Hoja de trabajo 3

Manipulación de Strings

1. Prueba las siguientes instrucciones en el Shell de Python y ve documentando en la tabla de abajo qué hace cada una de las instrucciones o métodos de manipulación de strings.

 “hola”\*3

 palabra=”soy un string”

 print(len(palabra))

 print(palabra[0])

 print(palabra[-1])

 print(palabra[2])

 print(palabra[1:5])

 print(palabra[4:])

 print(palabra[ :3]

 print(palabra[0:6:2])

 print(palabra[:6:2])

 print(palabra[0::2])

 print(palabra[::-1])

 print(palabra.capitalize())

 print(palabra)

Nota: Los strings son inmutables y se demuestra en lo que pasó con las dos últimas instrucciones, ¿qué significa que sea inmutable?

 print(palabra.count('o'))

 print(palabra.endswith("tring"))

 print(palabra.endswith("a"))

 "tr" in palabra

 print(palabra.find(“y”))

 print(palabra.find("ai"))

 print(palabra.islower())

 otra=palabra.upper()

 print(otra)

 print(otra.islower())

 print(otra.isupper())

 print(palabra.replace(“s”, “c”))  print(palabra)

**1**

MCC Fabiola Uribe Plata

 unaMas=palabra.replace(“s”, “c”)

 print(unaMas)

 palabra.startswith("s")

 palabra.startswith("c")

 modificada=palabra.capitalize()

 print(modificada)

 print(modificada.swapcase())

 print(palabra.title())

|  |  |
| --- | --- |
| **Instrucción, función o método** | **Sirve para** |
| **Palabra=soy un string** |  |
| **‘hola’\*3** | **holaholahola** |
| **Print(len(palabra))** | **13** |
| **print(palabra[0])** | **s** |
| **print(palabra[-1])** | **g** |
| **print(palabra[2])** | **y** |
| **print(palabra[1:5])** | **oy u** |
| **print(palabra[4:])** | **un string** |
| **print(palabra[ :3]** | **soy** |
| **print(palabra[0:6:2])** | **syu** |
| **print(palabra[:6:2])** | **syu** |
| **print(palabra[0::2])** | **syu tig** |
| **print(palabra[::-1])** | **gnirts nu yos** |
| **print(palabra.capitalize())** | **Soy un string** |
| **Print(palabra)** | **soy un string** |
| **Print(palabra.count(‘o’))** | **1** |
| **Print(palabra.endswith(‘tring’))** | **True** |
| **Print(palabra.endswith(‘a’))** | **False** |
| **‘tr’ in palabra** | **True** |
| **Print(palabra.find(‘y’))** | **2** |
| **Print(palabra.find(‘ai’))** | **-1** |
| **Print(palabra.islower())** | **True** |
| **Print(palabra.lower())** | **soy un string** |
| **Otra=palabra.upper()** |  |
| **Print(Otra)** | **SOY UN STRING** |
| **Print(Otra.islower())** | **False** |
| **Print(Otra.isupper())** | **True** |
| **Print(palabra.replace(‘s’,’c’))** | **coy un ctring** |
| **unaMas=palabra.replace(‘s’,’c’))** |  |
| **Print(unaMas)** | **coy un ctring** |
| **Palabra.startswith(‘s’)** | **True** |  |
| **Palabra.startswith(‘c’)** | **False** |  |
| **Modificada=palabra.capitalize()** |  |  |
| **Print(Modificada)** | **Soy un string** |  |
| **Print(palabra.title())** | **Soy Un String** |  |

2. Crea la función palindrome, la cual recibe un string, devolverá True si este string es palindrome. Es un palíndrome cuando la palabra se lee igual en ambos sentidos, por ejemplo “anita lava la tina”. Debe funcionar para palabras o frases. ¿Cómo puedo quitarle los espacios al string?

**2**

MCC Fabiola Uribe Plata