

# Introducción a C# .NET

Módulo 4

# Arreglos multidimensionales

## Arreglos de más de una dimensión

Sólo a modo informativo y fuera del alcance de este curso, se menciona que los arreglos pueden tener **más de una dimensión**.

Un arreglo de dos dimensiones, es una matriz, y los elementos se acceden desde dos coordenadas, una que representa la fila y otra que representa la columna. Veamos:



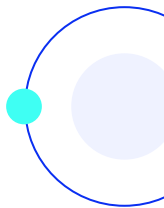
		Columnas								
Filas	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
	0	1	2	3	4	5	6	7	8	9

- Esta matriz tiene 2 filas y 10 columnas.
- Contiene números enteros positivos y negativos.
- El índice de la fila va desde 0 a 1 (son 2 filas).
- El índice de la columna va desde 0 a 9 (son 10 columnas).

Por ejemplo:

Creamos una matriz de 2 filas y 10 columnas, la llenamos con números aleatorios positivos menores a 100 (con el generador de números Random), y a la vez vamos mostrando el valor de cada elemento de la matriz:

```
Matriz
Matriz[0,0] = 0
Matriz[0,1] = 97
Matriz[0,2] = 2
Matriz[0,3] = 24
Matriz[0,4] = 91
Matriz[0,5] = 82
Matriz[0,6] = 84
Matriz[0,7] = 17
Matriz[0,8] = 9
Matriz[0,9] = 22
Matriz[1,0] = 73
Matriz[1,1] = 12
Matriz[1,2] = 29
Matriz[1,3] = 32
Matriz[1,4] = 9
Matriz[1,5] = 57
Matriz[1,6] = 72
Matriz[1,7] = 50
Matriz[1,8] = 48
Matriz[1,9] = 80
```



```
static void Main(string[] args)
{
    const int MaxFilas = 2;
    const int MaxColumnas = 10;
    Random objRandom = new Random();

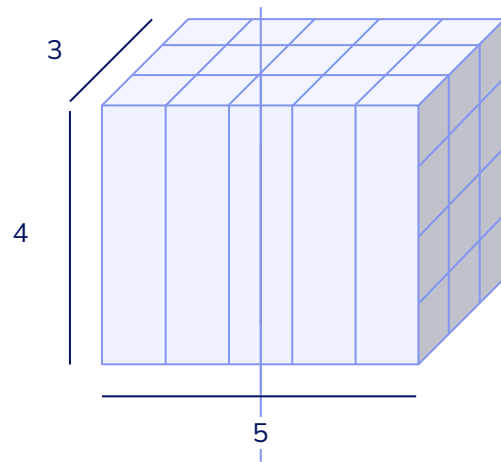
    Console.WriteLine("Matriz");

    int[,] Matriz = new int[MaxFilas , MaxColumnas];

    for (int i = 0; i < MaxFilas ; i++)
    {
        for (int j = 0; j < MaxColumnas ; j++)
        {
            Matriz[i, j] = objRandom.Next(100);
            Console.WriteLine($"Matriz[{i},{j}] = {Matriz[i, j]}");
        }
    }
    Console.ReadKey();
}
```

Si se imaginara una dimensión más, se tendría un cubo que es un arreglo de tres dimensiones. Sus elementos se acceden con **tres coordenadas**, la primera que representa la fila, la segunda que representa la columna, y una tercera que representa la profundidad dentro del cubo.

En la imagen de la derecha se ilustra un cubo de 4 filas, 5 columnas y 3 niveles de profundidad.



Un array de tres dimensiones (4 x 5 x 3)



Por ejemplo:

Creamos un cubo de 4 filas, 5 columnas y 3 niveles de profundidad, lo llenamos con números aleatorios positivos menores a 100 (con el generador de números Random), y a la vez vamos mostrando el valor de cada elemento del cubo:



Cubo

```
Cubo [0,0,0] = 89
Cubo [0,0,1] = 14
Cubo [0,0,2] = 52
Cubo [0,1,0] = 0
Cubo [0,1,1] = 82
Cubo [0,1,2] = 26
Cubo [0,2,0] = 76
Cubo [0,2,1] = 90
Cubo [0,2,2] = 59
Cubo [0,3,0] = 95
Cubo [0,3,1] = 5
Cubo [0,3,2] = 19
Cubo [0,4,0] = 58
Cubo [0,4,1] = 68
Cubo [0,4,2] = 93
Cubo [1,0,0] = 14
Cubo [1,0,1] = 46
Cubo [1,0,2] = 88
```



```
static void Main(string[] args)
{
    const int MaxFilas= 4;
    const int MaxColumnas = 5;
    const int MaxNiveles = 3;

    Random objRandom = new Random();

    Console.WriteLine("Cubo" + Environment.NewLine);

    int[, ,] Cubo = new int[MaxFilas, MaxColumnas , MaxNiveles];

    for (int i = 0; i < MaxFilas; i++)
    {
        for (int j = 0; j < MaxColumnas; j++)
        {
            for (int k = 0; k < MaxNiveles; k++)
```



...

```
        {  
            Cubo[i, j, k] = objRandom.Next(100);  
            Console.WriteLine($"Cubo [{i},{j},{k}] = {Cubo[i, j, k]}");  
        }  
    }  
    Console.ReadKey();  
}
```



**¡Sigamos  
trabajando!**