Name: Pannawit krutnak ID 6688052 Sec: 1

Introduction to Application Layer and Basic Socket Programming

ITCS226 Introduction to Computer Networks 2/2024

Lab 5: Work Sheet

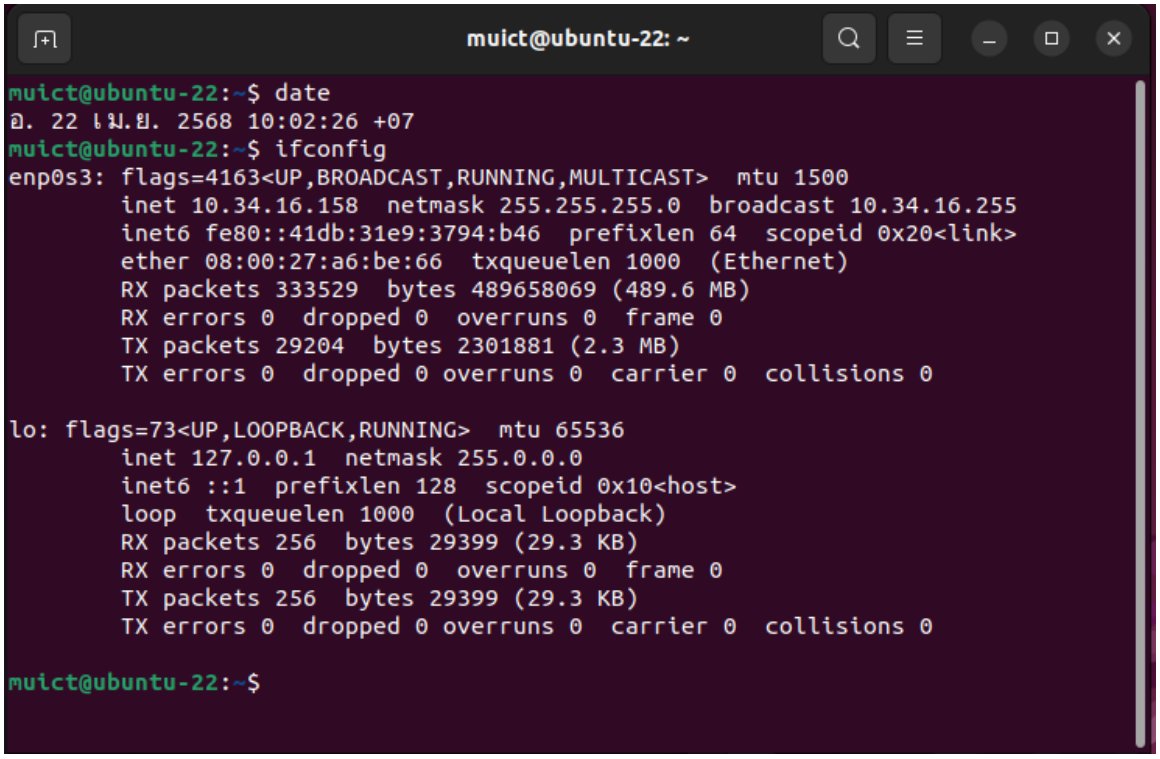
Instructions: Work on this sheet and save as PDF with the file name "Lab5-StudentID.pdf" by replacing Student ID with your Student ID, and upload to MyCourses.

Part 1: Basic system & networking commands

Read and follow the instructions in slide #3-9 to set up Ubuntu on Virtual Box

1. Type the following commands on the Terminal in Ubuntu. Screen capture the result and put your screenshot here.

```
clear
dat
ifconfig
```



Name: Pannawit krutnak ID 6688052 Sec: 1

2. What are the HWaddr (ether) and the IP address (inet) of your computer. (Hints: *not* the loopback one)
    - HW address: 08:00:27:a6:be:66
    - IP address: 10.34.16.158
    - Is this public or private address: private

3. Executed the command **`netstat -lt`** in Ubuntu?
    - Capture the output screen shot.

```
muict@ubuntu-22:~$ netstat -lt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN
tcp        0      0 localhost:ipp           0.0.0.0:*               LISTEN
tcp6       0      0 ip6-localhost:ipp       [::]:*                  LISTEN
muict@ubuntu-22:~$
```

    - What is the result of this command? Explain.

      It is TCP ports that are currently open and waiting for connections.

