



Modelos Analíticos con Python



Agenda

- 1. Variables, listas y diccionarios
- 2. Operaciones básicas y Operadores lógicos
- 3. Ciclos for y while
- 4. Funciones
- 5. Clases, métodos y atributos





Los conceptos básicos de Python

- 1. Variables, listas y diccionarios
- 2. Funciones
- 3. Clases, métodos y atributos



Un lenguaje de Programación viene con...

Tipos de datos básicos

Números, texto, variables de verdad (bool), etc.

Estructuras de datos

Podemos hacer "conjuntos" de cosas y agruparlas de formas específicas. ¡Y vienen con funcionalidades propias! Ejemplo: listas.

Funciones propias

Ejemplo: print(), type(), etc.



Variables y Tipos de Datos

En un lenguaje de programación, a los datos se los guarda en forma de variables. A cada variable debemos darle un nombre único que la identifique:

```
In []: a = 5
In []: un_nombre_cualquiera = 12.7
In []: b = 'Hola!'
In []: nueva_variable = True
Entero
Float
String
Boolean
```

A estas variables pueden se le pueden asignar distintos tipos de datos.

Listas (list)



Una estructura de dato muy importante en Python son las **listas**. Una lista consiste en una serie de elementos ordenados, es una estructura que es mutable

```
In [47]: lista_1 = [2, 4.7, True, 'Texto']
type(lista_1)

Out[47]: list
In [49]: lista_2 = [0, lista_1, 'Mas texto']
print(lista_2)

[0, [2, 4.7, True, 'Texto'], 'Mas texto']
```

Los elementos pueden ser de distintos tipos.

Incluso puede haber listas dentro de listas.

Nota: Las listas se definen con corchetes [] o con la clase lista = list()

Tuplas (tuple)



Las **tuplas** en python consiste en una serie de elementos, las cuales son inmutables, es decir, no se pueden modificar por indexación

```
In [1]: tuple1 = ("disco",10,1.2 )
tuple1

Out[1]: ('disco', 10, 1.2)

In [2]: type(tuple1)

Out[2]: tuple
```

Los elementos pueden ser de distintos tipos.

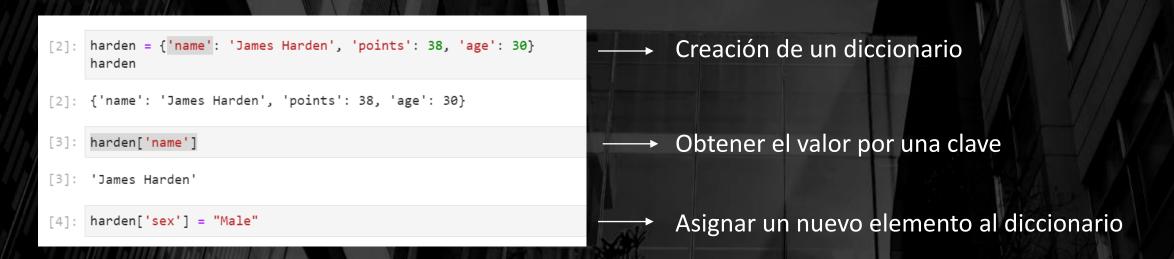
Incluso puede haber tuplas dentro de tuplas.

Las tuplas se definen con paréntesis () o con la clase t = tuple()

Diccionarios



Los Diccionarios en python son estructura que se componen de dos elementos Clave y valor

