


Conceptos Programación en Python



Contenido

- **Google Colab**
 - **Conocimientos básicos de Algoritmos y programación**
 - **Estructuras de control (if y for)**
 - **Funciones**
 - **Librería**
- 
- A decorative graphic consisting of multiple layers of blue dots forming a wavy, horizontal pattern that spans the width of the slide, positioned behind the list items.

Python



colab



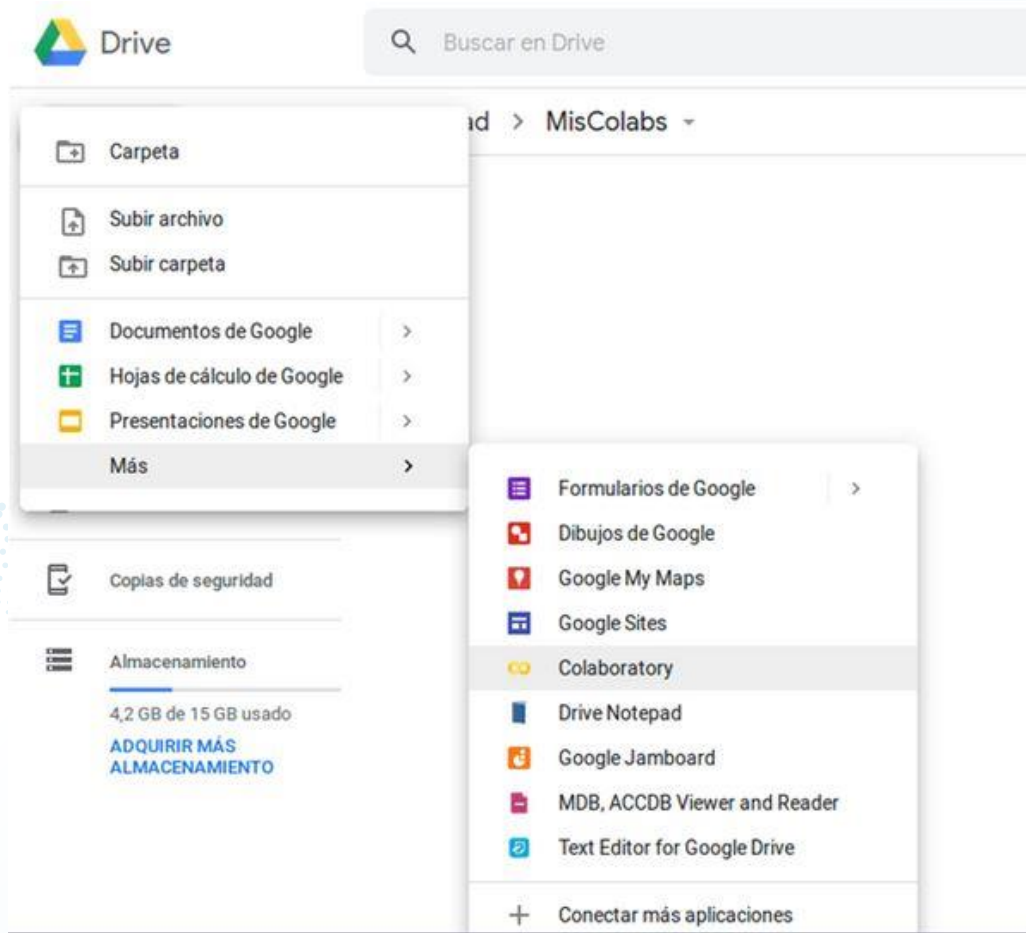
Permite escribir y ejecutar código de Python en un navegador. Se necesita:

- Cuenta en google
- Ir al siguiente url:

