

linTIC





#### CICLO I: Fundamentos de programación en Python





# Sesión 2: Introducción a Python







### Objetivos de la sesión

Al finalizar esta sesión estarás en capacidad de:

- 1. Identificar los tipos de variables más usados en Python
- 2. Modificar instrucciones en Python para resolver problemas específicos
- 3. Crear variables que representen diferentes tipos de datos y hacer operaciones sobre sus valores



#### ¿Qué es Python?



**Definición**: Python es un lenguaje programación de propósito general muy poderoso y flexible, y fácil de aprender.

#### Principales ventajas de Python:

- Es de código abierto (certificado por la OSI).
- Es interpretable y compilable.
- Es fácil de aprender gracias a que su sintaxis es bastante legible para los humanos.
- Es un lenguaje maduro (29 años).
- Es fácilmente extensible e integrable en otros lenguajes (C, java).







# Aplicaciones hechas con Python

















# VARIABLES Y OPERADORES ARITMÉTICOS





#### Variables en Python

Python es un lenguaje de programación orientado a objetos y su modelo de datos también está basado en objetos. Para cada dato que aparece en un programa, Python crea un objeto que lo contiene. Cada objeto contiene:

Un identificador único

 Un número entero distinto para cada objeto.

Un tipo de datos

• Entero, decimal, cadena de caracteres, etc.

Un valor

El propio dato





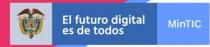


#### Operadores aritméticos en Python

Un operador aritmético toma dos operandos como entrada, realiza un cálculo y devuelve el resultado.

OPERADOR	DESCRIPCIÓN	USO
+	Realiza Adición entre los operandos	12 + 3 = 15
- 4	Realiza Sustracción entre los operandos	12 - 3 = 9
*	Realiza Multiplicación entre los operandos	12 * 3 = 36
d	Realiza División entre los operandos	12 / 3 = 4
%	Realiza un módulo entre los operandos	16 % 3 = 1
**	Realiza la potencia de los operandos	12 ** 3 = 1728
//	Realiza la división con resultado de número entero	18 // 5 = 3







# TIPOS DE DATOS EN PYTHON









Booleanos: Verdadero o falso

Numéricos: Enteros, punto flotante y complejos

Cadenas de caracteres: Letras

# Otros tipos de datos:

- Secuencias
- Mapas
- Conjuntos
- Iteradores
- Clases
- Excepciones

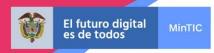






# ESTRUCTURAS ALGORÍTMICAS SECUENCIALES







#### ESTRUCTURAS SECUENCIALES

Es la estructura más sencilla de todas. En este tipo de estructura una acción, paso o instrucción sigue a otra en secuencia, es decir, no se ejecuta la instrucción si la anterior no se ha ejecutado (la salida de una, es la entrada de la siguiente y así sucesivamente, hasta el fin del proceso). Una estructura secuencial se representa de la siguiente forma:

Inicio

Acción1

Acción2

AcciónN Fin





### Asignación

**Definición:** En términos computacionales, consiste en el paso de valores a una variable. En otras palabras, la asignación consiste, en dar valor a una variable. Se representa con el símbolo □.

La asignación se puede clasificar de la siguiente forma:

- Simples: Almacena un valor constante a una variable. Ejemplo: a←5
- Contador: Para llevar un conteo del número de veces que se realiza un proceso.
   Ejemplo: a ← a+1
- Acumulador: Es un sumador en un proceso. Ejemplo: a ← a+b
- De trabajo: Guarda el resultado de una operación matemática o expresión. TIC 2022
   Ejemplo: a ← c+b\*2/4





## Estructuras de Entrada y Salida

**Lectura (Entrada):** La lectura consiste en recibir o capturar desde un dispositivo de entrada (p.ej. el teclado) un valor. En pseudocódigo se representa así:

#### Sintaxis:

Leer a, b Donde "a" y "b" son las variables que recibirán los valores

**Escritura (Salida):** Consiste en mostrar por medio de un dispositivo de salida (p.ej. monitor o impresora) un resultado o mensaje. En pseudocódigo se representa así:

#### Sintaxis:

Escribir "El resultado es:", R Donde "El resultado es:" es un mensaje que se desea aparezca y R es la variable que contiene un valor.







# Ejercicio para entender secuenciales

#### Definición del Problema:

Suponga que un individuo desea invertir su capital en un banco y desea saber cuánto dinero ganara después de un mes si el banco paga a razón de 2% mensual.

#### Análisis del Problema:

Datos de entrada: capital a invertir (cap\_inv)

Proceso: gan ← cap\_inv \* 0.02

Datos de Salida: Ganancia (gan)

#### Diseño del Algoritmo:

Inicio

Real cap\_inv, gan

Leer cap\_inv

gan ← cap\_inv \* 0.02

Escribir gan

Fin

#### Codificación:





#### Prueba de escritorio

#### Inicio

```
a \leftarrow 2

b \leftarrow 3

c \leftarrow 4

x1 \leftarrow (-b + (b^2 - 4^*a^*c)^{(1/2)}/(2^*a)

x2 \leftarrow (-b - (b^2 - 4^*a^*c)^{(1/2)}/(2^*a)

escribir x1

escribir x2
```

Fin





**IGRACIAS**POR SER PARTE DE ESTA EXPERIENCIA DE APRENDIZAJE!



