

# Inspira Crea Transforma

# **ESCUELA DE CIENCIAS APLICADAS E INGENIERÍA INGENIERÍA AGRONÓMICA**

## **ST0299 PENSAMIENTO COMPUTACIONAL II Guía para configurar un entorno y probar dispositivos (Sensores / Actuadores)**

**Yomin Jaramillo M**

Docente | Escuela de Ingeniería | Ingeniería Agronómica

Correo: [yejaramilm@eafit.edu.co](mailto:yejaramilm@eafit.edu.co)

# Actividades Previas realizadas

Las siguientes actividades ya se realizaron, si no ha realizado este trabajo debe hacerlo para que las siguientes actividades sean mas efectivas

# Instalación de IDE Arduino

Para programar la tarjeta necesitas el entorno de desarrollo Arduino, el cual lo puedes descargar del siguiente:

- <https://www.arduino.cc/en/software>

Ejecuta el instalador del programa y sigue los pasos de instalación:

- Acepta los términos y condiciones de la licencia
- Selecciona todas las opciones para que instale todos los complementos y drivers necesarios.
- Selecciona la ruta de instalación y presiona “install”.
- Monte uno de los programas que realizó en el simulador y realice la compilación, si tiene errores corrija hasta que quede compilado con éxito.,

# Programar tarjeta

Identifique los componentes de la tarjeta y en donde están instalados los diferentes sensores.

Conectar la tarjeta (board) a uno de los puertos del USB.

Configure por la opción de herramientas el puerto y seleccione la tarjeta Arduino indicada, en esta caso es una tarjeta Arduino Mega

Cargue el programa que prende y apaga el LED, al cargar debe de poder ver el LED titilando.

Puedes usar el siguiente enlace de apoyo:

- <https://arduino.cl/como-instalar-arduino-en-windows/#:~:text=%2D%20Instala%20el%20entorno%20de%20desarrollo,instalaci%C3%B3n%20y%20presiona%20%E2%80%9Cinstall%E2%80%9D.>

# Probar el entorno Arduino con Sensor

# Actividades Previas

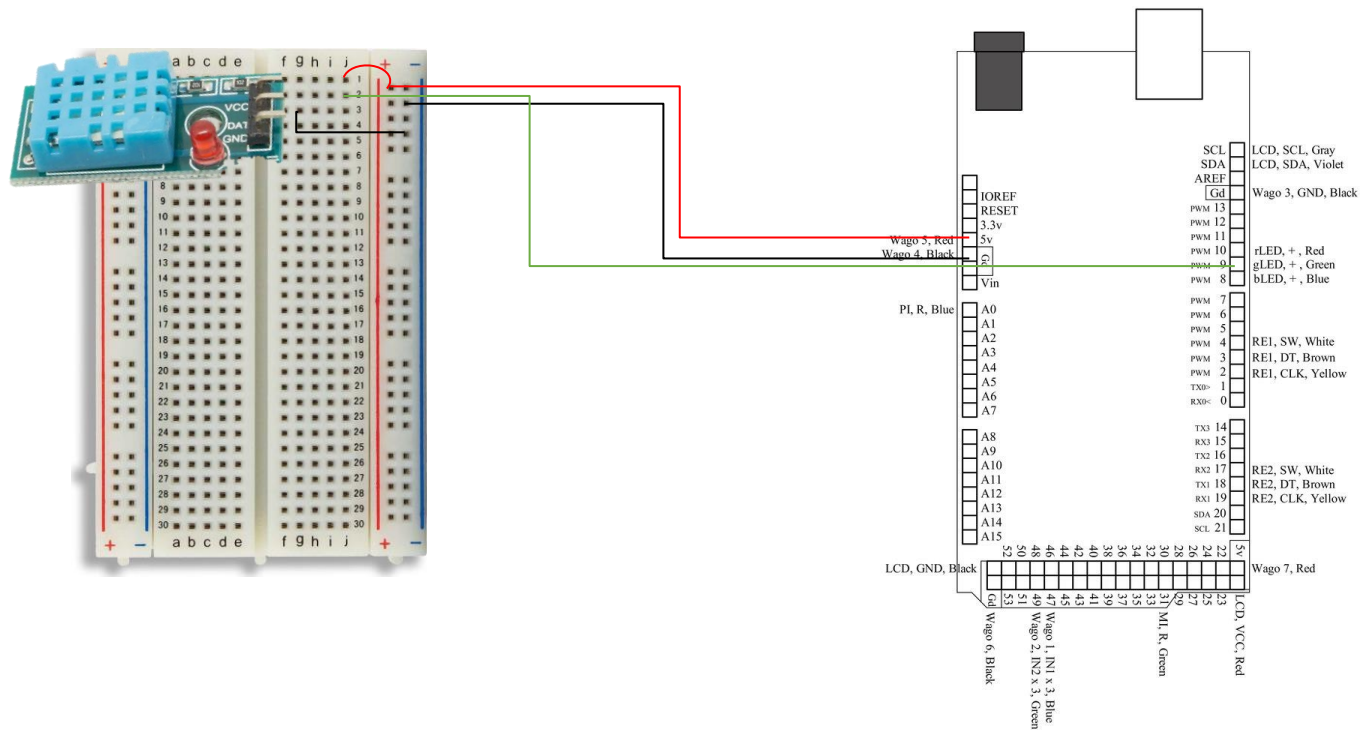
Debe tener instalado el IDLE de Arduino en su máquina, se recomienda sea la versión 1.8.x o preferiblemente la 2.x

