

# Java Standard Web Programming

Módulo 2



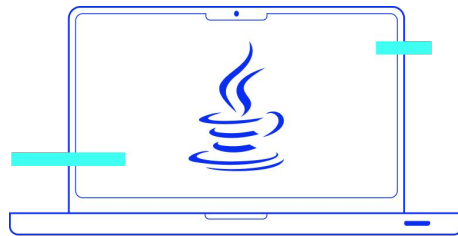
# Encapsulamiento

# Encapsulamiento

Es uno de los pilares de la **POO** y su filosofía es hacer que los atributos de las clases se puedan editar sólo a través de métodos.

Básicamente, el encapsulamiento se usa para dar **más seguridad a las aplicaciones**, para evitar que la información se filtre y para dar mucho más orden en las clases.

Es simplemente **la capacidad de ocultar los atributos de una clase**.



# Private

Para encapsular, se deben colocar los atributos de clase privados, para así ocultarlos.

```
private String color;  
private String marca;  
private String patente;  
private boolean encendido;
```

## Acceder o modificar esos atributos

No es posible acceder directamente a los datos configurados como **private**. Es necesario crear métodos para modificarlos y accederlos.

Estos métodos se crean en base a un estándar de identificación. Se denominan **Setters** y **Getters**, y siempre son **public**.



## Setter

Del inglés *Set*, que significa **establecer**. Sirve para **asignar un valor a un atributo**, pero de forma explícita, a través de un parámetro del mismo tipo. El **Setter** nunca retorna un valor (Siempre es `void`).



## Getter

Del inglés *Get*, que significa **obtener**. Sirve para **obtener (recuperar o acceder) el valor ya asignado a un atributo** y utilizarlo, estos métodos son de función y devuelven el tipo y el valor del atributo.

El **Getter** de los atributos boolean cambia la palabra **get** por **is** al ser un tipo de dato *verdad/falso* es más lógico preguntar “**esfalso**”, “**esverdadero**”.



**¡Sigamos  
trabajando!**

