WEEK 9
SOCKET SERVER DAN CLIENT

Arranged By:

Rajendra Rakha Arya Prabaswara
(1941720080/20)

PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

PRACTICUM 1

1. Create the server.py using nano or anotherone, just copy paste it from jbbsheet

```
 GNU nano 2.9.3                                server.py

import socket
from threading import Thread


# Multithreaded Python server
class ClientThread(Thread):

    def __init__(self, ip, port):
        Thread.__init__(self)
        self.ip = ip
        self.port = port
        print("Incoming connection from " + ip + ":" + str(port))

    def run(self):
        while True:
            try:
                data = conn.recv(2048)
                if len(data) == 0:
                    break

                print("length: " + str(len(data)))
                print("Server received data:", data)
                # MESSAGE = input("Input response:")
                MESSAGE = "OK"
                conn.send(MESSAGE.encode("utf8"))  # echo
            except Exception as e:
                print(e)
                break


TCP_IP = "0.0.0.0"
TCP_PORT = 1886
BUFFER_SIZE = 20

tcpServer = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
tcpServer.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
tcpServer.bind((TCP_IP, TCP_PORT))
threads = []

while True:
    tcpServer.listen(4)
    print("Server started on " + TCP_IP + " port " + str(TCP_PORT))
    (conn, (ip, port)) = tcpServer.accept()
    newthread = ClientThread(ip, port)
    newthread.start()
    threads.append(newthread)

for t in threads:
    t.join()

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo  M-A Mark Text
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Linter ^_ Go To Line M-E Redo  M-6 Copy Text
```

2. Change the file permission of server.py to make it excutable using chmod +x

```
-rwxr-xr-x 1 root  root       1267 May 17 03:43 server.py
```

3. Execute server.py

```
Server started on 0.0.0.0 port 1886
```

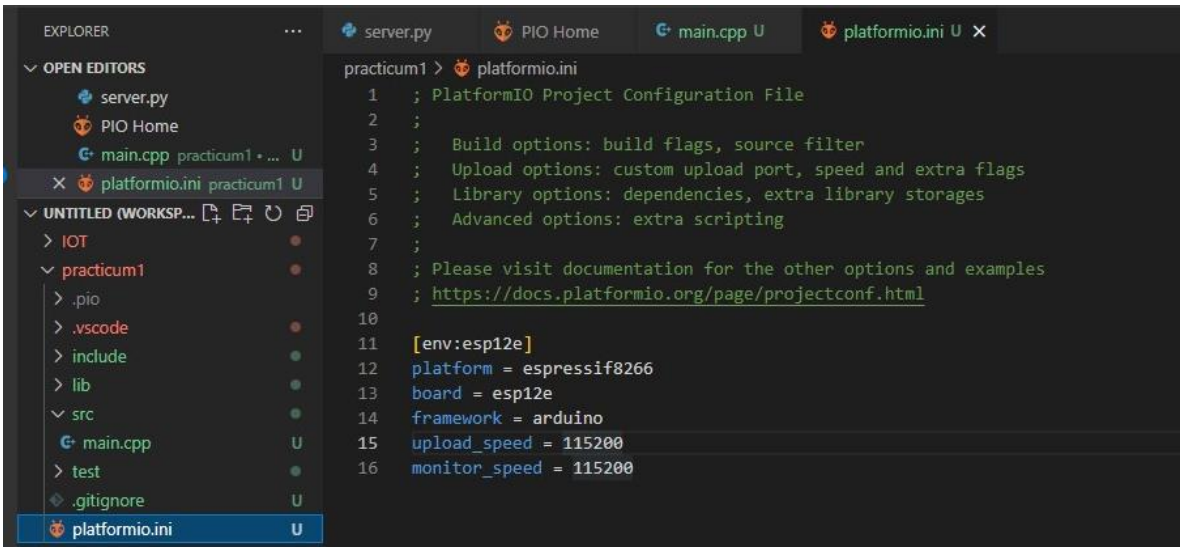
4. Test Connection by sending message

```
Server started on 0.0.0.0 port 1886
Incoming connection from 192.168.3.25:10593
Server started on 0.0.0.0 port 1886
length: 1
Server received data: b'h'
length: 1
Server received data: b'a'
length: 1
Server received data: b'l'
length: 1
Server received data: b'l'
length: 1
Server received data: b'o'
length: 1
Server received data: b'a'
length: 1
Server received data: b'b'
length: 1
Server received data: b'd'
length: 1
Server received data: b'u'
length: 1
Server received data: b'l'
```

```
Telnet 192.168.3.247
OKaOKlOKlOKoOKaOKbOKdOKuOKlOK
```

After successfully running socket sever, then it is necessary to create a socket client that runs on the controller or ESP8266 Amica or Lolita that you have. Create the following code To be able to communicate with the server socket, ESP8266 already has a wifi module ready to use. Change the code below to suit your needs.

