

```

# GUI_chatclient.py
# GUI 채팅 클라이언트

from socket import *
from tkinter import *
from tkinter.scrolledtext import ScrolledText
from threading import *

class ChatClient:
    client_socket = None

    def __init__(self, ip, port):
        self.initialize_socket(ip, port)
        self.initialize_gui()
        self.listen_thread()

    #TCP socket을 생성하고 server에게 연결
    def initialize_socket(self, ip, port):
        self.client_socket = socket(AF_INET, SOCK_STREAM)
        remote_ip = ip
        remote_port = port
        self.client_socket.connect((remote_ip, remote_port))

    #message를 전송하는 callback 함수
    def send_chat(self):
        #송신자 이름
        senders_name = self.name_widget.get().strip() + ":"
        #송신 메시지 창에서 메시지를 읽어 온다
        data = self.enter_text_widget.get(1.0, 'end').strip()
        #송신자 이름과 메시지를 결합
        message = (senders_name + data).encode('utf-8')
        #전송 메시지를 자신의 수신 메시지 창에도 표시
        self.chat_transcript_area.insert('end',
                                         message.decode('utf-8') + '\n')
        self.chat_transcript_area.yview(END)
        #메시지 전송
        self.client_socket.send(message)
        #송신 메시지 창을 비운다
        self.enter_text_widget.delete(1.0, 'end')
        return 'break'

```

```

#위젯을 배치하고 초기화한다
def initialize_gui(self):
    self.root = Tk()
    fr = []
    #5개의 프레임 생성
    for i in range(0,5):
        fr.append(Frame(self.root))
        fr[i].pack(fill=BOTH)

    #화면 라벨
    self.name_label = Label(fr[0], text='사용자 이름')
    self.recv_label = Label(fr[1], text = '수신 메시지:')
    self.send_label = Label(fr[3], text = '송신 메시지:')
    #[전송] 버튼
    self.send_btn = Button(fr[3], text='전송', command=self.send_chat)
    #수신 메시지 창
    self.chat_transcript_area = ScrolledText(fr[2], height =20, width=60)
    #송신 메시지 창
    self.enter_text_widget = ScrolledText(fr[4], height =5, width=60)
    #사용자 이름 입력 창
    self.name_widget = Entry(fr[0], width =15)

    #위젯 배치
    self.name_label.pack(side=LEFT)
    self.name_widget.pack(side=LEFT)
    self.recv_label.pack(side=LEFT)
    self.send_btn.pack(side=RIGHT, padx=20)
    self.chat_transcript_area.pack(side=LEFT, padx=2, pady=2)
    self.send_label.pack(side=LEFT)
    self.enter_text_widget.pack(side=LEFT, padx=2, pady=2)

    #Thread를 생성하고 채팅을 시작한다
def listen_thread(self):
    t = Thread(target=self.receive_message, args=(self.client_socket,))
    t.start()

#Server로부터 message를 수신하고 수신 메시지 창에 표시한다
def receive_message(self, so):
    while True:
        buf = so.recv(256)
        if not buf:

```

```

        break
    #수신 메시지 창에 표시
    self.chat_transcript_area.insert('end',buf.decode('utf-8') + '\n')
    #화면을 아래로 이동
    self.chat_transcript_area.yview(END)
so.close()

if __name__ == "__main__":
    ip = input("server IP addr: ")
    if ip == '':
        ip = '127.0.0.1'
    port = 2500
    ChatClient(ip, port)
    mainloop()

```