**WEEK 5 – CN LAB**

**NAME: - DIVYANSHU SHARMA**

**PES1UG20CS806**

**Simple Client-Server Application using Network Socket Programming**

**Task 1:**

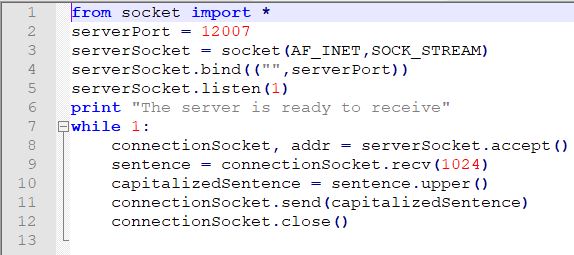
1. Create an application that will
2. Convert lowercase letters to uppercase

• e.g. [a…z] to [A…Z]

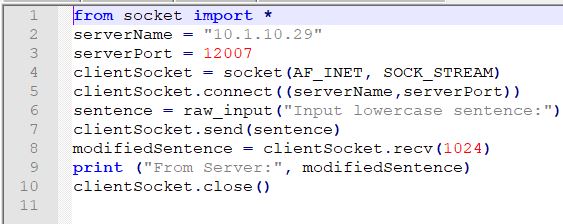
• code will not change any special characters, e.g. &\*!

1. If the character is in uppercase, the program must not alter
2. Create Socket API both for client and server.
3. Must take the server address and port from the Command Line Interface (CLI).
   1. **TCP Connection**

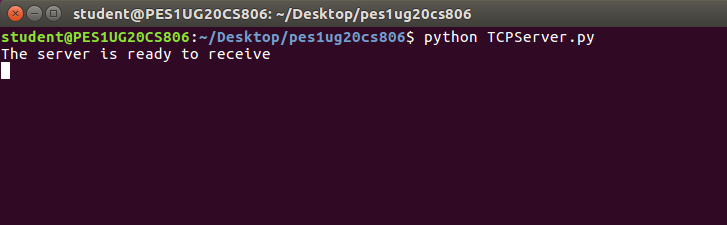
* A TCP connection can be made between two machines with the help of a socket interface using the socket library on Python.
* To create a TCP socket interface, the type of socket needs to be set as SOCK\_STREAM.
* The type of addresses needs to be set as AF\_INET which corresponds to IPv4.
* Once the server socket application is created, it needs to be hosted and hence needs to bind to a host IP and port number using the bind () function.
* Similarly, the client socket application needs to connect to a host using the IP address and port number.
* The socket can now listen for incoming connections as well as send messages to connected host machines.
  + 1. **TCP Server**