Para un mejor manejo, abra un directorio en su equipo, puede ser en el raíz de la materia, le sugiero lo llame ***ProyectoPCII***

Dentro de este directorio, cree dos subdirectorios: el primero llámelo ***Arduino*** y el Segundo ***Python***

Luego de tener ya listo su entorno, ensamble en una de las tarjetas el siguiente esquema con el sensor HT11, sensor de temperatura y humedad

# Ensamble de dispositivos





# Preparar para probar

Para hacer pruebas debe seguir los siguientes pasos:

- Revise que todo le haya quedado bien conectado
- Conectar la plaqueta con el cable azul a un puerto USB de su PC
- Descargar el archivo ***01.AdquirirDatosBasico.ino***
- Cree una subcarpeta en el directorio que ha creado para el proyecto del curso en la carpeta **Arduino**
- Copie el archivo descargado en esta carpeta
- Al darle doble click al archivos **.ino** dentro de la respectiva carpeta y se le abrirá el IDLE de Arduino con el código
- Vamos a probar como funciona esta lógica y realizar las configuraciones requeridas

# Probar el programa

Al ingresar al IDLE de Arduino, su código está disponible para ser cargado a la plaqueta.

Configure la plaqueta, para esto abra en el menú la opción de herramientas y:

- Seleccione la plaqueta Arduino Mega
- Seleccione el puerto, el le indica en cual está conectada su plaqueta

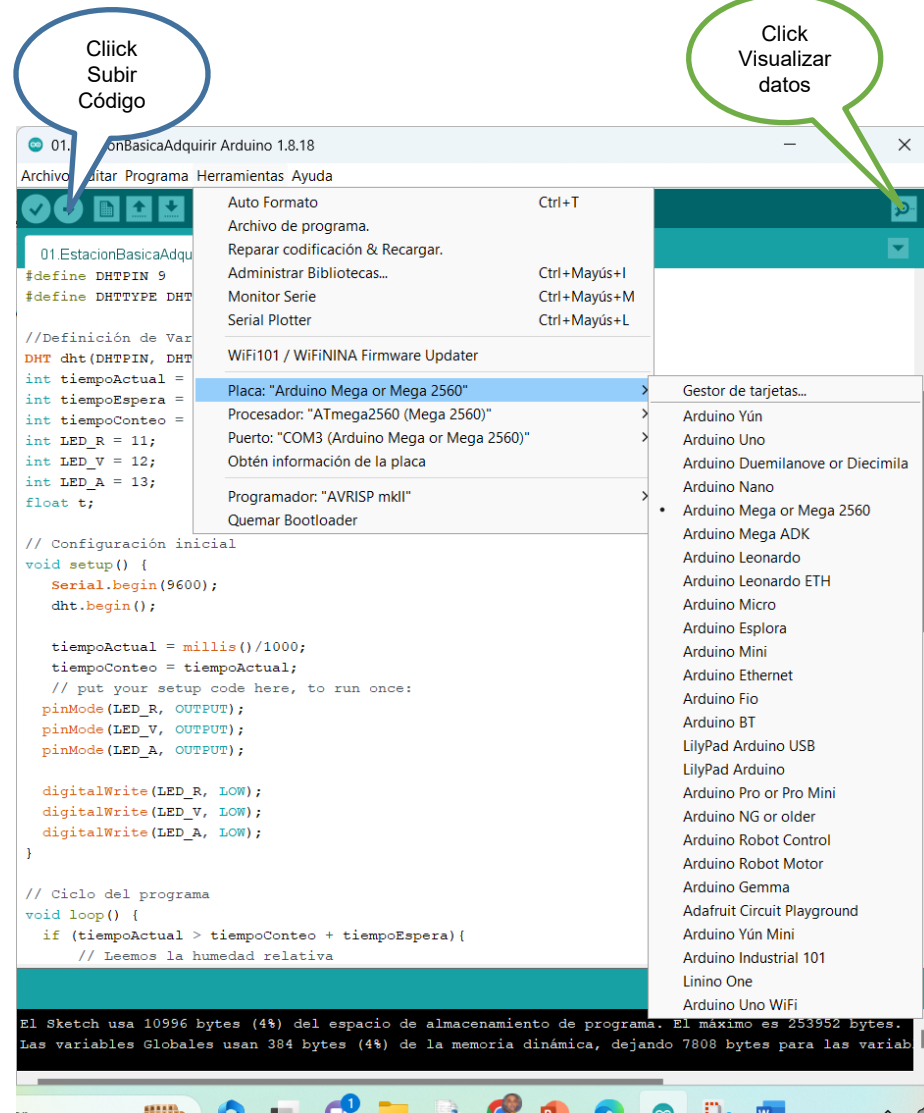
Luego de esto cargue el código, dando click en el ícono subir y espere que aparezca mensaje subido

