

## GUÍA DE EJERCICIOS

### CADENAS DE CARACTERES

1. Escribe un programa que concatene dos strings ingresados por el usuario.

**Entrada:**

Hola  
Mundo

**Salida:**

HolaMundo

2. Escribe un programa que imprima la longitud de un string dado por el usuario.

**Entrada:**

Python

**Salida:**

6

3. Escribe un programa que imprima el primer y el último carácter de un string dado por el usuario.

**Entrada:**

Python

**Salida:**

Primer carácter: P  
Último carácter: n

4. Escribe un programa que reciba un string y dos números enteros. El programa debe imprimir el substring correspondiente a esos índices.

**Entrada:**

Python  
1  
4

**Salida:**

yth

5. Escribe un programa que convierta un string a mayúsculas y otro a minúsculas.

**Entrada:**

Hola Mundo

**Salida:**

En mayúsculas: HOLA MUNDO

En minúsculas: hola mundo

6. Escribe un programa que cuente cuántas veces aparece un carácter específico en un string.

**Entrada:**

banana

a

**Salida:**

El carácter 'a' aparece 3 veces

7. Escribe un programa que verifique si una subcadena está contenida en un string.

**Entrada:**

abracadabra

cad

**Salida:**

cad está en abracadabra

8. Escribe un programa que invierta un string sin usar métodos especiales de Python.

**Entrada:**

Python

**Salida:**

nohtyP

9. Escribe un programa que reciba dos strings y determine cuál es mayor lexicográficamente.

**Entrada:**

manzana

naranja

**Salida:**

naranja es mayor que manzana

10. Escribe un programa que extraiga todas las vocales de un string dado por el usuario.

**Entrada:**

programación

**Salida:**

Las vocales son: o, a, a, i, ó

11. Escribe un programa que reciba un string y reemplace manualmente todos los espacios por guiones (-).

**Entrada:**

Hola Mundo

**Salida:**

Hola-Mundo

12. Escribe un programa que cuente cuántas palabras hay en un string (considera que las palabras están separadas por espacios).

**Entrada:**

Hola mundo desde Python

**Salida:**

Número de palabras: 4

Ejercicio 13: Palíndromo

13. Escribe un programa que determine si un string es un palíndromo (se lee igual de izquierda a derecha que de derecha a izquierda).

**Entrada:**

radar

**Salida:**

Es un palíndromo

14. Escribe un programa que reciba un string y convierta la primera letra de cada palabra a mayúscula (como un título).

**Entrada:**

python es divertido

**Salida:**

Python Es Divertido

15. Escribe un programa que reciba un string y cuente cuántos caracteres son letras, cuántos son números y cuántos son otros tipos de caracteres.

**Entrada:**

Hola123!

**Salida:**

Letras: 4

Números: 3

Otros: 1

16. Implementar un juego interactivo de "Ahorcado" en Python para practicar el manejo de strings, listas y bucles. El objetivo es adivinar una palabra secreta antes de quedarse sin intentos.

**Instrucciones:**

1. Definir una palabra secreta en el código
2. Mostrar la palabra secreta: Muestra al usuario la palabra en forma de guiones bajos (\_) representando las letras no adivinadas. La longitud de la palabra debe ser igual a la cantidad de guiones.
3. Solicitar letras: El programa debe permitir que el usuario introduzca una letra por turno. Verifica que la letra no haya sido adivinada previamente.
4. Verificar si la letra está en la palabra:
  - Si la letra está en la palabra, debes mostrarla en todas las posiciones correspondientes donde aparezca esa letra.
  - Si la letra no está en la palabra, reduce el número de intentos restantes y muestra un mensaje al jugador.
  - Límite de intentos: El jugador tiene un número limitado de intentos (por ejemplo, 6). Cada vez que falla, pierde un intento.
5. Fin del juego:
  - El juego termina si el jugador adivina todas las letras de la palabra o si se le acaban los intentos.
  - Si el jugador gana, muestra un mensaje de victoria.
  - Si pierde, muestra la palabra secreta.

**Detalles a tener en cuenta:**

**Entrada del usuario:** Asegúrate de que el programa acepte solo una letra por turno. Puedes manejar letras mayúsculas y minúsculas usando el método `.lower()` para estandarizar las entradas.

**Validación de entrada:** Evita que el usuario adivine la misma letra dos veces.

**Mostrar el progreso:** Muestra el progreso del jugador en cada turno, actualizando las letras adivinadas y los intentos restantes.

**Mensaje final:** Al final del juego, proporciona retroalimentación sobre si el jugador ganó o perdió, y muestra la palabra secreta.