

Variables simples y complejas en Python

1) Datos simples

Los números enteros son aquellos que no tienen decimales, tanto positivos como negativos (además del cero). En Python se pueden representar mediante el tipo int (de integer, entero) o el tipo long (largo).

```
# type(entero) daría int
entero = 23
```

```
# type(entero) daría long
entero = 23L
```

Los números reales son los que tienen decimales. En Python se expresan mediante el tipo float.

```
real = 0.2703
```

El tipo de dato carácter.

```
caracter = "a"
```

El tipo de dato cadena de caracteres.

```
cadena = "Mi cadena"
```

El tipo de dato booleano puede tomar los valores de verdadero (True) o falsedad (False)

```
booleano = True
```

2) Datos complejos

Tupla

Una tupla es una colección de datos de tipos diferentes que no se pueden modificar una vez creados. Para declarar una tupla, se usan paréntesis dentro de los cuales deben separarse por comas los elementos que la forman:

```
>>> tupla = (22, "hola kitties" , 2.5)
>>> tupla
(22, 'hola kitties', 2.5)
>>>
```

Los elementos de una tupla son accesibles a través del índice que ocupan en la misma siendo 0 (cero) el índice del primer elemento:

```
>>> tupla[0]
22
>>>
```

len() nos regresa el número de elementos de la misma.

```
>>> len(tupla)
3
>>>
```

Listas

Una lista es similar a una tupla sólo que con la gran diferencia de que permite modificar los datos una vez creados y en lugar de ir entre paréntesis va entre corchetes:

```
>>> mi_lista = ["hola kitties", 22, "hago listas" , 2.5]
```

A las listas se accede igual que a las tuplas por su número de índice:

```
>>> print mi_lista[1]
22
>>> print mi_lista[-2]
hago listas
>>>
```

Además, a diferencia de las tuplas permiten agregar nuevos valores:

```
>>> mi_lista.append('Nuevo Dato')
>>> print mi_lista
['hola kitties', 22, 6, 2.5, 'Nuevo Dato']
>>>
```

Diccionarios

Un conjunto de claves(palabras) y valores(definiciones).

```
mi_diccionario = {'paco' : 44, 'pepe' : 55, 'antonia' : 'muchos menos'}
>>> mi_diccionario['antonia']
'muchos menos'
>>>
```

un diccionario permite borrar cualquier entrada:

```
del(mi_diccionario['paco'])
```

Y como en las listas modificar valores:

```
>>> mi_diccionario['pepe'] = 'no se sabe'
>>> print mi_diccionario
{'antonia': 'muchos menos', 'pepe': 'no se sabe'}
>>>
```