Para ver si está funcionando, abra la consola de puerto serían y deberá ver los datos que se están leyendo

Estudie la lógica y anote preguntas para el profesor

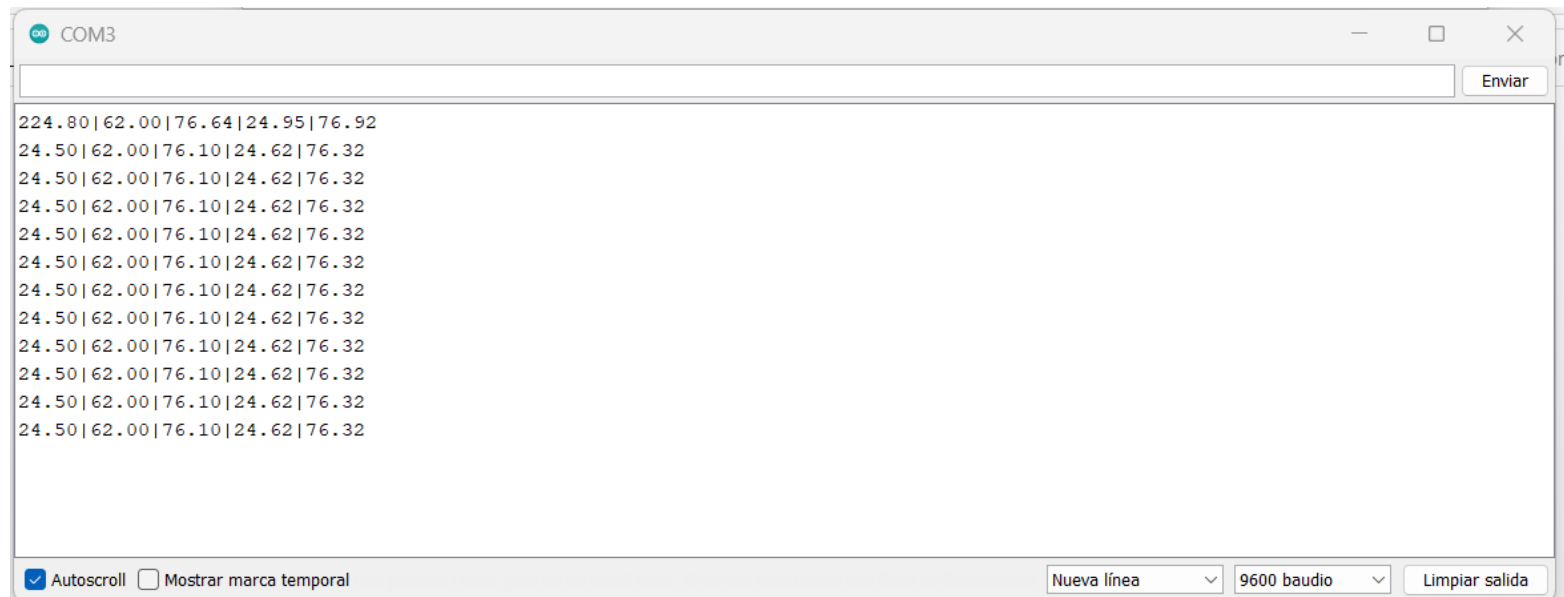
Complemente su instalación con LEDS para el semáforo y ver como cambia dependiendo de la temperatura.

