**CLIENT SIDE:**

from tkinter import \*

from socket import \*

import \_thread

# initialize server connection

def initialize\_client():

# initialize socket

s = socket(AF\_INET, SOCK\_STREAM)

# config details of server

host = '172.16.23.223' ## to use between devices in the same network eg.192.168.1.5

port = 80

# connect to server

s.connect((host, port))

return s

# update the chat log

def update\_chat(msg, state):

global chatlog

chatlog.config(state=NORMAL)

# update the message in the window

if state==0:

chatlog.insert(END, 'YOU: ' + msg)

else:

chatlog.insert(END, 'OTHER: ' + msg)

chatlog.config(state=DISABLED)

# show the latest messages

chatlog.yview(END)

# function to send message

def send():

global textbox

# get the message

msg = textbox.get("0.0", END)

# update the chatlog

update\_chat(msg, 0)

# send the message

s.send(msg.encode('ascii'))

textbox.delete("0.0", END)

# function to receive message

def receive():

while 1:

try:

data = s.recv(1024)

msg = data.decode('ascii')

if msg != "":

update\_chat(msg, 1)

except:

pass

def press(event):

send()

# GUI function

def GUI():

global chatlog

global textbox

# initialize tkinter object

gui = Tk()

# set title for the window

gui.title("Client Chat")

# set size for the window

gui.geometry("380x430")

# text space to display messages

chatlog = Text(gui, bg='white')

chatlog.config(state=DISABLED)

# button to send messages

sendbutton = Button(gui, bg='orange', fg='red', text='SEND', command=send)

# textbox to type messages

textbox = Text(gui, bg='white')

# place the components in the window

chatlog.place(x=6, y=6, height=386, width=370)

textbox.place(x=6, y=401, height=20, width=265)

sendbutton.place(x=300, y=401, height=20, width=50)

# bind textbox to use ENTER Key

textbox.bind("<KeyRelease-Return>", press)

# create thread to capture messages continuously

\_thread.start\_new\_thread(receive, ())

# to keep the window in loop

gui.mainloop()

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

chatlog = textbox = None

s = initialize\_client()

GUI()

**SERVER SIDE:**

from tkinter import \*

from socket import \*

import \_thread

# initialize server connection

def initialize\_server():

# initialize socket

s = socket(AF\_INET, SOCK\_STREAM)

# config details of server

host = '172.16.37.128' ## to use between devices in the same network eg.192.168.1.5

port = 80

# initialize server

s.bind((host, port))

# set no. of clients

s.listen(1)

# accept the connection from client

conn, addr = s.accept()

return conn

# update the chat log

def update\_chat(msg, state):

global chatlog

chatlog.config(state=NORMAL)

# update the message in the window

if state==0:

chatlog.insert(END, 'YOU: ' + msg)

else:

chatlog.insert(END, 'OTHER: ' + msg)

chatlog.config(state=DISABLED)

# show the latest messages

chatlog.yview(END)

# function to send message

def send():

global textbox

# get the message

msg = textbox.get("0.0", END)

# update the chatlog

update\_chat(msg, 0)

# send the message

conn.send(msg.encode('ascii'))

textbox.delete("0.0", END)

# function to receive message

def receive():

while 1:

try:

data = conn.recv(1024)

msg = data.decode('ascii')

if msg != "":

update\_chat(msg, 1)

except:

pass

def press(event):

send()

# GUI function

def GUI():

global chatlog

global textbox

# initialize tkinter object

gui = Tk()

# set title for the window

gui.title("Server Chat")

# set size for the window

gui.geometry("380x430")

# text space to display messages

chatlog = Text(gui, bg='white')

chatlog.config(state=DISABLED)

# button to send messages

sendbutton = Button(gui, bg='orange', fg='red', text='SEND', command=send)

# textbox to type messages

textbox = Text(gui, bg='white')

# place the components in the window

chatlog.place(x=6, y=6, height=386, width=370)

textbox.place(x=6, y=401, height=20, width=265)

sendbutton.place(x=300, y=401, height=20, width=50)

# bind textbox to use ENTER Key

textbox.bind("<KeyRelease-Return>", press)

# create thread to capture messages continuously

\_thread.start\_new\_thread(receive, ())

# to keep the window in loop

gui.mainloop()

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

chatlog = textbox = None

conn = initialize\_server()

GUI()