

# Taller personalizado de Programación y Robótica en CMM BML – 2025 - 2026

Clase 2 – Primeros pasos en  
Python (sin IDE)

**xx Octubre 2025**



Voluntario : J.C. Santamaria

# Curso 25-26 Clase 2 – Índice 90 minutos

---

1. ¿Qué es un lenguaje de programación? – 10'

---

2. ¿Por qué aprender Python como 1er lenguaje? – 10'

---

3. Python: Referencias para Alumnos – 10'

---

4. Probamos un intérprete de Python on line – 20'

---

5. Estructuras básicas en programas: ejemplos en lenguaje de bloques-5'

---

6. Reto CL2: Haz en clase un programa que sume 2 números -25'

---

**Resumen: Tópicos cubiertos en esta clase- 5'**

# ¿Qué es un lenguaje de programación?



Los lenguajes de programación son la forma en que **los seres humanos podemos dar instrucciones a un equipo informático**. Un “paquete” de estas instrucciones es lo que se denomina programa.

Los Lenguajes de programación tienen **REGLAS SINTACTICAS ABSOLUTAMENTE RIGIDAS**, a diferencia de los lenguajes humanos, para evitar ambigüedad y así no ejecutar acciones erróneas

En la **programación visual** se emplean bloques gráficos y formas geométricas que podemos arrastrar por la pantalla y ensamblar (como en un puzle). Ejemplos: Scratch, Blockly, App Inventor, MakeCode, ArduinoBlocks, Alice, Kodu, EduBlocks, etc.

Pero este tipo de lenguajes **NO SON PRACTICOS en usos reales**



En la **programación textual (o basada en texto)**, la **secuencia de instrucciones que componen un algoritmo se especifica con instrucciones de texto**, y por eso necesitamos conocer el vocabulario (instrucciones, palabras clave...); la sintaxis (operadores, marcadores...) para construir ‘frases’ (sentencias) y ‘cuentos’ (programas) correctos. Además, hay que conocer como ‘alimentar’ estos programas al equipo informático.

Ejemplos: Python, Processing, C++, Java, etc.

**Algoritmo : ‘receta’ para resolver un problema**, se puede escribir en diferentes lenguajes de programación

# ¿Por qué aprender Python como 1er lenguaje?

## Algunas Razones

- **Fácil de aprender y de usar.** Beneficia tanto a los principiantes como a los expertos.
  - **Sintaxis legible.**
  - **Creación rápida de programas: *es un lenguaje Interpretado***
- **Ampliamente usado en el mundo real**
  - **Desarrollo WEB**
  - **Data science**
  - **#1 en Inteligencia Artificial**
  - **Automatizacion**
- Funciona en los principales sistemas operativos y plataformas informáticas + **Existe una versión para microcontroladores → microPython**
- **Gratis** (al igual que su código fuente y sus bibliotecas).
- **Libre y de código abierto** (open source).
- Amplias **bibliotecas**.
- **Extensible e integrable** con otros lenguajes

## Algunos videos para ampliar info

1. [1 video corto en castellano que resume lo esencial](#)
2. **Video muy completo en Ingles**, incluye detalles técnicos al final y recomienda Thonny como entorno de desarrollo IDE (el que usaremos durante el curso)



# Python: Referencias para Alumnos

- **En el curso seguiremos el libro [Invent your own computer games with Python 4ed- Al Sweigart](#)**
  - Para referencias del lenguaje seguiremos [esta web](#) (es la + sencilla) o el [oficial en castellano](#)
  - Para **alumnos con formación en programación que lean en ingles**, la web de [Real Python](#) tiene unos excelentes tutoriales
- Python con interfaz grafico (GUI)
- [Create Graphical User Interfaces with Python](#) de Laura Sach
  - [Making Games with Python & Pygame](#) de Al Sweigart

# Probamos un intérprete de Python on line

**Sin instalar nada**, podemos probar a hacer y ejecutar programas en Python usando un intérprete on line

- 1- Entramos y probamos el modo terminal (`>_`) con una suma y un print
- 2- Escribimos un programa 'hola mundo' **en la ventana de arriba** y lo ejecutamos con el botón RUN
- 3- Vemos algunas opciones como: abrir, guardar, compartir, pantalla completa, .....



<https://www.online-python.com/>

Un interprete de Python online gratuito puede ser útil si no tenemos ordenador propio, o no queremos instalar un IDE en nuestro ordenador . Pero al cabo de un tiempo o vamos a versiones de pago on line, o instalamos un IDE

# Probamos un intérprete de Python on line -2

Ahora un programa algo más complicado : Saluda, nos pregunta el nombre , lo guarda y nos responde con “Encantado Pepe” o “Encantado Juan”. Lo hacemos con el profesor paso a paso



```
main.py P_holaplus.py +
5 # Fecha JCSP 2023 02 06
6 # Licencia : CC BY-NC-SA 4.0
7 # https://creativecommons.org/licenses/by-nc-s
8
9 print('¡Hola clase! Soy un programa en Python')
10 print('¿Cuál es tu nombre?')
11 miNombre = input()
12 print('Encantado de conocerte, ' + miNombre)

Ln: 1, Col: 1

Run Share Command Line Arguments

¡Hola clase! Soy un programa en Python
¿Cuál es tu nombre?
Jose Carlos
Encantado de conocerte, Jose Carlos

>_

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

