

Los Java Generic Methods son métodos genéricos de Java que no pertenecen a una clase concreta . Esto en un primer momento nos puede resultar chocante pero a veces ocurre. Recordemos que la forma de construir clases genéricas es :

CURSO SPRING BOOT
GRATIS
APUNTATE!!

```
package com.arquitecturajava;

import java.util.ArrayList;
import java.util.List;

public class Bolsa<T> {

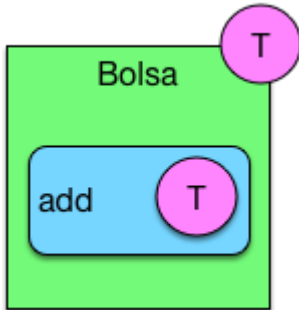
    private List<T> lista= new
    ArrayList<T>();

    public void add(T item) {

        lista.add(item);
    }

}
```

El método add es un método genérico ya que se apoya en el tipo T que la clase define.



Java Generic Methods

Sin embargo cuando nos referimos a métodos genéricos estamos hablando de métodos que son genéricos sin pertenecer a una clase genérica. Esto puede parecer raro en un primer momento , lo veremos más claro en código:

```
package com.arquitecturajava;
```

```
import java.util.ArrayList;  
import java.util.Arrays;  
import java.util.Collection;
```

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

```
        Collection<String> lista= new  
        ArrayList<String>(Arrays.asList("A","B","C"));
```

```
        Collection<String> nueva= duplicarColleccion(lista);
```

```
        nueva.forEach(x->System.out.println(x));
```

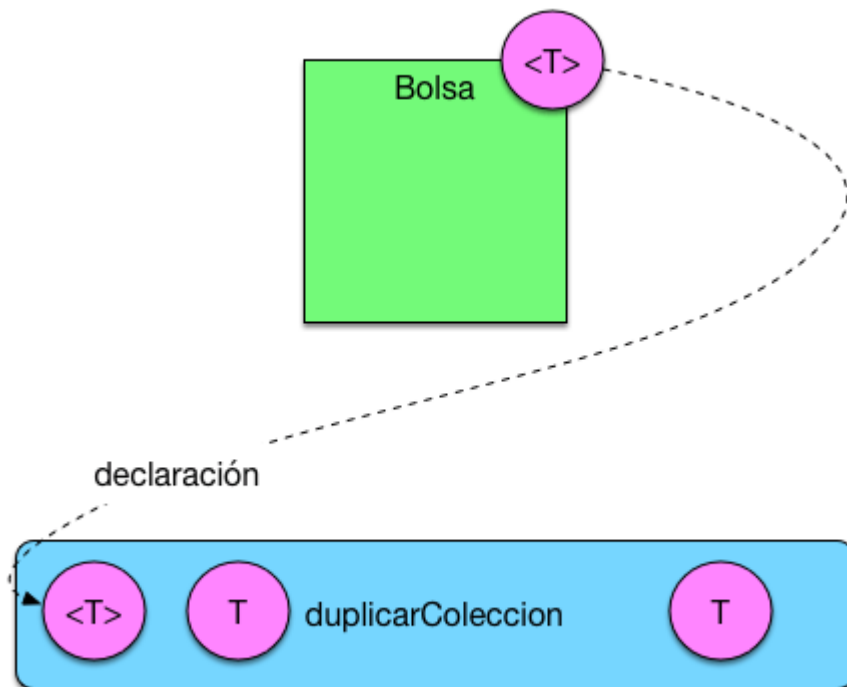
```
}

public static <T> Collection<T>
duplicarColeccion(Collection<T> coleccion) {

    coleccion.addAll(coleccion);
    return coleccion;

}
}
```

En este caso el método `duplicarColeccion` se encarga de volver a copiar los mismos elementos en la colección para tenerlos duplicados. Es un método genérico ya que admite cualquier tipo de colección cambiando el tipo `T`. Pero además no pertenece a ninguna clase en concreto ya que el tipo `T` se declara de forma explícita en el método. Para entenderlo de forma más sencilla es como si la declaración `<T>` de la clase parara al propio método.



Es un código que cuesta entender ya que parece que tiene dos parámetros de retorno genéricos y no solo uno. En primer lugar `<T>` y en segundo lugar `Collection<T>`. Los Java Generic Methods aportan una mayor flexibilidad a nuestra programación

**CURSO SPRING BOOT
GRATIS
APUNTATE!!**

Otros artículos relacionados:

[Java Generics](#)

[Java Generics WildCard<?>](#)

[Oracle Java Generics](#)