import socket

import threading

def handle\_client(client\_socket, client\_address):

while True:

message = client\_socket.recv(1024).decode('utf-8')

if message == "exit":

break

broadcast\_message(message, client\_socket)

print(f"Client {client\_address} disconnected.")

client\_socket.close()

def broadcast\_message(message, sender\_socket):

for client\_socket in clients:

if client\_socket != sender\_socket:

client\_socket.send(message.encode('utf-8'))

def start\_server():

server\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server\_socket.bind(('localhost', 8888))

server\_socket.listen(5)

print("Server started. Listening for connections...")

while True:

client\_socket, client\_address = server\_socket.accept()

clients.append(client\_socket)

print(f"Client {client\_address} connected.")

client\_thread = threading.Thread(target=handle\_client, args=(client\_socket, client\_address))

client\_thread.start()

clients = []

start\_server()