```
# Taller Prog. y Robótica en Colegio Santo Domingo – 2024 -2025 - Clase 1
# Nombre de programa : P_holaplus.py
# Resumen: se presenta, pregunta nombre y responde encantado .nombre
# Creditos : adaptacion de JCSP basado en "Invent with python" ed4 - cap. 2
# https://inventwithpython.com/invent4thed/chapter2.html
# Fecha JCSP 2023 02 06
# Licencia : CC BY-NC-SA 4.0
# https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es
```

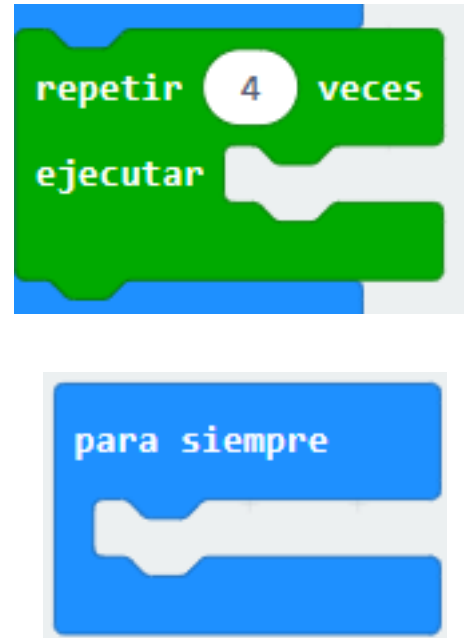
```
print('¡Hola clase! Soy un programa en Python')
print('¿Cuál es tu nombre?')
miNombre = input()
print('Encantado de conocerte, ' + miNombre)
```

# Estructuras básicas en programas: ejemplos en lenguaje de bloques

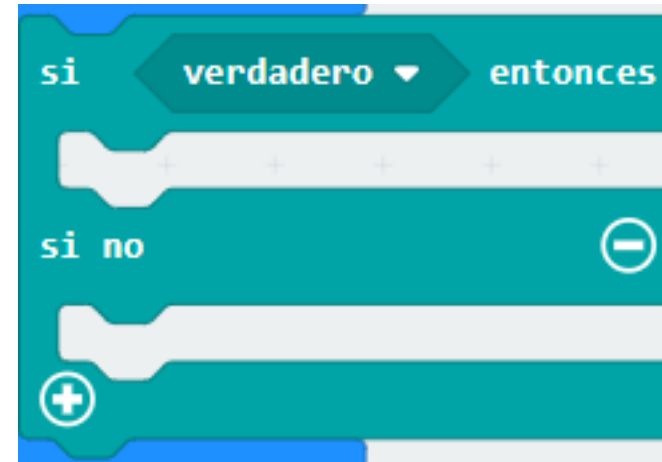
## SECUENCIA



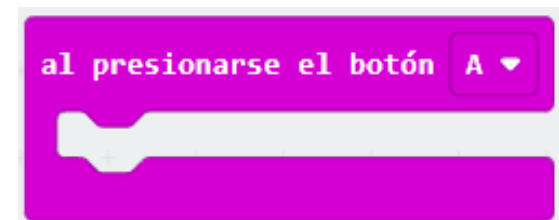
## BUCLE



## SELECCION



## EVENTO






# Reto CL2: Haz en clase un programa que sume 2 números

## Ayuda para: Escribir un programa que pida 2 números y los sume

- 1. Usa el programa 'Hola plus' como base
- 2. Input() devuelve lo que se introduzca como TEXTO, esto no lo podemos sumar directamente, usa la función int( pon dentro el texto o variable)
- 3. Tienes que guardar los dos números en una variable ejemplo

Numero1 = int(input())

- 4. En un print puedes hacer operaciones dentro y se mostrara el resultado
- 5. La salida del programa podría ser algo como



```
¡Hola clase! Dime 2 números y los sumaré
Dime el 1er número: 5
Dime el 1er número: 3
La suma es = 8
```

# Tópicos cubiertos en esta clase

## Estos son los tópicos de Python que hemos trabajado

- ☐ Interactive Shell : >>>
- ☐ Operadores (básicos) : +, -
- ☐ Numero enteros y de coma flotante
- ☐ Expresiones de cadena , frases, encerradas entre comillas simples o dobles
- ☐ Guardar valores en variables con '=' (cuidado no es que comparador de igualdad '==')
- ☐ Escribir un programa, guardarlo y ejecutarlo
- ☐ Comentarios: #
- ☐ Funciones incorporadas de python: print, input

————— Estamos trabajando el —————

➤ [Capítulo 2 de Invent .....Python](#)