
CLASE 1

Introducción a Python

Diego Sarceño Ramírez
4 de octubre de 2023

1. Shell de Python

Esta es una herramienta muy útil al estar aprendiendo a programar y, también, en el desarrollo de algún programa o proyecto. Esto es debido a que sirve para el testeado de trozos de código, funciones, objetos, etc. Debido a esto es, más que recomendable, necesario el saber utilizarla.

La forma de abrir la terminal en *Windows* de manera sencilla es abriendo el ejecutor de programas con la combinación de teclas *Windows + R* y escribir "cmd", esto abre la terminal de *Windows*, en la terminal se escribe "python" y luego de darle Enter, se abre la consola/shell de python.

En *Mac* y *Linux* se puede encontrar en el buscador de aplicaciones.

2. Calculadora

Python, es un lenguaje con una sintaxis muy intuitiva, lo que simplifica realizar operaciones como las de las calculadoras. Para esto se realizará una importación (más adelante en el curso se profundizará en los módulos) del módulo **math** que viene incluido con la instalación de python. La importación se realizará así:

```
1 import math
```

Este módulo incluye las funciones matemáticas más conocidas, tales como el seno, coseno, tangente, sus inversas, logaritmo, entre otras.

Ahora, se verá cómo realizar las operaciones básicas y la sintaxis para algunas funciones utilizando el paquete **math**.

```
1 # @Author: Diego Sarceno
2 # Date: 01.10.2020
3
4 import math
5
6 #----- CALCULADORA -----
7
8 # Suma y Resta
9 a + b
10 a - b
11
12 # Multiplicacion y Division
13 a * b
14 a / b
15
16 # Exponenciacion
```

```
17 a ** b
18
19 # Funciones del modulo MATH
20 math.sin(a)
21 math.cos(a)
22 math.exp(b)
23 math.log(a)
```

3. Tipos de Objetos

Como en todo lenguaje de programación y en casi cualquier ámbito de la vida cotidiana, existe una clasificación de los objetos a utilizar, los más utilizados en python y los que vienen por "defecto" son:

Tipos	Representación	Sintaxis
String o cadena (<i>str</i>)	Conjunto de caracteres	"Hola Mundo!" 'Hola Mundo!'
Entero (<i>int</i>)	Numero entero	12341
Flotante o Real (<i>float</i>)	Cualquier numero real	179.9195, 3.14, 3
Booleanos (<i>bool</i>)	Verdadero o Falso	True, False
Listas (<i>list</i>)	Objetos delimitados por corchetes	[<i>'a'</i> , 3, <i>True</i>]
Tuplas (<i>tuple</i>)	Objetos delimitados por parentesis	(<i>'a'</i> , 3, <i>True</i>)
Diccionarios (<i>dict</i>)	Asocia una llave a un valor en específico	{ <i>'key'</i> : <i>'value'</i> }

Cuadro 1: Tipos de Objetos

A diferencia de otros lenguajes, en python no es necesario especificar de que tipo de objeto se trata al momento de definir la variable.

3.1. Funciones basicas

Las dos primeras funciones, incluidas al descargar python, que utilizaremos de aquí en adelante muy repetidamente, son:

- Print: Esta función recibe como valor casi cualquier cosa, e imprime lo que dicha cosa haga o sea.

```
1 print('Hola Mundo!')
```

- Type: Esta función es muy útil puesto que nos dice el tipo del objeto que estamos tratando o utilizando.

```
1 type(4)
```

4. Strings o Cadenas

Las cadenas son objetos muy conocidos y muy útiles. Estos poseen operadores como la aritmética conocida y que hace cosas bastante intuitivas. Sumar dos cadenas es concatenar estas, multiplicar una cadena por un entero es concatenar dicha cadena con sí misma ese número de veces. Sin embargo, una de las características más interesantes e importantes de las cadenas es su iterabilidad, las cadenas son objetos iterables, esto quiere decir que podemos navegar por los caracteres que forma la cadena. Esto se hace de la siguiente forma:

```

1 a = 'Hola Mundo'
2 a[1:8:2]

```

Esto devolvera la siguiente cadena: **'oaMn'**. Lo que hace obvio el hecho de que hace cada parte, **a[caracter inicial:caracter final:paso]**, hay que recordar que para el caracter final el numero colocado siempre se le resta 1 puesto que el contador de caracteres/posiciones empieza en 0.

5. Diccionarios

En este objeto no se profundizara en el resto del curso, con lo que se definiria la sintaxis y como utilizarlos. Los diccionarios son una estructura de datos que permite almacenar valores de cualquier tipo como valores y acceder a ellos por medio de una referencia llamada llave. La sintaxis para definir un diccionario en python es:

```

1 diccionario = {'key':'value',
2 'key1':'value2',
3 'key3':'value3',
4 'key4':'value4'
5 }

```

El modo de acceder a los valores del diccionario es, como fue mencionado, por medio de la llave, es decir, se 'valua' el diccionario con el valor de la llave y este devolvera el valor que almacena la llave. Usando el diccionario anterior ejecute el siguiente comando:

```

1 print(diccionario['key1'])

```

Lo que devolvera: **value2**.

6. Números Complejos

Los números complejos son algo indispensable no solo para los matemáticos, también para los físicos; por ende, la aritmética de los números complejos viene incluida en python, entonces, podemos utilizar la aritmética vista en la sección 2. Tomando $u = a + bi$ y $w = c + di$

$$\begin{aligned}
 \operatorname{Re}\{u\} &= a & \operatorname{Im}\{u\} &= b \\
 \bar{u} &= a - bi \\
 u \pm w &= (a \pm c) + (b \pm d)i \\
 uw &= (ac - bd) + (bc + ad)i \\
 u/w &= \frac{ac + bd}{c^2 + d^2} + \frac{bc - ad}{c^2 + d^2}i \\
 |u| &= \sqrt{a^2 + b^2}
 \end{aligned}$$

La parte de aritmética básica, para los números complejos, es igual que para los números reales, pero se añaden ciertos métodos específicos para los numeros complejos, estos son: (Cabe recalcar que en python la unidad imaginaria es representada con la letra j)

```
1 # @Author: Diego Sarceno
2 # Date: 08.10.2020
3
4 #-----CALCULADORA DE COMPLEJOS-----
5
6 # Creacion de complejos
7 u = 3 + 4j
8 u = complex(3,4)
9
10 # Parte REAL y parte IMAGINARIA
11 pRe = u.real
12 pIm = u.img
13
14 # Modulo y Conjugado
15 mod = abs(u)
16 conj = u.conjugate()
```