

## Ejemplos: Arreglos Unidimensionales

### Ejemplo 4.1

Construye el programa en Java que, en tiempo de programación, cree un arreglo con 5 edades, 4 gastos y 3 nombres. Que sume el total de los gastos, y después que muestre los elementos de cada arreglo.

#### Explicación

1. Declarar y crear el arreglo EDADES[ ] con 5 elementos de tipo entero.

```
int edades[] = {17, 18, 19, 20, 21};
```

en la *posición 0* se guarda la edad 17

en la *posición 1* la edad 18

en la *posición 2* la edad 19

en la *posición 3* la edad 20

y en la *posición 4* la edad 21

El arreglo tendrá el tamaño según la cantidad de elementos que se escribieron, y una vez creado el arreglo, ya no puede crecer su tamaño.

2. Declarar y crear el arreglo GASTO[ ] con 4 elementos de tipo real (parte fraccionaria).

```
float []gasto = {5.25F, 10.5F, 15.75F, 20.0F };
```

3. Declarar y crear el arreglo NOMBRE[ ] con 3 elementos de tipo cadena de caracteres.

```
String nombre[] = {"Maty", "Mau", "Jana" };
```

4. Una vez que se han creado los arreglos con valores, es posible hacer operaciones con ellos, por ejemplo, sumar los 4 gastos del arreglo.

Para ello se empieza con el primer gasto que se encuentra en la posición 0 hasta llegar al último gasto, que se encuentra en la posición 3.

```
float suma = gasto[0]+gasto[1]+gasto[2]+gasto[3];
```

Si se intenta sumar el gasto[4] marcará un error porque el arreglo solo llega hasta la posición 3.

5. Mostrar la tercera edad, que se encuentra en la posición 2 porque el arreglo empieza en la posición 0.

```
System.out.println(edades[2]); // mostrará 19
```

6. Mostrar los elementos de cada arreglo.

La sentencia **Arrays.toString(nombre\_arreglo)** devuelve una representación de cadena del contenido del arreglo.

```
System.out.println(Arrays.toString(edades));
```

```
System.out.println(Arrays.toString(gasto));
```

```
System.out.println(Arrays.toString(nombre));
```

## Ejemplo 4.2

Construye el programa en Java que cree un arreglo para guardar 5 edades leídas del teclado. Después mostrar las edades mediante un ciclo.

### Explicación

1. Declarar y crear el arreglo EDADES[ ] de tipo entero para 5 elementos.

```
int edades[] = new int[5];
```

*Importante:* el arreglo creado tiene capacidad para 5 elementos, que se van a guardar de la posición 0 a la posición 4.

2. Introducir cada una de las edades (desde la posición 0 a la posición 4).

```
System.out.print("Introduce la edad 1: ");
edades[0] = Leer.nextInt();
System.out.print("Introduce la edad 2: ");
edades[1] = Leer.nextInt();
System.out.print("Introduce la edad 3: ");
edades[2] = Leer.nextInt();
System.out.print("Introduce la edad 4: ");
edades[3] = Leer.nextInt();
System.out.print("Introduce la edad 5: ");
edades[4] = Leer.nextInt();
```

3. Es posible mostrar una edad en particular, por ejemplo, mostrar la edad 3 (como el arreglo empieza en la posición 0, entonces la edad 3 es la que se guardó en la posición 2).

```
System.out.print("Edad 3 = " + edades[2]);
```

4. Utilizar un ciclo para mostrar cada uno de los elementos del arreglo.

El índice *i* empieza en la posición 0, porque el primer elemento del arreglo se guardó en la posición 0.

**edades.length** contiene el tamaño del arreglo, en este caso tendrá el valor 5.

```
for (int i = 0; i < edades.length; i++) {
    System.out.println("Edad "+i+": "+edades[i]);
}
```

### Ejemplo 4.3

Construye el programa en Java que mediante un ciclo guarde en un arreglo N gastos leídos del teclado. Después mostrar cada uno de los gastos y la suma de todos los gastos

#### Explicación

1. Obtener los N gastos solicitando al usuario que introduzca el valor de N.

```
System.out.print("¿Cuántos gastos son? ");  
int N = Leer.nextInt();
```

2. Declarar y crear el arreglo GASTOS[] para N elementos de tipo real (con parte fraccionaria).

```
float gastos[] = new float[N];
```

3. Utilizar un ciclo para leer cada uno de los gastos del arreglo.

El índice *i* empieza en la posición 0, porque el primer elemento del arreglo se guardará en la posición 0.

**gastos.length** contiene el tamaño del arreglo, en este caso es igual al valor leído para la variable N.

```
for (int i = 0; i < gastos.length; i++) {  
    System.out.print("Introduce el gasto "+i+": ");  
    gastos[i] = Leer.nextFloat();  
}
```

En cada iteración del ciclo, el índice *i* irá recorriendo cada posición del arreglo, por lo que, al momento de leer el gasto éste se guardará en la posición del valor de *i*.

4. Mostrar cada uno de los gastos, lo cual se hará con un ciclo.

El índice *i* empieza en la posición 0, porque el primer gasto se guardó en la posición 0.

**gastos.length** contiene el tamaño del arreglo, en este caso es igual al valor leído para la variable N.

```
for (int i = 0; i < gastos.length; i++) {  
    System.out.println("Gasto "+i+": "+gastos[i]);  
}
```

5. Obtener la suma de todos los gastos y mostrar su resultado.

Declarar el acumulador suma al principio del programa, e inicializarlo en cero.

```
float suma=0;
```

Acumular los gastos, ya sea en el ciclo de la lectura de los gastos o en el ciclo donde se están mostrando.

```
for (int i = 0; i < gastos.length; i++) {  
    System.out.println("Gasto "+i+": "+gastos[i]);  
    suma = suma + gastos[i];  
}
```

Mostrar el total de gastos, lo cual debe hacerse fuera del ciclo donde se calculó.

```
System.out.println("Total de gastos $" + suma);
```