NAME: SAKSHI CHHEDA
ROLL NO: 7
UID: 2018130005
BATCH: A
CLASS: T.E. COMPS

EXPERIMENT 8

AIM: To implement socket programming and creating a basic client-server model.

THEORY:
Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while another socket reaches out to the other to form a connection. The server forms the listener socket while the client reaches out to the server. They are the real backbones behind web browsing. In simpler terms, there is a server and a client.

A Simple Client-Server Program

1. Server

- Socket programming is started by importing the socket library and making a simple socket.
- Next, we made a socket instance and passed it two parameters. The first parameter is AF_INET and the second one is SOCK_STREAM. AF_INET refers to the address family ipv4. The SOCK_STREAM means connection oriented TCP protocol.
- Then we reserved a port on our pc, in this case we reserved port 8000.
- After that we binded our server to the specified port. A server has a bind() method which binds it to a specific ip and port so that it can listen to incoming requests on that ip and port.
- After that we put the server into listen mode. A server has a listen() method which puts the server into listen mode. This allows the server to listen to incoming connections. 5 here means that 5 connections are kept waiting if the server is busy and if a 6th socket trys to connect then the connection is refused.
- At last we make a while loop and start to accept all incoming connections using accept() method that initiates a connection with the client and at last send a thank you message to all connected sockets.

2. Client

- We start by importing the socket library and making a simple socket.
- Next, we made a socket instance and passed it two parameters. The first parameter is AF_INET and the second one is SOCK_STREAM. AF_INET refers to the address family ipv4. The SOCK_STREAM means connection oriented TCP protocol.
- Then we connect to port 8000, this is the port on which our server runs.
- Lastly we receive data from the server and print the received data.

CODE:

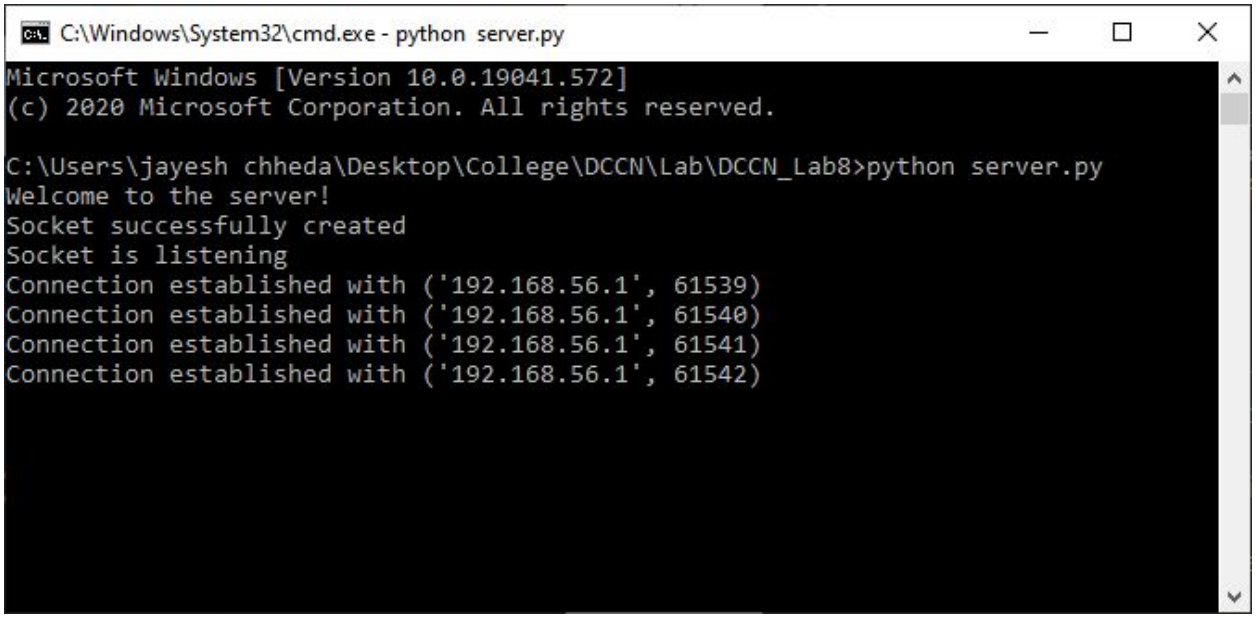
1. server.py

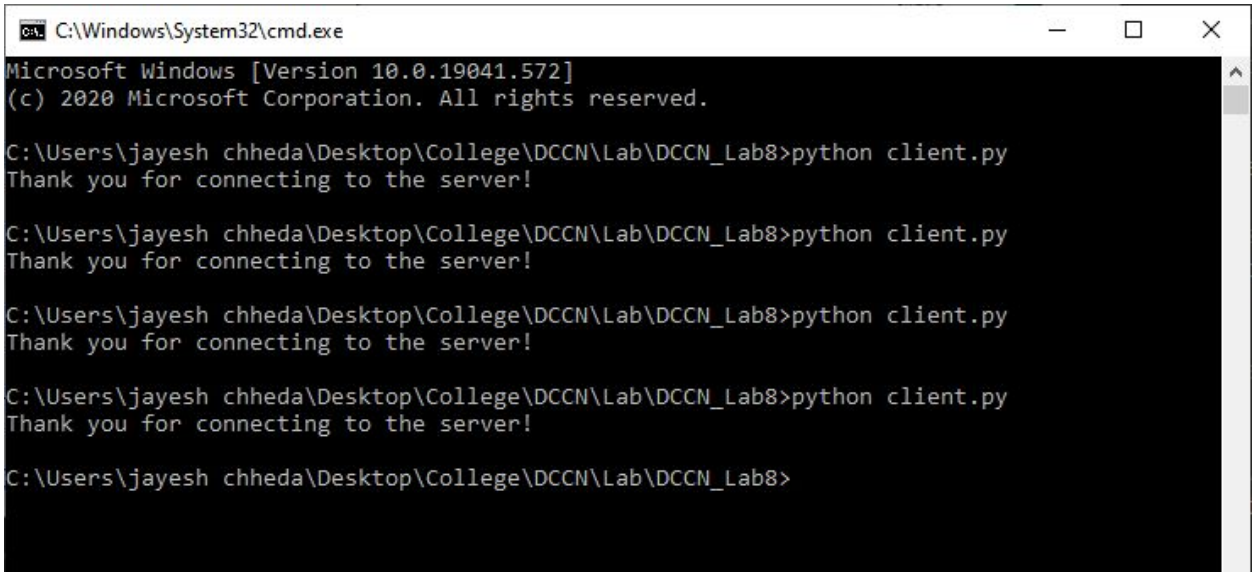
```
# Importing socket module
import socket

print('Welcome to the server!')

# Creating a socket object
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print("Socket successfully created")

# Reserved the port
port = 8000

# Binding server to the specified port
s.bind((socket.gethostname(), port))

# Putting the socket into listening mode
s.listen(5)
print("Socket is listening")

while True:
        # Establishing connection with client
        clientsocket, address = s.accept()
        print(f'Connection established with {address}')

        # Sending message to the client
        clientsocket.send(bytes('Thank you for connecting to the server!', 'utf-8'))
```

2. client.py

```
# Importing socket module
import socket

# Creating a socket object
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# Connecting to sever
s.connect((socket.gethostname(), 8000))

# Receiving data from the server
msg = s.recv(1024)

# Printing the received data
print(msg.decode('utf-8'))
```

OUTPUT :





CONCLUSION :

From this experiment, I understood the basics of socket programming and also learned how to implement a basic client-server model in python.