Si quiere ver variaciones, sople el sensor varias veces y verá que cambia la temperatura



# Visualización de datos

- Si todo está correcto, los datos en la consola del puerto serial se debe ver así:



# Configurar e integración con Python

# Inicio con entorno Python

Debemos realizar lo siguiente antes de seguir adelante con la integración con Programa en Python:

Asegurar que se tiene Python y los paquetes de Python pySerial instalado en su ordenador

Instalar un editor de texto para escribir códigos Python

Asegurar que los componentes de hardware estén configurados, como se defina en el diseño del sistema

Se conecta el Arduino al ordenador mediante un cable USB

# Instalar Python

En primer lugar, deberá descargar una copia de Python desde :  
<https://www.python.org/getit/>

En descargas está el archivos de instalación ejecute el archivo descargado para iniciar la instalación

Debe tener cuidado con la versión de Python que está descargando.

Compruebe si el sistema operativo es de 32 bits o 64 bits.

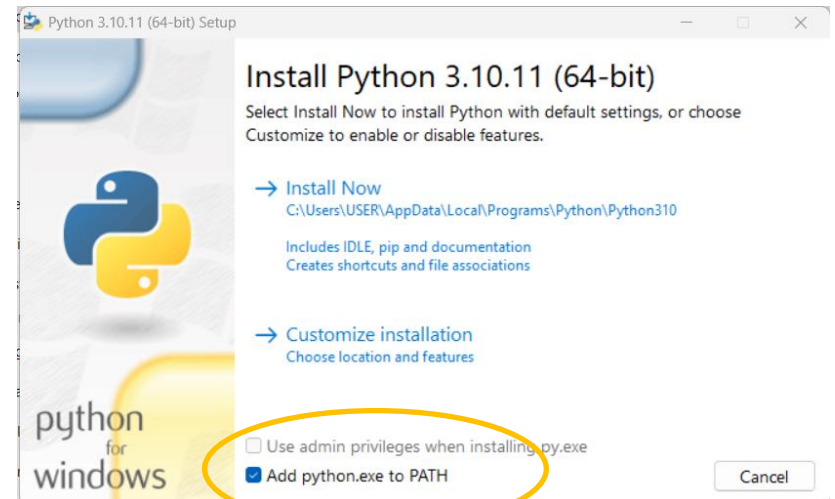
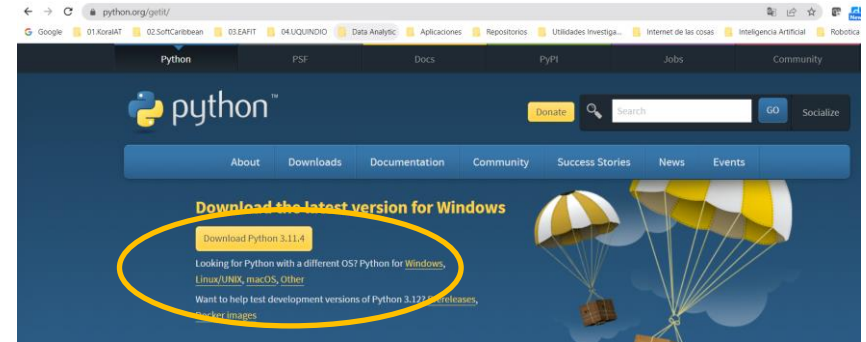
Instalar una versión 3.x de Python, recomiendo las versión 3.9.x, aunque puede ser la superior que actualmente es la 3.11.x

Se recomienda que se instale la versión de 32-bit por compatibilidad de librerías, aunque también se puede instalar las de 64-bit

En la instalación, se muestra una opción que permite chequear la opción que le configure la ruta del Python para poder ejecutar de cualquier directorio los comando.

