

# Inspira Crea Transforma

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

## **ST0299 PENSAMIENTO COMPUTACIONAL II Procesamiento de datos adquiridos**

**Yomin Jaramillo M**

Docente | Escuela de Ingeniería | Ingeniería Agronómica

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

# Repaso de Programas para Arduino y Python

# Repasando Arduino

El lenguaje de programación de Arduino se puede dividir en tres grandes partes:

Estructura,

Valores (variables y constantes),

Funciones.

## Estructuras

- Sketc
- Control
- Operadores

## Valores

- Constantes
- Tipos de Datos
- Conversión

## Funciones

- I/O Digitales
- I/O Analógicas
- Matemáticas

Referencia: <https://www.arduino.cc/reference/es/>

# Ejemplo

**//Declaración de librerías**

#include <librería.h>

**//Definición de constantes**

#define VARIABLE valor

**//Definición de Variables**

int tiempoActual = 0;

int tiempoConteo = 0;

void setup() {     **// Configuración inicial**

    Serial.begin(9600);

**// Inicialice salidas**

}

void loop() {     **// Ciclo del programa**

**//Lógica para adquirir y controlar**

}

# Repasando Python

Python es un lenguaje de programación que se utiliza en el desarrollo de software, aplicaciones web, ciencia de datos y machine learning.

Es multiparadigma, lo que significa que admite programación imperativa, orientada a objetos y funcional.

Es multiplataforma, por lo que se puede utilizar en Windows, Linux y Mac OS.

Es dinámico e interpretado, lo que significa que el tipo de las variables se decide en tiempo de ejecución.

Es fuertemente tipado, lo que significa que no se puede usar una variable en un contexto fuera de su tipo.

# Programas en Python

- Los archivos fuente de Python usan la extensión ".py" y se denominan "módulos".
- La estructura es muy estándar y está compuesta de:
  - Definición de Librerías: múltiples librerías disponibles para los programas, como:
    - `import sys`
  - Definición de variables: no se definen tipos, lo que se asigne representa el tipo, el nombre debe ser nemotécnico usando notación camello o con guion bajo, como
    - `puerto_arduino = "COM3"`
    - `cantidadDatos = 20`
  - Definición de funciones: es la forma de modularizar más fácil los programas, también se puede lograr con orientación a objetos, ejemplo
    - `def procesar_datos(numero, datos)`

# Otras características de Python

- Cadenas
  - Python tiene una clase de cadena integrada llamada `str` con muchas funciones prácticas. Ejemplo:
    - `datos = "estos es una cadena"`
    - `datos_arduino = 'dato1;dato2;dato3'`
- Listas
  - Python tiene un excelente tipo de lista integrada llamada `list`. Los literales de lista se escriben entre corchetes `[ ]`. Ejemplo
    - `titulos = [Fecha, 'Variable, 'Valor]`
- Utilidades
  - Tiene varias utilidades que permiten ayudan a realizar funciones. Ejemplos:
    - Interactuar con el sistema de archivos: `os`, `os.path`
    - El módulo `*subprocess*` es una forma sencilla de ejecutar un comando externo y capturar su resultado.
    - Una excepción representa un error de tiempo de ejecución que detiene la ejecución normal en una línea en particular y transfiere el control al código de manejo de errores



# Visualización de datos de forma tabular

# Librería Pandas

Pandas es una librería de Python especializada en la manipulación y el análisis de datos.

Ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales, es como el Excel de Python.

Es un software libre distribuido bajo la licencia BSD.

El nombre deriva del término "datos de panel", término de econometría que designa datos que combinan una dimensión temporal con otra dimensión transversal.

# Características de Pandas

Define nuevas estructuras de datos basadas en los arrays de la librería NumPy pero con nuevas funcionalidades.

Permite leer y escribir fácilmente ficheros en formato CSV, Excel y bases de datos SQL.

Permite acceder a los datos mediante índices o nombres para filas y columnas.

Ofrece métodos para reordenar, dividir y combinar conjuntos de datos.

Permite trabajar con series temporales.

Realiza todas estas operaciones de manera muy eficiente.

# Tipos de datos en pandas

Pandas dispone de tres estructuras de datos diferentes:

- Series: Estructura de una dimensión.
- DataFrame: Estructura de dos dimensiones (tablas).
- Panel: Estructura de tres dimensiones (cubos).

Estas estructuras se construyen a partir de arrays de la librería NumPy, añadiendo nuevas funcionalidades.

Ampliar información: <https://aprendeconalf.es/docencia/python/manual/pandas/>

# Librería Numpy

NumPy es una librería de Python especializada en el cálculo numérico y el análisis de datos, especialmente para un gran volumen de datos.

Incorpora una nueva clase de objetos llamados arrays que permite representar colecciones de datos de un mismo tipo en varias dimensiones, y funciones muy eficientes para su manipulación.

La ventaja de Numpy frente a las listas predefinidas en Python es que el procesamiento de los arrays se realiza mucho más rápido (hasta 50 veces más) que las listas, lo cual la hace ideal para el procesamiento de vectores y matrices de grandes dimensiones.

# Creación de Arreglos con Numpy

Para crear un array se utiliza la siguiente función de NumPy

- `np.array(lista)` : Crea un array a partir de la lista o tupla lista y devuelve una referencia a él. El número de dimensiones del array dependerá de las listas o tuplas anidadas en lista:
- Para una lista de valores se crea un array de una dimensión, también conocido como vector.
- Para una lista de listas de valores se crea un array de dos dimensiones, también conocido como matriz.
- Para una lista de listas de listas de valores se crea un array de tres dimensiones, también conocido como cubo.
- Y así sucesivamente. No hay límite en el número de dimensiones del array más allá de la memoria disponible en el sistema.

**Ampliar información:** <https://aprendeconalf.es/docencia/python/manual/numpy/>

# FIN DE TEMA

**Procesamiento de datos adquiridos**