4. Use the command **`nano /etc/hosts`**. What are written inside this file? Explain.

```
  GNU nano 6.2                        /etc/hosts
127.0.0.1       localhost
127.0.1.1       ubuntu-22

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

                [ File '/etc/hosts' is unwritable ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

It contains mapping of hostname and IP address.

Name: Pannawit krutnak ID 6688052 Sec: 1

Part 2: Setting up and Configuring HTTP Server + Using Curl HTTP commands

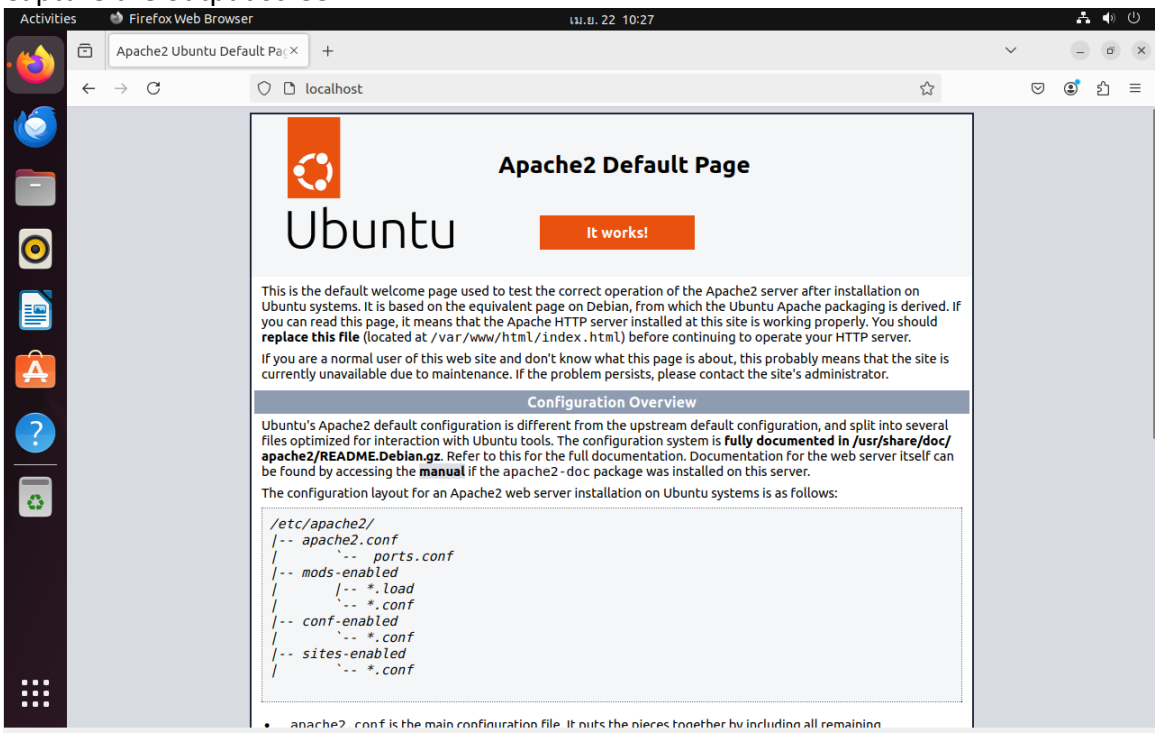
Follow the instructions in Slide #10-11 to set up Apache HTTP Server on Ubuntu

5. Once the Apache2 installation is done, capture the screenshot after executing the command `netstat -nlt` in Ubuntu, and highlight the entry related to the web server?

