1- O que é um processo cliente?

Res. : Processo cliente é como chamamos o processo que é responsável pelo lado cliente, ou seja será o processo que irá se conectar com o servidor.

2 - O que é um processo servidor?

Res. : Processo servidor é como o próprio nome já sugere o processo responsável por ser o server, ou seja ele ficara escutado na sua porta por clientes que desejam se conectar a ele.

3 - A função socket() do módulo ‘socket’ de Python é responsável por criar um socket no processo tanto para protocolo TCP, quanto UDP. Como diferenciar se o socket a ser criado é TCP e UDP?

Res. : Existe diferença na hora de criar o socket no código :

s = socket.socket (socket\_family, socket\_type) - No socket type para tcp usamos SOCK\_STREAM e para udp usamos SOCK\_DGRAM

4 - Para sockets TCP, responda:

1. Que sequência de chamadas de funções em Python deve ser realizada pelo cliente? (Não precisa especificar os parâmetros)

Res. : socket() -> connect() -> recv() send() -> close()

1. Que sequência de chamadas de funções em Python deve ser realizada pelo servidor? (Não precisa especificar os parâmetros)

Res. : socket() -> bind() -> listen() -> accept -> send() recv() -> close()

1. Quais destas funções são bloqueantes, isto é, o processo fica esperando?

Res. : connect(), recv() send() ,listen()

5 - Para sockets UDP, responda:

1. Que sequência de chamadas de funções em Python deve ser realizada pelo cliente? (Não precisa especificar os parâmetros)

Res. : socket() -> sendto() recvfrom() -> close()

1. Que sequência de chamadas de funções em Python deve ser realizada pelo servidor? (Não precisa especificar os parâmetros)

Res. : socket() -> bind() -> recvfrom() sendto() -> close()

1. Quais destas funções são bloqueantes, isto é, o processo fica esperando?

Res. : Nenhuma

6 - Para que serve o comando socket.bind()?

Res. : associa o socket a um endereço e porta locais. Em Python, é necessário no lado servidor.

7 - Em sockets Python, como é representado um endereço de um processo remoto?

Res. :

Pelo IP e pela porta