**Assignment**t **1**

**Batch No : B1-1M3E**

**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 a alarm in case of**

# import standard python modules.

import time

# import adafruit dht library.

import Adafruit\_DHT

# import Adafruit IO REST client.

from Adafruit\_IO import Client, Feed

# Delay in-between sensor readings, in seconds. DHT\_READ\_TIMEOUT = 5

# Pin connected to DHT22 data pin

DHT\_DATA\_PIN = 26

# Set to your Adafruit IO key.

# Remember, your key is a secret,

# so make sure not to publish it when you publish this code! ADAFRUIT\_IO\_KEY = 'YOUR\_AIO\_KEY'

# Set to your Adafruit IO username.

# (go to https://accounts.adafruit.com to find your username).

ADAFRUIT\_IO\_USERNAME = 'YOUR\_AIO\_USERNAME'

# Create an instance of the REST client.

aio = Client(ADAFRUIT\_IO\_USERNAME, ADAFRUIT\_IO\_KEY)

# Set up Adafruit IO Feeds.

temperature\_feed = aio.feeds('temperature') humidity\_feed = aio.feeds('humidity')

# Set up DHT22 Sensor.

dht22\_sensor = Adafruit\_DHT.DHT22

while True:

humidity, temperature = Adafruit\_DHT.read\_retry (dht22\_sensor, DHT\_DATA\_PIN)

if humidity is not None and temperature is not None: print('Temp={0:0.1f}\*C Humidity={1:0.1f}%'.format(temperature, humidity))

# Send humidity and temperature feeds to Adafruit IO

temperature = '%.2f'%(temperature)

humidity = '%.2f'%(humidity) aio.send(temperature\_feed.key, str(temperature)) aio.send(humidity\_feed.key, str(humidity))

else:

print('Failed to get DHT22 Reading, trying again in ', DHT\_READ\_TIMEOUT, 'seconds')

# Timeout to avoid flooding Adafruit IO time.sleep(DHT\_READ\_TIMEOUT)