



APRENDERAPROGRAMAR.COM

## MÉTODOS EN JAVA TIPO PROCEDIMIENTO (...VOID) Y TIPO FUNCIÓN (...RETURN). INVOCACIÓN. (CU00624B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

**Resumen:** Entrega nº24 curso Aprender programación Java desde cero.

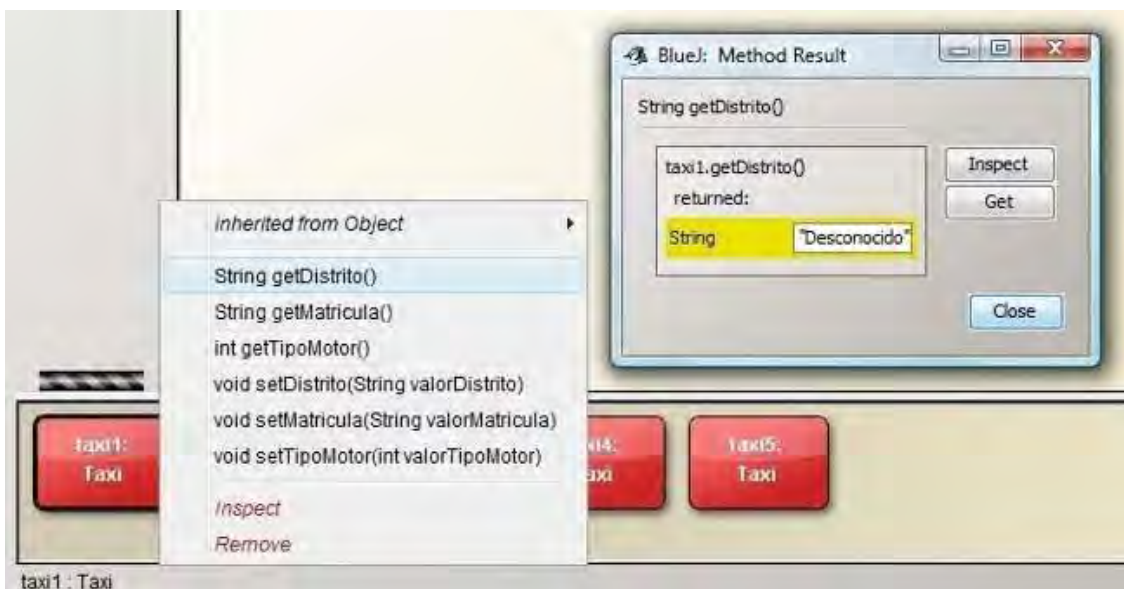
Autor: Alex Rodríguez

### MÉTODOS EN JAVA TIPO PROCEDIMIENTO (...VOID) Y TIPO FUNCIÓN (...RETURN)

Volvamos sobre los objetos taxi creados y que deben aparecer en el banco de objetos en la parte inferior izquierda de la pantalla del entorno de desarrollo. Si pulsamos con botón derecho del ratón sobre ellos se nos despliegan los métodos (operaciones) disponibles para cada objeto.

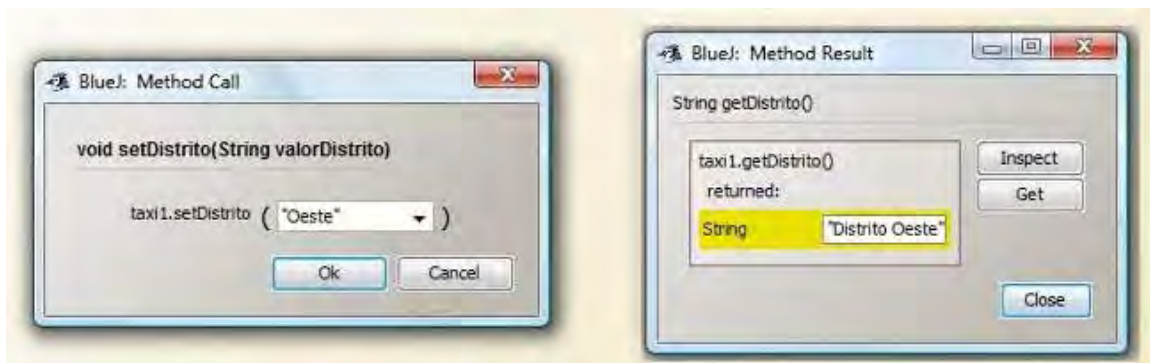


Pulsa sobre el taxi1 y elige la opción *String getDistrito*.



