

***{desafío}***  
***latam\_***



# Introducción a la programación \_

**Sesión Presencial 2**

# Itinerario

## Activación de conceptos

## Desarrollo Desafío

## Panel de discusión

# Activación de conceptos

# ¿Qué versión de Python debemos utilizar?

- 2.7
- 3.7
- Se puede usar cualquier versión
- Ninguna de las anteriores

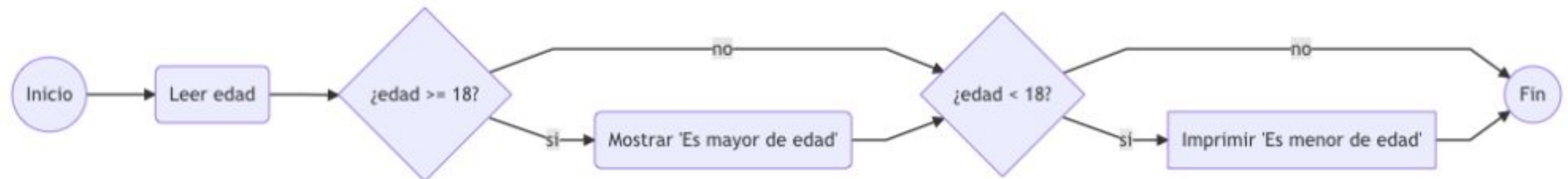
# ¿Qué es un algoritmo?

- Bloque de código que ejecuta una acción.
- Programa simple que se ejecuta en la terminal.
- Serie finita de pasos para resolver un problema.
- Error en el programa.

# ¿A qué corresponde la imagen?

## ¿Cuántas decisiones hay?

- Un diagrama de flujo
- Un programa
- Un algoritmo
- Ninguna de las anteriores



# ¿A qué corresponden las palabras “gatito” y “33”?

- Son variables
- Son el nombre del script
- Son argumentos
- “gatito” es el nombre de una variable y “33” su valor

```
Python 3.6.5 [Anaconda, Inc.] (default, Mar 29 2018, 13:32:41) [MSC v.1900  
64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>> mi_script.py gatito 33
```

# Python, características e importancia

- Lenguaje flexible y potente.
- Buena elección para comenzar a programar.
- Relevante en la industria.
- De sintaxis y lectura sencilla, siguiendo algunas normas.



# Trabajando desde la consola

- Desde Linux/Mac: Abrir terminal, y escribir “python”.
- Desde Windows: Abrir Anaconda prompt, y escribir “python”.

```
[isz:~] isz% python
Python 3.6.5 |Anaconda, Inc.| (default, Apr 26 2018, 08:42:37)
[GCC 4.2.1 Compatible Clang 4.0.1 (tags/RELEASE_401/final)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

# Editor de texto

- Crear archivo con extensión “.py”
- Ejecutar script

```
[isz:~/Desktop] isz% python mi_primer_programa.py  
[isz:~/Desktop] isz% █
```

# Variables

- Tienen un nombre y un valor.
- Siguen convenciones.
- Pueden ser de distinto tipo, dependiendo del valor asignado.
- Cada variable, corresponde a un **objeto**, y tienen métodos asociados, dependiendo del tipo de objeto.

```
1 numero = 20
2 soy_una_variable = "Bienvenidos a Introducción a Python"
3
4 # Esto no es correcto
5 usar espacios = "No es posible usar espacios para nombrar una variable"
```

File "<ipython-input-1-22634ce40c83>", line 5

```
    usar espacios = "No es posible usar espacios para nombrar una variable"
      ^
```

SyntaxError: invalid syntax

```
1 # Se debe respetar mayúsculas y minúsculas.
2 # Las variables por convención comienzan con minúscula.
3 print(Numero)
```

-----  
NameError Traceback (most recent call last)

<ipython-input-2-dd55f125c425> in <module>()

```
    1 # Se debe respetar mayúsculas y minúsculas.
    2 # Las variables por convención comienzan con minúscula.
----> 3 print(Numero)
```

NameError: name 'Numero' is not defined



# Strings

- Palabras o cadenas de texto entre comillas.
- Se pueden concatenar usando operador "+".
- Se les puede aplicar funciones como len(), upper() y count().
- Se pueden interpolar usando ".format()".

