

## CHECKPOINT 4

### ***Cuál es la diferencia entre una lista y una tupla en Python?***

Aunque ambas pertenecen a las colecciones de datos, tienen diferencias importantes entre ellas.

Las listas se definen utilizando corchetes [ ] : lista = ['a','b','c'], son mutables, se pueden añadir, modificar o eliminar elementos. Poseen métodos (append(), pop(),remove() ) que permiten hacer estas operaciones de forma sencilla. Se utilizan cuando se necesita una estructura de datos flexible.

Las tuplas se definen utilizando paréntesis (): tupla = ('a','b','c'), son inmutables, por lo tanto no se pueden modificar. También son más rápidas que las listas y se utilizan para almacenar datos cuyo valor no va a cambiar.

### ***¿Cuál es el orden de las operaciones?***

El orden de las operaciones matemáticas se conoce con el nombre de PEMDAS ( PEDMAS también es válido ) y es:

Paréntesis ()

Exponente \*\*

Multiplicación \*

División /

Suma +

Resta -

### ***¿Qué es un diccionario Python?***

Es una estructura para almacenar datos en formato par [clave : valor], que forma parte de las colecciones de datos.

Son mutables, por lo tanto permiten modificar sus elementos o valores.

La clave ha de ser única, mientras que el valor puede repetirse.

Se pueden anidar diccionarios dentro de diccionarios, lo cual las convierte en una estructura de datos muy flexible y potente.

Poseen métodos propios para trabajar con ellos de manera sencilla: keys(), values(), items()

### ***¿Cuál es la diferencia entre el método ordenado y la función de ordenación?***

La diferencia entre `sort()` y `sorted()` esta en como se aplican y trabajan con los datos.

`Sort()` : En un método que se aplica a objetos de tipo lista[], lo que hace es ordenar alfabéticamente la lista, modificándola. Devuelve el valor 'none'.

`Sorted()`: Es una función que se puede aplicar sobre cualquier objeto iterable, devuelve un nuevo objeto (copia) ordenado, pero no modifica el original

### ***¿Qué es un operador de reasignación?***

Es un tipo de operador que permite realizar cálculos de forma acumulativa ( sumando su valor en una variable) en una sola línea. Por ejemplo:

`numero = 6`

`numero *=2` seria igual a `→ numero = numero *2`