```
muict@ubuntu-22:~$ netstat -nlt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 ::1:631                 :::*                    LISTEN
```

- How is this command different form `netstat -lt` in Q3?
  It show ip and port

6. Open a Firefox web browser in Ubuntu and type in **http://localhost** in the URL tab. Capture the output screen.



7. Now use the web browser on your host OS (e.g., Windows), type in the Ubuntu's IP address you got from Q1. Capture the output screen?

Name: Pannawit krutnak ID 6688052 Sec: 1

Is it different from the results in Q6?
It have same result

8. Where is the *defaul*t DocumentRoot? (Hint: check the configuration file **`/etc/apache2/sites-available/000-default.conf`**).

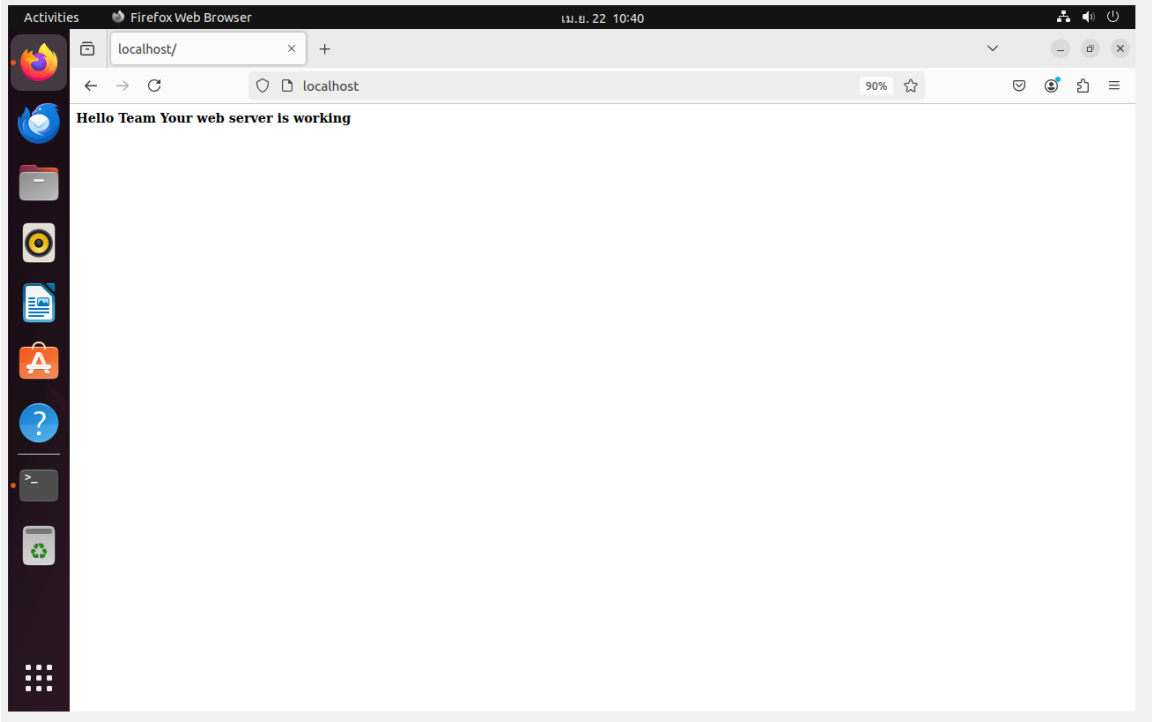
```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
```

What is the purpose of the of the DocumentRoot directory?
Ans: It is the main folder where your website's files are stored.

Follow the instructions listed in Slide #12-14 to configure Apache2 web server.

9. After changing the DocumentRoot and create index.html filr, open a Firefox web browser and type in **http://localhost** in the URL tab. Capture the screen shot of your browser.

Name: Pannawit krutnak ID 6688052 Sec: 1



Use curl to get a web page using HTTP and HTTPS. Read how to use curl in slide 15-17

10. Request a web page from `https://www.ict.mahidol.ac.th`, capture the screenshot of the response header.

