

# Arreglos Bidimensionales en Programación

## Elaboración y Recorridos

Bienvenidos a esta clase sobre arreglos bidimensionales en Python. Exploraremos su creación, manipulación y recorrido. Este conocimiento es fundamental para manejar datos estructurados en programación.

**Ing. Alfredo Colcha O., Mgs.**

# Objetivos de la Clase

- 1 Comprender Arreglos Bidimensionales  
Aprenderemos qué son los arreglos bidimensionales y su importancia en programación.
- 2 Crear y Manipular en Python  
Exploraremos la sintaxis para crear y modificar arreglos bidimensionales en Python.
- 3 Recorrer con Bucles  
Dominaremos técnicas para recorrer arreglos bidimensionales utilizando bucles anidados.



# Concepto de Arreglos Bidimensionales

## Definición

Un arreglo bidimensional es una estructura de datos que organiza elementos en filas y columnas. Es como una tabla o matriz.

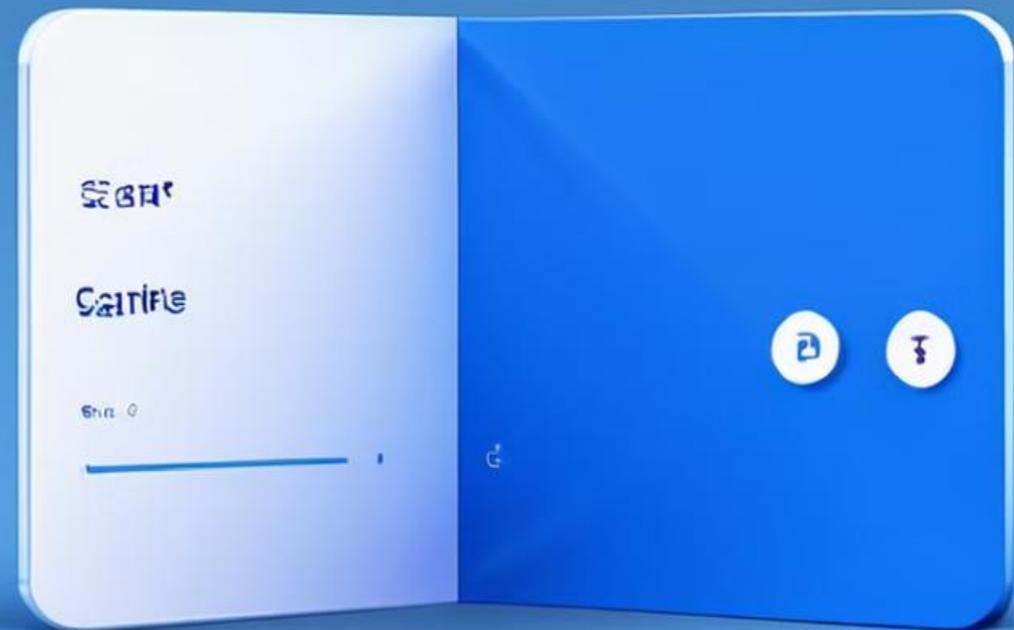
## Ejemplos

Se utilizan para representar tableros de juegos, hojas de cálculo, imágenes pixeladas y más.

## Ventajas

Permiten organizar datos de manera intuitiva y facilitan operaciones complejas en programación.

