

Métodos String

string.capitalize()

El método `capitalize()` devuelve una copia de la cadena con su primera letra en mayúscula.

```
texto = "hugo araya carrasco"
print (texto.capitalize())
Hugo Araya Carrasco
```

string.endswith(sufijo)

El método `endswith()` devuelve `True` si la cadena termina con el sufijo especificado.

```
texto = "mi diario python"
print(texto.endswith("python"))
True
print(texto.endswith("thon"))
True
Print(texto.endswith("py"))
False
```

string.find(sub)

Devuelve el índice más bajo de la cadena en la subcadena `sub`.

```
texto = "mi diario python"
texto.find("mi")
0
texto.find("m")
0
texto.find("i")
1
texto.find("python")
10
texto.find("py")
10
texto.find("p y")
-1
```

string.format()

Nos permite realizar una operación de formato de cadena. La cadena en la que se llama a este método puede contener texto literal o campos de reemplazo delimitados por llaves {}. Cada campo de reemplazo contiene el índice numérico de un argumento posicional o el nombre de un argumento de palabra clave. Devuelve una copia de la cadena donde cada campo de reemplazo se reemplaza con el valor de la cadena del argumento correspondiente.

```
print("La suma de 1 + 2 es {0}".format(1+2))  
'La suma de 1 + 2 es 3'
```

string.index(sub)

El método index es muy similar al método find. Con la diferencia de que cuando no se encuentra la subcadena, index lanza un ValueError.

```
texto = "mi diario python"  
texto.index("mi")  
0  
texto.index("PYTHON")  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
ValueError: substring not found
```

string.isalpha()

Devuelve verdadero si todos los caracteres de la cadena son alfanuméricos y hay al menos un carácter, de lo contrario es falso.

```
texto = "mi diario python"  
print(texto.isalpha())  
False  
print("midiariopython".isalpha())  
True
```

Nota: "mi diario python" a lanzado False, porque los espacios no son un carácter alfanumérico.

string.isdigit()

El método isdigit() devuelve True si todos los caracteres de la cadena son dígitos.

```
texto = "mi diario python"  
digitos = "12345"
```

```
print(texto.isdigit())
False
print(digitos.isdigit())
True
```

string.isspace()

El método `isspace()` devuelve True si solo hay caracteres de espacio en blanco.

```
print(" ".isspace())
True
print("  a".isspace())
False
```

string.lower()

El método `lower` devuelve una copia de la cadena con todos sus caracteres en minúsculas.

```
print("Hola Mundo".lower())
'hola mundo'
print("PYTHON".lower())
'python'
```

string.upper()

El método `upper()` devuelve la una copia de la cadena con todos su caracteres en mayúsculas.

```
texto = "mi diario python"
print(texto.upper())
'MI DIARIO PYTHON'
```

string.lstrip(chars)

El método `lstrip` devuelve una copia de la cadena con los caracteres iniciales eliminados. El argumento `chars` es una cadena que especifica el conjunto de caracteres que se eliminarán.

```
web = "www.pythondiario.com"
web.lstrip("w.")
'pythondiario.com'
```

string.replace (old, new[, maxsplit])

Devuelve una copia de la cadena en la que se han sustituido todas las apariciones de old por new. Si se proporciona el argumento opcional maxsplit, sólo se sustituyen las primeras maxsplit apariciones.

```
texto = "Hola como estan todos lo digo todo"
print(texto.replace("todo","nada"))
Hola como estan nadas lo digo nada
```

rstrip ()

Devuelve una copia de la cadena con el espacio al final suprimido.

```
texto= "Hola"
print(texto.rstrip("a"))
Hol
```

split ([sep [,maxsplit]])

Devuelve una lista de las palabras de la cadena, usando sep como delimitador de palabras. Si se indica maxsplit, se devolverán como mucho maxsplit valores (el último elemento contendrá el resto de la cadena). Si no se especifica sep o es None, cualquier espacio en blanco sirve de separador.

```
texto = "Hola como estan todos lo digo todo"
print(texto.split())
['Hola', 'como', 'estan', 'todos', 'lo', 'digo', 'todo']
```

swapcase ()

Devuelve una copia de la cadena con las mayúsculas pasadas a minúsculas y viceversa.

```
texto = "HoLa"
print(texto.swapcase())
hOlA
```

Funciones útiles.

str()

Esta función convierte un número en string.

int()

Esta función convierte un string formado por números en un número entero.

Float()

Esta función convierte un string en un número real.

DESAFIOS

1. Para un numero N menor de 100. Mostrar la suma de los cuadrados de los números que están separados entre si cuatro posiciones.
2. Imprimir 10 veces la serie de números de 1 a 10.
3. Para un número N imprimir su tabla de multiplicar.
4. Identificar si la suma de los dígitos de un numero es par o impar.
5. Solicitar un número e imprimir los dígitos pares de este.
6. Los números de las claves de dos cajas fuertes están mezcladas en un número entero llamado clave maestra.
Determine ambas claves, la primera clave se construye con los dígitos impares de la clave maestra y la segunda con los pares.
Ejemplo: Clave Maestra= 12345, clave1=135, clave2=24.
7. Escribir un programa que pida (lea de teclado) dos palabras y diga si riman o no.
 - Si coinciden las tres últimas letras tiene que decir que riman.
 - Si coinciden sólo las dos últimas tiene que decir que riman un poco y
 - Si no, que no riman.