

Procesamiento de datos

Actividad 2

Consigna 1

- 1 Generar un array de 3 dimensiones y mostrar el output en una captura de pantalla.

```
import numpy as np
three_dimensions_list = [[1, 2, 3], [4, 5, 6]], [[7, 8, 9], [10, 11, 12]]
array = np.array(three_dimensions_list)
print(array)
```

[32] ✓ 0.0s Python

```
... [[ 1  2  3]
      [ 4  5  6]]

      [[ 7  8  9]
       [10 11 12]]]
```

Consigna 2

- 2 - Ubicar el valor de un elemento en el array de 3 dimensiones con la función correspondiente y compartir el output en una captura de pantalla.

```
element = array[0,0,0]
print(element)
```

[5] ✓ 0.0s Python

```
... 1
```

Consigna 3

- 3 - Utilizando append, agregamos 3 valores nuevos al array y compartamos output en una captura de pantalla.

```
np.append(array, (101, 102, 103))
```

[13] ✓ 0.0s Python

```
... array([[ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 101,
           102, 103]])
```

Consigna 4

- 4 - Generar un nuevo array de 3 dimensiones, pero con el formato requerido para que el output en una captura de pantalla pueda mostrarnos un resultado como (3,3) usando shape.

```
new_two_dimensions = np.array([[1, 1, 1], [2, 2, 2], [3, 3, 3]])
new_three_dimensions = np.array([[[1, 1, 1], [2, 2, 2], [3, 3, 3]], [[4, 4, 4], [5, 5, 5], [6, 6, 6]], [[7, 7, 7], [8, 8, 8], [9, 9, 9]]])
print(new_two_dimensions.shape)
print(new_three_dimensions.shape)
```

[26] ✓ 0.0s Python

```
... (3, 3)
      (3, 3, 3)
```

Consigna 5

- 5 - Ahora, con la librería de pandas, crear una serie llamada (marcas_productos) y compartir el output en una captura de pantalla donde pueda mostrarnos la lista.

```
import pandas as pd

marcas_productos = pd.Series()

print(marcas_productos)
```

[29] ✓ 0.0s Python

... Series([], dtype: object)

Consigna 6

- 6 - Generar una nueva serie llamada (color_cantidad) utilizando index para generar o modificar la columna inicial. Luego, compartir el output en una captura de pantalla donde pueda mostrarnos la lista con los nombres de la

```
color_cantidad = pd.Series([7,5,3],index=['Amarillo','Azul','Rojo'])
print(color_cantidad)
```

[30] ✓ 0.0s Python

... Amarillo 7
Azul 5
Rojo 3
dtype: int64