5. Change the Platformio.in setting



6. Copy & Paste From Example Code

7. Build Code And Upload to NodeMCU

```
10
11  void connect_wifi()
12  {
13    Serial.printf("Connecting to %s ", ssid);
14    WiFi.begin(ssid, password);
15    while (WiFi.status() != WL_CONNECTED)
16    {
17      delay(500);
18      Serial.print(".");
19    }
20    Serial.println(" connected");
21  }
```

TERMINAL PROBLEMS 5 OUTPUT DEBUG CONSOLE PlatformIO: Mo

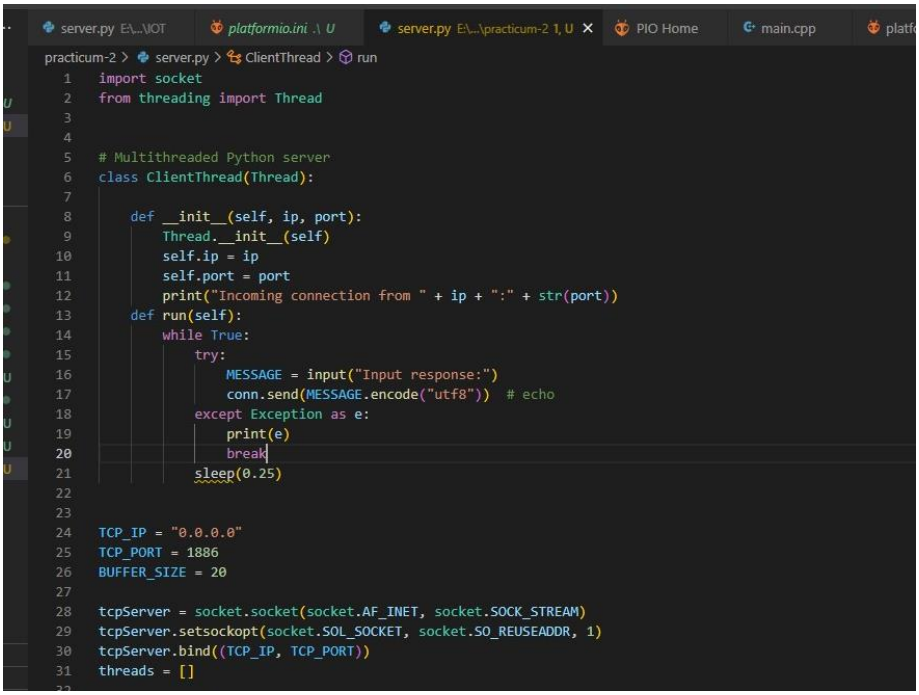
```
[Sending a request]
[Response:]
OK

[Disconnected]

[Connecting to 192.168.3.247 ... connected]
[Sending a request]
[Response:]
```

8. Observe in Server

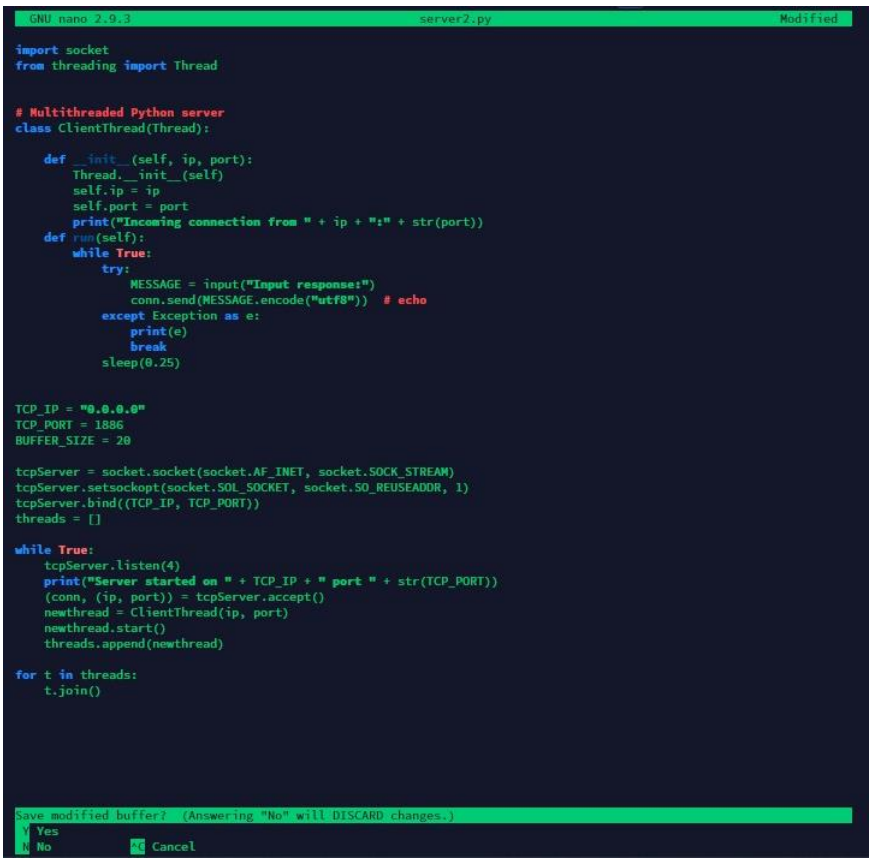
```
Server received data: b'Hai from ESP8266'
Incoming connection from 192.168.3.233:57016
Server started on 0.0.0.0 port 1886
length: 16
Server received data: b'Hai from ESP8266'
Incoming connection from 192.168.3.233:64527
Server started on 0.0.0.0 port 1886
length: 16
Server received data: b'Hai from ESP8266'
Incoming connection from 192.168.3.233:50361
Server started on 0.0.0.0 port 1886
length: 16
Server received data: b'Hai from ESP8266'
Incoming connection from 192.168.3.233:60007
Server started on 0.0.0.0 port 1886
length: 16
```

PRACTICUM 2

1. Create Server.py



2. Change Server.py in server



3. Run The Server.py

```
Server started on 0.0.0.0 port 1886
Incoming connection from 192.168.3.233:59950
Server started on 0.0.0.0 port 1886
Input response:
```

4. Change The Code Like The Example Code
5. Sucess