



## Programación Básica

### Clase 10

---

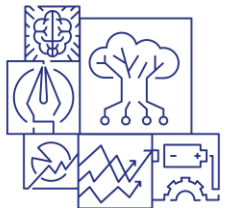
## Arreglos multidimensionales



# Agenda

## Arreglos multi dimensionales

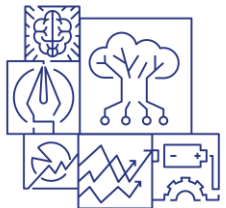
- Concepto y utilización
- Creación
- Escritura y lectura
- Ejemplos



# Arreglos multi dimensionales

Los arreglos multi dimensionales tienen por condición que almacenan un arreglo dentro de otros arreglos, lo que permite crear una estructura que si la visualizáramos tendría dos o más dimensiones.

	0	1	2
0	23	54	12
1	21	81	14
2	16	20	21



# Definición de un arreglo

Como se puede observar en el ejemplo, se crea un arreglo y cada uno de los ítems de este son otro arreglo, esto crea un arreglo de dos dimensiones de 3 x 3 elementos.

Arreglo más externo

```
edades = [ [23, 54, 12], [21, 81, 14], [16, 20, 21] ]
```

Arreglo      Arreglo      Arreglo



# Asignar un valor al arreglo

Se debe hacer referencia a una combinación (fila, columna) aunque no necesariamente un arreglo es gráfico, se presenta como una cuadrícula para facilitar su entendimiento.

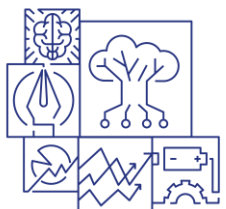
	0	1	2
0	23	54	12
1	21	81	14
2	16	20	21

`edades[0][1] = 45`

fila

columna

valor



# Asignar un valor al arreglo

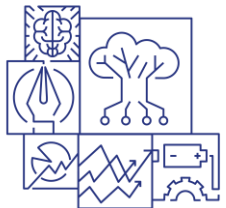
```
c = int(input("Columnas: "))  
f = int(input("Filas: "))  
valores = []  
for i in range(c):  
    valores.append([0] * f)
```

Si se deseara crear un arreglo de dos dimensiones (conocido como matrices) con valores por defecto en cero para luego ser cargados, se podría hacer de la siguiente manera.

Columnas: 4

Filas: 3

[[0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]]



# Un ejemplo completo

```
#Se definen los elementos para almacenar
mayor = 0
```

```
carros = []
for i in range(7):
    carros.append([0] * 3)
```

```
#Se llena la matriz
```

```
for p in range(7):
    print("Provincia ", (p + 1))
    for m in range(3):
        cantidad = int(input(" Cantidad para el cuatrimestre " + str(m+1) + ": "))
        carros[p][m] = cantidad
```

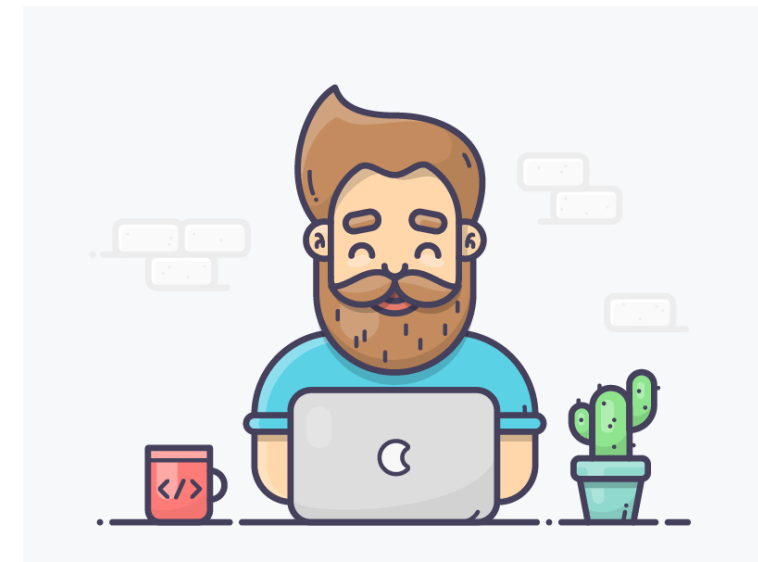
```
#Se busca la provincia mayor
```

```
for p in range(7):
    suma = 0
    for m in range(3):
        suma += carros[p][m]
    if (suma > mayor):
        mayor = suma
print("La mayor cantidad en una provincia es: ", mayor)
```

En este ejemplo se solicita la cantidad de vehículos vendidos por cuatrimestre por cada provincia, se almacenan en un arreglo de dos dimensiones, cada dimensión representa o una provincia o un cuatrimestre. Finalmente muestra la provincia con mayor cantidad de autos vendidos

# Ejercicio # 01

De acuerdo al ejemplo anterior, modifique el ejercicio para que muestre la cantidad mínima en un cuatrimestre sin importar la provincia.



