import socket

import sys

import threading

def read\_config():

with open("config.txt", "r") as file:

host, port = file.read().splitlines()

return host, int(port)

def chat\_server():

host, port = read\_config()

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

server\_socket.bind((host, port))

server\_socket.listen(1)

print(f"Servidor ouvindo em {host}:{port}...")

conn, addr = server\_socket.accept()

print(f"Conexão estabelecida com {addr}")

def receive\_messages():

while True:

msg = conn.recv(1024).decode()

if not msg:

break

print(f"Cliente: {msg}")

threading.Thread(target=receive\_messages, daemon=True).start()

while True:

conn.send(input("Servidor: ").encode())

def chat\_client():

host, port = read\_config()

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

client\_socket.connect((host, port))

print(f"Conectado ao servidor {host}:{port}")

def receive\_messages():

while True:

msg = client\_socket.recv(1024).decode()

if not msg:

break

print(f"Servidor: {msg}")

threading.Thread(target=receive\_messages, daemon=True).start()

while True:

client\_socket.send(input("Cliente: ").encode())

if \_\_name\_\_ == "\_\_main\_\_":

if len(sys.argv) < 2:

print("Uso: python script.py server|client")

sys.exit(1)

role = sys.argv[1].lower()

if role == "server":

chat\_server()

elif role == "client":

chat\_client()

else:

print("Opção inválida! Use 'server' ou 'client'")