```
muict@ubuntu-22:~$ curl -I https://www.ict.mahidol.ac.th
HTTP/1.1 200 OK
Date: Tue, 22 Apr 2025 03:39:34 GMT
Server: Apache/2.4.41 (Ubuntu)
Link: <https://www.ict.mahidol.ac.th/wp-json/>; rel="https://api.w.org/"
Link: <https://www.ict.mahidol.ac.th/wp-json/wp/v2/pages/37547>; rel="alternate"
; type="application/json"
Link: <https://www.ict.mahidol.ac.th/>; rel=shortlink
X-TEC-API-VERSION: v1
X-TEC-API-ROOT: https://www.ict.mahidol.ac.th/wp-json/tribe/events/v1/
X-TEC-API-ORIGIN: https://www.ict.mahidol.ac.th
Content-Type: text/html; charset=UTF-8
Set-Cookie: cookiesession1=678A8C58A46C1EBCCD7545F4EAA76607;Expires=Wed, 22 Apr
2026 03:42:29 GMT;Path=/;HttpOnly
```

- What is the status code, the server being used, and the operating system running on the server?
  Ans: Status code is 200, Server using Apache/2.4.41 on ubuntu

Name: Pannawit krutnak ID 6688052 Sec: 1

11. Use curl to request `index.html` on your localhost. Capture the **request** and **response** header. (Hint: use verbose option)

- Screen shot of HTTP request header:
  What is the user-agent in the HTTP request header?
  Ans: Agent is curl/7.81.0

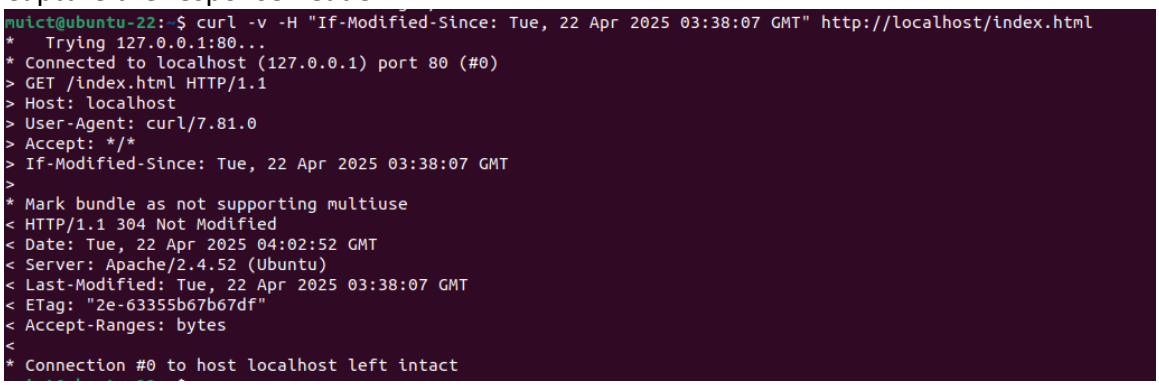
```
muict@ubuntu-22:~$ curl -v http://localhost/index.html
*   Trying 127.0.0.1:80...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /index.html HTTP/1.1
> Host: localhost
> User-Agent: curl/7.81.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Tue, 22 Apr 2025 03:57:09 GMT
< Server: Apache/2.4.52 (Ubuntu)
< Last-Modified: Tue, 22 Apr 2025 03:38:07 GMT
< ETag: "2e-63355b67b67df"
< Accept-Ranges: bytes
< Content-Length: 46
< Content-Type: text/html
<
<b>Hello Team Your web server is working </b>
* Connection #0 to host localhost left intact
```

- Screen shot of HTTP response header:

