

## Paso de parámetros

Ya hemos visto que a las funciones le podemos pasar una serie de parámetros que serán los datos de entrada a la función. Sin embargo, tenemos que tener presente que existen técnicamente dos formas de pasar parámetros:

- Paso por valor (visto en la Unidad 4)
- Paso por referencia

Es importante entender bien en qué se diferencia y por tanto vamos a verlo en detalle.

### Por referencia

La función no hace una copia, sino que hace referencia a los parámetros de entrada. Esto implica que las modificaciones que hagamos con los valores de estos parámetros se reflejarán en la función origen de la llamada.

Sin embargo, en Java esto no es posible con tipos de datos primitivos, sino con objetos. Los tipos objeto que hemos tratado hasta ahora son los Arrays, Strings, etc.

En el siguiente ejemplo creamos una función que recibe un parámetro tipo array de enteros y multiplica por 2 cada uno de los elementos del array:

```
public static void arrayPorDos(int[] miArray) // devuelve un entero
{
    for(int i=0; i < miArray.length; i++)
        miArray[i] = miArray[i] * 2;
}

public static void main(String[] args)
{
    int[] miArray = {2,3,4};

    System.out.println("Main: Contenido array antes de llamar a la función");
    for(int i=0; i < miArray.length; i++)
        System.out.print(miArray[i] + " ");

    arrayPorDos(miArray);

    System.out.println("\nMain: Contenido array después de llamar a la función");
    for(int i=0; i < miArray.length; i++)
        System.out.print(miArray[i] + " ");
}
```

Salida:

```
Main: Contenido array antes de llamar a la función
2 3 4
Main: Contenido array después de llamar a la función
4 6 8
```

Puedes observar que la función `arrayPorDos` no devuelve ningún resultado. Sin embargo, las modificaciones que realiza sobre el array que se le pasa por parámetro se ven reflejadas en el bloque de la función desde donde se hace la llamada.

El siguiente código hace lo mismo, pero destacamos en rojo aquello que no tendría sentido hacer: no hace falta que la función devuelva el array.

```
public static int[] arrayPorDos(int[] miArray) // devuelve un entero
{
    for(int i=0; i < miArray.length; i++)
        miArray[i] = miArray[i] * 2;

    return miArray;
}

public static void main(String[] args)
{
    int[] miArray = {2,3,4};

    System.out.println("Main: Contenido array antes de llamar a la función");
    for(int i=0; i < miArray.length; i++)
        System.out.print(miArray[i] + " ");

    miArray = arrayPorDos(miArray);

    System.out.println("\nMain: Contenido array después de llamar a la función");
    for(int i=0; i < miArray.length; i++)
        System.out.print(miArray[i] + " ");
}
```