```python
'''Python_Server.py'''

# Writer: Writer
# Writing Date: 21/09/21
# Comment: A system of caring for patients. It is a real-time monitoring system.


#### Import Library ####
import socket
import threading
import time
import serial

#### Setting Global Variable ####
BPM = 0
SPO2 = 0
DHT = 0
err_dht = ""
err_max = ""

##########################################################################################################

#### Setting Funtions ####


#### Main - Sensors Value output ####

## Use Socket Library [bind-listen-accept-sendall-close] ##

def run_server(msg_list, host="192.168.0.11", port=4000):
    with socket.socket() as s:
        s.setsockopt(socket.SOL_SOCKET,socket.SO_REUSEADDR,1)
        s.bind((host, port))
        s.listen(2)
        conn, addr = s.accept()
        conn.sendall(msg_list.encode())
        print("[", addr[0], "] is connected.Wn>> The value is ", msg_list,"WnWn")
        conn.close()


#### Thread - Max30102 Value input ####

## Read Value -> Decode Data -> Input Decoded data to [Global Variable] - BPM and SPO2  ##

def thr_import_max():
    global BPM
    global SPO2
    global err_max

    while True:
        val_max_sum = 0
        count = 0
        for _ in range(10):
            try:
                val_max = ser_max.readline()
                val_max = val_max.decode('utf-8')[:len(val_max) - 2]
                val_max_split = val_max.split(",")

                err_max = ""

                if int(val_max_split[1]) is not 0:
```

```
                    bpm_value = val_max_split[1]
                if int(val_max_split[2]) > 80:
                    val_max_sum += int(val_max_split[2])
                    count += 1
            except ValueError:
                err_max = "Max30102 had error. Plz Check Value and Restart!"
        if count is not 0:
            val_max_avr = val_max_sum // count

        try:
            BPM = bpm_value
            SPO2 = str(val_max_avr)

        except:
            pass


#### Thread - DHT22 Value input ####

## Read Value -> Decode Data -> Input Decoded data to [Global Variable] - DHT  ##

def thr_import_dht():
    global DHT
    global err_dht

    while True:
        try:
            val_dht = ser_dht.readline()
            val_dht = val_dht.decode('utf-8')[:len(val_dht) - 2]
            DHT = val_dht

            err_dht = ""
        except:
            err_dht = "DHT22 had error. Plz Check Value and Restart!"


#### Check and modify Datas ####

## Print Menu, [ Run Server | Print Sensors Values | Check Sensors Available ] ##

def checkValue():

    while True:
        valueType = int(input("Wn0. Run ServerWn1. DHT22 Value printWn2. Max30102 Value printWn3. CheckWn4. Max30102 - Available CheckWn5. All Value printWnCheck value type : "))
        if valueType == 1:
            val_dht = ser_dht.readline()
            val_dht = val_dht.decode('utf-8')[:len(val_dht) - 2]
            for _ in range(3):
                print(val_dht + "WnWn")

        elif valueType == 2:
            val_max_sum = 0
            val_max_avr = 0
            count = 0
            for _ in range(10):
                val_max = ser_max.readline()
                val_max = val_max.decode('utf-8')[:len(val_max) - 2]
                val_max_split = val_max.split(",")
                if int(val_max_split[2]) > 80:
                    val_max_sum += int(val_max_split[2])
```

```
                    count += 1
                print(val_max)
            if count is not 0:
                val_max_avr = val_max_sum // count
            print("Wn" + val_max_split[1] + "," + str(val_max_avr) + "WnWn")

        elif valueType == 3:
            value_dht_available = ser_dht.readable()
            print(str(value_dht_available) + "WnWn")

        elif valueType == 4:
            value_max_available = ser_max.readable()
            print(str(value_max_available) + "WnWn")

        elif valueType == 5:
            val_dht = ser_dht.readline()
            val_dht = val_dht.decode('utf-8')[:len(val_dht) - 2]
            val_max = ser_max.readline()
            val_max = val_max.decode('utf-8')[:len(val_max) - 2]
            print(val_dht+val_max + "WnWn")

        elif valueType == 0:
            break

        else:
            print("plz again!WnWn")

#########################################################################################################

#### Run Server ####


#### Pre-Funtion - Setting OS and Ports ####

## Select OS - Linux or Windows ##
## And Input Port number or Select Default Values ##

print("Select Serial Port Menu!Wn")
while True:
    os_select = int(input("1. LinuxWn2. WindowsWn3. Linux DefaultWn4. Windows DefaultWnSelect your
    if os_select == 1:
        max_port = input("Enter Max30102 Port: /dev/rfcomm")
        tty_port = input("Enter DHT22 Port: /dev/ttyUSB")
        break
    elif os_select == 2:
        max_port = input("Enter Max30102 Port: COM")
        tty_port = input("Enter DHT22 Port: COM")
        break
    elif os_select == 3:
        max_port = input("Enter Max30102 Port: /dev/rfcomm")
        tty_port = 0
        break
    elif os_select == 4:
        max_port = 11
        tty_port = 15
        break
    else:
        print("Plz again")

## OS [Linux, Windows] Select And Input Values ##
if os_select == 1 or os_select == 3:
```

```
183     ser_max = serial.Serial("/dev/rfcomm"+max_port, 115200)
184     ser_dht = serial.Serial("/dev/ttyUSB"+tty_port, 9600)
185 elif os_select == 2 or os_select == 4:
186     ser_max = serial.Serial("COM"+max_port, 115200)
187     ser_dht = serial.Serial("COM"+tty_port, 9600)
188 
189 
190 #### Main Funtion - Check Values, Run Tread, Run Socket Server ####
191 
192 if __name__ == '__main__':
193 
194     checkValue()
195 
196     dht_thr = threading.Thread(target=thr_import_dht)
197     max_thr = threading.Thread(target=thr_import_max)
198 
199     dht_thr.start()
200     max_thr.start()
201 
202     time.sleep(20)
203 
204     val = DHT + "," + str(BPM) + "," + str(SPO2)
205     print("₩n₩nReady to connect.₩n>>> " + val)
206 
207     while 1:
208         if int(BPM) > 20:
209             try:
210                 val = DHT + "," + str(int(BPM) - 20) + "," + str(SPO2)
211             except:
212                 pass
213         else:
214             val = DHT + "," + str(BPM) + "," + str(SPO2)
215 
216         if err_max is not "":
217             print(err_max)
218         if err_dht is not "":
219             print(err_dht)
220 
221         run_server(val)
```