```
1 nombre = 'Carlos'
2 apellido = 'Santana'
```

```
1 # Concatenación
2 print("Mi nombre es " + nombre + " " + apellido)
```

Mi nombre es Carlos Santana

```
1 # Interpolación
2 print("Mi nombre es {} {}".format(nombre, apellido))
```

Mi nombre es Carlos Santana

```
1 print(nombre.count("a"))
2 print(len(apellido))
3 print(apellido.upper())
```

1  
7  
SANTANA

# Integers

- Números enteros.
- Se les puede aplicar operaciones matemáticas respetando la precedencia.
- No pueden concatenarse directamente a un String (Tener en cuenta cuando se usa input o argv).

```
1 a = 10
2 b = 2
3 print(a * b)
```

20

```
1 print(a / b + 15)
```

20.0

```
1 cadena = "2"
2 print(cadena + 2)
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-9-b3bb090f63c1> in <module>()
      1 cadena = "2"
----> 2 print(cadena + 2)

TypeError: must be str, not int
```

# Otros tipos de dato/objeto

- Float: Números decimales.
- Boolean: True o False.

## Métodos que se pueden aplicar a los objetos

- type()
- Str(), int(), float()

# Métodos

- Aprender a escribir y leer documentación.
- Método nativo / Método asociado a un tipo de objeto.
- Parámetros y retorno.

```
Signature: len(obj, /)
Docstring: Return the number of items in a container.
Type:      builtin_function_or_method
```

```
Docstring:
S.upper() -> str

Return a copy of S converted to uppercase.
```



# Manejo básico de flujo

“If” evalúa una condición booleana

Operador	Nombre	Ejemplo	Resultado
==	Igual a	2 == 2	true
!=	Distinto a	2 != 2	false
>	Mayor a	3 > 4	false
>=	Mayor o igual a	3 >= 3	true
<	Menor a	4 < 3	false
<=	Menor o igual a	3 <= 4	true

IMPORTANTE: No confundir “asignación” con “comparación”

# Manejo básico de flujo

“elif” y “else”:

```
valor1 = int(input("Ingrese valor 1"))
valor2 = int(input("Ingrese valor 2"))

if valor1 > valor2:
    print("valor1 {} es mayor".format(valor1))
elif valor1 == valor2:
    print("Ambos valores son iguales")
else:
    print("valor2 {} es mayor".format(valor2))
```

¿Cuál es la diferencia entre este código ...

```
1 palabra = input("Ingrese una palabra")
2 largo = len(palabra)
3
4 if largo <= 4:
5     print("Pequeña")
6 elif largo < 10:
7     print("Mediana")
8 else:
9     print("Larga")
```

... y este ?

```
1 palabra = input("Ingrese una palabra")
2 largo = len(palabra)
3
4 if largo <= 4:
5     print("Pequeña")
6
7 if largo < 10:
8     print("Mediana")
9 else:
10    print("Larga")
```



# Refactorización

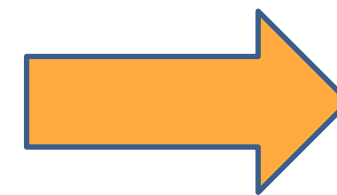
## Operadores lógicos

Permiten simplificar el uso de "if"

Operator	Nombre	Ejemplo	Resultado
&&	y (and)	false && true	Devuelve true si ambos operandos son true, en este ejemplo se devuelve false.
	o (or)	false    true	Devuelve true si al menos uno de los operandos es true, en este ejemplo devuelve true.
!	no (not)	!false	Devuelve lo opuesto al resultado de la evaluación, en este ejemplo devuelve true.

```
edad = 30
zurdo = True

if edad >= 30:
    if zurdo is True:
        print("Es zurdo y mayor de edad")
```



```
edad = 30
zurdo = True

if edad >= 18 and zurdo is True:
    print("Es mayor de edad y zurdo")
```

***/\* Desafío \*/***

# Panel de discusión

**{desafío}**  
**latam\_**

*Academia de  
talentos digitales*

[www.desafiolatam.com](http://www.desafiolatam.com)