```
muict@ubuntu-22:~$ curl -i http://localhost/index.html
HTTP/1.1 200 OK
Date: Tue, 22 Apr 2025 03:58:14 GMT
Server: Apache/2.4.52 (Ubuntu)
Last-Modified: Tue, 22 Apr 2025 03:38:07 GMT
ETag: "2e-63355b67b67df"
Accept-Ranges: bytes
Content-Length: 46
Content-Type: text/html
```

-

Name: Pannawit krutnak ID 6688052 Sec: 1

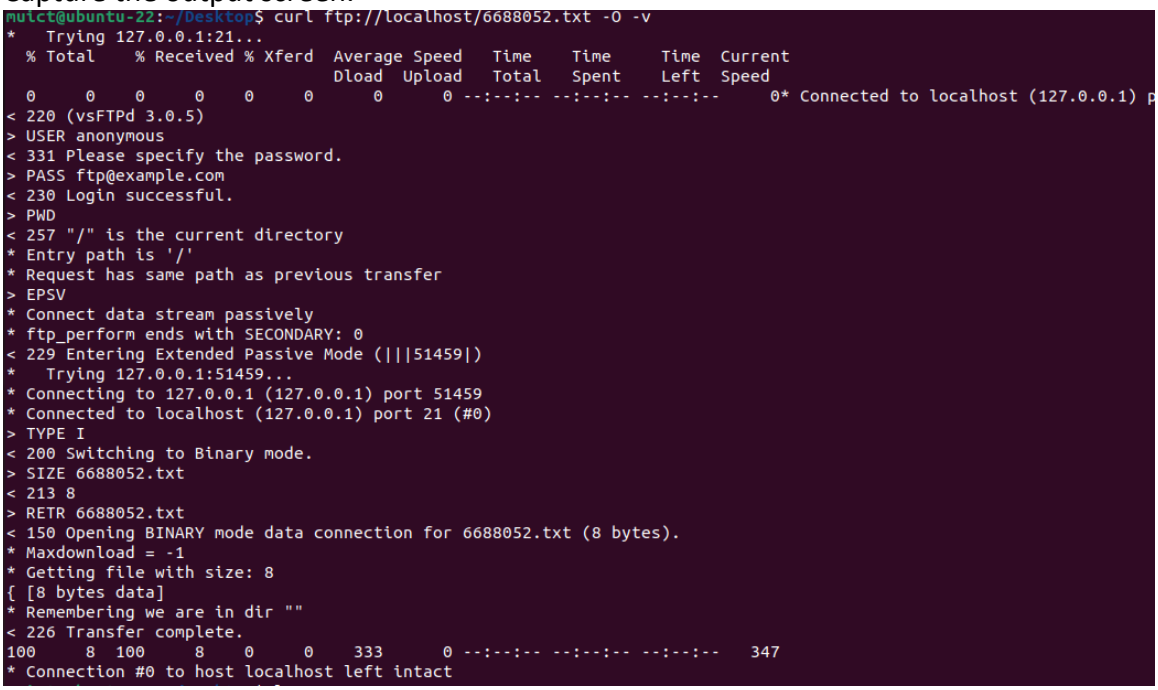
12. Use curl to request `index.html` again, but add a conditional header **If-Modified-Since** to the request message. Set the **date** parameter to the date you obtained from the response in Q11.
    - Capture the response header.
    - 
      ```
      muict@ubuntu-22:~$ curl -v -H "If-Modified-Since: Tue, 22 Apr 2025 03:38:07 GMT" http://localhost/index.html
      *   Trying 127.0.0.1:80...
      * Connected to localhost (127.0.0.1) port 80 (#0)
      > GET /index.html HTTP/1.1
      > Host: localhost
      > User-Agent: curl/7.81.0
      > Accept: */*
      > If-Modified-Since: Tue, 22 Apr 2025 03:38:07 GMT
      >
      * Mark bundle as not supporting multiuse
      < HTTP/1.1 304 Not Modified
      < Date: Tue, 22 Apr 2025 04:02:52 GMT
      < Server: Apache/2.4.52 (Ubuntu)
      < Last-Modified: Tue, 22 Apr 2025 03:38:07 GMT
      < ETag: "2e-63355b67b67df"
      < Accept-Ranges: bytes
      <
      * Connection #0 to host localhost left intact
      ```
    - What is the status code of the response message.
        - 304

Name: Pannawit krutnak ID 6688052 Sec: 1

Part 3: Setting up and Configuring FTP Server

Follow the instructions in slide #18 to set up FTP server on Ubuntu, and read how to use FTP curl commands in slide #19-20

13. Use curl to download the **6588xxx.txt** from the FTP server. Hint: server IP is 127.0.0.1, and use **–v** to see the details of FTP message exchange.
    - Capture the output screen.



```
nuict@ubuntu-22:~/Desktop$ curl ftp://localhost/6688052.txt -O -v
*   Trying 127.0.0.1:21...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0* Connected to localhost (127.0.0.1) p
< 220 (vsFTPd 3.0.5)
> USER anonymous
< 331 Please specify the password.
> PASS ftp@example.com
< 230 Login successful.
> PWD
< 257 "/" is the current directory
* Entry path is '/'
* Request has same path as previous transfer
> EPSV
* Connect data stream passively
* ftp_perform ends with SECONDARY: 0
< 229 Entering Extended Passive Mode (|||51459|)
*   Trying 127.0.0.1:51459...
* Connecting to 127.0.0.1 (127.0.0.1) port 51459
* Connected to localhost (127.0.0.1) port 21 (#0)
> TYPE I
< 200 Switching to Binary mode.
> SIZE 6688052.txt
< 213 8
> RETR 6688052.txt
< 150 Opening BINARY mode data connection for 6688052.txt (8 bytes).
* Maxdownload = -1
* Getting file with size: 8
{ [8 bytes data]
* Remembering we are in dir ""
< 226 Transfer complete.
100     8  100     8    0     0    333      0 --:--:-- --:--:-- --:--:--   347
* Connection #0 to host localhost left intact
```

    - Based on the result, answer the following questions
        - Is this FTP connection passive or active?
          passive
        - What is the port the server used for data connection?
          21
        - What is the type of data being transferred?
          I
        - What is the FTP command uses to indicate the download file operation? (Hint: FTP command is in uppercase)
          RETR

Name: Pannawit krutnak ID 6688052 Sec: 1

Part 4: Socket programming by Python

Learn socket programing from the Lecture 11.

14. Follow the instructions in slide 21-22, and write a **tcpserver.py** and **tcpclient.py** programs

- Paste the source codes of both client and server programs here.

Server:

