

Práctica 3 Comunicación, gestión y representación de datos de sensores con Python.

3.1 Objetivos de la práctica:

- Familiarizarse con Python como lenguaje de programación
- Gestionar datos procedentes de comunicaciones serie
- Trabajar con estos datos en formato csv para almacenarlos en ficheros y redireccionarlos vía internet
- Operar con los datos y representarlos gráficamente.

Si no conoces Python, este tutorial contiene prácticamente todo lo necesario para esta práctica:

<https://uniwebsidad.com/libros/python>

Como entorno de desarrollo recomendamos Anaconda y su editor Spyder:

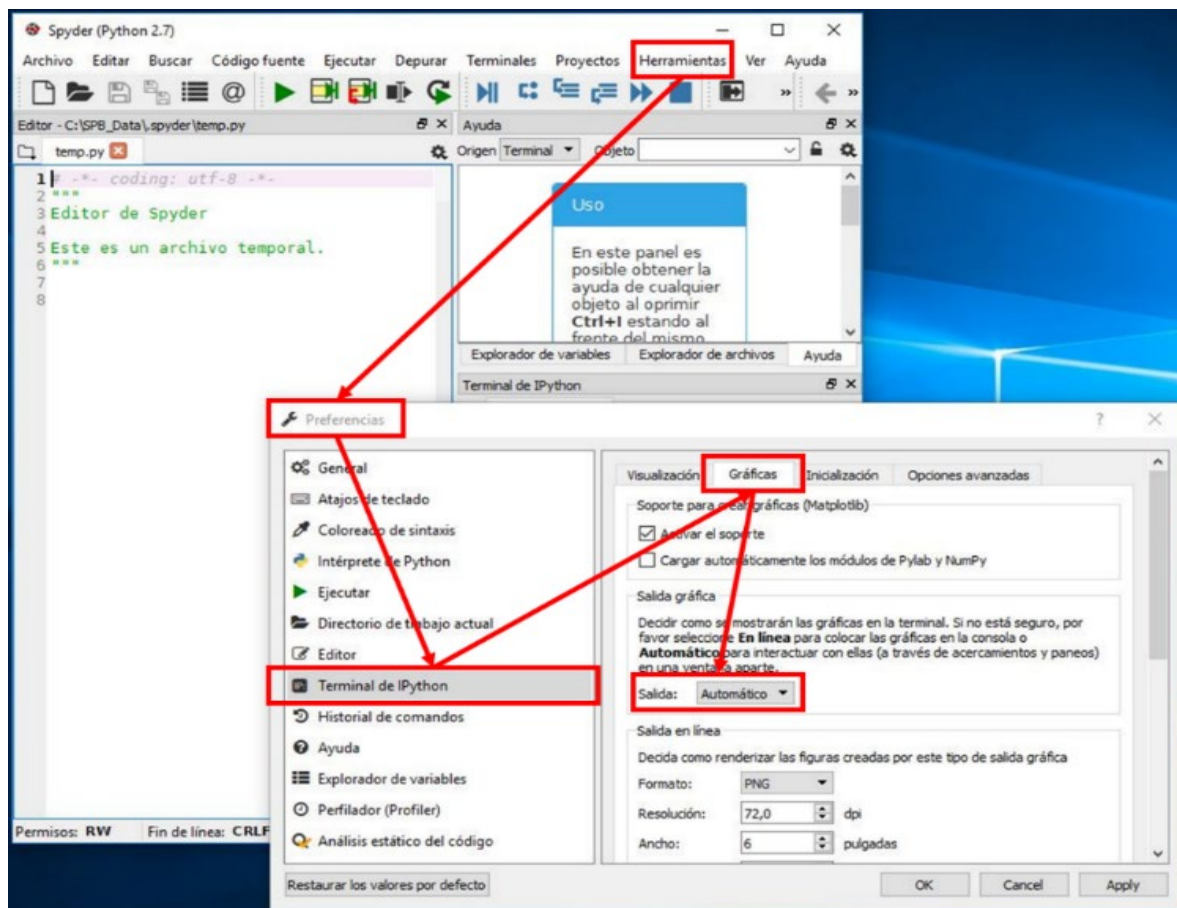
<https://www.anaconda.com/download>

Partimos del apartado 5 de la primera práctica en la que se conecta un sensor inercial por I2C, se muestrea la aceleración y se mandan los datos vía UART cada 50 ms al PC.

3.2 Tareas a realizar:

1. Conecta el Nano 33 BLE con el programa cargado y comprueba la recepción de datos en el puerto serie con un programa de puerto serie como TERMITE.
Material de apoyo: https://www.compuphase.com/software_termite.htm
2. Programa en Python un programa que acceda al puerto serie y muestre por pantalla los datos en tiempo real (igual que el terminal). Necesitarás instalar el objeto pyserial como administrador.
3. Modifica el programa para que almacene los datos en fichero .txt separando cada variable con “;” y con retorno de carro al final de cada muestra para abrirlo desde Excel. Previamente, guarda un fichero con texto fijo para asegurarte que generas el fichero adecuadamente.

4. Modifica el programa para que acumule los datos durante 5 segundos, calcule el promedio y desviación estándar y los represente en tiempo real. Necesitarás instalar la librería Matplotlib (u otra de representación de gráficas) y configurar Spyder para que no genere una gráfica detrás de otra, sino que permita la actualización automática de la gráfica tras cada muestra plotada.



3.3 Hitos evaluables de la práctica:

1. Enseñar el proceso de captura de datos, explicarlo y mostrar el csv resultante en Excel de la tarea 3
2. Enseñar el proceso de generación de las gráficas correspondientes de la tarea 4