Lab session 1 (Part II)

Target set of this session:

-to make distinction between

service, interface, and protocol

- To implement a connection-oriented service
- To use the socket library available in Python.

the socket library available in Python

The main functions are:

- socket(): to create an object representing the connection
- accept(): a blocking call to wait for incoming connection requests; if successful, the call returns a new socket for a separate connection
- connect(): to set up a connection to a specified party
- close(): to tear down a connection
- send(), recv(): to send and receive data over a connection, respectively

To Do

Implement a service that allows two communicating parties (a client and a server) to reliably send and receive data over a connection.

Hints

Server

```
1 from socket  import *
2 s = socket(AF_INET, SOCK_STREAM)
3 (conn, addr) = s.accept()  # returns new socket and addr. client
4 while True:                # forever
5   data = conn.recv(1024)   # receive data from client
6   if not data: break       # stop if client stopped
7   conn.send(str(data)+"*") # return sent data plus an "*"
8 conn.close()               # close the connection
```

Client

```
1 from socket  import *
2 s = socket(AF_INET, SOCK_STREAM)
3 s.connect((HOST, PORT)) # connect to server (block until accepted)
4 s.send('Hello, world')  # send some data
5 data = s.recv(1024)     # receive the response
6 print data               # print the result
7 s.close()               # close the connection
```