

Creado por:

Isabel Maniega

## Estructura de datos de almacenamiento

En el presente documento se han incluido varios de los archivos de clase.

Incluye:

- Tuplas,
- Listas,
- Diccionarios,
- etc

## Tuplas

```
In [1]: # 0, 1, 2, 3 ... Izquierda a derecha
# -1, -2, -3, ... Derecha a Izquierda
A = (10, 20, 30, 40)
A
```

Out[1]: (10, 20, 30, 40)

```
In [2]: A[2]
```

Out[2]: 30

```
In [3]: A[0], A[1], A[2], A[3]
```

Out[3]: (10, 20, 30, 40)

```
In [4]: # No se puede ni modificar
# No se puede apendizar
```

## Arrays

```
In [6]: import numpy as np
```

```
In [7]: B = np.array([1, 2, 3, 4])
B
```

Out[7]: array([1, 2, 3, 4])

```
In [8]: B[0]
```

Out[8]: 1

```
In [9]: B[0] = 20
B
```

Out[9]: array([20, 2, 3, 4])

```
In [10]: B1 = np.append(B, 50)
B1
```

Out[10]: array([20, 2, 3, 4, 50])

```
In [11]: # se puede modificar
# se puede añadir/ apendizar
```

**o concatenar...**

```
In [12]: B2 = np.array([60])
B2
```

Out[12]: array([60])

```
In [14]: # unir dos arrays
B3 = np.concatenate((B1, B2))
B3
```

Out[14]: array([20, 2, 3, 4, 50, 60])

## Listas

```
In [15]: C = list((1, 2, 3, 4))
C
```

Out[15]: [1, 2, 3, 4]

```
In [16]: D = [0, 2, 3, 5, 7, 8]
D
```

Out[16]: [0, 2, 3, 5, 7, 8]

```
In [17]: D[2]
```

Out[17]: 3

```
In [18]: D[2] = 100
D
```

Out[18]: [0, 2, 100, 5, 7, 8]

```
In [19]: D.append(200)
D
```

Out[19]: [0, 2, 100, 5, 7, 8, 200]

```
In [20]: # Eliminar por valor concreta
D.remove(200)
D
```

Out[20]: [0, 2, 100, 5, 7, 8]

```
In [21]: # Eliminar por posición concreta
D.remove(D[2])
D
```

Out[21]: [0, 2, 5, 7, 8]

```
In [23]: L = [1, 1, 2, 3, 6, 3, 3, 8, 10, 10, 5, 1]
L
```

Out[23]: [1, 1, 2, 3, 6, 3, 3, 8, 10, 10, 5, 1]

```
In [24]: # ordenar los valores
L.sort()
L
```

Out[24]: [1, 1, 1, 2, 3, 3, 3, 5, 6, 8, 10, 10]

```
In [25]: # Eliminar valores duplicados y colocarlos
L = set(L)
L
```

Out[25]: {1, 2, 3, 5, 6, 8, 10}

```
In [26]: type(L)
```

Out[26]: set

```
In [28]: L[0] = 100
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-28-9772e816e59b> in <module>
----> 1 L[0] = 100

TypeError: 'set' object does not support item assignment
```

```
In [29]: L.append(50)
```

```
-----
AttributeError                            Traceback (most recent call last)
<ipython-input-29-3dc50f5ae633> in <module>
----> 1 L.append(50)

AttributeError: 'set' object has no attribute 'append'
```

```
In [30]: L1 = list(L)
L1
```

Out[30]: [1, 2, 3, 5, 6, 8, 10]

```
In [31]: L1[0] = 100
L1
```

Out[31]: [100, 2, 3, 5, 6, 8, 10]

```
In [32]: L1.append(500)
L1
```

Out[32]: [100, 2, 3, 5, 6, 8, 10, 500]

