Programming with Python sockets modules: Do the following simple tasks?

[1] Printing your machine's name and IPv4 address?

[2] Retrieve a remote machine's IP address and convert the IP address to different format?

[3] Setting and getting the default socket timeout, the program should include how to handle the socket error gracefully?

[4] Finding the service name, given the port and protocol of the remote host (server)?

[5] Printing the current time from the internet time server with the help of NTP? Also write an SNTP client that prints the current time from the internet time server received with the SNTP protocol?

[6] Modifying sockets send/receive buffer size and changing the socket to blocking/non-blocking mode?

[7] Write a program that demonstrates the reuse socket addresses?

[8] Write a simple TCP echo client/server application with the help of TCP socket object. The server wait for the client to be connected and send some data to the server. When the data is received, the server echoes the data to the client.

[9] Write a simple UDP echo client/server application with the help of TCP socket object. The server wait for the client to be connected and send some data to the server. When the data is received, the server echoes the data to the client.

[10] Write a program that is a TCP server that returns a HTTP response to a browser that displays the client's IP address and the number of times it has connected to the server. Test your program with a standard Web browser like the Internet Explorer.

Q No. [8] In details

Write a TCP client and server in the python. When the server is run, it establishes a socket on which it listens for a TCP connection request. When the client is run, it sends a TCP connection request to the server. The server accepts the connection request, and waits to receive a message. The client sends a text string for a message, and then waits to receive a response from the server. The server responds with a text string and then closes the socket connection.

Simple TCP Server

When the server executes, it establishes a socket and listens for a TCP connection request. When it receives a connection request, it sends a text string and then terminates. Both the port number and the text string message are specified on the command line.

Simple TCP Client

When the client runs, it establishes a socket and sends a connection request message to the IP address of a running server. After connecting to the server, it reads a stream of text data from the socket until it reaches a new line character. It prints the text and terminates. Both the server's port number and IP address are specified on the command line.