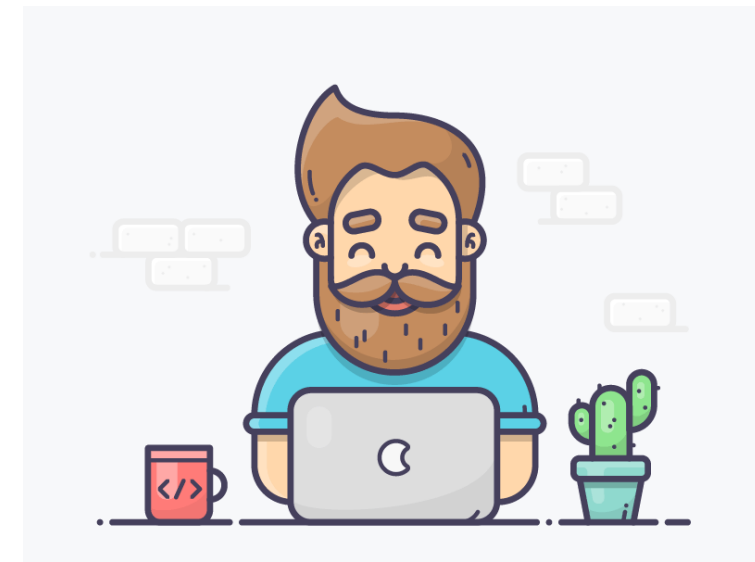
20 minutos



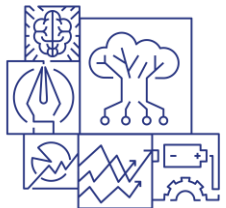


## Ejercicio # 02

Comparta con su profesor, algunos casos en los que considera que es necesario utilizar un arreglo de dos dimensiones para almacenar los datos generados en un programa.

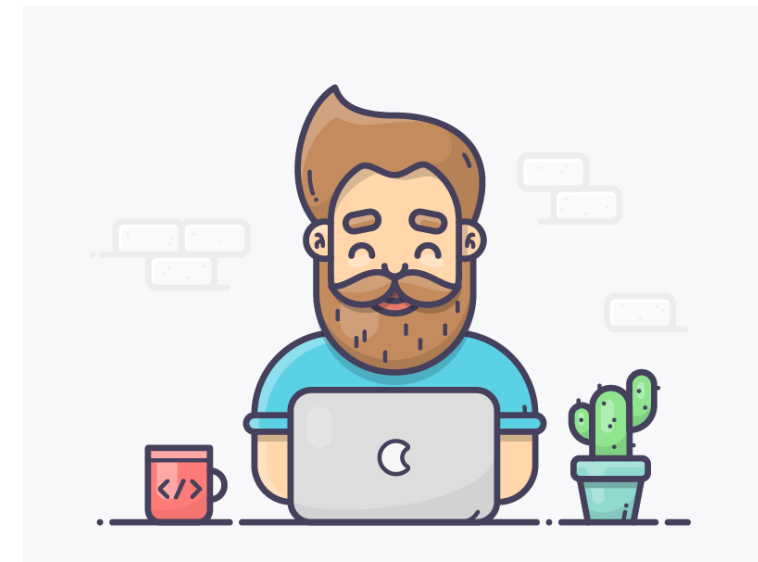


30 minutos

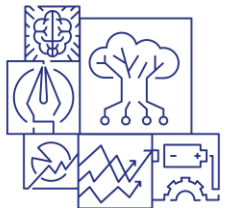


# Ejercicio # 03

De los casos encontrados anteriormente, desarrolle un ejercicio relacionado, su profesor podrá orientarlo al respecto.



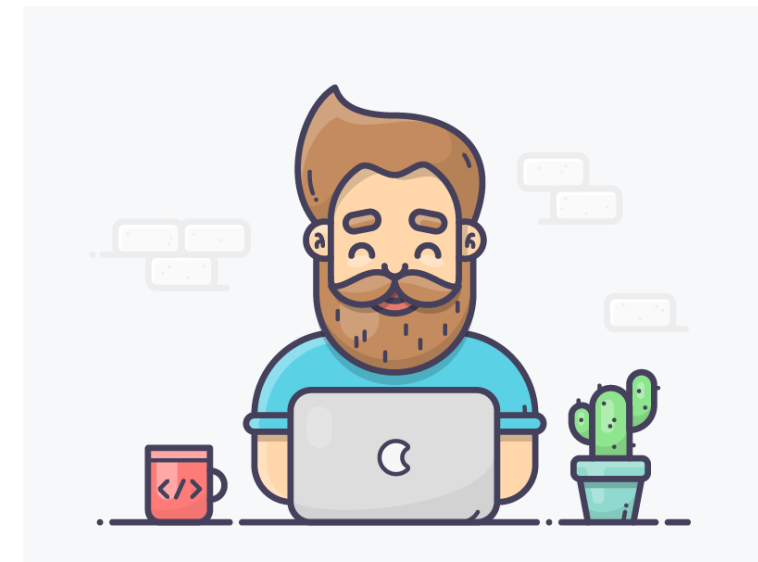
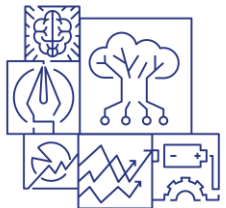
30 minutos



## Ejercicio # 04

Desarrolle un programa que almacene un valor 0 en los 60 espacios de un autobús en sus 4 horarios y que le solicite al usuario la posición que desea reservar, remplazando el valor por un 1 (que representa vendido).

¿Qué otro valor se podría utilizar?



40 minutos

Existen arreglos con mayor cantidad de dimensiones, son de mucha utilidad cuando se tienen más categorizaciones para los datos, pero están fuera del alcance de este curso.

