Lab 5 Network Programming

Monday, March 11, 2019 12:05 PM

Concurrency

What's the thread???

It's a lightweight process.

What's multi-threading???

- It's a mechanism allows the usage of the available advances of hardware.
- It's the advance in software that meets the advance in hardware.

- In the previous programs ===> the whole server is reserved for only one client.
- now we want to give only a thread for the client, that thread serves that client.
  - one thread for sending,
  - one for receiving.
    - We'll create two threads for each client as if there's only one thread for a client, there'll be a problem of no multiple messages to be sent in a row.

SERVER:-

```
# -*- coding: utf-8 -*-
"""
Created on Mon Mar 18 12:17:08 2019

@author: M7md_Karam
"""

from _thread import *
import threading
from socket import *

def client_thread(c): #===> we'll create another thread within this function
    #create a new thread, using a detailed way with same parameters.
    receive=threading.Thread(target= receive_thread, arg=(c,))
    #starting the thread
    receive.start()
    while True:
        c.send(input("server: ").encode('utf-8'))

def receive_thread(c):
   while True:
      x = c.recv(500)
      print(x.decode('UTF-8'))

s=socket(AF_INET, SOCK_STREAM)
host = '127.0.0.1'
port = 7000
s.bind ((host, port))
s.listen(5)
while True:
    c, add=s.accept()
    print("connection from", add[0])
    #create thread for serving that session number.
    #start_new_thread("function", "the arguments of the function") ===>
    that's a shortcut, that may cause some problems.
    start_new_thread(client_thread,(c)) ===> second parameter is a tuple.
    #receive.join() is a method for closing the thread before exiting the
    program.
```

Client:

```
# -*- coding: utf-8 -*-
"""
Created on Mon Mar 18 12:54:51 2019

@author: M7md_Karam
"""
from socket import *
import threading

def receive_thread(s):
    while True:
        x=s.recv(500)
        print(x.decode('utf-8'))

s=socket(AF_INET, SOCK_STREAM)
host = '127.0.0.1'
port = 7000
s.connect((host,port))
receive=threading.Thread(target=receive_thread,args=(s,))
receive.start()
while True:
    s.send(input("client: ").encode('utf-8'))
```