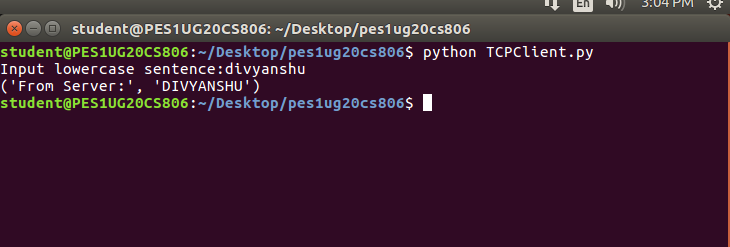
* + 1. **TCP Client**



* + 1. **TCP Connection between Server and Client**

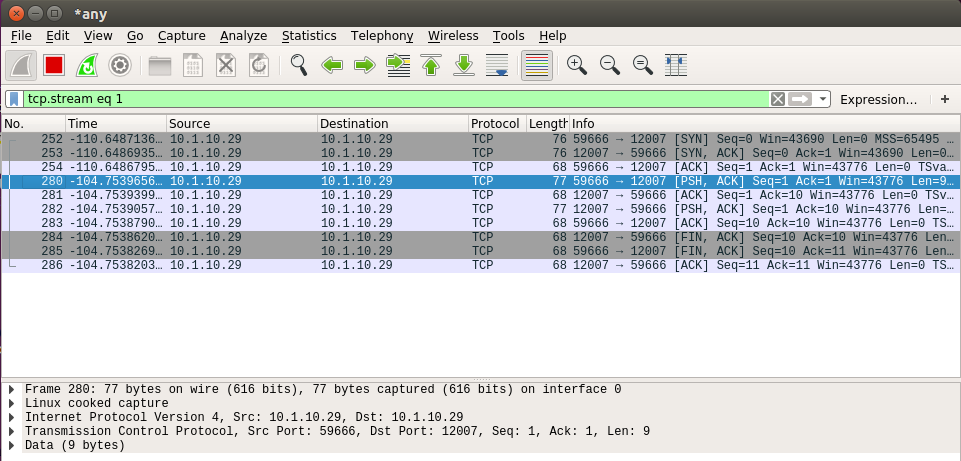


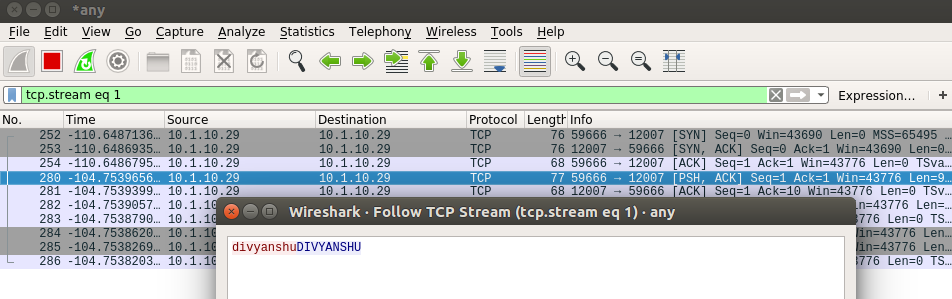
**TCP Server**



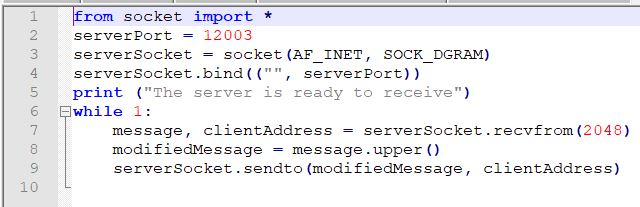
**TCP Client**

* + 1. **Wireshark Capture for TCP Connection**

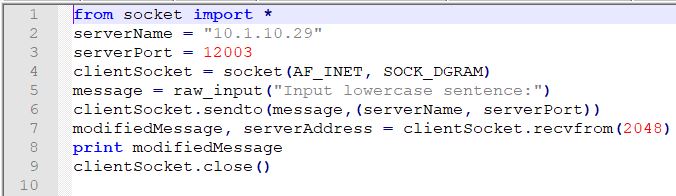




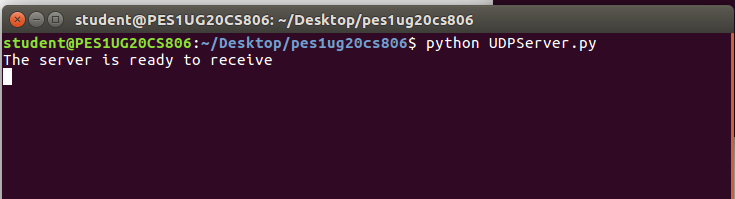
* 1. **UDP Connection**
* A UDP connection can be made between two machines with the help of a socket interface using the socket library on Python.
* A UDP connection can be made between two machines with the help of a socket interface using the socket library on Python3.
* The type of addresses needs to be set as AF\_INET which corresponds to IPv4.
* Once the server socket application is created, it needs to be hosted and hence needs to bind to a host IP and port number using the bind () function.
* Similarly, the client socket application needs to connect to a host using the IP address and port number.
* The socket can now listen for incoming connections as well as send messages to connected host machines.
  + 1. **UDP Server**



