

Practica #4 Sensor de Temperatura DS18B20

- 1.- Crear, Insertar y Consultar en la base de datos, todos los datos generados por el sensor de temperatura, así como la fecha y hora de nuestro sistema, en MongoDB Atlas (si tienen algún problema. Lo pueden hacer en la base local).
- 2.- Limitar la inserción de datos cuando la temperatura sea mayor a 28 grados centígrados u 82.4 grados Fahrenheit en caso de decidir usar Fahrenheit. La pantalla debe imprimir todas las temperaturas generadas por el sensor.
- 3.- Mostrar en pantalla los encabezados de Fecha Hora Temperatura.
- 4.- Enviar evidencia en foto de, código completo, resultado en pantalla (terminal raspberry), Resultado en la Base de Datos, y conexiones.

Nota: les dejo el código Base y el diagrama de conexión.

```
## Se declaran 3 variables que apuntan a la direccion de los datos del sensor
import os
import glob
import time

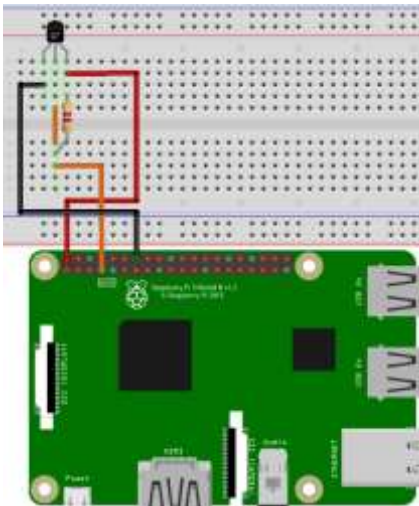
os.system('sudo apt-get install')
os.system('sudo apt-get install')

base_dir = "/tmp/temperatura"
device_folder = glob.glob(base_dir + "/*")
device_file = device_folder[0]

## con esta funcion se lee el archivo que contiene la salida de la temperatura. Se leen todas las lineas
## y luego se devuelve para que el código que llama a esta funcion pueda usarlo.
def read_temp_raw():
    f = open(device_file, "r")
    lines = f.readlines()
    f.close()
    return lines

## Esta funcion se encarga de procesar los datos de la anterior. Primero nos aseguramos que la primera linea traiga un 02, esto significa
## que habra una linea con una temperatura.
def read_temp():
    lines = read_temp_raw()
    while lines[0].strip()[-1:] != "02":
        time.sleep(0.1)
        lines = read_temp_raw()
    equals_pos = lines[1].find("=")
    if equals_pos != -1:
        temp_string = lines[1][equals_pos+2:]
        ## Es la posicion de la '=' y se agrega 2 a la posicion, así esto se obtiene
        ## los valores de temp reales.
        temp_c = float(temp_string) / 1000.0
        ## Convertimos a Centígrados.
        temp_f = temp_c * 9.0 / 5.0 + 32.0
        ## Convertimos a Fahrenheit.
        return temp_c, temp_f

while True:
    ## Esta linea siempre es verdadera por lo que se ejecutara para siempre hasta que el prog sea interrumpido por un error o por el usuario
    print(read_temp())
    time.sleep(1)
```



A este código le hace falta importar las librerías GPIO, el setwarnings, el setmode, el setup, y declarar el pin. Favor de agregárselo como lo vimos en clase.