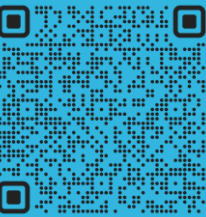




MATRICES

Algoritmos y estructuras de datos I

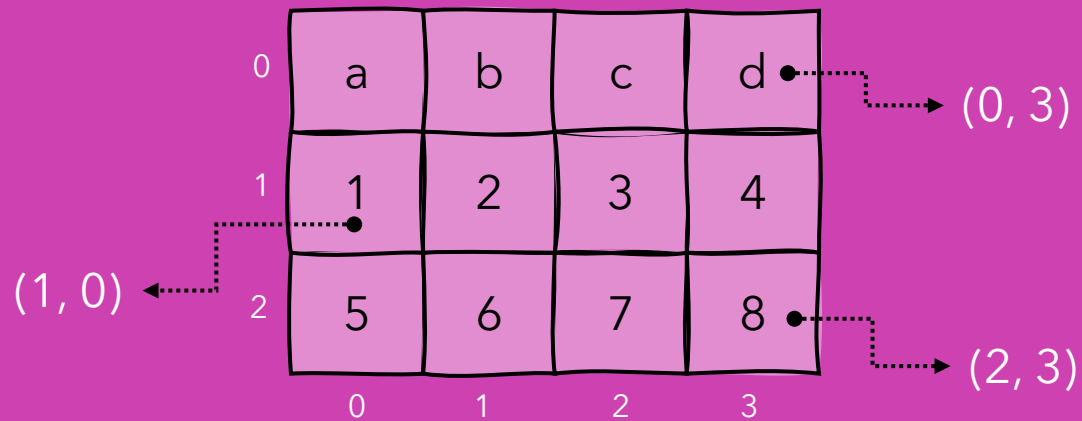


¿QUÉ ES UNA MATRIZ?

Son una estructura de datos de dos dimensiones de elementos contiguos, relacionados por un mismo nombre, donde los elementos son distinguidos por dos índices (normalmente denominado como fila y columna).

En la representación, se muestra la representación de una matriz de tres filas y cuatro columnas.

Tanto las filas como las columnas, siempre empieza en el índice 0, y terminan en el índice (tamaño - 1); dónde tamaño se refiere a la longitud de la fila o columna de la matriz.

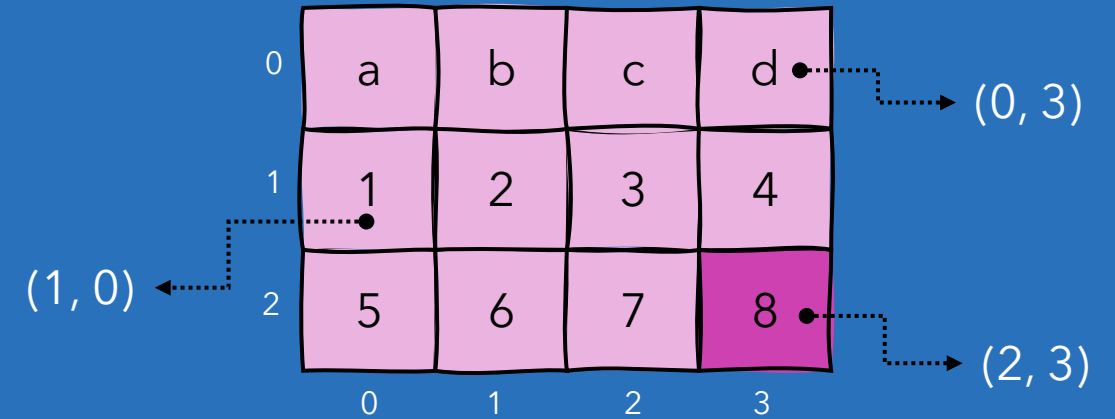


¿CÓMO CREAR UNA MATRIZ?

Su implementación la realizaremos mediante la composición de una lista de listas, teniendo en cuenta que contendrá listas en su interior.

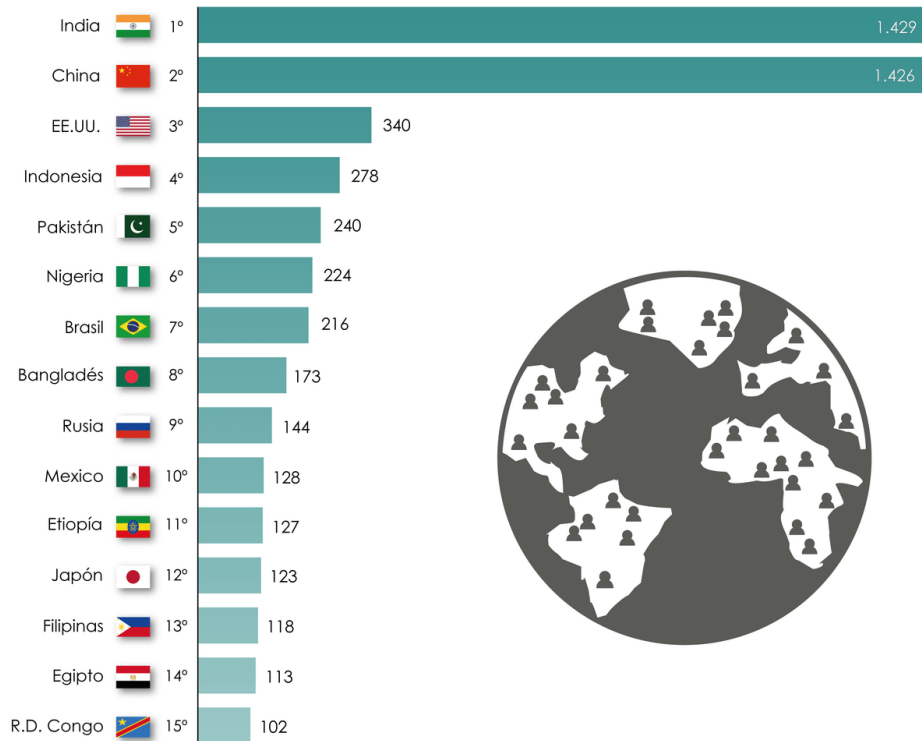
Es posible acceder a una posición (fila y columna) de la matriz usando el nombre de la variable y corchetes "[" indicando en su interior el índice que quieres obtener, utilizando corchetes para cada lista.