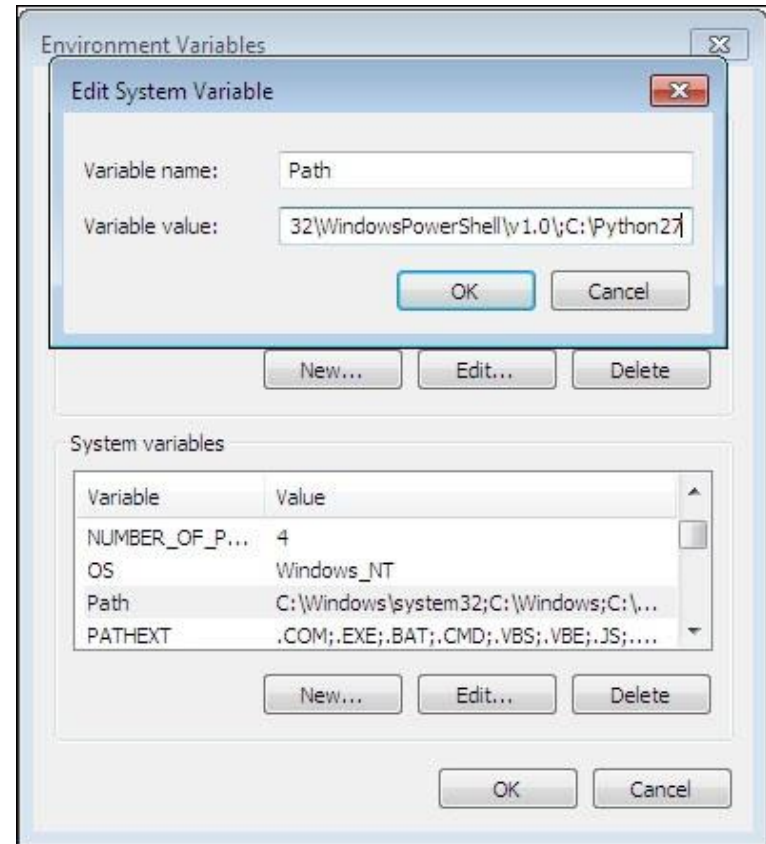
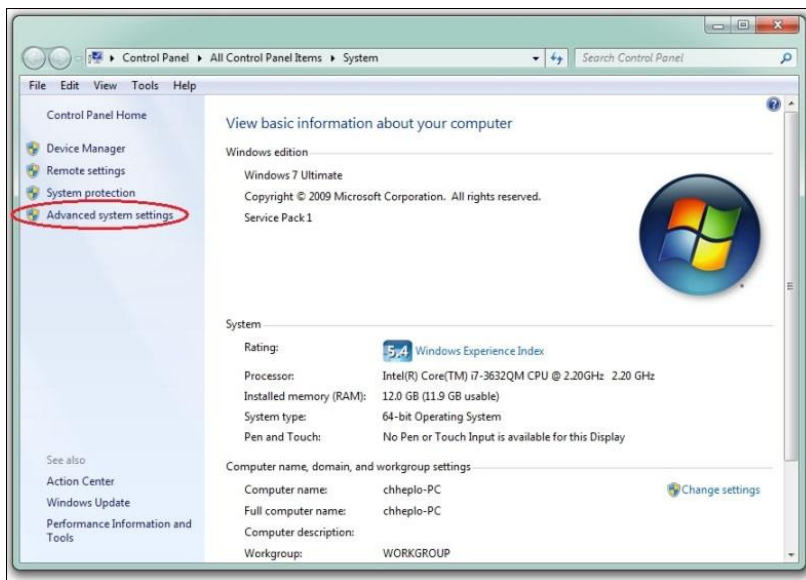
Si no lo hace, debe adicionar en las variables de entorno el PATH de Python, donde quedó instalado.