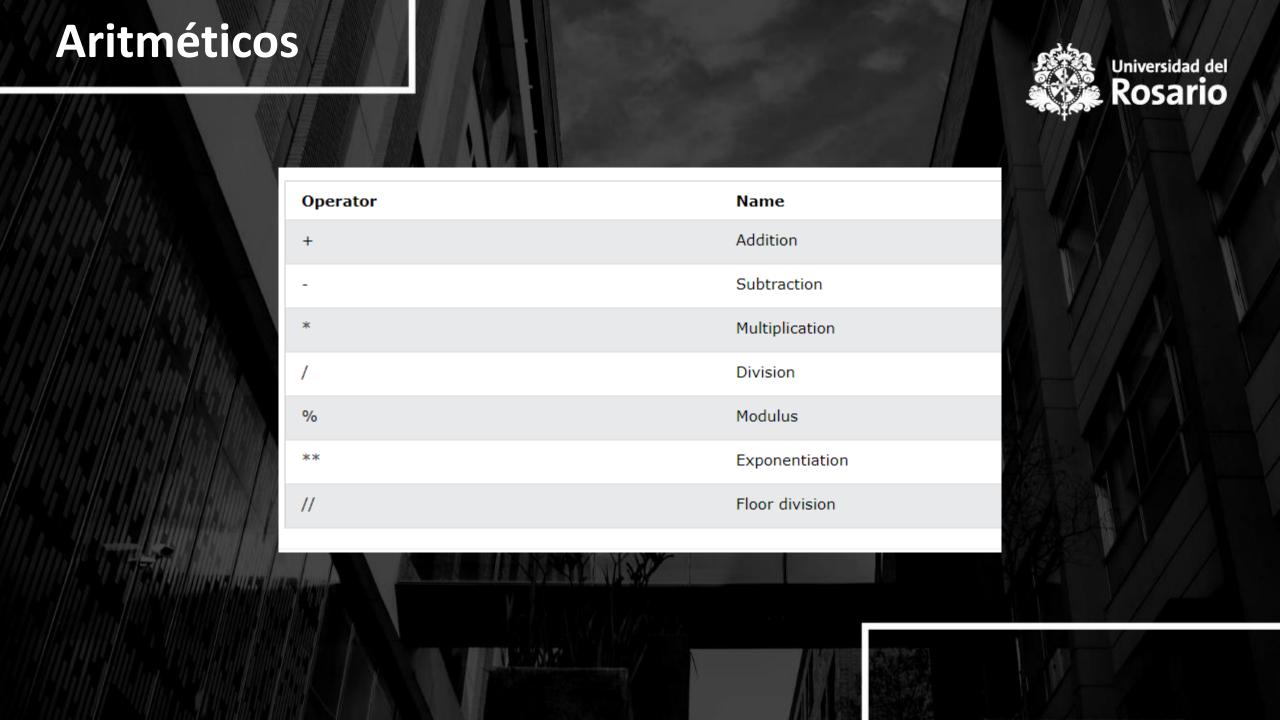


Los diccionarios se definen con llaves {key:value, key2:value2} o con la clase d = dict()





Operaciones Básicas: Operadores Aritméticos, de asignación de comparación y comparativos

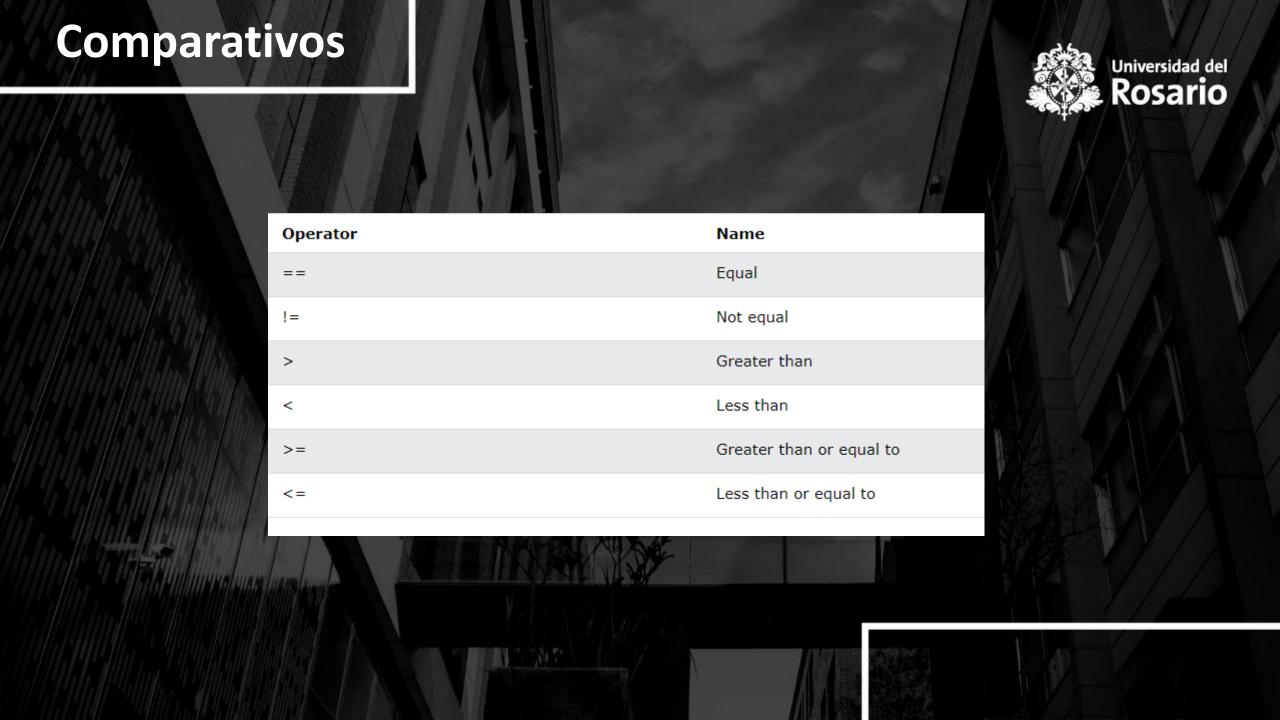


Asignación



Operator	Example	Same As
=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3
%=	x %= 3	x = x % 3
//=	x //= 3	x = x // 3
**=	x **= 3	x = x ** 3
&=	x &= 3	x = x & 3
=	x = 3	x = x 3
^=	x ^= 3	x = x ^ 3

Pueden explorarlos en: https://www.w3schools.com/python/python_ope rators.asp









El Zen de Python

- 1. Belleza sobre fealdad
- 2. Simple es mejor que complejo
- 3. Complejo es mejor que complicado
- 4. La capacidad de un código de ser leído cuenta
- 5. Debe existir una, y preferiblemente solo una, manera obvia de hacerlo
- 6. Ahora es mejor que nunca

Clases

Clase: son un especie de molde o la plantilla para crear objetos.

Atributos: son las "características" que van a tener los objetos.

Métodos: son acciones que pueden hacer nuestros objetos creados con una clase en especifica.



Gracias