

Strings

En Python, un string (cadena de caracteres) es un tipo de datos que representa una secuencia de caracteres. Se puede crear un string usando comillas simples (") o comillas dobles ("). Por ejemplo:

```
mi_string = 'Hola, ¿cómo estás?'
```

Los strings son objetos inmutables en Python, lo que significa que no se pueden modificar después de ser creados. Cualquier operación que parezca que modifica un string, como la concatenación o la eliminación de caracteres, en realidad crea un nuevo string..

Hay varias operaciones y métodos que se pueden usar con los strings en Python. Aquí hay algunos ejemplos:

- Concatenación: los strings se pueden unir usando el operador +:

```
saludo = 'Hola'
nombre = 'Juan'
mensaje = saludo + ', ' + nombre
print(mensaje) # salida: Hola, Juan
```

- Acceso a caracteres: se puede acceder a un carácter específico de un string usando el operador de indexación [] y un índice numérico que indica la posición del carácter (los índices comienzan en cero):

```
• mi_string = 'Python'
• print(mi_string[0]) # salida: P
```

- Métodos de manipulación de cadenas: hay varios métodos que se pueden usar para manipular los strings en Python. Algunos de ellos son upper(), lower(), split(), join() y strip().

GUIA EJERCICIOS BASICOS CON STRINGS

1. Escribir una función que reciba un string y devuelva el mismo string todo en mayúsculas.
2. Escribir una función que reciba un string y devuelva el mismo string todo en minúsculas.
3. Escribir una función que tome dos strings y devuelva un nuevo string que contenga ambos strings concatenados, separados por un espacio.
4. Escribir una función que tome un string y devuelva el número de caracteres que tiene el string.
5. Escribir una función que tome un string y un carácter y devuelva el número de veces que aparece ese carácter en el string.
6. Escribir una función que tome un string y un carácter y devuelva una lista con todas las palabras en el string que contienen ese carácter.
7. Escribir una función que tome un string y devuelva el mismo string con los espacios eliminados
8. Escribir una función que reciba dos string (nombre y apellido) y devuelva un diccionario con el nombre y apellido, eliminando los espacios del comienzo y

el final y colocando la primer letra en mayúscula

9. Escribir una función que tome una lista de nombres y los una en una sola cadena separada por un salto de línea, por ejemplo: ["Juan", "María", "Pedro"]
-> "Juan\nMaría\nPedro".
10. Escribir una función que tome un nombre y un apellido y devuelva un email en formato "inicial_nombre.apellido@utn-fra.com.ar".
11. Escribir una función que tome una lista de palabras y devuelva un string que contenga todas las palabras concatenadas con comas y "y" antes de la última palabra. Por ejemplo, si la lista es ["manzanas", "naranjas", "bananas"], el string resultante debería ser "manzanas, naranjas y bananas"..
12. Escribir una función que tome un número de tarjeta de crédito como input, verificar que todos los caracteres sean numéricos y devolver los últimos cuatro dígitos y los primeros dígitos como asteriscos, por ejemplo: "**** *
**** 1234".
13. Escribir una función que tome un correo electrónico y elimine cualquier carácter después del símbolo @, por ejemplo: "usuario@gmail.com" -> "usuario".
14. Escribir una función que tome una URL y devuelva solo el nombre de dominio, por ejemplo: "https://www.ejemplo.com.ar/pagina1" -> "ejemplo"..
15. Escribir una función que tome una cadena de texto y devuelva solo los caracteres alfabéticos, eliminando cualquier número o símbolo, por ejemplo: "H0l4, m4nd0!" -> "Hl, mnd."

16. Escribir una función que tome una cadena de texto y la convierta en un acrónimo, tomando la primera letra de cada palabra, por ejemplo: "Universidad Tecnológica Nacional Facultad Regional Avellaneda" -> "UTNFRA".
17. Escribir una función que tome un número binario y lo convierta en una cadena de 8 bits, rellenando con ceros a la izquierda si es necesario, por ejemplo: "101" -> "00000101".
18. Escribir una función que tome una cadena de caracteres y reemplace todas las letras mayúsculas por letras minúsculas y todas las letras minúsculas por letras mayúsculas, por ejemplo: "HoLa" -> "hOIA"
19. Escribir una función que tome una cadena de caracteres y cuente la cantidad de dígitos que contiene, por ejemplo: "1234 Hola Mundo" -> contiene 4 dígitos.
20. Escribir una función que tome una lista de direcciones de correo electrónico y las una en una sola cadena separada por punto y coma, por ejemplo: ["juan@gmail.com", "maria@hotmail.com"] -> "juan@gmail.com;maria@hotmail.com".
21. Crear una función que reciba como parámetro un string y devuelva un diccionario donde cada clave es una palabra y cada valor es un entero que indica cuántas veces aparece esa palabra dentro del string.