**Experiment No.8**

Code:

Server Side

import socket

if \_\_name\_\_ == '\_\_main\_\_':

# Defining Socket

host = '127.0.0.1'

port = 8080

totalclient = int(input('Enter number of clients: '))

sock = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

sock.bind((host, port))

sock.listen(totalclient)

# Establishing Connections

connections = []

print('Initiating clients')

for i in range(totalclient):

conn = sock.accept()

connections.append(conn)

print('Connected with client', i+1)

fileno = 0

idx = 0

for conn in connections:

# Receiving File Data

idx += 1

data = conn[0].recv(1024).decode()

if not data:

continue

# Creating a new file at server end and writing the data

filename = 'output'+str(fileno)+'.txt'

fileno = fileno+1

fo = open(filename, "w")

while data:

if not data:

break

else:

fo.write(data)

data = conn[0].recv(1024).decode()

print()

print('Receiving file from client', idx)

print()

print('Received successfully! New filename is:', filename)

fo.close()

# Closing all Connections

for conn in connections:

conn[0].close()

Client Side

import socket

# Creating Client Socket

if \_\_name\_\_ == '\_\_main\_\_':

host = '127.0.0.1'

port = 8080

sock = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

# Connecting with Server

sock.connect((host, port))

while True:

filename = input('Input filename you want to send: ')

try:

# Reading file and sending data to server

fi = open(filename, "r")

data = fi.read()

if not data:

break

while data:

sock.send(str(data).encode())

data = fi.read()

# File is closed after data is sent

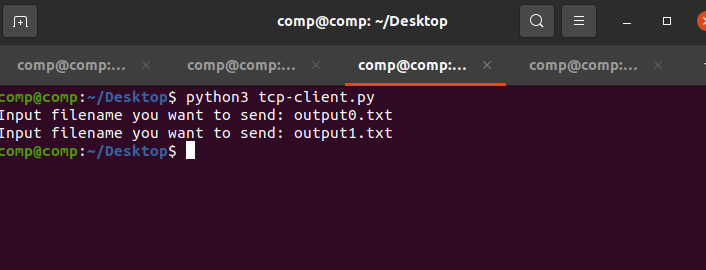
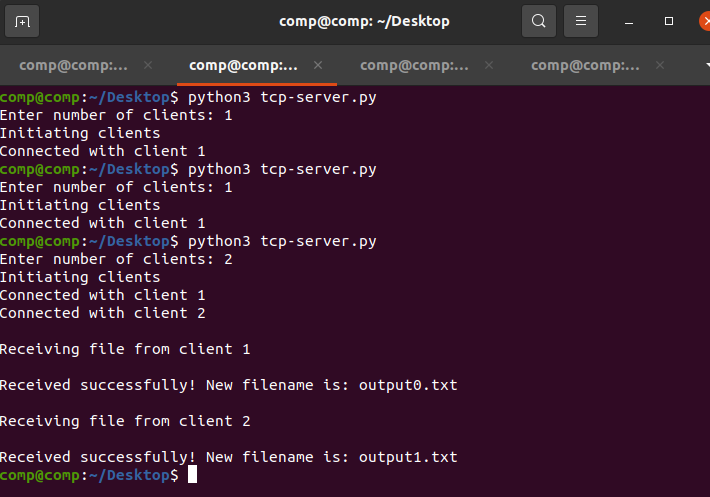
fi.close()

except IOError:

print('You entered an invalid filename!\

Please enter a valid name')

Output:



tcp-server.py

Open with

Files Created:

Output0.txt

First Line

Second Line

First Line

Second Line

Output1.txt

First Line

Second Line

First Line

Second Line