|  |  |
| --- | --- |
| **Función** | **Sirve para** |
| Print(Len()) | Imprime la longitud de string. |
| Print(palabra[0]) | Imprime la letra en la posición 0 de la string que está en palabra |
| print(palabra[1:5]) | Imprime las letras en la posición 1 y 5 |
| print(palabra[4:]) | Imprime las letras de la posición 4 al final |
| print(palabra[0:6:2]) o print(palabra[:6:2]) | Imprime las letras en la posición 0 6 y 2 |
| print(palabra[0::2]) | Imprime cada dos espacio. |
| print(palabra[::-1]) | Escribe la frase al revés |
| print(palabra.capitalize()) | Pone mayúscula a la primera |
| print(palabra.count('o')) | Cuenta las o que están en la palabra |
| print(palabra.endswith("tring")) | Comprueba si termina en tring y regresa un valor booleano |
| print(palabra.find(“y”)) | Imprime la posición de y en palabra |
| palabra.islower() | Checa si está en minúsculas y devuelve un valor booleano |
| palabra.upper() | Cambia la palabra a mayúsculas |

1. **¿Qué instrucción pondrías para “esconder” una palabra, es decir, tener otra con la misma cantidad, pero de asteriscos en vez de letras? Por ejemplo: Si la palabra es “fotosíntesis”, obtener su versión “escondida” como \*\*\*\*\*\*\*\*\*\*\*\***

palabra= “Mi casa es azul”

len(palabra)

palabraAst= “\*” x len(palabra)

1. **¿Qué instrucción de Python utilizarías para saber si un carácter es una vocal? Es una sola línea de código y utiliza una de las instrucciones que viste la clase pasada.**

def esVocal(caracter):  
 return caracter in “aeiouAEIOU”

1. **Una vez que tienes la respuesta anterior, ¿recuerdas cómo hiciste la función esVocal(carater) en tu examen rápido? ¿Cómo la escribirías ahora?**

def esVocal(carácter):

return caracter in “aeiouAEIOU”

1. **a= “hipopótamo” ¿Por qué a[3]= “s” nos marcaría un error? ¿Por qué no Podemos cambiar caracteres de un string?**
2. **¿Qué instrucción utilizarías para obtener sólo los caracteres de índice impar de un string?**

palabra[1::2]

1. **¿Qué instrucción utilizarías para obtener los últimos 3 caracteres de un string pero en sentido inverso? Por ejemplo, si el string es “girasol”, obtener “los”.**

palabra[:-4:-1]

1. **¿Qué instrucción utilizarías para obtener la cantidad de vocales que tiene un string?**

palabra.count(“a”)

1. **Una vez que tengas la respuesta anterior, escribe una función que se llame cuantasVocales(cadena) la cual recibe un string y regresa la cantidad de vocales que tiene el string de entrada.**

def cuantasVocales(cadena):  
 palabra=palabra.lower()  
 retorno palabra.count(“a”)+palabra.count(“e”)+palabra.count(“i”)+palabra.count(“o”)+palabra.count(“u”)

st=input(“Teclea palabra o frase”)

num=cuantasVocales(st)

print(“Tu string tiene”, num, “vocales”)

1. **Escribe una función llamada suprimeVocales(string) que reciba un string y nos devuelva un string con sólo las consonantes del string de entrada, por ejemplo, suprimeVocales(“hipopótamo”) la función nos devolvería el string “hpptm”**

def suprimeVocales(string):  
 palabra= palabra.lower()  
 palabra=palabra.replace(“a”, “”)  
 palabra=palabra.replace(“e”, “”)

palabra=palabra.replace(“i”, “”)

palabra=palabra.replace(“o”, “”)

palabra=palabra.replace(“u”, “”)

return palabra

st=input(“Teclea palabra o frase”)

print(suprimeVocales(st))

1. **En un programa, le pedimos al usuario que nos teclee la respuesta si o no, ¿qué instrucción utilizarías para que no importando si el usuario usó mayúsculas, minúsculas o una combinación de ellas para escribir si o no, el programa siempre identifique las respuestas?**
2. **Pensemos que estamos procesando un archivo de texto, ¿qué función de Python para manipular strings te serviría si quieres identificar palabras que comienzan con la letra “C”?**