## Otras cosas importantes

```
In [33]: listado = [3, 7, 5, 1]
listado
```

Out[33]: [3, 7, 5, 1]

```
In [34]: min(listado)
```

Out[34]: 1

```
In [35]: max(listado)
```

Out[35]: 7

```
In [36]: # Recomendación listas:
L = [1, 23, 26, 45, 48, 1, 3, 6, 8]
L
```

Out[36]: [1, 23, 26, 45, 48, 1, 3, 6, 8]

```
In [37]: # separar en grupos de 3 en 3:
L = [
    1, 23, 26,
    45, 48, 1,
    3, 6, 8
]
L
```

Out[37]: [1, 23, 26, 45, 48, 1, 3, 6, 8]

```
In [8]: test = [1, 2, 3, 4, 5, 6]
test
```

Out[8]: [1, 2, 3, 4, 5, 6]

```
In [9]: test.reverse()
test
```

Out[9]: [6, 5, 4, 3, 2, 1]

```
In [10]: test1 = enumerate(test)
test1
```

Out[10]: <enumerate at 0x7fe9f451cc00>

```
In [11]: list(test1)
```

Out[11]: [(0, 6), (1, 5), (2, 4), (3, 3), (4, 2), (5, 1)]

## Transformación de entre ellas (Array/Lista)

**Numpy array a lista**

```
In [38]: import numpy as np
```

```
In [39]: E = np.array([1, 3, 4, 25, 1, 6, 7])
E
```

Out[39]: array([ 1, 3, 4, 25, 1, 6, 7])

```
In [40]: E1 = E.tolist()
E1
```

Out[40]: [1, 3, 4, 25, 1, 6, 7]

**Lista a numpy Array**

```
In [42]: F = [2, 3, 6, 8, 4, 12, 25]
F
```

Out[42]: [2, 3, 6, 8, 4, 12, 25]

```
In [43]: F1 = np.array(F)
F1
```

Out[43]: array([ 2, 3, 6, 8, 4, 12, 25])

```
In [45]: type(E1), type(F1)
```

Out[45]: (list, numpy.ndarray)

## Matrices

### Ejemplo 1: Teoría

```
In [46]: import numpy as np
```

```
In [47]: a = np.array([
    [10, 20, 30, 40],
    [50, 60, 70, 80],
    [90, 100, 110, 120]
])
a
```

Out[47]: array([[ 10, 20, 30, 40],
 [ 50, 60, 70, 80],
 [ 90, 100, 110, 120]])

### Ejemplo 2: Suma de matrices

```
In [49]: # np.array([ [fila1], [fila2]])
matriz1 = np.array([[1, 2], [3, 4]])
matriz1
```

Out[49]: array([[1, 2],
 [3, 4]])

```
In [51]: matriz2 = np.array([[5, 6], [7, 8]])
matriz2
```

Out[51]: array([[5, 6],
 [7, 8]])

```
In [52]: matriz_suma = matriz1 + matriz2
matriz_suma
```

Out[52]: array([[ 6, 8],
 [10, 12]])

### Ejemplo 3: Multiplicación y división de matrices

```
In [53]: # np.array([ [fila1], [fila2]])
matriz1 = np.array([[1, 2], [3, 4]])
matriz1
```

Out[53]: array([[1, 2],
 [3, 4]])

```
In [54]: matriz2 = np.array([[5, 6], [7, 8]])
matriz2
```

Out[54]: array([[5, 6],
 [7, 8]])

```
In [56]: # Multiplicación
matriz_multiplicacion = 2*matriz1
matriz_multiplicacion
```

Out[56]: array([[2, 4],
 [6, 8]])

```
In [57]: # División
matriz_division = matriz_multiplicacion / 2
matriz_division
```

Out[57]: array([[1., 2.],
 [3., 4.]])

```
In [58]: matriz_division = matriz2 / 2
matriz_division
```

Out[58]: array([[2.5, 3. ],
 [3.5, 4. ]])

Creado por:

Isabel Maniega

In [ ]: