

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 en tus 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 lograr esto debemos colocar los atributos de la clase privados para así ocultarlos.

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



Acceder o modificar esos atributos

Los datos al ser **private** no los podemos acceder directamente debemos crear unos métodos tanto como para modificarlos como para accederlos.

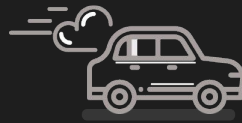
Estos métodos deben ser creados con un estándar de identificación y se denomina Setters y Getters y siempre deben ser **public**.

Setters: Del Inglés Set, que significa establecer, nos sirve para asignar un valor a un atributo , pero de forma explícita, a través de un parámetro del mismo tipo, además el Setter nunca retorna nada (Siempre es void).

Getters: Del Inglés Get, que significa obtener, nos 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.

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

```
public String getColor() {  
    return color;  
}  
  
public void setColor(String color) {  
    this.color = color;  
}  
  
public String getMarca() {  
    return marca;  
}  
  
public void setMarca(String marca) {  
    this.marca = marca;  
}  
  
public String getPatente() {  
    return patente;  
}  
  
public void setPatente(String patente) {  
    this.patente = patente;  
}  
  
public boolean isEncendido() {  
    return encendido;  
}  
  
public void setEncendido(boolean encendido) {  
    this.encendido = encendido;  
}
```



Acceder o modificar esos atributos

Damos valores a los atributos a través de los Seters

```
// creamos o instanciamos los objetos
Auto auto1 = new Auto();

//le damos valores a los atributos del auto 1
auto1.setColor("Rojo");
auto1.setMarca("Ferrari");
auto1.setPatente("ABC-188");

//Encendemos el Auto1
auto1.setEncendido(true);

//le damos valores a los atributos del auto 2
Auto auto2 = new Auto("Plateado", "Audi", "ZBG-999", true);

// Apagamos el Auto2
auto2.setEncendido(true);

// mostramos en consola el mensaje
System.out.println("La patente del Auto1 es :" + auto1.getPatente());
System.out.println("La patente del Auto2 es :" + auto2.getPatente());
```



Obtenemos los valores a través de los Getters