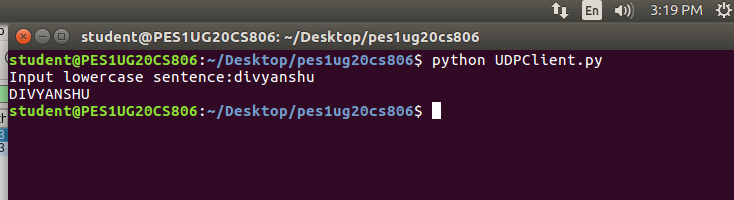
* + 1. **UDP Client**



* + 1. **UDP Connection between Server and Client**

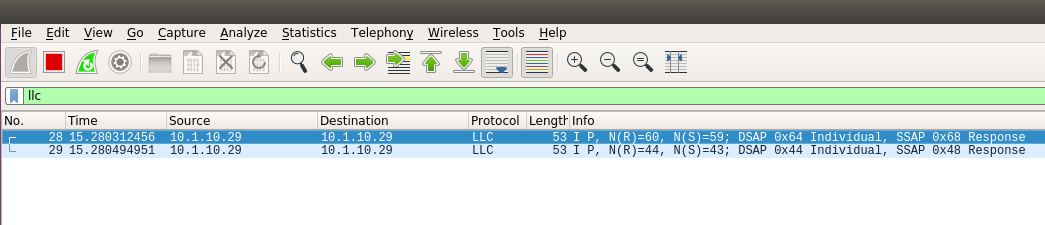


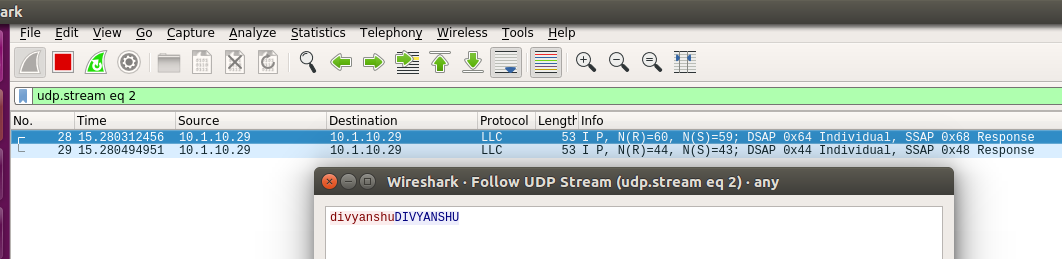
**UDP Server**



**UDP Client**

* + 1. **Wireshark Capture for UDP Connection**





* 1. **Problems**

***Q1****.* ***Suppose you run TCPClient before you run TCPServer. What happens? Why?***

***Answer:***This will lead to a ***ConnectionRefusedError***, since the server socket application we are trying to connect has not been initiated and is not listening for connections on the given port number. Therefore, any connection requests sent by a client machine at that IP address and port number immediately fail since the connection gets refused. A TCP connection can be established between two socket interfaces only when a host machine listens to requests on a given IP address and port number and accepts connections made by another machine at the same address and port.

**Q2.** ***Suppose you run UDPClient before you run UDPServer. What happens? Why?***

***Answer:*** No error will be obtained since UDP does not require a prior connection to be set up between the host machines for data transfer to begin. It is a connectionless protocol which transfers packets of data to a destination IP and port number without verifying the existence of the connection. Hence, it is prone to data integrity issues such as loss of packets. If any packets of data are sent before the server is executed, the packets are lost forever and will not reach the server socket application. However, if any packets of data are sent after the server is executed, the client will be able to send packets to a destination server and also receive response packets in return.

**Q3.** ***What happens if you use different port numbers for the client and server sides?***

***Answer:*** This will lead to a ***ConnectionRefusedError*** for a TCP connection, since the server socket application we are trying to connect to is not listening for requests at the same port number as the one the client socket application is trying to connect with. Hence, the connection between the two socket interfaces is never setup and the connection are downright refused. However, on a UDP connection, since no prior connection is required to be established between the host machines for data transfer to take place, no error as such is obtained. Any messages sent by the client are lost since the destination server does not exists.

