

Curso	Engenharia da Computação	Período	9A
Disciplina	Sistemas Computacionais Distribuídos	Data	18/04/2024
Professor	Prof. Marcílio F. Oliveira Neto		
EXERCÍCIO PARA ENTREGAR			

**Para cada exercício, crie os scripts necessários e capture um print da tela durante a execução para anexar na atividade.**

1) Dado o fragmento de código a seguir, faça:

```
import socket as sc

hots = '127.0.0.50'
port = 8585

servidor = sc.socket(family=sc.AF_INET, type=sc.SOCK_DGRAM)
```

- Um script, utilizando o código acima, que permita o servidor receber a conexão de um cliente a fim de trocar mensagens. O servidor deverá enviar a mensagem “Conectado” logo no início, e a cada mensagem recebida, retornar “Ok”.
- Um cliente para se conectar com o servidor criado. As mensagens do cliente serão inseridas via teclado. O cliente deverá executar até que o comando “sair” seja inserido.

2) Sobre soquetes TCP, faça:

- Faça, com base no script CLIENTE abaixo, uma implementação que troca mensagens com um servidor. Exiba as mensagens recebidas na tela.
- Faça, com base no script SERVIDOR abaixo, uma implementação que troca mensagens com o cliente construído.

CLIENTE	SERVIDOR
<pre>import socket, threading as td  host = 'localhost' port = 7072  #função responsável por receber mensagens do servidor, #dado a conexão com ele, e exibir na tela a mensagem def recebe_msg(____):     while True:         _____         _____         _____         _____  #função responsável por enviar mensagens ao servidor, #dado a conexão com ele def envia_msg(____):     while True:         _____         _____         _____         _____  #instancie o soquete e faça a conexão com o servidor, #após isso, inicie a execução das threads para executar #as funções acima. _____ _____ _____ _____ _____</pre>	<pre>import socket as sc import threading as td  host = 'localhost' port = 8888  #instancie o soquete e execute as funções adequadas para ouvir #conexões _____ _____ _____  def trata_msg(____):     try:         while True:             #receber e enviar dados do cliente             _____             _____             _____             _____         except:             _____.close()  while True:     #tratar o aceite da conexão e executar a função na thread     try:         _____         _____         _____         _____     except:         _____.close()</pre>

- 3) Crie um servidor em Flask que:
- Tenha um endpoint capaz de receber um json e salvar em uma variável global interna do servidor.
  - Tenha um endpoint capaz de retornar todos os itens da variável global do servidor.
  - Tenha um endpoint capaz de filtrar e retornar um único item contido na lista global, sob a chave "dados".
  - Construa um cliente, utilizando o módulo requests, que interaja com o servidor.

A estrutura do json é dada abaixo:

```
{
  "total":200,
  "url": "http://api.engcomp2024.com",
  "dados":
  [
    {
      "relevante" : "Conteudo 1"
    },
    {
      "relevante": "Conteudo 2"
    },
  ]
}
```