

Sobrecarga de métodos en Java

¿Qué es la sobrecarga (Overloading)?

En **Java**, dentro de una misma clase, pueden ser definidos dos o más métodos con el mismo nombre pero con parámetros diferentes. Cuando esto ocurre, se dice que el método está **sobrecargado**.

Cuando se llama a un método **sobrecargado**, **Java** utiliza el tipo y / o número de argumentos como guía para determinar qué versión del método debe ejecutar. Por lo tanto, **los métodos sobrecargados deben ser diferentes en el tipo y / o en el número de parámetros**:

```
miMetodoSobrecargado(2,"Hola");
```

¿Qué método
elige el
compilador?

```
miMetodoSobrecargado(int, String){...implementación}  
miMetodoSobrecargado(int, String,int){...implementación}  
miMetodoSobrecargado(String){...implementación}  
miMetodoSobrecargado(int,int){...implementación}
```

El primer método coincide en número de argumentos y en los tipos por lo que es el que elegirá el compilador

Normas en la sobrecarga (Overloading)

1

Las firmas deben ser diferentes: número de argumentos y/o tipos diferentes:

Firma del método

```
public String miMetodo(int arg1, String arg2) {...cosas que hace el método... }
```

NO

De esta forma **evitamos la ambigüedad**. De hecho **Java** no la consiente.

2

De la primera norma se deduce la segunda: no podemos sobrecargar dos métodos basándonos únicamente en el tipo de retorno. El tipo de retorno no está incluido en la firma del método.

Normas en la sobrecarga (Overloading)

3

Dos firmas con el mismo número de argumentos, mismos tipos pero en distinto orden se considerarán diferentes, por lo tanto válidos (aunque en la práctica no tiene ningún sentido. Siempre se diseña un método para un conjunto dado de parámetros que debe ocurrir independientemente del orden de los parámetros):

```
public String miMetodo(int arg1, String arg2) {...cosas que hace el método... }  
public String miMetodo(String arg1, int arg2) {...cosas que hace el método... }
```

4

Siempre que respetemos las normas anteriores es perfectamente posible sobrecargar métodos Static.

Resumen sobrecarga (Overloading)

Para **sobrecargar** un método, los argumentos pasados deben **DIFERIR** por lo menos en uno (aunque puede diferir en todos...) de los siguientes aspectos:

Número de argumentos

1

```
add(int, int);  
add(int, int, int);
```

Tipo de los argumentos

2

```
add(int, int);  
add(int, float);
```

Orden del tipo de los argumentos

3

```
add(float, int);  
add(int, float);
```