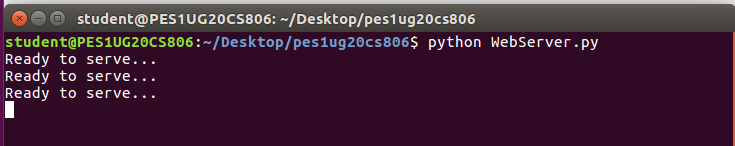
**Task 2: Web Server**

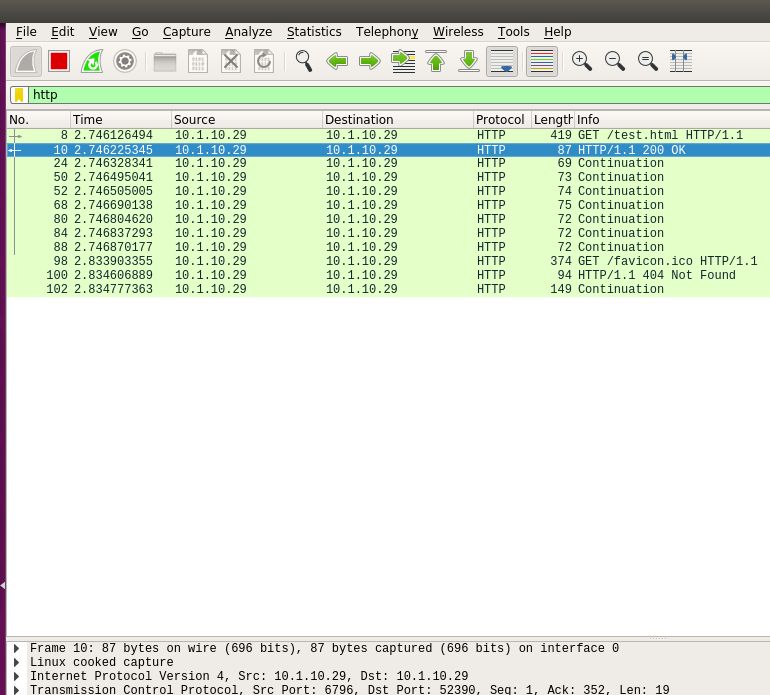
In this assignment, you will develop a simple Web server in Python that is capable of processing only one request. Specifically, your Web server will

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.

For this assignment, the companion Web site provides 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.

