10 DE NOVIEMBRE DE 2025

EJERICICOS PYTHON 9: ARRAYS

ENRIQUE GONZÁLEZ

CAMPUSFP
1º DAM GETAFE

ENRIQUE GONZÁLEZ
1º DAM GETAFE

Contenido

ENRIQUE GONZÁLEZ
1º DAM GETAFE

Ejercicio 1: Sumar dos arrays elemento por elemento

Enunciado: Crea dos arrays de Numpy de tamaño 5, llenos de números enteros. Luego, calcula la suma de ambos arrays elemento por elemento y muestra el resultado.

Motivo para usar arrays: Con Numpy, podemos sumar directamente dos arrays usando la operación +, sin necesidad de recorrer cada elemento manualmente como tendríamos que hacer con listas.

CÓDIGO:

```
import numpy as np

#EJERCICIO 1: SUMA 2 ARRAYS

array1=np.array([1, 2, 3, 4, 5])

array2=np.array([10, 20, 30, 40, 50])


suma=array1 + array2

print(suma)
```

COMPROBACIÓN:

```
PS C:\Users\CampusFP\AppData\Local\Programs\Microsoft VS Code> &
[11 22 33 44 55]
PS C:\Users\CampusFP\AppData\Local\Programs\Microsoft VS Code>
```

Ejercicio 2: Multiplicar cada elemento de un array por un número

Enunciado: Crea un array de Numpy de tamaño 6 con valores enteros de tu elección. Luego, multiplica cada elemento del array por 3 y muestra el resultado.

Motivo para usar arrays: La multiplicación escalar con arrays es mucho más eficiente en Numpy, ya que la operación se aplica directamente a cada elemento sin necesidad de un bucle.

CÓDIGO:

```
#EJERCICIO 2:MULTIPLICAR ARRAY POR 3

arrayx3=np.array([3, 4, 66, 12, 9, 22])

por3=arrayx3 * 3

print(por3)
```

COMPROBACIÓN:

```
[  9  12 198  36  27  66]
PS C:\Users\CampusFP\AppData\Local\Programs\Microsoft VS Code>
```

Ejercicio 3: Calcular el promedio de un array

Enunciado: Crea un array de Numpy con 10 números enteros de tu elección. Calcula el promedio de los elementos y muestra el resultado.

Motivo para usar arrays: Numpy permite calcular el promedio de los elementos de un array con la función np.mean() de forma rápida y eficiente, algo que con listas requeriría escribir más código.

CÓDIGO:

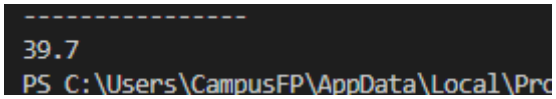
```
#EJERCICIO 3:CALCULAR MEDIA

arraymedia=np.array([1, 22, 34, 56, 77, 99, 2, 3, 98, 5])

media=np.mean(arraymedia)

print(media)
```

COMPROBACIÓN:



Ejercicio 4: Filtrar elementos mayores a un valor dado

Enunciado: Crea un array de Numpy con 8 números enteros. Luego, crea un nuevo array que solo contenga los elementos mayores a 5 y muéstralo.

Motivo para usar arrays: Con Numpy, puedes aplicar filtros de manera muy rápida usando operaciones de comparación en una sola línea. Filtrar listas de esta forma requeriría escribir bucles y condicionales adicionales.

CÓDIGO:

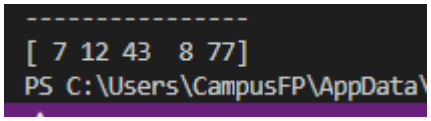
```
#EJERCICIO 4: MAYORES DE 5

array5=np.array([4, 1, 7, 12, 43, 2, 8, 77])

mayor5=array5[array5>5]

print(mayor5)
```

COMPROBACIÓN:



Ejercicio 5: Elevar al cuadrado cada elemento de un array

Enunciado: Crea un array de Numpy con 5 elementos enteros. Calcula el cuadrado de cada elemento y muestra el resultado.

Motivo para usar arrays: Numpy permite aplicar operaciones matemáticas, como la potenciación, a cada elemento de un array en una sola operación. Esto es mucho más eficiente que hacer un bucle para elevar cada elemento al cuadrado en una lista.

CÓDIGO:

```
#EJERCICIO 5: ELEVAR AL CUADRADO

cuadrado=np.square([4, 22, 12, 2, 66])

print(cuadrado)
```

COMPROBACIÓN: