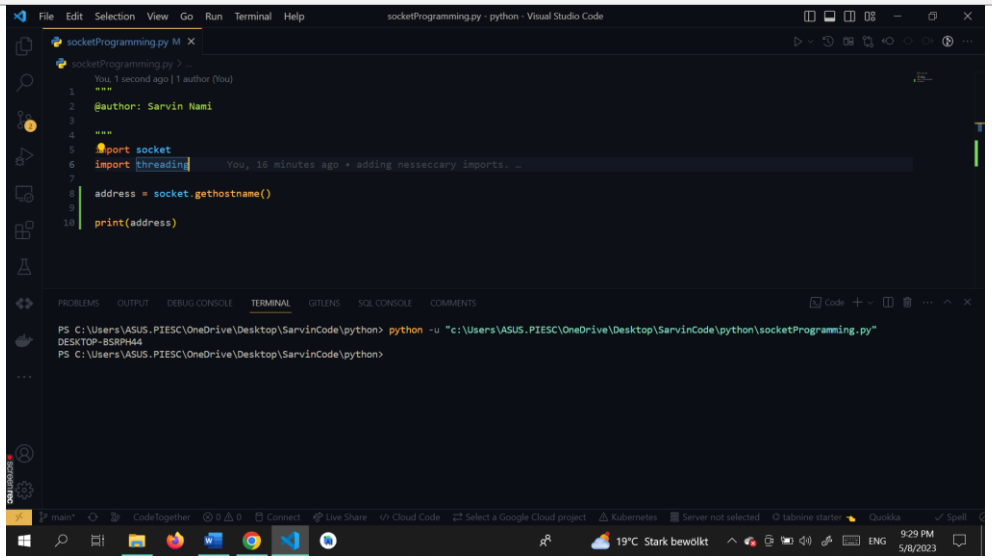
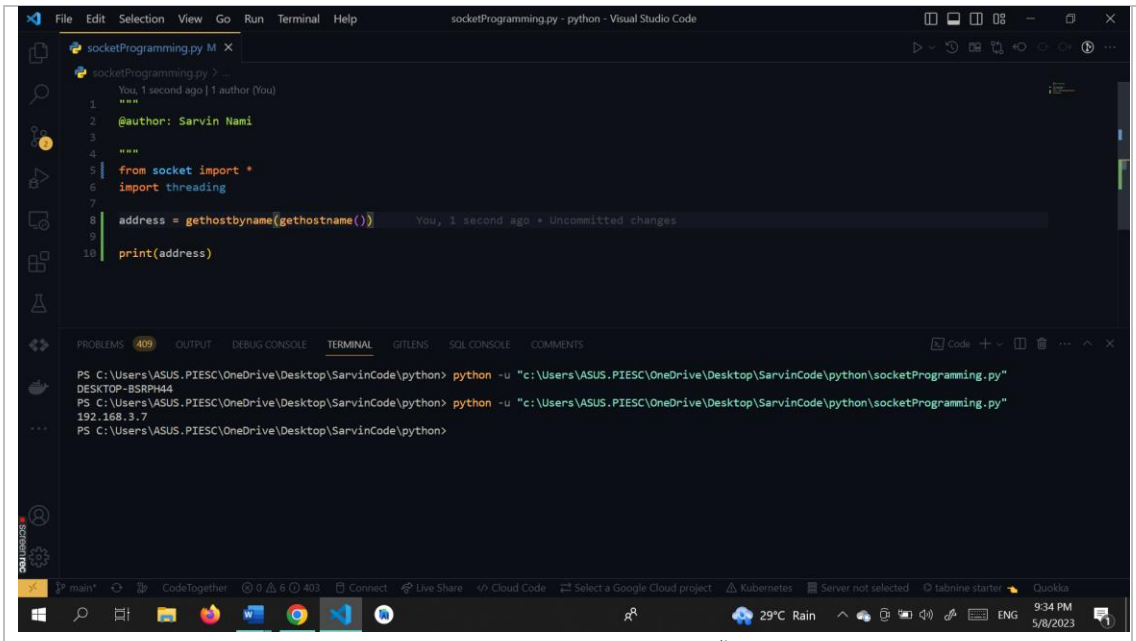
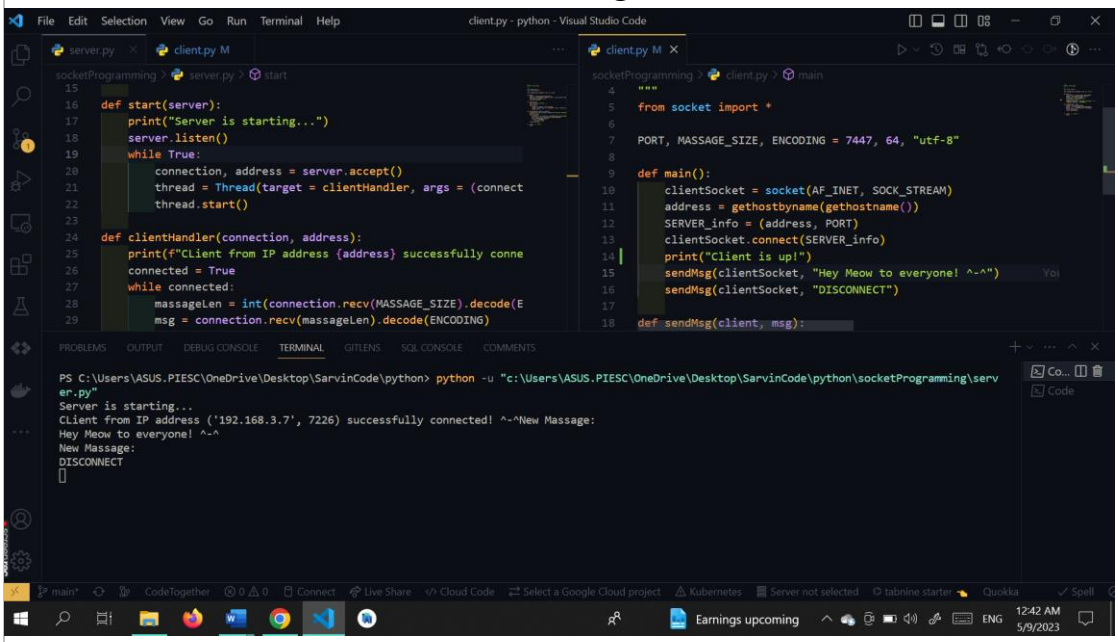
# فرم گزارش کار آزمایشگاه شبکه

