

# Arrays en Java

## 2. Array multidimensional en Java

```
TipodeDato[][] nombre=new TipodeDato[tamaño][tamaño];  
//el primer "tamaño" son filas, y el segundo "tamaño" son  
columnas
```

## Tipos de Arrays en Java

1. Array de una dimensión
2. Array multidimensional

## Array

Almacena una colección secuencial de un tamaño fijo de elementos del mismo tipo. Un array o matriz se utiliza para almacenar una colección de datos, pero a menudo es más útil pensar en una matriz como una colección de variables del mismo tipo.

En lugar de declarar variables individuales, como numero0, numero1, ..., y numero99, se declara una matriz como por ejemplo: numeros y se utiliza numeros[0], numeros[1] y ..., numeros[99] para representar variables individuales dentro de esa matriz.

## 1. Array de una dimensión en Java

Instanciación de un Array en Java

```
TipodeDato [] nombre=new TipodeDato[tamaño];
```

## Instanciar un arreglo dentro de un array

```
matriz[0][0]=1;  
matriz[0][1]=2;  
matriz[0][2]=3;  
matriz[1][0]=4;  
matriz[1][1]=5;  
matriz[1][2]=6;  
matriz[2][0]=7;  
matriz[2][1]=8;  
matriz[2][2]=9;
```

## Ejemplo de array de dos dimensiones

```
public class EjemploArray2D {  
    public static void main(String[] args) {  
        // Crear un array de dos dimensiones  
        int[][] matriz = new int[3][3];  
        // Inicializar los elementos del array  
        for (int i = 0; i < matriz.length; i++) {  
            for (int j = 0; j < matriz[i].length; j++) {  
                matriz[i][j] = i * j;  
            }  
        }  
        // Imprimir el array  
        for (int i = 0; i < matriz.length; i++) {  
            for (int j = 0; j < matriz[i].length; j++) {  
                System.out.print(matriz[i][j] + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

## Ejemplo de un array de una dimensión en Java

Nota: Recuerda: La primera posición del array es 0.

```
class ArrayUnidimensional {  
    public static void main(String[] args) {  
        // Crear un array unidimensional  
        int[] array = new int[5];  
        // Inicializar los elementos del array  
        array[0] = 10;  
        array[1] = 20;  
        array[2] = 30;  
        array[3] = 40;  
        array[4] = 50;  
        // Imprimir el array  
        for (int i = 0; i < array.length; i++) {  
            System.out.print(array[i] + " ");  
        }  
    }  
}
```

# Arrays en Java

## 2. Array multidimensional en Java

`TipodeDato[][] nombre=new TipodeDato[tamaño][tamaño];`  
//el primer "tamaño" son filas, y el segundo "tamaño" son columnas

*Declara variables dentro de un array multidimensional*

```
miArray[0][0]=1;
miArray[0][1]=2;
miArray[0][2]=3;
miArray[1][0]=4;
miArray[1][1]=5;
miArray[1][2]=6;
miArray[2][0]=7;
miArray[2][1]=8;
miArray[2][2]=9;
```

## Ejemplo de array de dos dimensiones

```
public class ProbandoArrays {
    public static void main(String[] args) {
        //Declara array de dos dimensiones
        int[][] miArray = new int[3][4];
        //Inicializa los elementos del array
        for (int i = 0; i < miArray.length; i++) {
            for (int j = 0; j < miArray[i].length; j++) {
                miArray[i][j] = i * j;
            }
        }
        //Imprime el array
        System.out.println(miArray);
    }
}
```

## Tipos de Arrays en Java

1. Array de una dimensión
2. Array multidimensional

## Array

Almacena una colección secuencial de un tamaño fijo de elementos del mismo tipo. Un array o matriz se utiliza para almacenar una colección de datos, pero a menudo es más útil pensar en una matriz como una colección de variables del mismo tipo.

En lugar de declarar variables individuales, como `numero0`, `numero1`, ... y `numero99`, se declara una matriz como por ejemplo: `numeros` y se utiliza `numeros[0]`, `numeros[1]` y ..., `numeros[99]` para representar variables individuales dentro de esa matriz.

## 1. Array de una dimensión en Java

Instanciación de un Array en Java

`TipodeDato [] nombre=new TipodeDato[tamaño];`

## Ejemplo de un array de una dimensión en Java

Zero Based: La primera posición del array es la [0]

