

## Lección 2

1. Recorre cadenas por medio de índices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> &
Esta es una cadena: Hola Mundo!
Primer caracter de la cadena: H
Ultimo caracter de la cadena: !
```

2. Imprime las posiciones de la cadena por índices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/
Esta es una cadena: Hola Mundo!
Posicion 0: H
Posicion 1: o
Posicion 2: l
Posicion 3: a
Posicion -1: !
Posicion -2: o
```

3. Hacemos slicing de la cadena

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/
Esta es una cadena: Hola Mundo!
Silice: Hola
Silice: Mundo
Silice: Mundo!
Silice: Hola
Silice: !
Silice: a Mund
```

4. Invierte la cadena

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> &
Esta es una cadena: Hola Mundo!
Invertir: !odnuM aloH
```

5. Muestra solo las posiciones pares

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1>
Esta es una cadena: Hola Mundo!
Posicion pares: Hl ud!
```

6. Imprime la cadena al revés, saltando de tres en tres hacia atrás

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes08.py"
Esta es una cadena: Hola Mundo!
Puedes adivinar que esta imprimiendo?: !n o
```

7. Invierte la cadena usando .join y reversed

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes09.py"
Esta es una cadena: Hola Mundo!
Invertir 2da opcion: !odnuM aloH
```

8. No se pueden agregar elementos a las cadenas por medio de índices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes08.py"
Esta es una cadena: Hola Mundo!
Traceback (most recent call last):
  File "e:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1\leccion02\datatypes08.py", line 5, in <module>
    cadena[4] = '*'
    ~~~~~^~~~
TypeError: 'str' object does not support item assignment
```

9. Cambia directamente la cadena reescribiéndola en la variable de forma manual

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes10.py"
Esta es una cadena: Hola Mundo!
Se modifiko la cadena?: Hola*Mundo!
```

10. Tampoco se puede borrar una cadena por índices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes10.py"
Esta es una cadena: Hola Mundo!
Traceback (most recent call last):
  File "e:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1\leccion02\datatypes10.py", line 5, in <module>
    del cadena[4]
    ~~~~~^~~~
TypeError: 'str' object doesn't support item deletion
```

11. Modifica la cadena por slicing

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes11.py"
Esta es una cadena: Hola Mundo!
Se eliminaron varios caracteres?: Hola!
```

12. Se borró la variable

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes12.py"
Esta es una cadena: Hola Mundo!
Se eliminaron varios caracteres?: Hola!
```

13. {} y .format para introducir cadenas en cadenas de un print

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escom/3 Semestre/PCD/pcd3am1_2024/leccion02/datatypes13.py"
Esta es una cadena con formato: Hola Mundo !
Esta es una cadena con formato desordenado: Mundo ! Hola
Esta es una cadena con formato: Hola Mundo !
```

14. Cambia los formatos a binario, exponencial y flotante truncado a 4 dígitos

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Py
Binario del 256: 1101
Formato exponencial: 1.234345e+03
Flotante truncado a 4 dígitos: 3.1416
```

15. Una notación distinta para ajustar la alineación del texto

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C
Alineacion de cadenas:
|Izq      |   Cen   |         Der|
>> Mario  << soy yo, y tengo >>43 << años!
```

16. Cambia en automático lo más conveniente al hacer operaciones entre distintos tipos de datos numéricos

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C
<class 'int'>
<class 'float'>
5
0.6666666666666666
5.666666666666667
7
-1
12
0.75
0
3
```

17. Operaciones con complejos

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C
Suma: (3+8j)
Resta: (-1+8j)
Division: (1.307692307692308+0.5384615384615384j)
Multiplicacion: (-13+13j)
```

18. Experimentos con booleanos

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C
<class 'bool'>
a=True
b=False
34 == 34 : True
23 == 24: False
34 != 34 : False
23 != 24: True
x < y < z
x > y o y < z
Negacion
```

19. Imprime listas y su tamaño

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Lista Vacía: []
Tamaño Lista Vacía: 0
Lista Numeros: [1, 2, 3, 4, 5, 6]
Tamaño Lista Numeros: 6
Lista Letras: ['a', 'b', 'c']
Tamaño Lista Letras: 3
Lista Objetos: [1, '*', 0, True, 3.141592, False, 'Hola Mundo!', ['a', 'b', 'c']]
Tamaño Lista Objetos: 8
```

20. Matrices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Matriz completa: [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]]
Segundo Renglon: [5, 6, 7, 8]
Tercer Renglon, segunda Columna: 10
```

21. Imprime los elementos de una lista por índices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Ultima posicion 1: e
Ultima posicion 2: e
Ultima posicion 3: e
Penultima posicion d
```

22. Guarda los elementos de una cadena en una lista al tomar un separador

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Separada por espacios en una lista: ['Hola', 'Mundo!', 'Aqui', 'estoy', '...']
Otro separador: ['1', '2', '3', '4']
```

23. Append agrega elementos al final de una lista

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Separada por espacios en una lista: ['Hola', 'Mundo!', 'Aqui', 'estoy', '...']
Otro separador: ['1', '2', '3', '4']
```

24. Imprime la lista, la revierte y la imprime al revés. No es lo mismo revertirla que imprimirla al revés

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Lista original: ['IPN', 'UNAM', 'ANAHUAC', 'TEC', 'IBERO', 2, 3, 4]
Lista invertida: [4, 3, 2, 'IBERO', 'TEC', 'ANAHUAC', 'UNAM', 'IPN']
Lista original nuevamente: ['IPN', 'UNAM', 'ANAHUAC', 'TEC', 'IBERO', 2, 3, 4]
Volvemos a invertirla: ['IPN', 'UNAM', 'ANAHUAC', 'TEC', 'IBERO', 2, 3, 4]
```

25. Se eliminan elementos de la lista por índices

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Lista original: [1, 2, 3, 4, 2, 2, 5, 6, 2, 7]
Se elimino un elemento: [1, 3, 4, 2, 2, 5, 2, 7]
Se extrajo el ultimo elemento: 7 de la lista: [1, 3, 4, 2, 2, 5, 2]
```

26. Slicing

```
PS E:\Escom\3 Semestre\PCD\pcd3am1_2024\parcial1> & C:/Python312/python.exe "e:/Escritorio/Programas/Python312/Python/Python312/python.exe" e:/Escritorio/Programas/Python312/Python/Python312/python.exe
Lista original: [100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
Slice de 0 a 9: [100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
Slice de 0 a 2: [100, 200, 300]
Slice de Inicio a 2: [100, 200, 300]
Slice de 3 a 9: [400, 500, 600, 700, 800, 900, 1000]
Slice de 3 al final: [400, 500, 600, 700, 800, 900, 1000]
Slice de 4 a 6: [500, 600, 700]
Slice de Inicio a Fin: [100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
Slice posiciones pares: [100, 300, 500, 700, 900]
Slice posiciones impares: [200, 400, 600, 800, 1000]
Slice invertido impares: [1000, 800, 600, 400, 200]
Slice invertido pares: [900, 700, 500, 300, 100]
```