| نام و نام خانوادگی | سروین نامی | شماره دانشجویی | 9931103 | نام و شماره آزمایش | 6 |
|---|---|---|---|---|---|
| | | | | | |
| هدف آزمایش | برقراری ارتباط tcp بین کلاینت و سرور نوشته شده. | | | | |
| ابزارهای مورد نیاز | لپ تاپ مجهز به پایتون | | | | |
| شرح آزمایش |  <br>در ابتدا نام دستگاه را چاپ کردیم. | | | | |

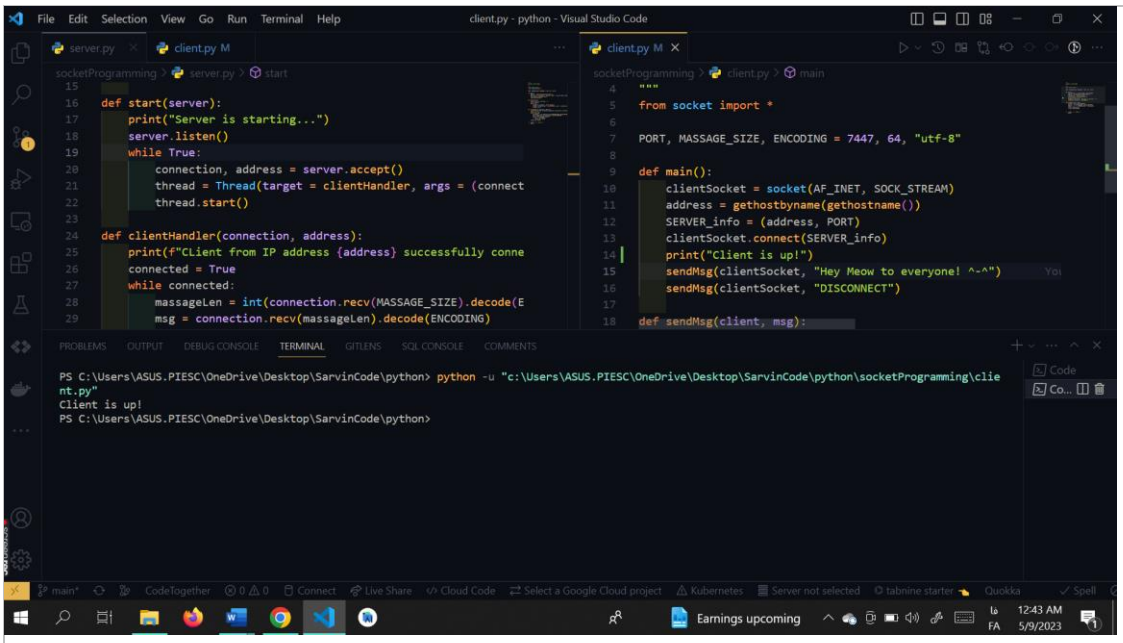



آدرس ip دستگاه من چاپ شد.

نتایج سمت سرور:




نتایج سمت کلاینت:




کد سمت سرور با کامنت برای توضیحات خط به خط:

```
"""
@author: Sarvin Nami

"""
from socket import *
from threading import *
# defining port number, our tcp massage size and our encoding mode.
PORT, MASSAGE_SIZE, ENCODING = 7447, 64, "utf-8"

def main():
    # the ip address of my laptop as server
    address = gethostbyname(gethostname())
    # defining host information and server socket
    HOST_info, serverSocket = (address, PORT), socket(AF_INET,
SOCK_STREAM)
    # binding the server socket via host info
    serverSocket.bind(HOST_info)
    # starting the server ^-^
    start(serverSocket)

def start(server):
    print("Server is starting…")
    # server is waiting for clients
    server.listen()
    while True:
        # every moment server is ready to accept connections from
clients
```

```
        connection, address = server.accept()
        # having a thread for each client
        thread = Thread(target = clientHandler, args = (connection,
address))
        thread.start()

# thread function
def clientHandler(connection, address):
    print(f"Client from IP address {address} successfully connected! ^-
^")
    connected = True
    while connected:
        # figuring out the length of out massage
        massageLen =
int(connection.recv(MASSAGE_SIZE).decode(ENCODING))
        # reading the massage to reach the defined length
        msg = connection.recv(massageLen).decode(ENCODING)
        print(f"New Massage:\n{msg}")
        # disconnect time T-T
        if msg == "DISCONNECT":
            connected = False
    # closing the connection
    connection.close()

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

کد سمت کلاینت با کامنت:

```
"""
@author: Sarvin Nami

"""
from socket import *

# defining basic setting as server
PORT, MASSAGE_SIZE, ENCODING = 7447, 64, "utf-8"

def main():
    # defining client socket
    clientSocket = socket(AF_INET, SOCK_STREAM)
    # the ip address of my laptop as client
    address = gethostbyname(gethostname())
    # defining server information
    SERVER_info = (address, PORT)
    # connecting client socket to the server
    clientSocket.connect(SERVER_info)
```

```
    print("Client is up!")
    # sending massages and ending the connection
    sendMsg(clientSocket, "Hey Meow to everyone! ^-^")
    sendMsg(clientSocket, "DISCONNECT")

def sendMsg(client, msg):
    # encoding the massage
    massage = msg.encode(ENCODING)
    # calculating massage length, encoding it, turning it to bytes and
sending it.
    msgLen = str(len(massage)).encode(ENCODING)
    msgLen += b' ' * (MASSAGE_SIZE - len(msgLen))
    client.send(msgLen)
    # sending the massage
    client.send(massage)


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

| | |
|---|---|
| به راحتی میتوان انواع ارتباطات برنامه های کاربردی را برای انواع انقالات tcp و udpتوسعه داد. | نتیجه-گیری |