

---

# MATRICES

*Una aplicación de  
las listas en Python*

# MATRICES

- Una matriz es un conjunto de valores organizados en filas y columnas
  - Fila: cada una de las líneas horizontales de valores
  - Columna: cada una de las líneas verticales de valores
  - Dimensión de una matriz (tamaño u orden): cantidad de filas por la cantidad de columnas (  $m \times n$  )

Valor: elemento o entrada de la matriz.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 7 \\ 4 & 9 & 2 \\ 6 & 0 & 5 \end{bmatrix}$$

Dimensión de la matriz A:  $4 \times 3$ .

Referencia a un elemento: por medio de la fila y la columna donde este

. Elemento en  $A_{4,2}$ : 0

. Elemento en  $A_{2,3}$ : 7

# SUBLISTAS Y MATRICES

---

- En programación las matrices se pueden implementar por filas o por columnas usando el concepto de sublistas
- Vamos a estudiar la implementación por filas usando el concepto de listas anidadas de Python

- Cada elemento de la lista es una sublista que representa una fila completa de la matriz

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

```
>>> matriz = [[1,2,3], [4,5,6], [7,8,9]]
```

**¿ Representación por columnas ?**

- Se puede seleccionar una fila completa usando solo un índice
- Se puede seleccionar un elemento dentro de la fila mediante el uso de la notación de doble índice

```
>>> matriz= [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
>>> matriz[1] ← Fila 2
[4, 5, 6]
>>> matriz[1][0] ← Fila 2
4
>>>
```

- Dada la siguiente matriz:

`a = [[2, 4, 6], [1, 5, 8], [8, 0, 2], [-1, 9, 4]]`

- ¿ Cómo determinamos la cantidad de filas ?
  - ¿ Cómo determinamos la cantidad de columnas ?
  - ¿ Cuál elemento está en la fila 3 columna 1 ?
  - ¿ Referencia de ese elemento usando listas Python ?
- 
- ¿Y cuando hay sublistas dentro de sublistas?
  - Tuplas y strings dentro de listas: al igual que las sublistas se usan dos índices.

# PRÁCTICA

---

## ■ Ejercicios

leer\_matriz

imprimir\_matriz

programa principal

## ***Función leer\_matriz***

Lee una matriz de tamaño  $m \times n$ : cada elemento de la matriz se va a leer por separado.

Para cada fila se leen (input) los elementos de dicha fila. Con la función append se agregan los elementos a la fila.

Cuando una fila es completada esta se agrega (append) a la matriz.

- Entradas:  $m, n$  ( $m$  filas,  $n$  columnas)
- Salidas: retorna la matriz leída



## ***Función imprimir\_matriz***

Recibe una matriz y la imprime.

- Entradas: una matriz
- Salidas: impresión de la matriz, cada fila en una línea separada. Ejemplo: la matriz

[ [2, 4, 6], [4, 3, 2] ] se imprimiría así:

2   4   6

4   3   2

---

## ***Programa principal***

En el programa principal leer  $m$  y  $n$ ,  
luego llamar a la función de lectura  
y por último llamar a la función de  
impresión.

- Desarrolle la función **matriz\_nula** que reciba una matriz y retorne el valor booleano True si es nula (o cero), de lo contrario retorne False. Una matriz es nula si todos los elementos son ceros.

$$\begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

Ejemplos del funcionamiento:

```
>>> matriz_nula([[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]])  
True  
>>> matriz_nula([[0, 0, 0], [0, 0, 0], [0, 1, 0]])  
False
```

# FUNCIÓN *enumerate*

---

- Aplicada a secuencias, conjuntos y diccionarios da automáticamente el número del elemento que esta procesándose

```
>>> for i, j in enumerate([25,20,30,5]):  
    print(i, j)
```

```
0 25  
1 20  
2 30  
3 5
```

---

# Más prácticas

- Ejercicios del libro de texto
- Laboratorio

---

*"... el odio, la venganza y la violencia producen más odio, más venganza y más violencia, producen una sociedad cada vez peor.*

*Lo bueno es que tenemos la tolerancia, el amor y el respeto con los cuales podemos construir un mundo mejor ..."*