**CODE:**

#include<DTH.h>

#define DHTPIN 2 // Define the pin where your DHT11 is connected

#define DHTTYPE DHT11 // Define the type of DHT sensor (DHT11 or DHT22)

DHT dht(DHTPΙΝ, DHTTYPE);

void setup() {

Serial.begin(9600);

dht.begin();

}

void loop() {

delay(2000); // Delay for 2 seconds between measurements

// Read temperature and humidity from the sensor

float temperature dht.readTemperature();

float humidity dht.readHumidity();

// Check if any reads failed and exit early (to try again)

if (isnan(temperature) || isnan(humidity)) {

Serial.println(&quot;Failed to read from DHT sensor&quot;);

return;

}

// Print the temperature and humidity values to the serial monitor

Serial.print(&quot;Temperature: &quot;);

Serial.print(temperature);

Serial.println(&quot; °C&quot;);

Serial.print(&quot;Humidity: &quot;);

Serial.print(humidity);

Serial.println(&quot; %&quot;);

// You can also convert to Fahrenheit if needed

// float fahrenheit (temperature 1.8) + 32;

// Serial.print(&quot;Temperature: &quot;);

//Serial.print(fahrenheit);

// Serial.println(&quot; "F&quot;);

}

import sys

import adafruit\_DHT as dht

from time import sleep

import urllib

import RPi GPIO as GPIO

GPIO.setmode(GPIO.BCM)

GPIO.setup(24, GPIO. IN)

GPIO.setwarnings(False)

#Smoke Sensor Digital output

def DHT11\_data():

#Reading from DHTII and storing the temperature and humidity humi, temp dht.read\_retry(11,23)

return humi, temp

#Enter Your API key here myAPI "ZQN4Z08XZ9HLOZY0"

#URL where we will send the data, Don't change it

baseURL = "https://api.thingspeak.com/update?api\_key=%s" %

while True:

humidity, temperature = DHT11\_data()

print (Temp: {0:0.1f} C Humidity: {0:0.1f} %.format(temperature, humidity))

input\_state = GPIO.input(24)

if input\_state == False:

smoke = 1

print('Smoke Detected")

print(smoke)

else:

smoke = 0

print(Smoke is not Detected")

print(smoke)

from urllib.request import urlopen

content = urlopen(baseURL + "&field1=%s&field2=%s&field3=%s"% (temperature,

humidity, smoke))

sleep(15)

#sudo pip3 install adafruit-circuitpython-dht

#sudo apt-get install libgpiod2

import time

import board

import digitalio

import adafruit\_character\_lcd.character\_led as characteriod

import adafruit\_dht

import RPI.GPIO as GPIO

#Initial the dht device, with data pin connected to:

dhtDevice = adafruit\_dht.DHT11(board.D19)

#Modify this if you have a different sized character LCD

lcd\_columns 16

lcd\_rows 2

#Raspberry Pi Pin Config:

led\_rsdigitalio.DigitalinOut(board.D5)

led\_en digitalio.DigitalInOut(board.D6)

lcd\_d4digitalio.DigitalInOut(board.D12)

lcd\_d5 digitalio.DigitalInOut(board.D13)

lcd\_d6digitalio.DigitallnOut(board.D16)

lcd\_d7digitalio.DigitalInOut(board.D17)

#Initialise the led class

led characteried.Character\_LCD\_Mono(

lcd\_rs, lcd\_en, lcd\_d4, lcd\_d5, lcd\_d6, lcd\_d7, lcd\_columns, lcd\_rows)

if\_\_name\_\_==‘\_\_main\_\_’:

while True:

try:

#Print the values to the serial port

temperature\_c = dhtDevice.temperature

temperature\_f = temperature\_c\*(9/5)+32

humidity = dhtDevice.humidity

print("Temp: {:.1f} F / {:.1f} C Humidity: { }%".format(temperature\_f, temperature\_c, humidity))

lcd.clear()

#lcd\_line\_1="Temperature:" + str(temperature\_c) + "C"

#lod\_line\_2= "\nHumidity:"+ str(humidity) + "%"

#lcd.message lcd\_line\_1 + lcd\_line\_2;

lcd.message("Temper:%. 1f C" %temperature\_c)

lcd.message("\nHumidity:%.IF" %humidity)

time.sleep(2.0)

except RuntimeError as error:

#Errors happen fairly often, DHT's are hard to read, just keep going

print(error.args[0])

time.sleep(2.0)

continue

except KeyboardInterrupt:

GPIO.cleanup()

print (Exiting Program')

exit()