

ASSIGNMENT-2

NAME:DHINESH P

TEAM ID:PNT2022TMID12783

QUESTION:

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

CODE:

```
import random

import time

while True:

    temp = random.randint(1,100)

    humi = random.randint(1,100)

    print("Temperature: ", temp)

    print("Humidity: ", humi)

    if temp >= 60 and (30 < humi > 60):

        print("Alarm ON!")

    else:

        print("Alarm OFF!")

        time.sleep(2)
```

OUTPUT:

The screenshot shows the OnlineGDB IDE interface. On the left is a sidebar with navigation links: OnlineGDB beta, code.compile.run.debug.share., IDE, My Projects, Classroom (new), Learn Programming, Programming Questions, Sign Up, and Login. Below these are social media icons for Facebook, Twitter, and a '+ 30.2K' button. A 'GOT AN OPINION?' survey banner is also present. The main editor area displays a Python script named 'main.py' with the following code:

```
1 '''
2
3 Online Python Compiler.
4 Code, Compile, Run and Debug python program online.
5 Write your code in this editor and press "Run" button to execute it.
6 '''
7
8 import random
9 import time
10 while True:
11     temp = random.randint(1,100)
12     humi = random.randint(1,100)
13     print("Temperature: ", temp)
14     print("Humidity: ", humi)
15     if temp >= 60 and (30 < humi > 60):
16         print("Alarm ON!")
17     else:
18         print("Alarm OFF!")
19     time.sleep(2)
```

Below the code editor, the output window shows the execution results:

```
input
Temperature: 14
Humidity: 45
Alarm OFF!
Temperature: 49
Humidity: 82
Alarm OFF!
Temperature: 99
Humidity: 64
Alarm ON!
```

The interface also includes a top toolbar with buttons for Run, Debug, Stop, Share, Save, and Beautify, and a language dropdown set to Python 3. An 'Activate Windows' watermark is visible in the bottom right corner.

This screenshot shows the same OnlineGDB IDE interface as the first one, but with a different set of output results. The Python code in the editor remains the same. The output window displays the following execution results:

```
input
Temperature: 14
Humidity: 45
Alarm OFF!
Temperature: 49
Humidity: 82
Alarm OFF!
Temperature: 99
Humidity: 64
Alarm ON!
Temperature: 13
Humidity: 58
Alarm OFF!
Temperature: 39
Humidity: 11
Alarm OFF!
Temperature: 91
Humidity: 44
Alarm OFF!
Temperature: 90
Humidity: 39
Alarm OFF!
Temperature: 75
Humidity: 58
Alarm OFF!
Temperature: 95
Humidity: 19
Alarm OFF!
Temperature: 77
Humidity: 32
Alarm OFF!
Temperature: 74
Humidity: 33
Alarm OFF!
```

The interface elements, including the sidebar, toolbar, and language dropdown, are consistent with the first screenshot. An 'Activate Windows' watermark is also present in the bottom right corner.