



# Curso de Java Standard



Ing. Octavio Robleto



octavio.robleto@gmail.com



<https://octaviorobleto.com>

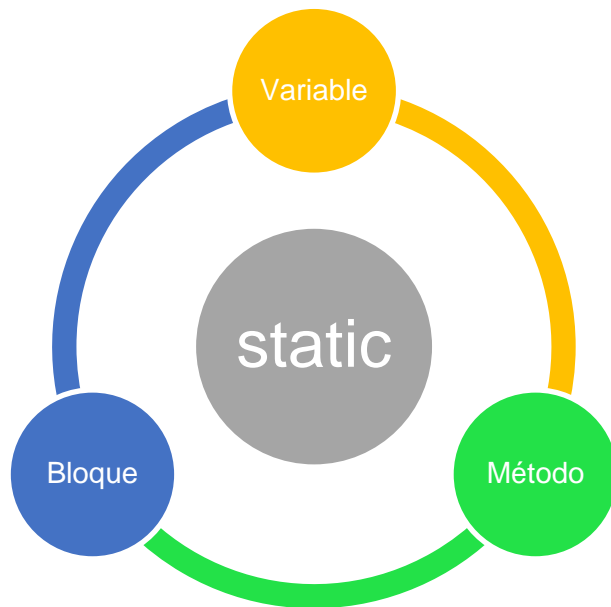


# Miembros de Clase

La palabra reservada **static** se usa principalmente para el manejo eficiente de la memoria de nuestro programa.

Cuando una clase, método o variable es declarada como estática se puede acceder a ella sin la necesidad de tener que instanciar un objeto de la clase.

Los casos mas conocidos son el método **main** y la salida por consola **System.out**.



# Variables y constantes estáticas

Conocidas también como variables y/o constantes de clase se pueden acceder sin la necesidad de instanciar un objeto de la clase, se usa en la mayoría de los casos para datos en común, por ejemplo, el nombre de la empresa, la versión del programa, etc.

Obtiene memoria solo una vez en el área de clase en el momento de la carga de la clase.

Es una palabra reservada adicional y no limita que se le pueda asignar un modificador de acceso según sea nuestro requerimiento.

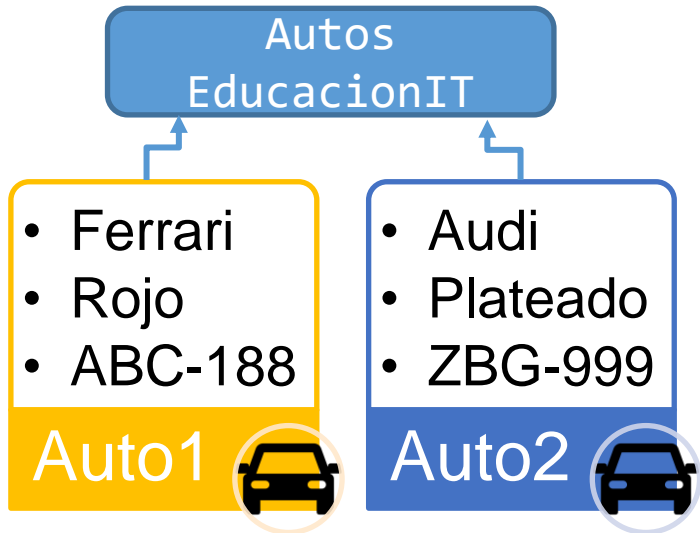
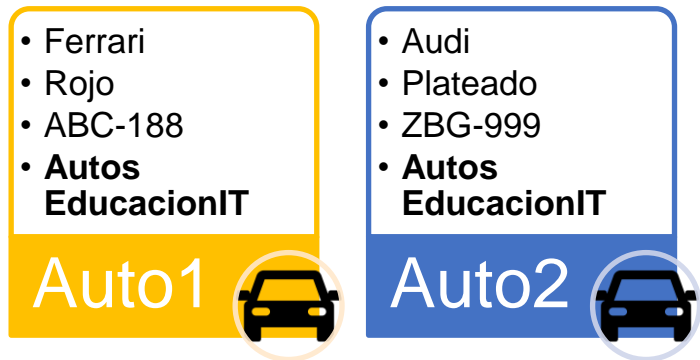
```
private String color;  
private String marca;  
private String patente;  
private boolean encendido;  
public static String concesionaria = "Autos EducacionIT";
```



# Mejora el manejo de memoria?

Para este ejemplo imaginemos que necesitamos indicar el mismo nombre de la concesionaria a todos los autos.

Con una constante estática asignamos un solo espacio de memoria para todos los objetos, en cambio cuando esa constante la hacemos miembro de instancia (único para cada objeto) reserva un espacio de memoria por cada uno de ellos (mas memoria)



# Métodos Estáticos

Conocidas también como métodos de clase, como las variables estáticas se puede acceder al método sin la necesidad de crear un objeto de la clase.

Un método estático puede acceder a un miembro de datos estáticos y puede cambiar su valor.

Hay dos cosas importantes a tener en cuenta:

1. El método estático no puede utilizar miembros de datos no estáticos ni llamar directamente a un método no estático.
2. **this** y **super** (que veremos mas adelante) no se pueden utilizar en un contexto estático.

```
public static void setConcesionaria(String concesionaria) {  
    Auto.concesionaria = concesionaria;  
}
```



# Bloque Estático

Es posible declarar bloques de código como estáticos, de tal manera que sean ejecutados cuando se cargue la clase.

Se utiliza para inicializar el miembro de datos estáticos.

Se ejecuta antes que el método principal en el momento de la carga de clases.

Si no se declara un bloque de este tipo de forma explícita, la JVM combina todos los campos estáticos en un bloque y los ejecuta durante la carga de clases.

```
static String version;  
  
static {  
    version = "V 1.0";  
    System.out.println("Asignacion de valores antes de iniciar");  
}
```