



ARREGLOS



Un array o matriz es simplemente una variable que puede contener valores múltiples, a diferencia de una variable regular que solo puede contener un único valor.

Los siguientes son algunos puntos importantes acerca de las matrices, arreglos o arrays de Java.

- En Java, todas las matrices se asignan dinámicamente. (Se analiza a continuación)
- Como las matrices/arrays son objetos en Java, cada array tiene asociado una variable de instancia de longitud (length) que contiene la cantidad de elementos que la matriz puede contener. (En otras palabras, length contiene el tamaño de la matriz.)
- Una variable array en Java se declara como otras variables con corchetes [] después del tipo de datos.



- Las variables en el array están ordenadas y cada una tiene un índice que comienza desde 0.
- El array Java también se puede usar como un campo estático, una variable local o un parámetro de método.
- El tamaño de un array debe especificarse mediante un valor int y no, long o short.
- La superclase directa de un tipo de array es Object.
- Cada tipo de array implementa las interfaces Cloneable y java.io.Serializable.
- El array puede contener tipos de datos primitivos, así como también objetos de una clase según la definición del array. En el caso de los tipos de datos primitivos, los valores reales se almacenan en ubicaciones de memoria contigua. En el caso de los objetos de una clase, los objetos reales se almacenan en heap.

Los arreglos se pueden definir como objetos en los que podemos guardar más de una variable, es decir, al tener un único arreglo, este puede guardar múltiples variables de acuerdo a su tamaño o capacidad, es importante recordar que las variables guardadas deben ser del mismo tipo, por ejemplo: Si tenemos un arreglo de tipo Numérico que puede almacenar 10 variables, solo podrá almacenar 10 números diferentes, no otras variables como caracteres o Strings.

Existen 2 tipos de arreglos, los unidimensionales, y los multidimensionales (generalmente 2 dimensiones y se les denomina matrices), en esta lección veremos cómo declarar y utilizar los arreglos unidimensionales, estos son más sencillos y fáciles de comprender, y pueden servirnos para muchas aplicaciones.

Declaración:

La estructura de declaración de un arreglo es la siguiente:

```
tipo_dato [] nombre_variable; //también puede declararse "tipo_dato  
nombre_variable[];"
```

Arrays unidimensionales

Un array unidimensional es una lista de variables relacionadas. Tales listas son comunes en la programación. Por ejemplo, puede usar un array unidimensional para almacenar los números de cuenta de los usuarios activos en una red. Otro



array podría usarse para almacenar los promedios de bateo actuales para un equipo de béisbol.

La forma general de declarar un arreglo unidimensional es:

tipo nombre-array[];

o

tipo [] nombre-array;

La declaración de un array tiene dos componentes: el tipo y el nombre.

tipo declara el tipo de elemento del array. El tipo de elemento determina el tipo de datos de cada elemento que comprende la matriz. Al igual que la matriz de tipo int, también podemos crear una matriz de otros tipos de datos primitivos como char, float, double..etc o tipo de datos definido por el usuario (objetos de una clase). Por lo tanto, el tipo de elemento para la matriz determina el tipo de datos que la matriz contendrá.

Ejemplo:

// ambas son declaraciones válidas

int intArray[];

int[] intArray;

//Tipo de datos primitivos

byte byteArray[];

short shortArray[];

boolean booleanArray[];

long longArray[];

float floatArray[];

double doubleArray[];

char charArray[];

//Tipos de datos definidos por el usuario

```
// una serie de referencias a objetos de  
// la clase MyClass (una clase creada por  
// el usuario)  
MyClass myClassArray[];  
Object[] ao,      // array de Object  
Collection[] ca; // array de Collection
```

Aunque la primera declaración anterior establece el hecho de que `intArray` es una variable de matriz, en realidad no existe una matriz. Simplemente le dice al compilador que esta variable (`intArray`) contendrá una matriz del tipo entero. Para vincular `intArray` con una matriz física real de enteros, debe asignar una usando `new` y asignarlo a `intArray`.