

Assignment -2

Assignment Date	20/09/2022
Student Name	Palani M
Student Roll Number	61071912130
Maximum Marks	2 Marks

Question:

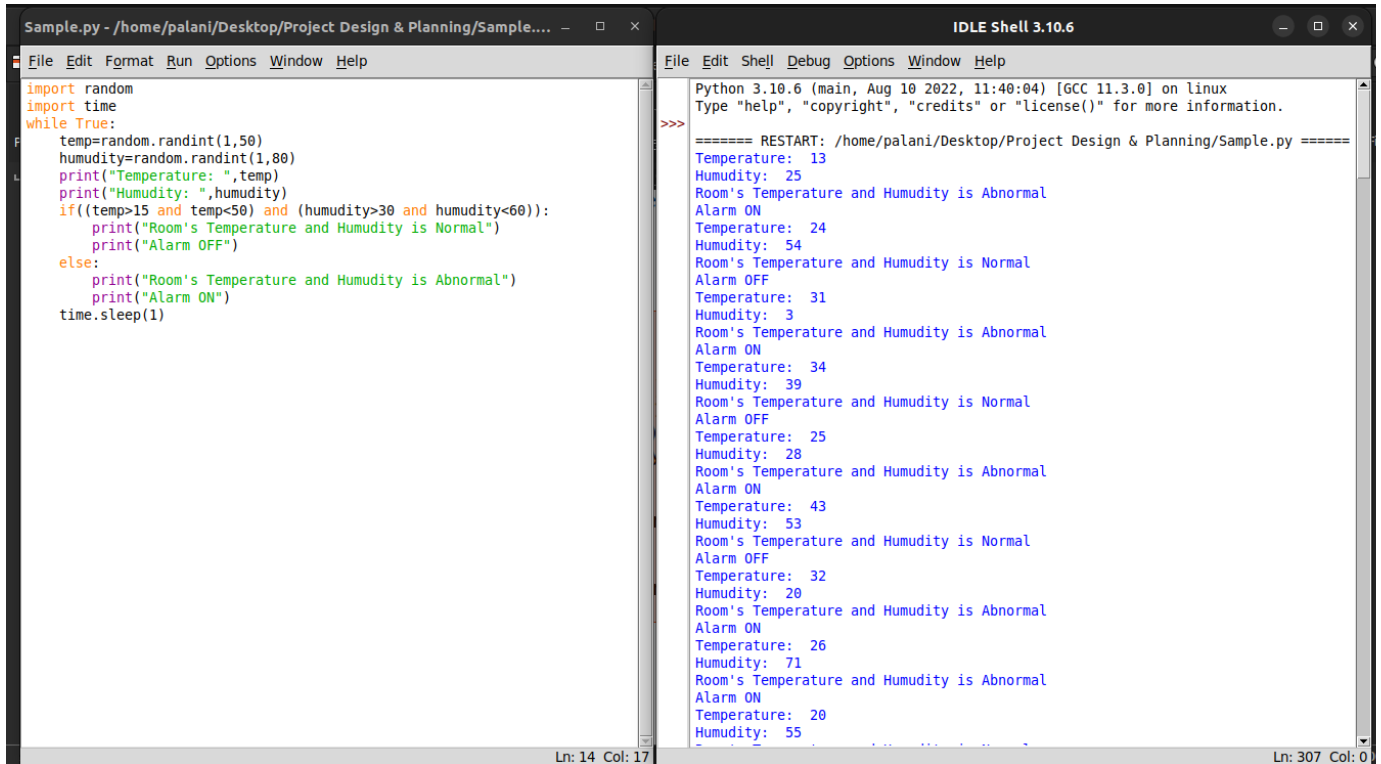
Build a python code, Assume u get temperature and humidity values (generated with random function to a variable) and write a condition to continuously detect alarm in case of high temperature.

Solution:

Python Code:

```
import random
import time
while True:
    temp=random.randint(1,50)
    humudity=random.randint(1,80)
    print("Temperature: ",temp)
    print("Humudity: ",humudity)
    if((temp>15 and temp<50) and (humudity>30 and
humudity<60)):
        print("Room's Temperature and Humudity is Normal")
        print("Alarm OFF")
    else:
        print("Room's Temperature and Humudity is Abnormal")
        print("Alarm ON")
    time.sleep(1)
```

Output:



The image shows a screenshot of an IDE with two windows. The left window, titled 'Sample.py - /home/palani/Desktop/Project Design & Planning/Sample....', contains a Python script. The script imports 'random' and 'time', then enters a 'while True' loop. Inside the loop, it generates random temperature and humidity values, prints them, and checks if they are within normal ranges. If both are normal, it prints 'Room's Temperature and Humudity is Normal' and 'Alarm OFF'. If either is outside the normal range, it prints 'Room's Temperature and Humudity is Abnormal' and 'Alarm ON'. It then sleeps for 1 second and repeats the process.

```
import random
import time
while True:
    temp=random.randint(1,50)
    humudity=random.randint(1,80)
    print("Temperature: ",temp)
    print("Humudity: ",humudity)
    if((temp>15 and temp<50) and (humudity>30 and humudity<60)):
        print("Room's Temperature and Humudity is Normal")
        print("Alarm OFF")
    else:
        print("Room's Temperature and Humudity is Abnormal")
        print("Alarm ON")
    time.sleep(1)
```

The right window, titled 'IDLE Shell 3.10.6', shows the output of the script. It starts with the Python version and GCC information. After a restart, it shows a series of temperature and humidity readings, along with the corresponding 'Room's Temperature and Humudity' status and 'Alarm' state. The output is as follows:

```
Python 3.10.6 (main, Aug 10 2022, 11:40:04) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/palani/Desktop/Project Design & Planning/Sample.py =====
Temperature: 13
Humudity: 25
Room's Temperature and Humudity is Abnormal
Alarm ON
Temperature: 24
Humudity: 54
Room's Temperature and Humudity is Normal
Alarm OFF
Temperature: 31
Humudity: 3
Room's Temperature and Humudity is Abnormal
Alarm ON
Temperature: 34
Humudity: 39
Room's Temperature and Humudity is Normal
Alarm OFF
Temperature: 25
Humudity: 28
Room's Temperature and Humudity is Abnormal
Alarm ON
Temperature: 43
Humudity: 53
Room's Temperature and Humudity is Normal
Alarm OFF
Temperature: 32
Humudity: 20
Room's Temperature and Humudity is Abnormal
Alarm ON
Temperature: 26
Humudity: 71
Room's Temperature and Humudity is Abnormal
Alarm ON
Temperature: 20
Humudity: 55
```