



APRENDERAPROGRAMAR.COM

CONCEPTO DE
PARÁMETROS FORMALES Y
ACTUALES EN MÉTODOS Y
CONSTRUCTORES JAVA.
EJEMPLO. (CU00628B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

Resumen: Entrega nº28 curso Aprender programación Java desde cero.

Autor: Alex Rodríguez

PARÁMETROS FORMALES Y PARÁMETROS ACTUALES EN MÉTODOS Y CONSTRUCTORES JAVA

Hemos visto que un método (o constructor) puede requerir un parámetro, con un tipo y nombre concretos. Por ejemplo tipo *String* y nombre del parámetro *valorDistrito*. A su vez *valorDistrito* podría ir tomando distintos valores a lo largo de una ejecución del método, por ejemplo “Norte”, “Oeste”, “Sur”, etc.



Vamos a definir dos conceptos relacionados con los parámetros:

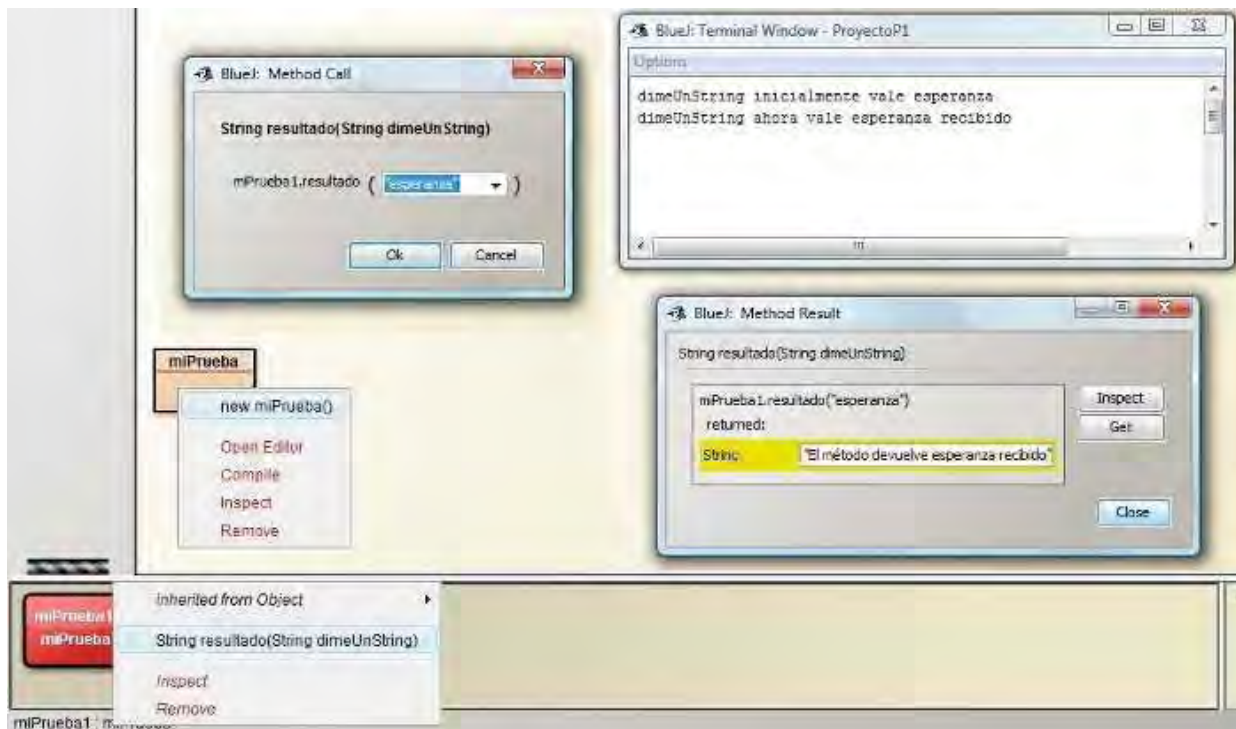
- a) **Parámetro formal:** es el nombre de un parámetro de un constructor o método tal y como se define en su cabecera, por ejemplo *valorDistrito* es un parámetro formal.
- b) **Parámetro actual:** es el valor concreto que tiene un parámetro en un momento dado.

Prueba a escribir y compilar el siguiente código:

```
//Esta clase es una prueba. Curso aprenderaprogramar.com Java desde cero
public class miPrueba {

    public String resultado (String dimeUnString) {
        System.out.println ("dimeUnString inicialmente vale " + dimeUnString);
        dimeUnString = dimeUnString + " recibido";
        System.out.println ("dimeUnString ahora vale " + dimeUnString);
        return "El método devuelve " + dimeUnString;
    }
}
```

Ahora crea un objeto de tipo *miPrueba* y ejecuta el método introduciendo una cadena, por ejemplo “esperanza”. El resultado será algo similar a esto:



El parámetro formal *dimeUnString* es de tipo *String* y inicialmente toma el valor que le hayamos pasado al constructor cuando creamos el objeto. Luego su valor actual pasa a ser el valor pasado al constructor + “recibido”. Finalmente el método nos devuelve una cadena de la que forma parte el parámetro.

Conclusión interesante: los parámetros de métodos o constructores también almacenan valores que pueden ir cambiando. Por eso diremos que también son variables al igual que los campos, aunque conceptualmente sean cosas distintas.

Ten en cuenta una cosa: un campo es una variable que está de forma permanente asociada a un objeto y cuyo ámbito (lugares donde podemos usarla) es toda la clase. Su tiempo de vida es indefinido mientras exista el objeto al que está ligado. En cambio **un parámetro formal tiene un ámbito limitado al método o constructor y tiempo de vida limitado** al tiempo durante el cual se ejecuta el método o constructor. En este sentido, decimos que un parámetro es un tipo de variable local (solo podemos usarla dentro del método) y temporal (se crea al comenzar la ejecución del método y se destruye al terminar su ejecución).

Para evitar confusiones, en general trataremos de evitar modificar el contenido de un parámetro dentro del código de un método.

Próxima entrega: CU00629B

Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) --> Cursos, o en la dirección siguiente:
http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188