<https://colab.research.google.com/notebooks/welcome.ipynb>

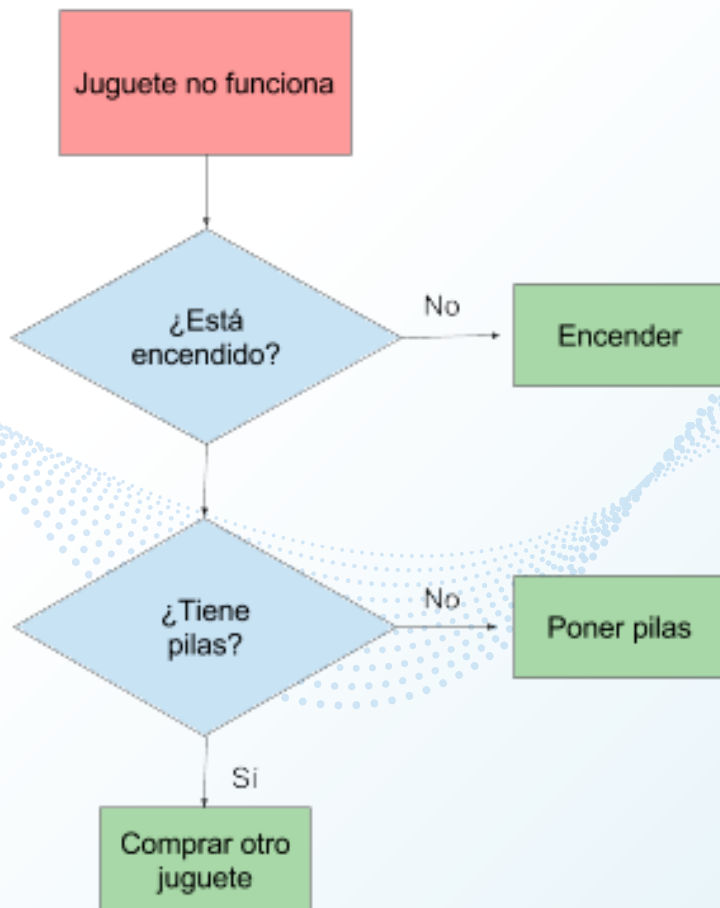


Colaboratory



Algoritmo

Conjunto de instrucciones o reglas definidas y no-ambiguas, ordenadas y finitas que permite, típicamente, solucionar un problema.

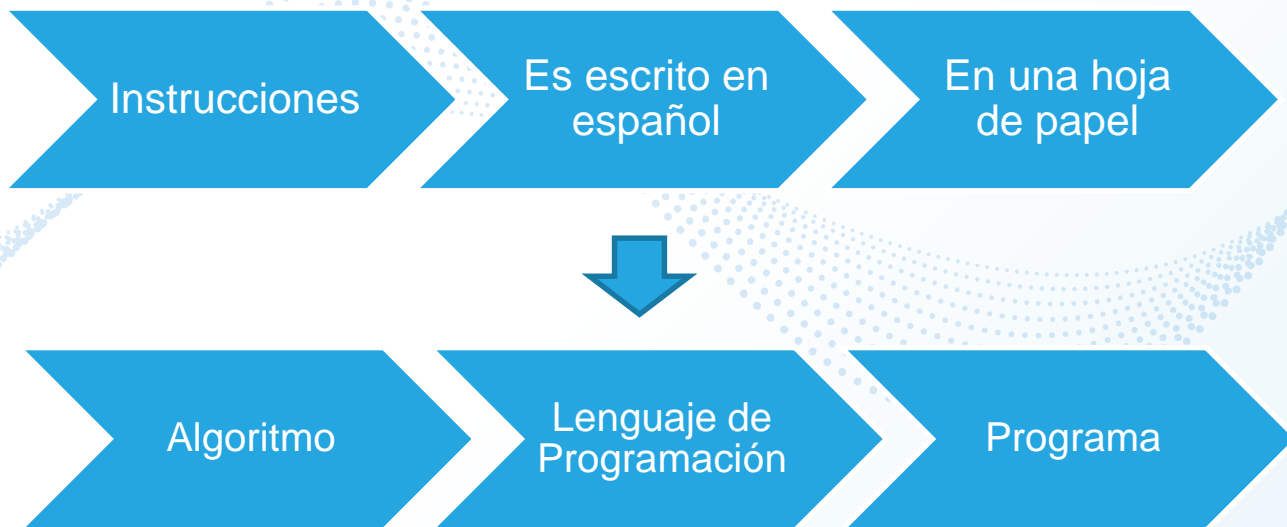


Programación

Es la implementación de un algoritmo en un determinado lenguaje de programación, para realizar un programa.



Ejemplo: Cómo armar una bicicleta



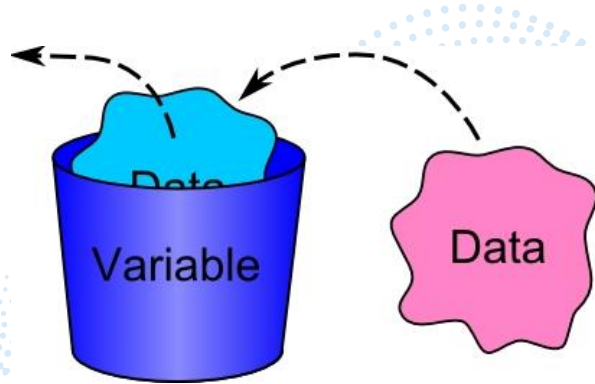
PROGRAMA

Componentes de un programa:

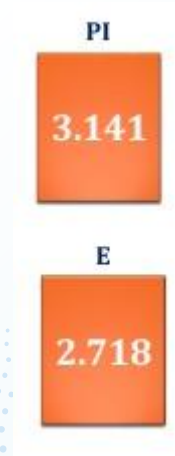
- Variables / Constantes
- Operadores
- Estructuras



Variables y Constantes



Una variable es un lugar donde almacenar un dato, tiene un nombre, un valor y un tipo.



