1) In Python we use the socket module. To create a TCP socket we need to create a new instance of the socket class with AF\_INET (IPv4) and SOCK\_STREAM (TCP). Then we use the connect method to connect to it and can use methods such as send to send data or recv to receive data over the TCP server.

2) A client socket is used to initiate a connection to a server socket because it sends a request to the server and waits for a response. While a server socket waits for incoming connections from client sockets, and then when the client socket connects to a server socket the server socket will create a new socket object to handle the connection. So in theory they work together with one handling the client and one handling the server.

3) In my code I had a line that directed the OS to let us reuse the same bind port, it was this line of code: s.setsockopt(socket.SOL\_SOCKET, socket.SO\_REUSEADDR, 1). So by setting the SO\_REUSEADDR to 1 (true) we tell the OS to let us reuse the bind port.

4) We get information about the incoming connection to the server socket. It specifically prints out the IP address and port number from the client that connected to the server.

5) The recv method is used for receiving data from the client socket. When the server is done sending HTTP requests it waits for a response from the client where the recv method blocks until the data is received from the client or the connection is closed.

6) https://github.com/lukewismer/CMPUT404-Lab2