

Ejercicios teóricos Python

1.- ¿Cuál es la diferencia entre una lista y una tupla en Python?

Desde el punto de vista de la sintaxis las listas se definen con corchetes '[''] y las tuplas con paréntesis '()'. Respecto a la funcionalidad, las listas son variables mutables y las tuplas son inmutables. Esto quiere decir que, en el caso de las listas, se pueden cambiar la mayoría de los aspectos como el orden o introducir/quitar elementos, haciendo que sean variables dinámicas que pueden ir cambiando mediante el uso del programa. Por otro lado, las tuplas no se pueden alterar mediante métodos. Si queremos cambiar un tupla tendremos que guardar una nueva en la misma variable o en otra nueva, por lo que hace falta volver a definir la variable de la tupla.

```
list = ['Manzana', 'Pera']  
tuple = ('Tarta de manzana', 'Pera al vino')
```

2.- ¿Cuál es el orden de las operaciones?

El orden de operaciones es el protocolo que define los pasos a seguir a la hora de resolver una ecuación matemática. Por ejemplo:

$$result = 4^2 - 3(2 + 5)/2$$

Según la regla BODMAS, el orden de las operaciones es la siguiente de arriba abajo:

- Brackets => Paréntesis o corchetes
- Order => Orden (potencias y raíces)
- Divisions => Divisiones
- Multiplication => Multiplicaciones
- Addition => Sumas
- Subtraction => Restas

3.- ¿Qué es un diccionario Python?

Los diccionarios en Python son estructuras de datos basados en pares clave-valor. Se podría decir que son como listas, pero con una complejidad de estructuración un poco mayor que permite una mejor organización de los datos. Permiten facilidad para almacenar datos y recuperarlos mediante el uso de la clave única. Los diccionarios, de la misma forma que las listas, son dinámicos o mutables, de forma que, pueden crecer y decrecer. También son anidados, permitiendo a un diccionario tener más diccionarios en su campo valor.

```
dict = {  
    'Manzana': 'Tarta de manzana',  
    'Pera': 'Pera al vino'  
}
```

4.- ¿Cuál es la diferencia entre el método ordenado y la función de ordenación?

La diferencia es la mutabilidad, mientras el método 'sort()' muta la lista, quiere decir que modifica la lista, el método 'sorted()' devuelve una lista ordenada nueva, pero sin alterar la lista original. Esto último quiere decir que si al usar 'sorted()' la nueva lista no se crea en alguna variable, la lista se perderá. En cambio con 'sort()' en el momento en el que se ejecuta el método la lista ya ha cambiado.

5.- ¿Qué es un operador de reasignación?

Un operador de reasignación es cuando se le declara un valor a una variable que ya contiene una variable. De forma que el valor anterior se borra y se pierde para ser reemplazado por el nuevo. Estos operadores se emplean mayoritariamente para cambiar el valor de variables inmutables como, por ejemplo, las tuplas. El operador de reasignación más común es el '='.

```
tuple += ('Bizcocho de limon',)
print(tuple)
```