

Bruno Miguel Díaz Huallullo

1. En listas, podemos utilizar a `.values()` y este nos va a mostrar una vista con todos los valores que contiene la lista creada.

Falso, en listas no existe `.values()` solo en diccionarios

2. En listas se puede realizar suma de listas utilizando el operador de `+` el cual me va a sumar cada elemento con el índice correspondiente al de la otra lista para tener finalmente una sola lista.

Falso, el operador `+` no suma cada elemento, si no une las listas

3. En Python, una lista no está limitada a que pueda tener elementos de un solo tipo de dato; dentro de una misma lista es posible almacenar valores enteros, flotantes, incluso otras listas o diccionarios.

Verdadero, las listas sí pueden almacenar distintos tipos de datos

4. En los datos tipo string pueden recorrerse utilizando un bucle `for`, ingresando carácter por carácter. Con estos puede contar cuántas veces aparece un carácter, cambiar su formato o hasta agregar o cambiar strings basado en ciertas condiciones que tu pueda indicar en tu programa

Verdadero, los datos tipo string sí permiten que un bucle `for` las recorra carácter por carácter

5. Python es un lenguaje que debe utilizar la indentación o sangrado para delimitar los bloques de código en las clases o funciones, por lo que tenemos que verificar que las líneas que queremos que pertenezcan a la función estén correctamente indentadas y generalmente son 4 espacios.

```
def observer (...):
```

```
    print("Hi everyone!")
```

Verdadero, la indentación o sangrado muestra que esa línea de código pertenece a un bloque de código en clases o funciones

6. En la siguiente función, al hacer el llamado del mismo y ejecutar el programa. Este no tendrá ningún tipo de error o excepción (Leer cuidadosamente el fragmento de este código).

```
def multiplicador():
```

```
    return 4 + spam*5
```

```
print(multiplicador())
```

Falso, la variable “spam” no está definida

7. El Python, cuando trabajamos con `try` no va a producir ninguna excepción, siempre va a ejecutar el `except` dentro de la sentencia construida

Falso, `except` solo es ejecutado si ocurre una excepción dentro del bloque `try`

8. Al trabajar con funciones estas no pueden ser usadas o llamadas desde otras funciones.

Falso, las funciones sí pueden ser llamadas desde otras funciones