# Sintaxis y Creación de Arreglos en Python



## Definición Básica

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

## Comprensión de Lista

```
matriz = [[0 for j in range(3)] for i in range(3)]
```

## Usando NumPy

```
import numpy as np matriz = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
```

# Operaciones Básicas con Arreglos

1

Acceso a Elementos

`elemento = matriz[filas][columna]` Ejemplo: `primer_elemento = matriz[0][0]`

2

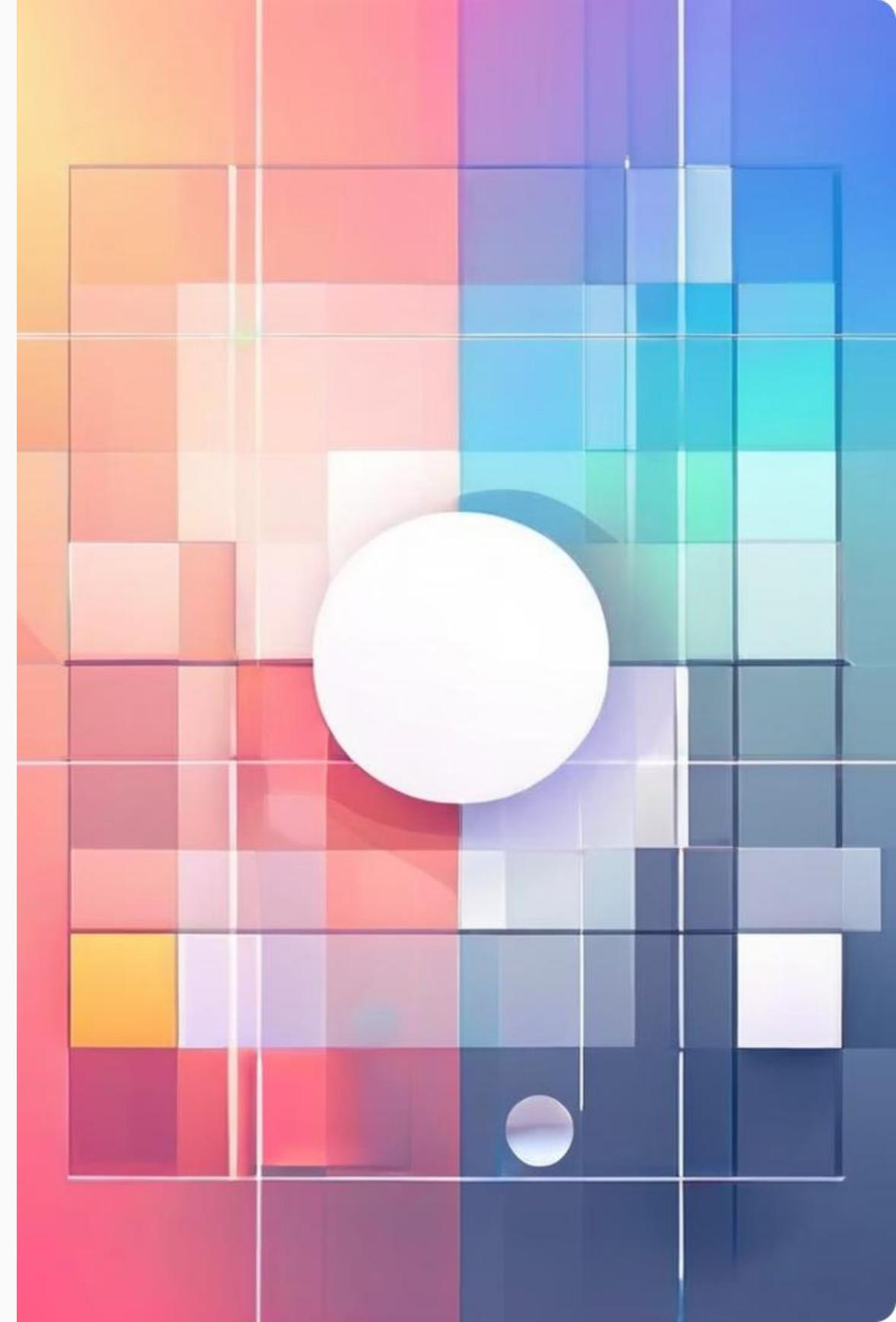
Modificación de Elementos

`matriz[filas][columna] = nuevo_valor` Ejemplo: `matriz[1][1] = 55`

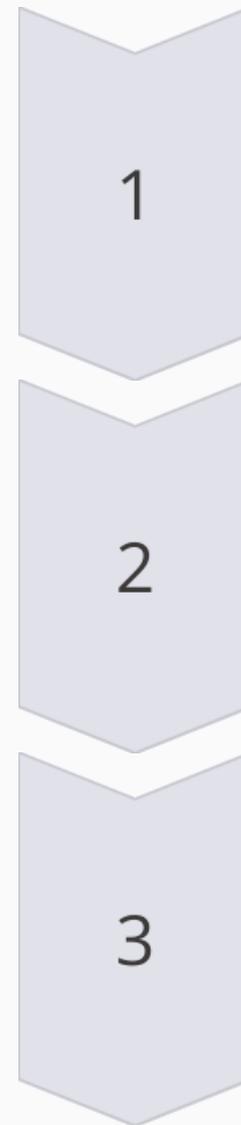
3

Recorrido Simple

`for fila in matriz: for elemento in fila: print(elemento)`



# Recorrer Arreglos con Bucles



## Bucle Externo

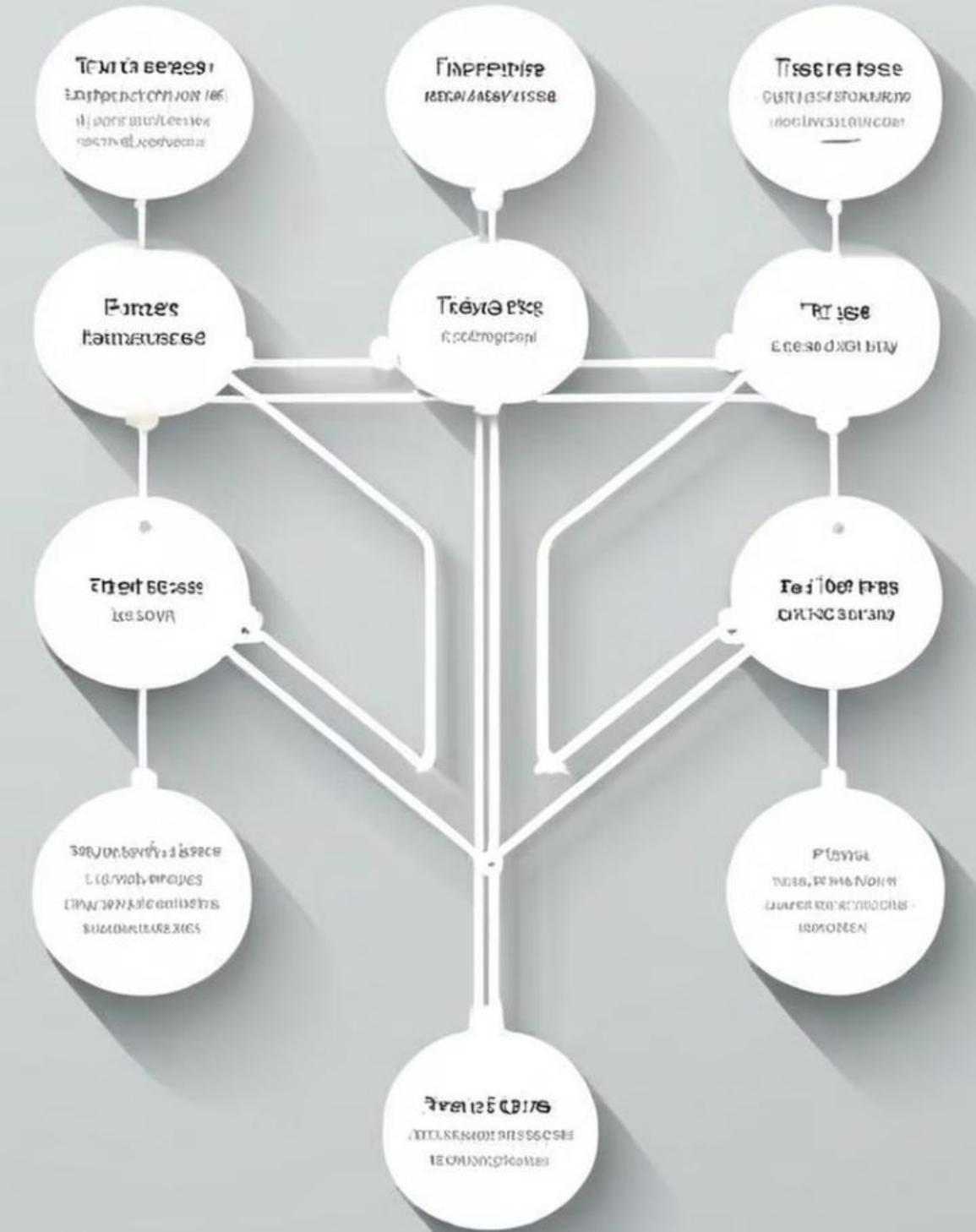
Itera sobre las filas del arreglo bidimensional.

## Bucle Interno

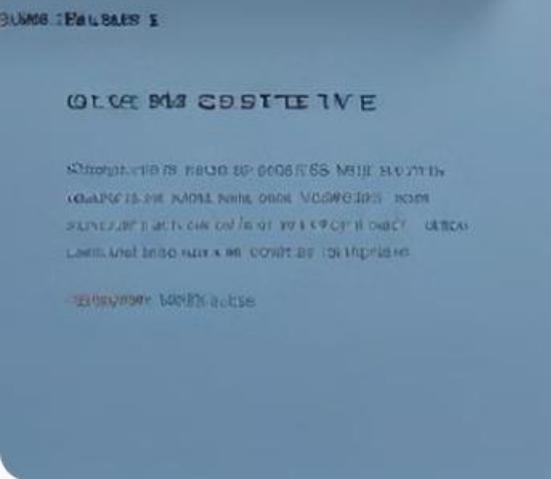
Recorre los elementos dentro de cada fila.

## Operaciones

Realiza acciones con cada elemento durante el recorrido.



# Ejemplos de Recorridos y Manipulación



Operación	Descripción	Ejemplo
Suma de Elementos	Suma todos los elementos de la matriz	suma += elemento
Valor Máximo	Encuentra el valor más alto en la matriz	if elemento > maximo: maximo = elemento
Contar Pares	Cuenta la cantidad de números pares	if elemento % 2 == 0: contador_pares += 1

# Resumen y Conclusiones



## Estructura

Los arreglos bidimensionales organizan datos en filas y columnas eficientemente.



## Manipulación

Python ofrece sintaxis clara para crear y modificar arreglos bidimensionales.



## Recorridos

Los bucles anidados son esenciales para procesar arreglos bidimensionales eficazmente.



## Aplicaciones

Los arreglos bidimensionales son cruciales en muchas áreas de la programación.