La ejecución de un método se denomina habitualmente **“invocación” del método o “llamada” al método**. Los métodos disponibles los define la clase, pero se invocan sobre cada objeto en particular. Al invocar el método *getDistrito()* se nos abre una ventana de BlueJ denominada *Method Result* donde nos indica: *returned String “Desconocido”*. Le hemos pedido al objeto que nos diga cuál es su distrito y nos devuelve “Desconocido”. La razón para ello es que en el constructor de la clase incluimos una línea de inicialización de distrito para todo objeto de tipo Taxi con el valor “Desconocido”. Si en vez de ese valor hubiésemos establecido otro, ese sería el que ahora obtendríamos.

Cierra esa ventana y repite el proceso eligiendo ahora la opción *void setDistrito (String valorDistrito)*. En el recuadro donde solicita el distrito escribe “Oeste”. No olvides incluir las comillas obligadas por tratarse de un String. Seguidamente, vuelve a invocar el método *getDistrito* y comprueba el resultado obtenido.



Los métodos que hemos definido en la clase Taxi podemos clasificarlos de la siguiente manera:

- a) **Métodos tipo función:** son métodos que nos devuelven algo. Un método es tipo función si comienza con un tipo (no consideramos ahora la palabra clave public). Por ejemplo *String getDistrito()* comienza con el tipo String lo que significa que nos devuelve una cadena de caracteres, mientras que *int getTipoMotor()* comienza con el tipo int lo que significa que nos devuelve un entero. Tener en cuenta que un método tipo función devuelve solo un dato u objeto como resultado, no varios. La devolución del resultado se expresa con la palabra clave *return* seguida del dato u objeto a devolver, por ejemplo *return tipoMotor;*. La sentencia *return* implica que termina la ejecución del código en el método y estará típicamente en la línea final. De existir una línea por detrás de una sentencia *return*, nunca llegaría a ejecutarse (tendremos que matizar esto porque el uso de condicionales nos permitirá tener más de un *return* en un método). De forma genérica:

```
//Comentario descriptivo de qué hace el método
public tipoValorDevuelto nombreDelMétodo (tipo parámetro1, tipo parámetro2...) {

    Código del método
    return ResultadoQueDevuelveElMétodo;
}
```

- b) **Métodos tipo procedimiento:** son métodos que realizan ciertas operaciones sin devolver un valor u objeto concreto. Un método es tipo procedimiento si comienza con la palabra clave *void* (que traducido del inglés viene siendo “vacío” o “nulo”). En estos casos podemos decir que el tipo de retorno es *void*. De forma genérica:

```
//Comentario descriptivo de qué hace el método
public void nombreDelMétodo (tipo parámetro1, tipo parámetro2...) {

    Código del método
}
```

En general un método con tipo de retorno void no llevará sentencia *return*, aunque en Java se permite que un método de este tipo incluya la sentencia *return;*. Si ocurre esto, la sentencia da lugar a que el método finalice en ese punto sin ejecutar ninguna línea más de código. Solo tiene sentido su uso asociado a que se cumplan ciertas condiciones.

**¿Puede un método ser al mismo tiempo función y procedimiento?** Formalmente debemos evitarlo. Podríamos tratar de agrupar operaciones, pero es una situación que trataremos de evitar. Nos plantearemos como objetivo que todo método tipo función se centre en devolvernos aquello que nos interesa sin realizar otros procesos. Igualmente buscaremos que todo procedimiento realice un proceso concreto y no varios. Cada método debe realizar una tarea concreta, específica y bien definida. Un método no debe ocuparse de dos tareas.

## EJERCICIO

Considera estás desarrollando un programa Java donde necesitas trabajar con objetos de tipo DiscoMusical. Define una clase DiscoMusical análoga a la que hemos visto para taxis, pero en este caso considerando los siguientes atributos de clase: titulo (String), autor (String), añoEdicion (int), formato (String), digital (boolean). Define un constructor y los métodos para poder establecer y obtener los valores de los atributos. Compila el código para comprobar que no presenta errores. Crea un objeto y comprueba sus métodos como hemos hecho con los objetos Taxi. Para comprobar la corrección de tu solución puedes consultar en los foros [aprenderaprogramar.com](http://www.aprenderaprogramar.com).

**Próxima entrega:** CU00625B

**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](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188)