

## ARREGLOS EN JAVA UNA DIMENSION

### Crear un arreglo

La forma de crear un arreglo en JAVA es como sigue:

- Definimos la variable como array usando paréntesis cuadrados.
- Creamos el arreglo usando la palabra reservada new, luego el tipo y finalmente el tamaño (también llamada dimensión).

#### Ejemplo 01:

```
public class Ejer040101 {  
    public static void main(String[] args) {  
        // Definir arreglos  
        // Crea apuntador inicial al arreglo  
        int [] a;  
  
        // Crea 5 espacios continuos para valores enteros  
        a = new int[5];  
    }  
}
```

Esta forma es hacerla línea a línea, para que vea el proceso que se lleva a cabo para la creación del arreglo.

## Ejemplo 02: Cargar datos al arreglo

```
public class Ejer040102 {  
  
    public static void main(String[] args) {  
        // Definir arreglos  
        // Crea apuntador inicial al arreglo  
        int [] a;  
  
        // Crea 5 espacios continuos para valores enteros  
        a = new int[5];  
        // Guarda valores en cada localidad del arreglo  
        a[0] = 20;  
        a[1] = 10;  
        a[2] = 2;  
        a[3] = 8;  
        a[4] = 50;  
  
    }  
}
```

En este ejemplo cargamos de forma estática un conjunto de valores a cada posición del arreglo. Debemos tener siempre en cuenta que se comienza en la posición CERO.

## Ejemplo 03: Cargar e imprimir

```
import java.util.Scanner;
public class Ejer0400103 {

    public static void main(String[] args) {
        // Definir arreglo

        int [] a = new int[5];
        Scanner teclado = new Scanner(System.in);

        // Ciclo para cargar datos
        for (int i = 0; i<5; i++){
            System.out.printf ("Escriba un numero para posicion: %d %n",i);
            a[i] = teclado.nextInt();
        }

        // Ciclo para desplegar contenido
        // Secuencia de datos almacenados
        for (int i = 0; i<5; i++){
            System.out.printf ("%n Escriba un numero para posicion: %d es %d %n",
                i, a[i]);
        }
    }
}
```

Aquí ya se vuelve dinámica la carga de datos y al mismo tiempo la crecion del arreglo se hace en una sola línea.

### Secuencia de Carga

```
General Output
-----Configuration:
Escriba un numero para posicion: 0
10
Escriba un numero para posicion: 1
4
Escriba un numero para posicion: 2
6
Escriba un numero para posicion: 3
1
Escriba un numero para posicion: 4
20
```

### Secuencia de impresión

```
General Output
Escriba un numero para posicion: 0 es 10
Escriba un numero para posicion: 1 es 4
Escriba un numero para posicion: 2 es 6
Escriba un numero para posicion: 3 es 1
Escriba un numero para posicion: 4 es 20
Process completed.
```

## MANEJO INDIVIDUAL DE CONTENIDO DE UN ARREGLO

Los arreglos pueden ser consultados en su totalidad (como lo hicimos en el programa anterior) o bien solo por posiciones específicas.

Si queremos saber el contenido de la posición 2 y 4 del arreglo, solo debemos de indicar el arreglo y su posición deseada.

```
public class Ejer040104 {  
    public static void main(String[] args) {  
        int [] a = new int[5];  
  
        a[0] = 20;  
        a[1] = 10;  
        a[2] = 2;  
        a[3] = 8;  
        a[4] = 50;  
  
        System.out.printf ("Contenido Posicion 2 es %d ",a[2]);  
        System.out.printf ("Contenido Posicion 4 es %d ",a[4]);  
    }  
}
```

### Resultado

```
General Output  
-----Configuration: <Default>-----  
Contenido Posicion 2 es 2 Contenido Posicion 4 es 50  
Process completed.  
  
Aqui coincide que la posicion 2 tenga un valor de 2 :)
```

## ARREGLO COMO OBJETO

Al ser un objeto el arreglo, podemos saber la dimensión de éste mediante una propiedad que precisamente tiene la longitud de este. Veamos un ejemplo:

```

import java.util.Scanner;
public class Ejer040105 {

    public static void main(String[] args) {
        // Definir arreglo
        int [] a = new int[3];
        Scanner teclado = new Scanner(System.in);

        int tope = a.length;

        // Ciclo para cargar datos
        for (int i = 0; i < tope; i++){
            System.out.printf ("Escriba un numero para posicion: %d %n",i);
            a[i] = teclado.nextInt();
        }

        // Ciclo para desplegar contenido
        // Secuencia de datos: 1, 2, 3, 4, 5
        for (int i = 0; i < a.length; i++){
            System.out.printf ("%n Escriba un numero para posicion: %d es %d %n",
                i, a[i]);
        }
        // TODO code application logic here
    }
}

```

**NOTA:** Observe como manejamos el limite final de la dimensión en el ciclo for, en función de la propiedad length.