

Cognome	BRAVO
Nome	STUDENTE
Matricola	

Il codice sotto riportato rappresenta un'applicazione Client-Server che fornisce informazioni meteorologiche. L'applicazione richiede che l'utente prima specifichi la città di riferimento nel formato corretto (es. Rome,it) e successivamente richieda una delle misure meteorologiche disponibili. Per fornire i dati il Server utilizza la funzione *'EstraiMeteo'*, che prendendo in input una città e un parametro ritorna il dato numerico corrispondente sotto forma di stringa. Si noti che dopo ogni richiesta l'utente può scegliere se richiedere un nuovo dato per una nuova città a sua scelta oppure chiudere la sessione.

### HINT:

Il codice seguente:

```
nome = Andrea
print(f'Hello {nome}')
```

stampa  
'Hello Andrea'

### Quesiti (6 pt):

**Q1** Completare il codice mancante nel Server e nel Client in maniera tale che:

1. Il server sia in ascolto sulla porta 12001 (0.5 pt)
2. Il servizio di trasporto utilizzato sia TCP (1.5 pt)

**Q2 (1 pt)** Il Server riesce a gestire client multipli (SI/NO)? Si motivi la risposta.

**Si, il Server è di tipo Multi-Threading.**

**Q3 (1 pt)** Che protocollo utilizza il Server per ottenere i dati meteorologici? Si motivi la risposta.

**HTTP, tramite la funzione GET dalla libreria requests.**

**Q4 (2 pt)** Si riporti ciò che appare sul terminale client nella situazione in cui il client voglia conoscere la temperatura presente a Palermo, supponendo che a Palermo si registri 31° al momento della richiesta e che il client voglia effettuare una sola richiesta.

**Benvenuto nel sistema meteo di FCI 2022**  
**Inserisci una città a tua scelta (es. Rome,it):Palermo,it**  
**Scegli una misura meteo (temp, pressure, humidity):temp**  
**Dal Server Meteo: 31°**  
**Si vuole procedere con una nuova richiesta (si/no)?no**  
**Arrivederci!**

## Client

```
from socket import *

serverName = 'localhost'
serverPort = 12001 (Q1.1)

clientSocket = socket(AF_INET, SOCK_STREAM) (Q1.2)
clientSocket.connect((serverName, serverPort))

clientSocket.settimeout(5)

print('Benvenuto nel sistema meteo di FCI 2022')

while True:
    msg = input('Inserisci una città a tua scelta (es. Rome,it):')
    clientSocket.send(msg.encode('utf-8'))

    try:
        feedback = clientSocket.recv(2048)
        feedback = feedback.decode('utf-8')

        if feedback == 'OK':
            msg = input('Scegli una misura meteo (temp, pressure, humidity):')
            clientSocket.send(msg.encode('utf-8'))

            meteo = clientSocket.recv(2048)
            meteo = meteo.decode('utf-8')

            print(f'Dal Server Meteo: {meteo}')

            close = input('Si vuole procedere con una nuova richiesta (si/no)?')
            clientSocket.send(close.encode('utf-8'))

            if close == 'no':
                print('Arrivederci!')
                break
    except:
        print('Timeout scaduto: chiusura sessione')
        break

clientSocket.close()
```

# Server

```
import requests
import json
from socket import *
from threading import Thread

def handler(connectionSocket):
    while True:

        citta = connectionSocket.recv(2048)
        connectionSocket.send('OK'.encode('utf-8'))

        par = connectionSocket.recv(2048)

        meteo = EstraiMeteo(citta.decode('utf-8'), par.decode('utf-8'))
        connectionSocket.send(meteo.encode('utf-8'))

        contd = connectionSocket.recv(2048)
        if contd.decode('utf-8') == 'no'
            break

    connectionSocket.close()

def EstraiMeteo(citta, parametro):

    get_text = f'http://api.openweathermap.org/data/2.5/weather?q={citta}&APPID=<appkeyserver>'
    r = requests.get(get_text)

    dict = json.loads(r.text)['main']
    par = dict[parametro]

    return str(par)

serverName = 'localhost'
serverPort = 12001 (Q1.1)
serverSocket = socket(AF_INET, SOCK_STREAM) (Q1.2)
serverSocket.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
serverSocket.bind(("", serverPort))
serverSocket.listen(1)

while 1:
    print('Server Pronto')

    wSocket, addr = serverSocket.accept() (Q1.2)
    thread = Thread(target=handler, args=(wSocket,))
    thread.start()
```