and purposes as part of the said Constitution,
namely:  </p><p><a name="1"><strong><h3>Amendment 1</h3></strong></a>

<p></p><p>Congress shall make no law respecting an establishment of
religion, or prohibiting the free exercise thereof; or
abridging the freedom of speech, or of the press; or the
right of the people peaceably to assemble, and to petition
```

Project

Assignment 1 C:\Users\timur\Documents

venv library root

gaia\_request.py

http\_server.py

Using a socket to GET a file.PNG

Using a socket to GET a file 1.PNG

External Libraries

Scratches and Consoles

gaia\_request.py

http\_server.py

```
1 import socket
2
3 mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM) # creates the socket
4 mysock.connect(('gaia.cs.umass.edu', 80)) # tcp connection is established
5
6 cmdsmall = 'GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\nHost:gaia.cs.umass.edu\r\n\r\n'
7 #mysock.send(cmdsmall) # the socket sends the get request over
8
9 cmdlarge = 'GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1\r\nHost:gaia.cs.umass.edu\r\n\r\n'
10 mysock.send(cmdlarge) # the socket sends the get request over
11
12 while True:
13     data = mysock.recv(2048) # receive 2048 bytes of data at a time
14     if len(data) < 1: # if no data is read, then break out of loop
15         break
16     print(data.decode(),end='') # print the decoded message back to us
17 mysock.close() # close the socket once the while loop is broken
```

Run: gaia\_request

↑

↓

↺

↻

🔍

🗑️

</p><p><a name="6"><strong><h3>Amendment VI</h3></strong></a>

<p></p><p>In all criminal prosecutions, the accused shall enjoy the right

to a speedy and public trial, by an impartial jury of the state

and district wherein the crime shall have been committed, which

district shall have been previously ascertained by law, and

to be informed of the nature and cause of the accusation;

to be confronted with the witnesses against him; to have

compulsory process for obtaining witnesses in his favor,

and to have the assistance of counsel for his defense.

</p><p><a name="7"><strong><h3>Amendment VII</h3></strong></a>

<p></p><p>In suits at common law, where the value in controversy shall

exceed twenty dollars, the right of trial by jury shall be

preserved, and no fact tried by a jury, shall be otherwise

reexamined in any court of the United States, than according

to the rules of the common law.

</p><p><a name="8"><strong><h3>Amendment VIII</h3></strong></a>

<p></p><p>Excessive bail shall not be required, nor excessive fines

imposed, nor cruel and unusual punishments inflicted.

</p><p><a name="9"><strong><h3>Amendment IX</h3></strong></a>

<p></p><p>The enumeration in the Constitution, of certain rights, shall

not be construed to deny or disparage others retained by the people.

</p><p><a name="10"><strong><h3>Amendment X</h3></strong></a>

<p></p>

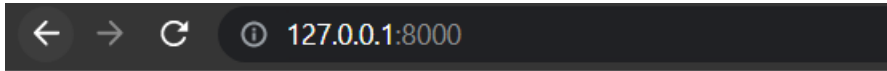
<p>The powers not delegated to the United States by the Constitution, nor prohibited

by it to the states, are reserved to the states respectively, or to the people.</p>

</body></html>

Process finished with exit code 0

3) Screenshots of the HTTP server running



Congratulations! You've downloaded the first Wireshark lab file!

```
# Class - CS372 Winter 2022
# This has been adapted from the book as well as using https://www.youtube.com/watch?v=80vynoW6fI sockets

import socket

serverSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM) # creates the tcp socket
serverSocket.bind(('127.0.0.1', 8080)) # binds the created socket to the host and port
serverSocket.listen(1) # this line has the server listen
print("Connect by ('127.0.0.1',8080)\n") # lets us know the server is running

connectionSocket, add = serverSocket.accept() # the accept method creates a new socket called connectionSocket and
sentence = connectionSocket.recv(1024) # get the response
print('Received:' + sentence) # print it to the console in the encoded format

response = "HTTP/1.1 200 OK\n\n" \
    "Content-Type: text/html; charset=UTF-8\n\n" \
    "<html>Congratulations! You've downloaded the first Wireshark lab file!</html>\n\n" # the response we will send to the user
print('\nSending >>>>>>>>') # the next three lines print the response the server is sending to the console
print(response)
print('<<<<<<<<')

connectionSocket.send(response.encode()) # send the response encoded to display as html to the user
connectionSocket.close() # close the socket
```

Run: http\_server

```
"C:\Users\timur\Documents\Oregon State School\CS 372\Assignment 1\env\Scripts\python.exe" "C:\Users\timur\Documents\Oregon State School\CS 372\Assignment 1\http_server.py"
Connect by ('127.0.0.1',8080)

Received: b'GET / HTTP/1.1\r\nHost: 127.0.0.1:8080\r\nConnection: keep-alive\r\nsec-ch-ua: " Not A;Brand";v="99", "Chromium";v="96", "Google Chrome";v="96"\r\nsec-ch-ua-mobile:
70\r\nsec-ch-ua-platform: "Windows"\r\nUpgrade-Insecure-Requests: 1\r\nUser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.110
Safari/537.36\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\nPurpose:
prefetch\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\nAccept-Language: en-US,en;q=0
.9\r\nCookie: tc_ptid=t42QX3FsT0X3VKdHb3y60; tc_ptidexpiry=1697581724842\r\n\r\n'

Sending >>>>>>>>
HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8

<html>Congratulations! You've downloaded the first Wireshark lab file!</html>

<<<<<<<<

Process finished with exit code 0
```

- 4)
- To run the code there are two options
    - Open PyCharm and hit run.
    - Open the console and navigate to the location the file is in. Enter `python http_server.py` to run on the console.
  - Once the code is running, open the web browser and go to `127.0.0.1:8080`. You will receive the message about downloading the wireshark file on the browser and the console will display the output in the screenshot.