```
import socket

STUDENT_ID = "6688052"
HOST = '127.0.0.1'
PORT = 42000

def start_server():
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
        s.bind((HOST, PORT))
        s.listen(3)
        print(f"Server listening on {HOST}:{PORT}")

        while True:
            conn, addr = s.accept()
            with conn:
                print(f"Connected by {addr[0]}:{addr[1]}")
                data = conn.recv(1024).decode()
                if data:
                    x_str, y_str = data.split(",")
                    x = int(x_str)
                    y = int(y_str)
                    print(f"Student ID: {STUDENT_ID}")
                    print(f"Received x={x}, y={y}")
                    result = x * y
                    conn.sendall(str(result).encode())

if __name__ == "__main__":
    start_server()
```

client:

```
import socket
import time

STUDENT_ID = "6688052"
```

Name: Pannawit krutnak ID 6688052 Sec: 1

```
HOST = '127.0.0.1'
PORT = 42000

def start_client():
    x = int(input("Enter first integer (x): "))
    y = int(input("Enter second integer (y): "))

    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
        s.connect((HOST, PORT))
        message = f"{x},{y}"
        s.sendall(message.encode())
        result = s.recv(1024).decode()

        print(f"Student ID: {STUDENT_ID}")
        print(f"Current Date & Time: {time.asctime()}")
        print(f"Result of x*y: {result}")

if __name__ == "__main__":
    start_client()
```

- Paste the screen output of server and client here. (Make sure your output looks like the provided example)

```
PS C:\Users\Student\Downloads\lab5\py> python3 .\tcpserver.py
Python was not found; run without arguments to install from the Microsoft Store, or disable this shortcut fro
m Settings > Apps > Advanced app settings > App execution aliases.
PS C:\Users\Student\Downloads\lab5\py> python tcpserver.py
Server listening on 127.0.0.1:42000
Connected by 127.0.0.1:59483
Student ID: 6688052
Received x=3, y=4
```

```
PS C:\Users\Student\Downloads\lab5\py> python .\tcpclient.py
Enter first integer (x): 3
Enter second integer (y): 4
Student ID: 6688052
Current Date & Time: Tue Apr 22 11:19:44 2025
Result of x*y: 12
PS C:\Users\Student\Downloads\lab5\py>
```

Name: Pannawit krutnak ID 6688052 Sec: 1

15. Change your Python programs from Q14 to support **UDP**, and named the programs as **udpserver.py** and **udpclient.py**. (see slide 23)

- Paste the source codes of both client and server programs here.

Server:

```
import socket

STUDENT_ID = "6688052"
HOST = '127.0.0.1'
PORT = 1112

def start_udp_server():
    with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
        s.bind((HOST, PORT))
        print(f"UDP server listening on {HOST}:{PORT}")

        while True:
            data, addr = s.recvfrom(1024)
            message = data.decode()
            print(f"Connected by {addr[0]}:{addr[1]}")
            if message:
                try:
                    x_str, y_str = message.split(",")
                    x = int(x_str)
                    y = int(y_str)
                    print(f"Student ID: {STUDENT_ID}")
                    print(f"Received x={x}, y={y}")
                    result = x * y
                    s.sendto(str(result).encode(), addr)
                except ValueError:
                    error_msg = "Invalid input format. Use: x,y"
                    s.sendto(error_msg.encode(), addr)

if __name__ == "__main__":
    start_udp_server()
```

Client:

```
import socket
import time

STUDENT_ID = "6688052"
HOST = '127.0.0.1'
```

Name: Pannawit krutnak ID 6688052 Sec: 1

```
PORT = 1112

def start_udp_client():
    x = int(input("Enter first integer (x): "))
    y = int(input("Enter second integer (y): "))

    with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
        message = f"{x},{y}"
        s.sendto(message.encode(), (HOST, PORT))
        result, _ = s.recvfrom(1024)

        print(f"Student ID: {STUDENT_ID}")
        print(f"Current Date & Time: {time.asctime()}")
        print(f"Result of x*y: {result.decode()}")

if __name__ == "__main__":
    start_udp_client()
```

- Paste the screen output of server and client here. (Make sure your output looks like the provided example)

```
PS C:\Users\Student\Downloads\lab5\py> python .\udpserver.py
UDP server listening on 127.0.0.1:1112
Connected by 127.0.0.1:51825
Student ID: 6688052
Received x=3, y=4
```

-

```
PS C:\Users\Student\Downloads\lab5\py> python .\udpclient.py
Enter first integer (x): 3
Enter second integer (y): 4
Student ID: 6688052
Current Date & Time: Tue Apr 22 11:24:33 2025
Result of x*y: 12
PS C:\Users\Student\Downloads\lab5\py>
```

-