

# ASSIGNMENT-2

---

**Name:** NITHISH T

**Topic:** Assignment on temperature and humidity sensing and alarm automation using python

**Code:**

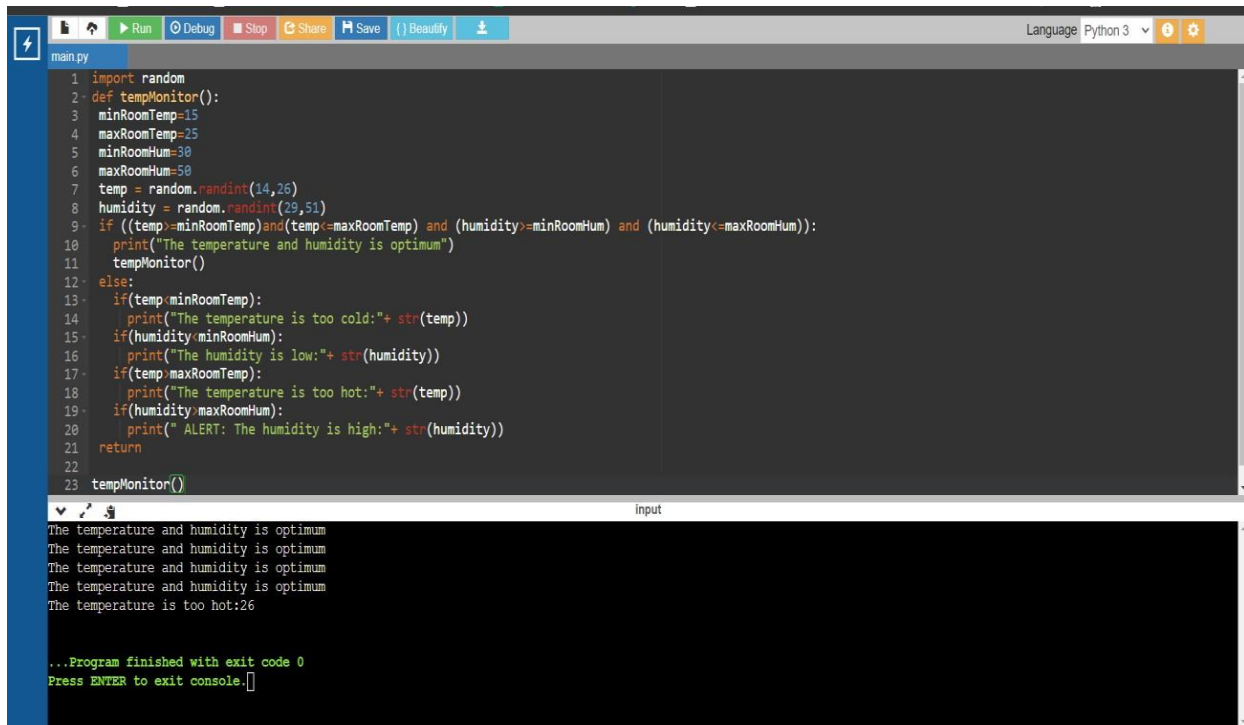
```
import random

def tempMonitor():
    minRoomTemp=15
    maxRoomTemp=25
    minRoomHum=30
    maxRoomHum=50
    temp = random.randint(14,26)
    humidity = random.randint(29,51)
    if ((temp>=minRoomTemp)and(temp<=maxRoomTemp) and
        (humidity>=minRoomHum) and (humidity<=maxRoomHum)):
        print("The temperature and humidity is optimum")
        tempMonitor()
    else:
        if(temp<minRoomTemp):
            print("The temperature is too cold:"+ str(temp))
        if(humidity<minRoomHum):
```

```
    print("The humidity is low:"+ str(humidity))
    if(temp>maxRoomTemp):
        print("The temperature is too hot:"+ str(temp))
    if(humidity>maxRoomHum):
        print(" ALERT: The humidity is high:"+
str(humidity))
    return

tempMonitor()
```