Es importante realizar el paso anterior para no tener que hacer la configuración de variables de entorno



# Configuración

No realice esto si chequeo la opción de incluir en el path de variables de entorno la ruta donde fue instalado Python



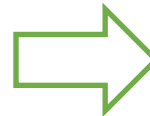
# Prueba la instalación de Python

Ingresa a una consola de Windows o PowerShell, busque en Windows CDM (Símbolo del Sistema) o PowerShell, recomendando CDM

Ejecute los siguientes comandos para validar que tiene Python instalado correctamente:

- `python -h`: debe mostrarle una lista de argumentos que puede utilizar con Python
- `Python -V`: le muestra la versión que fue instalada

Finalmente digite Python y de <enter>, esto debe entrar a una consola de Python donde puede realizar comando del lenguaje



```
Símbolo del sistema
C:\Users\USER>python -V
Python 3.10.11

C:\Users\USER>python -h
usage: python [option] ... [-c cmd | -m mod | file | -] [arg] ...
Options and arguments (and corresponding environment variables):
-b      : issue warnings about str(bytes_instance), str(bytearray_instance)
         and comparing bytes/bytearray with str. (-bb: issue errors)
-B      : don't write .pyc files on import; also PYTHONDONTWRITEBYTECODE=x
-c cmd  : program passed in as string (terminates option list)
-d      : turn on parser debugging output (for experts only, only works on
         debug builds); also PYTHONDEBUG=x
-E      : ignore PYTHON* environment variables (such as PYTHONPATH)
-h      : print this help message and exit (also -? or --help)
-i      : inspect interactively after running script; forces a prompt even
         if stdin does not appear to be a terminal; also PYTHONINSPECT=x
-I      : isolate Python from the user's environment (implies -E and -s)
-m mod  : run library module as a script (terminates option list)
-O      : remove assert and __debug__-dependent statements; add .opt-1 before
         .pyc extension; also PYTHONOPTIMIZE=x
-OO     : do -O changes and also discard docstrings; add .opt-2 before
         .pyc extension
-q      : don't print version and copyright messages on interactive startup
-s      : don't add user site directory to sys.path; also PYTHONNOUSERSITE
-S      : don't imply 'import site' on initialization
-u      : force the stdout and stderr streams to be unbuffered;
         this option has no effect on stdin; also PYTHONUNBUFFERED=x
-v      : verbose (trace import statements); also PYTHONVERBOSE=x
         can be supplied multiple times to increase verbosity
```



```
Símbolo del sistema - python
C:\Users\USER>python
Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr  5 2023, 00:38:17) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a=15
>>> b=280
>>> a*b
4200
>>> import pyserial
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'pyserial'
```



# Instalar Protocolo PySerial

# Revisar si cuenta con PIP

Para instalar librerías de Python, requiere **pip**

En caso que no esté instalado, debe realizar los siguientes pasos que se encuentran documentados en:

<https://pip.pypa.io/en/stable/installation/#python>

Realice los pasos que allí le indican, debe bajar el archivo **get-pip.py**: <https://bootstrap.pypa.io/get-pip.py>. Lo abre y dar click derecho y dice guardar como con el mismo nombre

Luego donde bajó el archivo ejecute lo que se indica en el gráfico

## get-pip.py

This is a Python script that uses some bootstrapping logic to install pip.

- Download the script, from <https://bootstrap.pypa.io/get-pip.py>.
- Open a terminal/command prompt, `cd` to the folder containing the `get-pip.py` file and run:

Linux   MacOS   Windows

```
C:> py get-pip.py
```

# Instalar librería pySerial

La biblioteca pySerial permite la comunicación con Arduino encapsulando el acceso para el puerto serie.

Este módulo proporciona acceso a la configuración del puerto serie a través de propiedades Python y le permite configurar el puerto serie directamente a través del intérprete.

pySerial será el puente para cualquier comunicación futura entre el Python y Arduino.

Instale pySerial así desde la línea de comando de IDLE de Python o desde una consola de sistema operativo:

**> *pip install pyserial***

# Valide que la librería quedó instalada

- Ingrese al IDE por defecto de Python
- Realice las siguientes instrucciones:
  - `import serial` //No debe dar error
  - `puerto = 'COM7'` //No debe dar error
  - `arduino = serial.Serial(port, 9600)`
- Puede que le de error que el puerto no es el indicado
- Abra el IDLE de Arduino si lo ha cerrado y revise en que puerto tiene conectada la plaqueta, este es el que debe asignar a puerto, cambie, pruebe de nuevo y ya no le debe dar error

