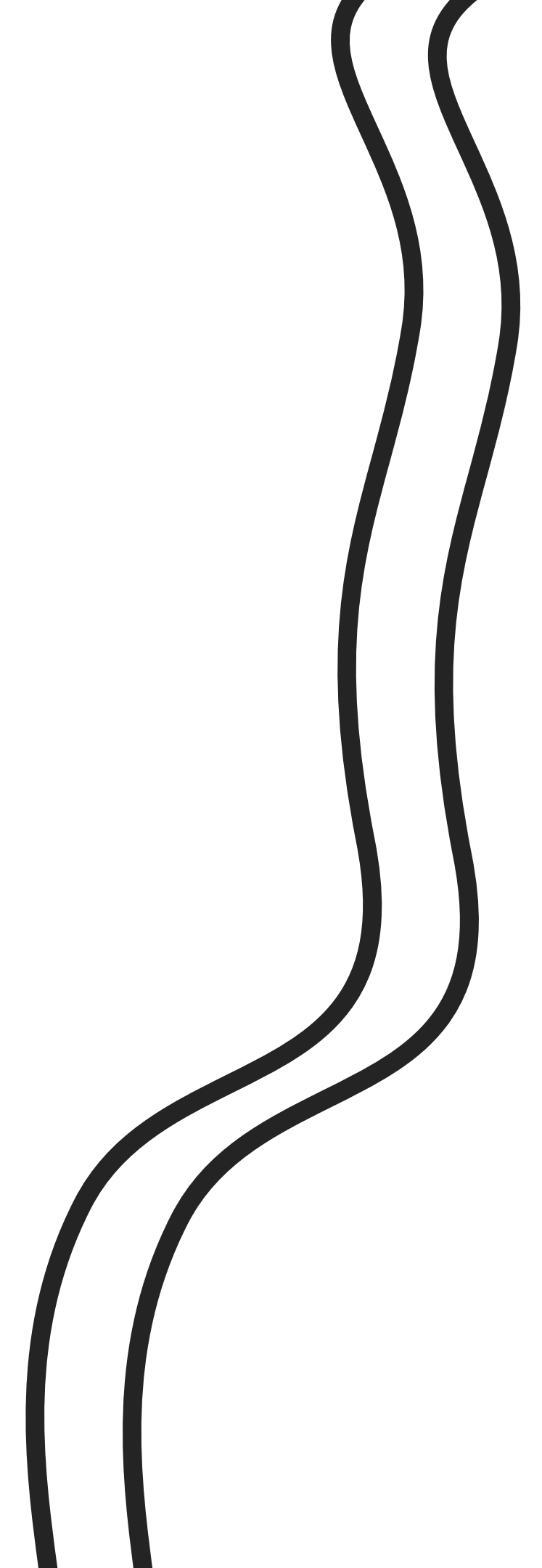


# Módulo 7 STRINGS



# EN ESTE MÓDULO



1

TIPO DE DATO

---

2

MANPULACIÓN DE STRINGS

---

3

FUNCIONES Y STRINGS

---

4

CUESTIONARIO

---

5

ASCII

---

# TIPO DE DATO

≧ String - str ≦

Recordemos

Información que almacena

Concatenación de str (+)

Repetición de str (\*)

# MANIPULACIÓN

*de str*

# SELECCIONAR CARACTER DE UN STR

① Es posible seleccionar ciertos elementos de un string según su posición en él.



Para esto usamos [ ]

Los strings son inmutables!  
`s[1] = "r"` arrojará error



② Los índices siempre comienzan desde 0.



El primer carácter o primera letra de un string tiene asociado el índice o posición 0.

EJ.

# SELECCIONAR CARACTER DE UN STR



\*Los espacios en blanco también son un caracter

ⒺJ. `a = "La casa es verde"`  
`print(a[5])`

\*Podemos seleccionar símbolos de derecha a izquierda, es decir partiendo desde el último, si usamos números negativos

ⒺJ. `print("Martes"[5])`  
`print("Martes"[-1])`

# RECORRIENDO UN STR

- ①. Recorramos un str de izquierda a derecha.  Intentemos con while
- ①. Recorramos un str de derecha a izquierda.  Intentemos con for

	H	o	l	a	,		¿	q	u	é		t	a	l	?
Índice positivo	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Índice negativo	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1



**SLICING** = **SELECCIONAR**  
**SUB-STRING** DE UN STR

① Es posible seleccionar varios caracteres a la vez a partir de un string.  Para esto usamos **string[inicio:final+1]**

Ej. `print("Martes"[1:4])`

Ej. Para el string x='abracadabra', qué caracteres se extraen al usar x[3:6]?

# OPERADOR IN

Es un operador booleano que toma dos string y devuelve True si el primero aparece como substring en el segundo.

EJ. `s = "hola, ¿Qué tal?"`

```
print("Qué" in s)
```

```
print("qué" in s)
```

# RECORRIENDO UN STR FOR

ej. 

```
prefijos = "CLMNPT"  
sufijo = "apa"  
  
for c in prefijos:  
    print(c + sufijo)
```

# COMPARACIÓN DE STR

Ej.

```
s = "hola"  
a = "Hola"  
  
print(s == a)  
print(s < a)  
print(s > a)
```

Las comparaciones se hacen por orden lexicográfico.



```
s = "S"  
a = "árbol"  
  
print(s < a)
```

# SÍMBOLOS ESPECIALES

① Algunos símbolos no pueden escribirse usando el teclado.



Usamos "\" y algún otro caracter para indicarlos

② Investigue:  
Cadenas de escape

③ Ej. `print("Un párrafo.\nOtro párrafo.")`

\n : indica una nueva linea



"new line"

Y si quiero escribir un "\" en mi string ?

# FUNCIONES *y strings*

# MÉTODOS PARA STR

## ① len(str):

retorna el largo del string que se le entregue como parámetro



①.

```
print(len("Martes"))  
print(len("Yo soy."))
```

se usan de modos  
distintos!



## ② upper(): Pasa todo un string a mayúscula



①.

```
print("Martes".upper())
```

## ③ lower(): Pasa todo un string a minúscula



①.

```
print("MArtes".lower())
```

# MÉTODOS PARA STR

## ④ strip(a\_eliminar):

retorna un string donde se han removido los caracteres entregados como parámetros si estos se encuentran en los extremos del string.



ej.

```
a = "Bien. Martes a las 5."  
print(a.strip("."))
```



# MÉTODOS PARA STR

## ④ replace(viejo, nuevo):

retorna un string donde se han reemplazado los caracteres entregados como parámetros por otros que también se han entregado como parámetros.



ej.

```
a = "Hola!!1!"  
print(a.replace("1", "!"))
```

# CUESTIONARIO

*hasta 5 pp*

\*Dos preguntas buenas otorgan 1 pp. No hay intermedios.

# ASCII