



Graficar el clima

Creación: 2021



Contenido

Objetivo.....	3
Introducción	3
Requerimientos	4
Instalar Matplotlib.....	4
Elegir una estación meteorológica	4
Programación en Python	5
Bibliografía	7



Objetivo

Recopilar datos meteorológicos de una base de datos meteorológica de Raspberry Pi y utilizarlos para crear gráficos de datos de sensores a lo largo del tiempo usando Python.

Introducción

A principios de 2016 en una iniciativa de Raspberry en conjunto con Oracle se enviaron casi mil estaciones meteorológicas a escuelas de todo el mundo, listas para ensamblarse y comenzar a recopilar datos meteorológicos globales.

Cada estación meteorológica está equipada con los sensores que se muestran en la siguiente tabla:

Nombre del sensor	Propósito
Pluviómetro	Mide el volumen de lluvia que cae en milímetros.
Anemómetro	Mide la velocidad del viento en kilómetros por hora.
Veleta	Mide la dirección del viento en grados.
Sonda de temperatura del suelo	Mide la temperatura del suelo en grados Celsius.
Sensor de temperatura	Mide la temperatura del aire en grados Celsius.
Sensor de humedad	Mide la humedad relativa del aire como porcentaje.
Sensor de presión	Mide la presión atmosférica en pascales.
Sensor de calidad del aire	Mide la calidad del aire como porcentaje relativo.

Tabla 1 Sensores en cada estación meteorológica



Requerimientos

- Raspberry pi con Raspbian y conexión a internet

Instalar Matplotlib

1. Para instalar el software que se necesita, ejecutar los siguientes comandos en una terminal (mantener presionado ctrl + alt + t):

```
sudo apt-get install python3-matplotlib -y  
sudo apt-get install python3-requests -y
```

Elegir una estación meteorológica

Se puede obtener una lista de todas las estaciones meteorológicas que están actualmente en línea, utilizando una URL simple. Esto se debe a que la base de datos a la que todas las estaciones meteorológicas cargan datos tiene una API RESTful. Este es un método mediante el cual puede escribir código que usa solicitudes HTTP simples (como un navegador) para obtener los datos.

1. Copiar y pegar la siguiente URL en un navegador web:

<https://apex.oracle.com/pls/apex/raspberrypi/weatherstation/getallstations>

Se debe ver una web llena de datos en los que se pueden encontrar:

- **weather_stn_id**: El ID de la estación meteorológica
 - **weather_stn_name**: El nombre de la estación meteorológica
 - **weather_stn_lat**: La latitud de la estación meteorológica
 - **weather_stn_long**: La longitud de la estación meteorológica
2. Se pueden descargar los datos meteorológicos registrados para una estación meteorológica individual con bastante facilidad, solo se debe poner el ID de la estación meteorológica al final del siguiente link:

<https://apex.oracle.com/pls/apex/raspberrypi/weatherstation/getallmeasurements/>



Programación en Python

1. Una vez elegida la estación meteorológica abrir el IDE de Python haciendo clic en **Menú** (Logotipo de Raspberry pi), después en **Programación** y finalmente en **Thonny Python IDE**.

El módulo **request** maneja las llamadas HTTP, que se necesitan para obtener páginas web. **matplotlib** es una biblioteca que permite dibujar gráficos en Python. **dateutil** es un módulo que permite convertir fechas a diferentes formatos.

2. Escribir el siguiente código en el archivo Python para importar las funciones y métodos necesarios:

```
from requests import get
import matplotlib.pyplot as plt
from dateutil import parser
```

Después se debe declarar una variable para almacenar la URL de la llamada a la API y realizar la llamada.

3. Copiar el siguiente código sustituyendo el ID de la estación elegida anteriormente.

```
url='https://apex.oracle.com/pls/apex/raspberrypi/weatherstation/getallmeasurements/511028'
weather = get(url).json()
```

4. Si se quisiera encontrar la temperatura del aire ambiente específicamente, por ejemplo, se podría escribir lo siguiente:

```
temperatures = [record['ambient_temp'] for record in weather['items']]
timestamps = [parser.parse(record['reading_timestamp']) for record in
weather['items']]
```

5. Para graficar los datos, solo se necesitan cuatro líneas de código. La primera indicará los dos conjuntos de datos que está trazando, la segunda y tercera agrega etiquetas a los ejes Y y X respectivamente y la última mostrará los datos:

```
plt.plot(timestamps, temperatures)
plt.ylabel('Temperatura')
plt.xlabel('Tiempo')
plt.show()
```

6. Hacer clic en el icono **Run**, aparecerá la gráfica.

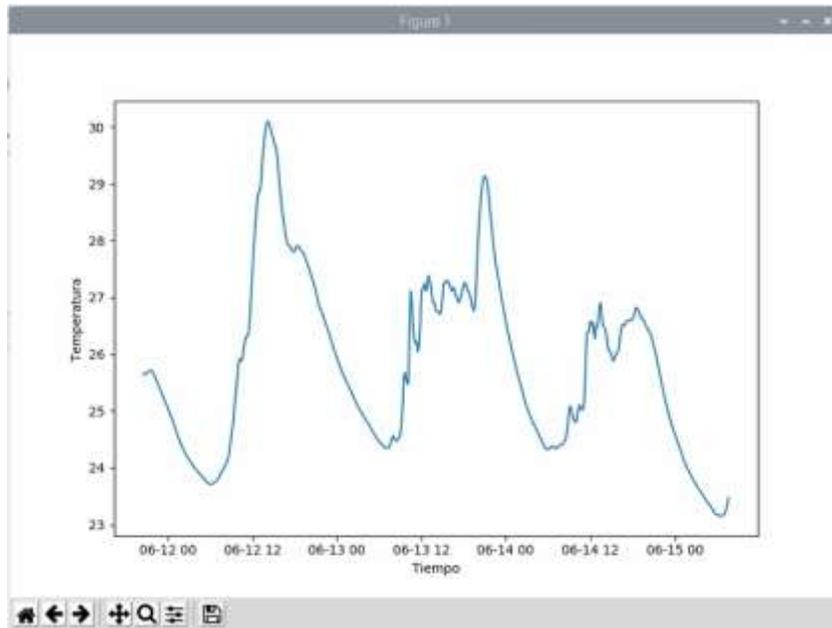


Figura 1 Grafica de tiempo contra temperatura

7. Si se desea graficar otro parámetro se debe modificar la línea de código:

```
temperatures = [record['ambient_temp'] for record in weather['items']]
```

Cambiando 'ambient_temp' por alguno de los siguientes:

- 'ground_temp'
- 'air_quality'
- 'air_pressure'
- 'humidity'
- 'wind_direction'
- 'wind_speed'
- 'wind_gust_speed'
- 'rainfall'

Y respectivamente la etiqueta del eje Y por:

- 'Temperatura del suelo'
- 'Calidad del aire'
- 'Presión del aire'
- 'Humedad'
- 'Dirección del viento'
- 'Velocidad del viento'
- 'Velocidad de la ráfaga de viento'
- 'Lluvia'



Bibliografía

Graphing the weather. (2021). Recuperado el 27 de febrero de 2021, de <https://projects.raspberrypi.org/en/projects/graphing-the-weather>