Una constante es un valor fijo que no puede ser alterado.

Variables

`nombre = "Juan Carlos";`

Nombre de la
Variable

Valor Inicial

Guardar *datos y estados*.

Asignar valores de una variable a otra.

Representar valores dentro de una expresión matemática.

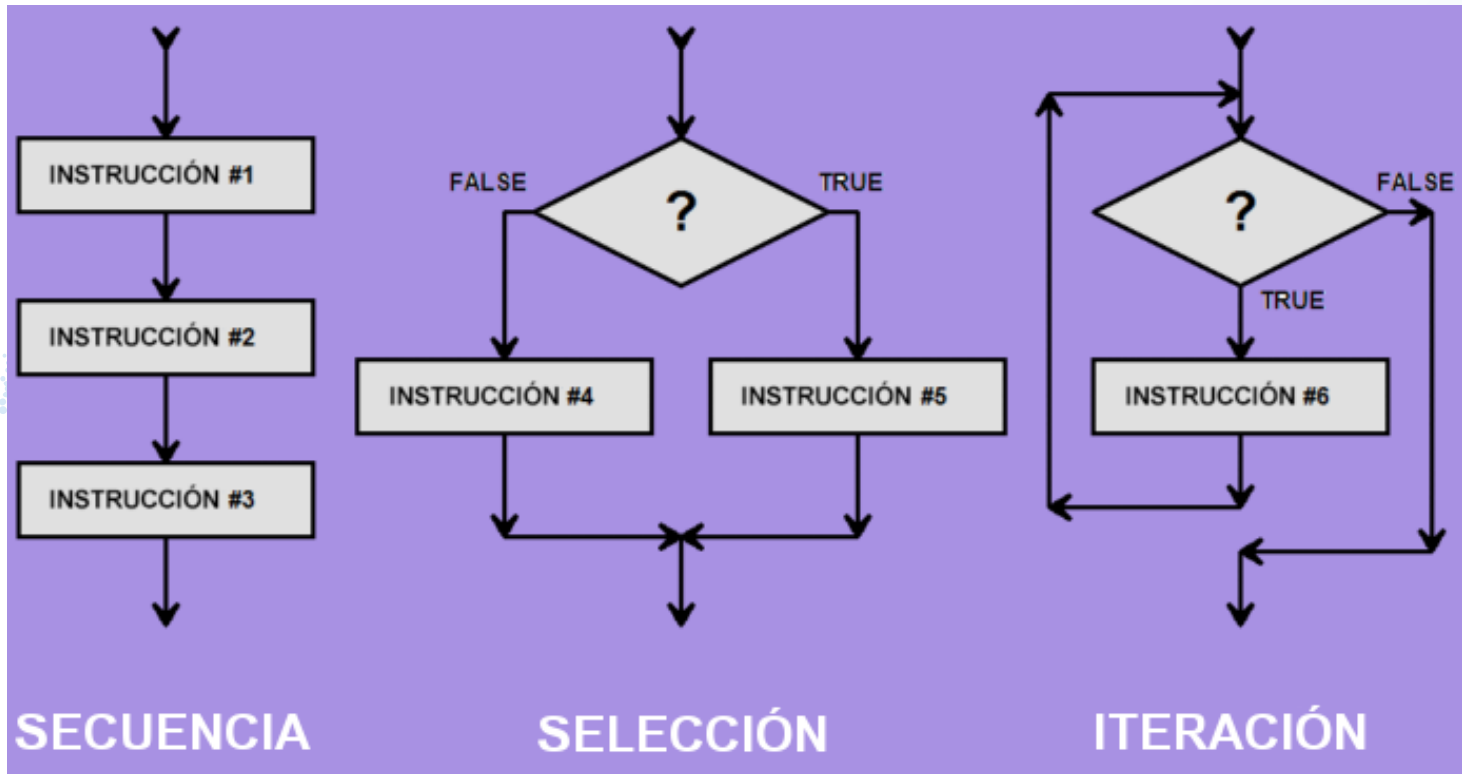
Mostrar valores por pantalla.