## Output



The screenshot shows a Python IDE with a file named 'main.py'. The code defines a function 'tempMonitor()' that generates random temperature and humidity values and checks them against predefined room ranges. The output console shows the results of five function calls, with the first four being 'optimum' and the fifth being 'too hot:26'.

```
1 import random
2 def tempMonitor():
3     minRoomTemp=15
4     maxRoomTemp=25
5     minRoomHum=30
6     maxRoomHum=50
7     temp = random.randint(14,26)
8     humidity = random.randint(29,51)
9     if ((temp>minRoomTemp)and(temp<=maxRoomTemp) and (humidity>=minRoomHum) and (humidity<=maxRoomHum)):
10        print("The temperature and humidity is optimum")
11        tempMonitor()
12    else:
13        if(temp<minRoomTemp):
14            print("The temperature is too cold:"+ str(temp))
15        if(humidity<minRoomHum):
16            print("The humidity is low:"+ str(humidity))
17        if(temp>maxRoomTemp):
18            print("The temperature is too hot:"+ str(temp))
19        if(humidity>maxRoomHum):
20            print(" ALERT: The humidity is high:"+ str(humidity))
21    return
22
23 tempMonitor()
```

Input

```
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature is too hot:26

...Program finished with exit code 0
Press ENTER to exit console.[]
```

```
main.py
1 import random
2 def tempMonitor():
3     minRoomTemp=15
4     maxRoomTemp=25
5     minRoomHum=30
6     maxRoomHum=50
7     temp = random.randint(14,26)
8     humidity = random.randint(29,51)
9     if ((temp>=minRoomTemp)and(temp<=maxRoomTemp) and (humidity>=minRoomHum) and (humidity<=maxRoomHum)):
10        print("The temperature and humidity is optimum")
11        tempMonitor()
12    else:
13        if(temp<minRoomTemp):
14            print("The temperature is too cold:"+ str(temp))
15        if(humidity<minRoomHum):
16            print("The humidity is low:"+ str(humidity))
17        if(temp>maxRoomTemp):
18            print("The temperature is too hot:"+ str(temp))
19        if(humidity>maxRoomHum):
20            print("ALERT: The humidity is high:"+ str(humidity))
21    return
22
23 tempMonitor()
```

input

```
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature and humidity is optimum
The temperature is too cold:14

...Program finished with exit code 0
Press ENTER to exit console.
```

```
main.py
1 import random
2 def tempMonitor():
3     minRoomTemp=15
4     maxRoomTemp=25
5     minRoomHum=30
6     maxRoomHum=50
7     temp = random.randint(14,26)
8     humidity = random.randint(29,51)
9     if ((temp>=minRoomTemp)and(temp<=maxRoomTemp) and (humidity>=minRoomHum) and (humidity<=maxRoomHum)):
10        print("The temperature and humidity is optimum")
11        tempMonitor()
12    else:
13        if(temp<minRoomTemp):
14            print("The temperature is too cold:"+ str(temp))
15        if(humidity<minRoomHum):
16            print("The humidity is low:"+ str(humidity))
17        if(temp>maxRoomTemp):
18            print("The temperature is too hot:"+ str(temp))
19        if(humidity>maxRoomHum):
20            print("ALERT: The humidity is high:"+ str(humidity))
21    return
22
23 tempMonitor()
```

input

```
The temperature and humidity is optimum
The humidity is low:29

...Program finished with exit code 0
Press ENTER to exit console.
```

The screenshot shows a web browser window with multiple tabs. The active tab is an online Python compiler. The code in the editor is a Python script named `main.py` that defines a `tempMonitor()` function. This function generates random temperature and humidity values and checks if they fall within specified ranges (15-25 for temperature, 30-50 for humidity). If they do, it prints an optimum message; otherwise, it prints a warning for being too cold, too hot, or high humidity. The function is called at the bottom of the script.

```
1 import random
2 def tempMonitor():
3     minRoomTemp=15
4     maxRoomTemp=25
5     minRoomHum=30
6     maxRoomHum=50
7     temp = random.randint(14,26)
8     humidity = random.randint(29,51)
9     if ((temp>=minRoomTemp)and(temp<=maxRoomTemp) and (humidity>=minRoomHum) and (humidity<=maxRoomHum)):
10        print("The temperature and humidity is optimum")
11        tempMonitor()
12    else:
13        if(temp<minRoomTemp):
14            print("The temperature is too cold:"+ str(temp))
15        if(humidity<minRoomHum):
16            print("The humidity is low:"+ str(humidity))
17        if(temp>maxRoomTemp):
18            print("The temperature is too hot:"+ str(temp))
19        if(humidity>maxRoomHum):
20            print(" ALERT: The humidity is high:"+ str(humidity))
21    return
22
23 tempMonitor()
```

The output window shows the program's execution: four "optimum" messages followed by an "ALERT: The humidity is high:51" message. The console ends with "...Program finished with exit code 0" and a prompt to press ENTER.

Paarthiban K201EC...py Assignment 2.docx 02:37 PM 28-09-2022