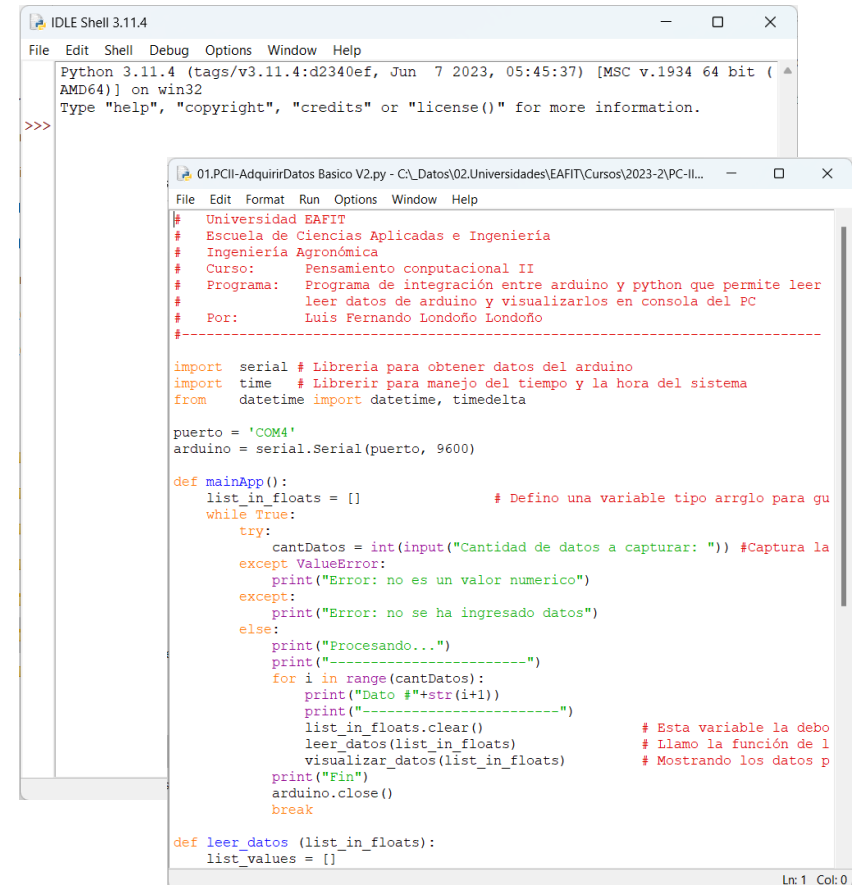
# Abrir programa ejemplo de Python

Baje de ejemplos de EAFIT interactiva el archivo: **01.PCII-AdquirirDatos Basico V2.py**

Coloque el archivo en la carpeta que abrió para el proyecto en el subdirectorio **Python**.

Abra este archivo en el IDLE de Python, puede ser de dos maneras:

- Desde el botón Windows digite Python y el le muestra para que abra el IDLE, desde ahí puede abrir el archivo desde File
- O en el directorio donde está el archivo, de click mouse derecho diga que va a abrir el archivo con (Show more options) IDLE Python



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>

01.PCII-AdquirirDatos Basico V2.py - C:\Datos\02.Universidades\EAFIT\Cursos\2023-2\PCII-IL...
File Edit Format Run Options Window Help
# Universidad EAFIT
# Escuela de Ciencias Aplicadas e Ingeniería
# Ingeniería Agronómica
# Curso: Pensamiento computacional II
# Programa: Programa de integración entre arduino y python que permite leer
# leer datos de arduino y visualizarlos en consola del PC
# Por: Luis Fernando Londoño Londoño
#-----
import serial # Libreria para obtener datos del arduino
import time # Libreria para manejo del tiempo y la hora del sistema
from datetime import datetime, timedelta

puerto = 'COM4'
arduino = serial.Serial(puerto, 9600)

def mainApp():
    list_in_floats = [] # Defino una variable tipo arreglo para gu
    while True:
        try:
            cantDatos = int(input("Cantidad de datos a capturar: ")) #Captura la
        except ValueError:
            print("Error: no es un valor numerico")
        except:
            print("Error: no se ha ingresado datos")
        else:
            print("Procesando...")
            print("-----")
            for i in range(cantDatos):
                print("Dato #" + str(i+1))
                print("-----")
                list_in_floats.clear() # Esta variable la debo
                leer_datos(list_in_floats) # Llamo la función de l
                visualizar_datos(list_in_floats) # Mostrando los datos p
            print("Fin")
            arduino.close()
            break

def leer_datos(list_in_floats):
    list_values = []