**SUCCESSFULLY ACCESS THE .html FILE FROM THE SERVER**

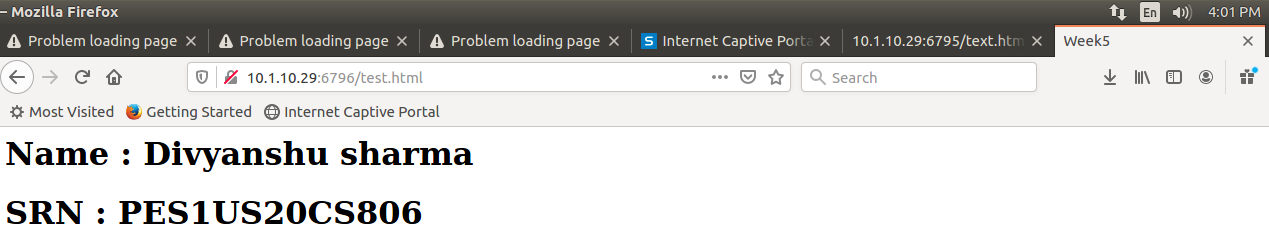




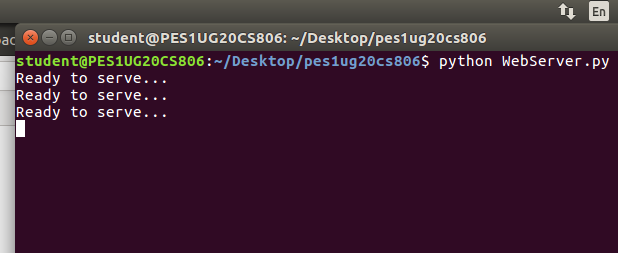
**Wireshark Capture**

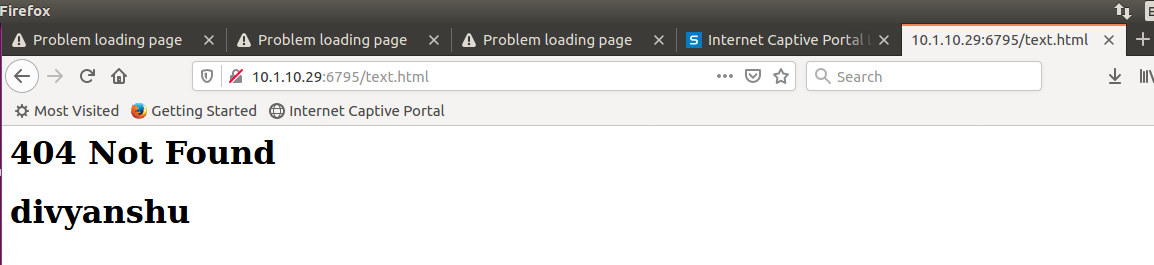


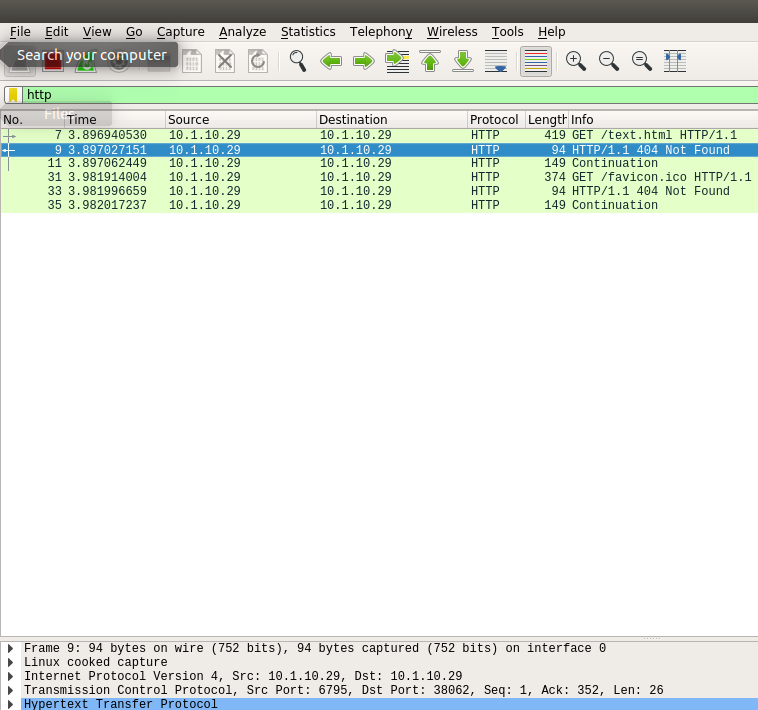
**TCP PROTOCOL**



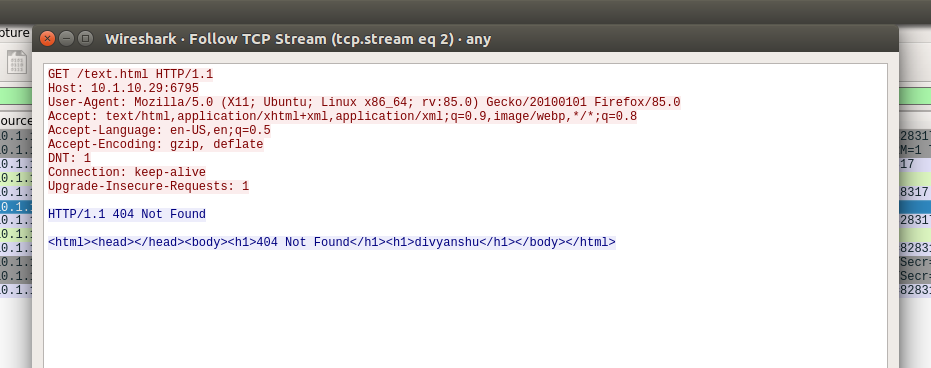
**UNSUCCESSFUL ACCESS OF FILE RETURNING *404 NOT FOUND***







**Wireshark Capture**



**TCP PROTOCOL**