```
class arrayProbado {
    public static void main(String[] args) {
        //Declara array de una dimensión
        int miArray[]=new int[5]; //declaración del array
        //Inicializa los elementos del array
        miArray[0]=1;
        miArray[1]=2;
        miArray[2]=3;
        miArray[3]=4;
        miArray[4]=5;
        //Imprime array
        for (int i = 0; i < miArray.length; i++) {
            System.out.println(miArray[i]);
        }
    }
}
```

# *Array*

**Almacena una colección secuencial de un tamaño fijo de elementos del mismo tipo. Un array o matriz se utiliza para almacenar una colección de datos, pero a menudo es más útil pensar en una matriz como una colección de variables del mismo tipo.**

**En lugar de declarar variables individuales, como numero0, numero1, ... y numero99, se declara una matriz como por ejemplo: numeros y se utiliza numeros[0], numeros[1] y ..., numeros[99] para representar variables individuales dentro de esa matriz.**



# ***Tipos de Arrays en Java***

- 1. Array de una dimensión**
- 2. Array multidimensional**

# ***1. Array de una dimensión en Java***

**Instanciación de un Array en Java**

**TipodeDato [] nombre=new TipodeDato[tamaño];**

# *Ejemplo de un array de una dimensión en Java*

Zero Based: La primera posición del array es la [0]

```
class arrayPrueba{
    public static void main(String args[]){

        int miArray[]=new int[5];//declaracion del array
        miArray[0]=10;//inicializacion de las posiciones del array
        miArray[1]=20;
        miArray[2]=70;
        miArray[3]=40;
        miArray[4]=50;

        //imprimo array
        for(int i=0;i<miArray.length;i++)//length es una propiedad de
        los arrays que indica su longitud
        System.out.println(miArray[i]);

    }
}
```

## *2.Array multidimensional en Java*

```
TipodeDato[][] nombre=new TipodeDato[tamaño][tamaño];  
//el primer "tamaño" son filas, y el segundo "tamaño" son  
columnas
```

## *Inicializar variables dentro de un array multidimensional*

```
miArray[0][0]=1;  
miArray[0][1]=2;  
miArray[0][2]=3;  
miArray[1][0]=4;  
miArray[1][1]=5;  
miArray[1][2]=6;  
miArray[2][0]=7;  
miArray[2][1]=8;  
miArray[2][2]=9;
```



```
public class PruebaArrays2D{  
    public static void main(String args[]){  
        int[][] miArray=new int[3][3];  
        //Declaro array de dos dimensiones  
        miArray[0][0]=1;  
        miArray[0][1]=2;  
        miArray[0][2]=3;  
        miArray[1][0]=4;  
        miArray[1][1]=5;  
        miArray[1][2]=6;  
        miArray[2][0]=7;  
        miArray[2][1]=8;  
        miArray[2][2]=9;  
  
        //Imprimir array de dos dimensiones  
        for(int i=0;i<3;i++){  
            for(int j=0;j<3;j++){  
                System.out.print(miArray[i][j]+" ");  
            }  
            System.out.println();  
        }  
    }  
}
```

# Arrays en Java

## 2. Array multidimensional en Java

`TipodeDato[][] nombre=new TipodeDato[tamaño][tamaño];`  
//el primer "tamaño" son filas, y el segundo "tamaño" son columnas

*Declara variables dentro de un array multidimensional*

```
miArray[0][0]=1;
miArray[0][1]=2;
miArray[0][2]=3;
miArray[1][0]=4;
miArray[1][1]=5;
miArray[1][2]=6;
miArray[2][0]=7;
miArray[2][1]=8;
```

## Ejemplo de array de dos dimensiones

```
public class ProbandoArrays {
    public static void main(String[] args) {
        //Declara array de dos dimensiones
        int[][] miArray = {
            {1, 2, 3},
            {4, 5, 6},
            {7, 8, 9}
        };
        //Imprime array de dos dimensiones
        System.out.println("Array de dos dimensiones:");
        for (int i = 0; i < miArray.length; i++) {
            for (int j = 0; j < miArray[i].length; j++) {
                System.out.print(miArray[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

## Array

Almacena una colección secuencial de un tamaño fijo de elementos del mismo tipo. Un array o matriz se utiliza para almacenar una colección de datos, pero a menudo es más útil pensar en una matriz como una colección de variables del mismo tipo.

En lugar de declarar variables individuales, como `numero0`, `numero1`, ... y `numero99`, se declara una matriz como por ejemplo: `numeros` y se utiliza `numeros[0]`, `numeros[1]` y ..., `numeros[99]` para representar variables individuales dentro de esa matriz.

## Tipos de Arrays en Java

1. Array de una dimensión
2. Array multidimensional

## 1. Array de una dimensión en Java

Instanciación de un Array en Java

`TipodeDato [] nombre=new TipodeDato[tamaño];`

## Ejemplo de un array de una dimensión en Java

Zero Based: La primera posición del array es la [0]

```
class arrayProbado {
    public static void main(String[] args) {
        //Declara array de una dimensión
        int miArray[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
        //Imprime array
        for (int i = 0; i < miArray.length; i++) {
            System.out.print(miArray[i] + " ");
        }
    }
}
```