El primer conjunto de corchetes accede a fila y el segundo a la columna.



```
matriz = [['a', 'b', 'c', 'd'],  
          [1, 2, 3, 4],  
          [5, 6, 7, 8]]  
  
print(matriz[0][3]) #Imprime d  
print(matriz[1][0]) #Imprime 1  
print(matriz[2][3]) #Imprime 8
```

Países más poblados (millones de habitante)



	País	Millones hab.
0	India	1429
1	China	1426
2	EE.UU.	340
	0	1

ALMACENAMIENTO DE DATOS

Las matrices son una herramienta fundamental en la representación de datos para generar gráficos y visualizaciones.

Su estructura bidimensional y su capacidad para almacenar valores numéricos o información en forma de cuadrícula las convierten en una base sólida para la creación de representaciones visuales.

```
matriz = [  
    ['India', 1429],  
    ['China', 1426],  
    ['EE.UU.', 340]  
]  
print(matriz[0][0]) #Imprime India  
print(matriz[0][1]) #Imprime 1429
```

RECORRER UNA MATRIZ

```
matriz = [['a', 'b', 'c', 'd'],  
          [1, 2, 3, 4],  
          [5, 6, 7, 8]]
```

```
for fila in range(0, len(matriz)):  
    for columna in range(0, len(matriz[fila])):  
        print("[", fila, "][", columna, "]")  
        print("->", matriz[fila][columna])
```

Por lo general, el recorrido de una matriz se realiza mediante el uso de un ciclo *for* anidado, que vaya accediendo uno por uno a los elementos que queremos acceder de la misma, tanto para la fila como para la columna.

Al ser listas con listas en su interior, seguiremos teniendo acceso a las mismas funciones para agregar, modificar y eliminar elementos como hemos visto anteriormente cuando trabajamos con listas de forma individual.

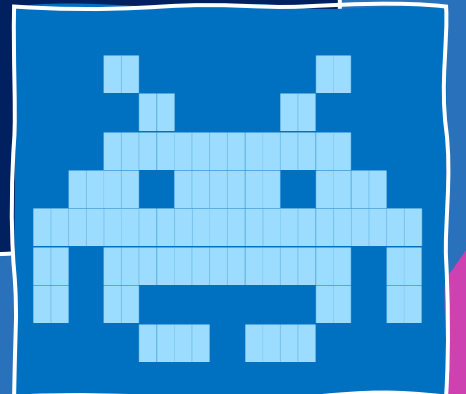
REPRESENTACIONES GRÁFICAS

Las matrices también pueden ser utilizadas para crear representaciones visuales. Un ejemplo claro es la generación de imágenes mediante matrices binarias, donde cada elemento indica la presencia o ausencia de un pixel.

En el siguiente código, se observa como una matriz define la forma de un cangrejo mediante una representación gráfica ASCII, donde cada 1 indica la presencia de un pixel y cada 0 indica su ausencia. Al imprimir la matriz en la consola, obtenemos una figura que muestra la forma del cangrejo del juego *Space Invaders*.

```
cangrejo = [
    [0,0,1,0,0,0,0,0,1,0,0],
    [0,0,0,1,0,0,0,1,0,0,0],
    [0,0,1,1,1,1,1,1,0,0],
    [0,1,1,0,1,1,1,0,1,1,0],
    [1,1,1,1,1,1,1,1,1,1],
    [1,0,1,1,1,1,1,1,0,1],
    [1,0,1,0,0,0,0,0,1,0,1],
    [0,0,0,1,1,0,1,1,0,0,0],
]

for fila in range(len(cangrejo)):
    for columna in
range(len(cangrejo[fila])):
        if cangrejo[fila][columna] == 1:
            print('█', end='')
        else:
            print(' ', end='')
    print()
```



CARGA Y VISUALIZACIÓN DE DATOS

El código permite al usuario ingresar información sobre varios alumnos.

Primero, solicita al usuario que indique la cantidad de alumnos que desea ingresar. Luego, en un bucle, solicita el nombre y el legajo de cada alumno y los agrega a una matriz.

Una vez que se han ingresado todos los datos, el código muestra la información de cada alumno, incluyendo su nombre y legajo.

```
matrizAlumnos = []

cantidadAlumnos = int(input("¿Qué cantidad de alumnos desea cargar? "))

for fila in range(0, cantidadAlumnos):
    alumno = []

    print("Ingrese los datos del alumno ", fila + 1)
    nombre = input("Nombre: ")
    legajo = input("Legajo: ")

    alumno.append(nombre)
    alumno.append(legajo)

    matrizAlumnos.append(alumno)

for fila in range(0, cantidadAlumnos):
    print("Datos del alumno ", fila + 1)
    print("Nombre:", matrizAlumnos[fila][0], "Legajo:", matrizAlumnos[fila][1])
```


AUTOEVALUACIÓN

La autoevaluación es crucial para identificar mejoras y fortalezas de forma objetiva, facilitando la fijación de metas y la toma de acciones para alcanzarlas.