Operadores

Operador	Operación	Ejemplo	Resultado
**	Potencia	4**3	64
*	Multiplicación	8.25*7	57.75
/	División	15/4	3.75
+	Suma	125.78 + 62.50	188.28
-	Resta	65.30 - 32.33	32.37
Mod	Módulo (residuo)	15 Mod 2	1
Div	División Entera	17 Div 3	5

Operador	Operación	Ejemplo	Resultado
=	Igual que	"hola"= "ola"	Falso
<>	Diferente	"a" <> "b"	Verdadero
<	Menor que	7 < 15	Verdadero
>	Mayor que	22 > 11	Verdadero
<=	Menor o igual que	15 <= 22	Verdadero
>=	Mayo o igual que	33 >= 20	Verdadero

Estructuras de Control



1. Estructuras de Control - Secuencia

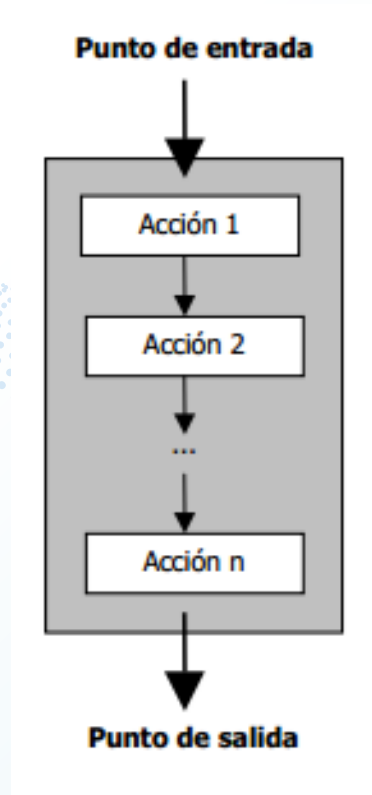
`a = 23`

`b = 12`

`c = 0`

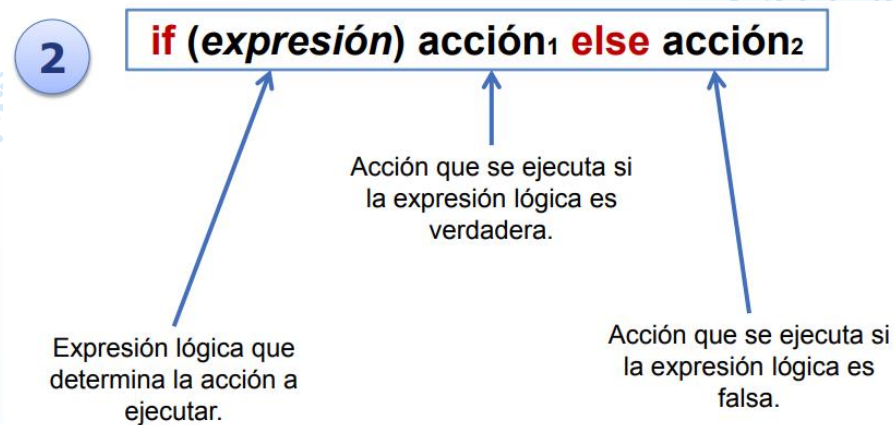
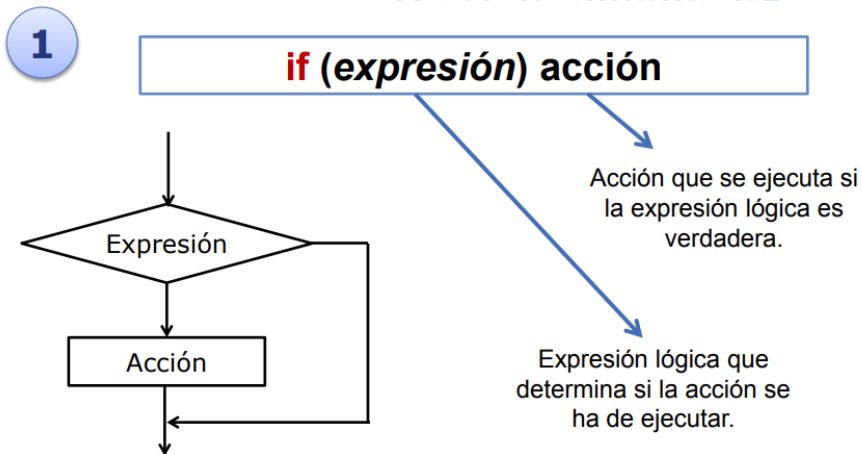
`c = a + b`

`print(c)`



2. Estructuras de Control - Selección

Sentencia IF (SI)



2. Estructuras de Control - Selección

Sentencia IF (SI)

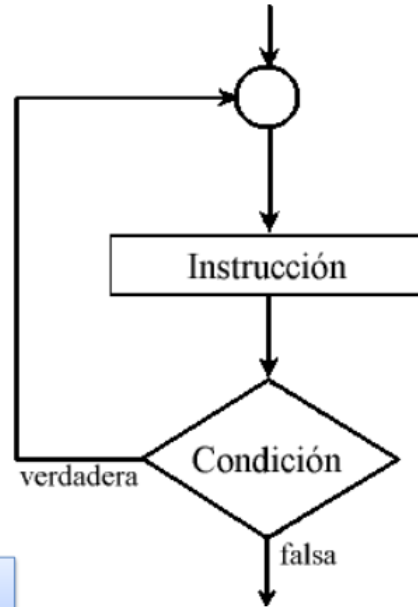
```
if (nombre=="María"):
    Print("Bienvenida María")
else:
    Print("Bienvenidos a todos")
```


3. Estructuras de Control - Iteración

SENTENCIA WHILE

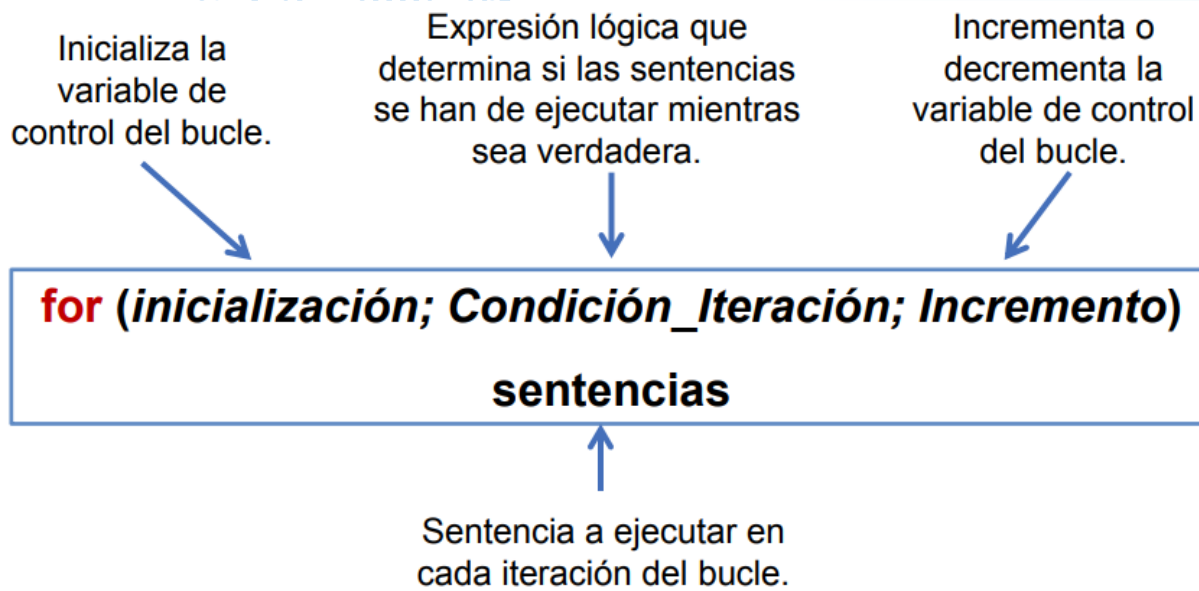
SENTENCIA FOR

SENTENCIA DO-WHILE



3. Estructuras de Control - Iteración

Sentencia FOR



3. Estructuras de Control - Iteración

Sentencia FOR

```
For i in range(3)  
    Print(f"Bienvenida {i}")
```

Funciones

Nombre de la función

Argumentos

Código de la función

```
def suma(a,b):  
    resultado = a+b  
    return resultado  
  
print(suma(5,6))
```

Invocación

Valor de retorno

The diagram illustrates the components of a function definition and its invocation. A central black box contains the code. Red arrows point from the labels 'Nombre de la función' to 'suma' and 'Argumentos' to 'a,b'. A blue arrow points from 'Código de la función' to the 'def' keyword. A purple arrow points from 'Invocación' to the 'print' statement. A pink arrow points from 'Valor de retorno' to the 'return' statement.

Uso de Librerías

Import math as mt

Print(mt.fsum([5,6]))



Requests



Pillow

tqdm



matplotlib



SQLAlchemy

django



Keras

NLTK 3.4



Gracias!

Alguna pregunta?

- Maria.limaylla@gmail.com