Ln: 1 Col: 0
```

# Ejecutar programa ejemplo

Revise que en una de las instrucciones se está asignando el puerto, cambie para el que usted tiene conectada la tarjeta

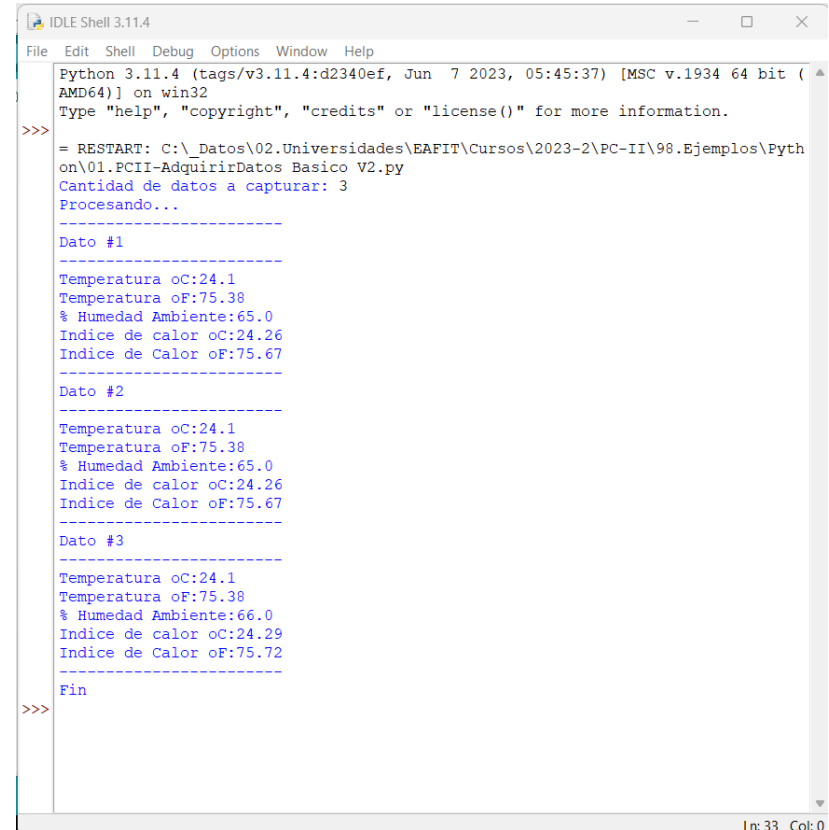
En la parte superior del IDLE de click sobre <Run> y seleccione <Run Module>

El programa le pide cantidad de datos a capturar, digite un valor, por ejemplo 5

EL programa le debe mostrar lo que en la imagen se presenta

Estudie el código y formule todas las preguntas que considere no comprende del código

Si da error, analice y trate de buscar las causas



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\_Datos\02.Universidades\EAFIT\Cursos\2023-2\PC-II\98.Ejemplos\Python\01.PCII-AdquirirDatos Basico V2.py
Cantidad de datos a capturar: 3
Procesando...
-----
Dato #1
-----
Temperatura oC:24.1
Temperatura oF:75.38
% Humedad Ambiente:65.0
Indice de calor oC:24.26
Indice de Calor oF:75.67
-----
Dato #2
-----
Temperatura oC:24.1
Temperatura oF:75.38
% Humedad Ambiente:65.0
Indice de calor oC:24.26
Indice de Calor oF:75.67
-----
Dato #3
-----
Temperatura oC:24.1
Temperatura oF:75.38
% Humedad Ambiente:66.0
Indice de calor oC:24.29
Indice de Calor oF:75.72
-----
Fin
>>>
```

# Retos

Vamos a modificar el código Arduino para controlar los tres LED que hemos integrado

Complemente el código de tal forma que depende de rangos de temperatura prenda los LED de diferente manera

Adicione el sensor de humedad, busque en internet como es la conexión o solicite asesoría al profesor.

Busque como se conectan los sensores de luminosidad análogo y el digital entregados por el profesor.

En equipo busquen como conectar el sensor de Co2 y el sensor de presencia.

Revise si hay error en Python, es porque está enviando menos datos de los que recibe Python y ajuste.

# FIN DE TEMA

**Configurando un entorno para desarrollo de  
sistemas ciber-físicos**