

Functions Explanation

1. server.py

1. `handle_client (conn,addr)`
 - this function is created to handle all communication between client and server. It can handle individual connection between client and server.
2. `start()`
 - this function will allow server to start listening for connection. It handles new connection and distribute those to where they need to go. When a connection happens, the handling of those connection will be passed to handle client in a new thread.
 - the while block in this function is set to repeat on loop until we want to stop listening for new connection or when the server came to a crash.
3. `bind()`
 - bind is used to associate socket with the server address. In this program, the address is localhost (the current server) and the port number is 12345.
4. `listen ()`
 - `listen()` is used to put the socket into server mode.
5. `accepts()`
 - `accepts()` is for waiting for an incoming connection. Also, it returns an open connection between the server and client.
6. `close()`
 - after the communication with a client is finished, connection is cleared using `close()`.

2. client.py

1. `send()`
 - After the connection is established using `connect()`, data can be sent through the socket with `send()` and received with `recv()`.
2. `randomMessage()`
 - Generate a random word each time client.py is run to provide a variety of words in the program.

Name: Iman Khaleda binti Zamri (이만)

Student ID: 2021079443

Running the program

1. Run server.py
 - a. VSCode terminal will output that the server is starting and is listening for connection on the localhost.

```
User/Desktop/socket programming/server.py"
[STARTING] server is starting...
[LISTENING] Server is listening on 192.168.0.15
```

2. Run client.py
 - a. Open command prompt on your laptop and paste the path at which the program's folder is stored. In this case it is at Desktop in a folder named "socket programming". Input "python client.py" as shown below.

```
C:\Users\User\Desktop\socket programming>python client.py
```

Program Outputs

3. Open VSCode to see that the terminal will output something like this.

```
User/Desktop/socket programming/server.py"
[STARTING] server is starting...
[LISTENING] Server is listening on 192.168.0.15
[NEW CONNECTION] ('192.168.0.15', 59655) connected.
[ACTIVE CONNECTIONS] 1
[('192.168.0.15', 59655)] apple
[('192.168.0.15', 59655)] popcorn
[('192.168.0.15', 59655)] apple
[('192.168.0.15', 59655)] DISCONNECT!
[NEW CONNECTION] ('192.168.0.15', 59658) connected.
[ACTIVE CONNECTIONS] 1
[('192.168.0.15', 59658)] hello
[('192.168.0.15', 59658)] mandarin
[('192.168.0.15', 59658)] jar
[('192.168.0.15', 59658)] DISCONNECT!
[NEW CONNECTION] ('192.168.0.15', 59659) connected.
[ACTIVE CONNECTIONS] 1
[('192.168.0.15', 59659)] lemon
[('192.168.0.15', 59659)] doll
[('192.168.0.15', 59659)] orange
[('192.168.0.15', 59659)] DISCONNECT!
```

4.
 - a. While the server host for connection will remain the same (the localhost), the address will change.
 - i. Address: store IP address and the port it came from of the new connection. In this case address position is at the highlighted part.
 1. [NEW CONNECTION] ('192.168.0.15', 57266) connected.
 - b. When all messages have been sent and received, the connection will be closed and the disconnect message will be sent.

Name: Iman Khaleda binti Zamri (이만)

Student ID: 2021079443

- i. `[('192.168.0.15', 57266)] DISCONNECT!`
 - c. Active connections tells us the number of connections that are active at one time
 - i. `[ACTIVE CONNECTIONS] 1`
5. In relation to no 3, when the VSCode terminal output all the messages, the command prompt will output that the message has been received.

```
C:\Users\User>cd Desktop\socket programming

C:\Users\User\Desktop\socket programming>python client.py
Message received
Message received
Message received
Message received

C:\Users\User\Desktop\socket programming>python client.py
Message received
Message received
Message received
Message received

C:\Users\User\Desktop\socket programming>python client.py
Message received
Message received
Message received
Message received
```