



Curso de Java Standard



Ing. Octavio Robleto



octavio.robleto@gmail.com



<https://octaviorobleto.com>

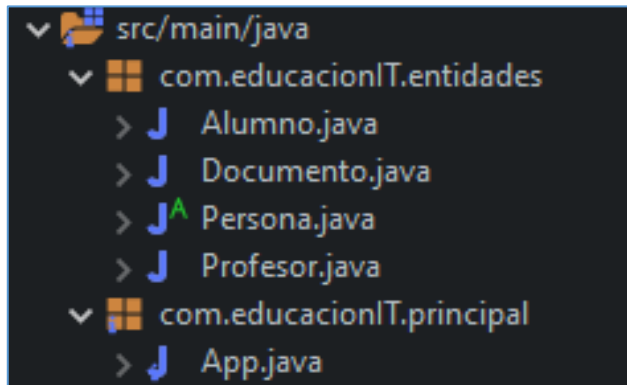


Paquetes

Los paquetes no son mas que carpetas dentro de nuestro proyecto Java que nos ayudan a organizar y contener un conjunto de clases relacionadas bien por finalidad, por ámbito o por herencia.

Los paquetes resuelven además, el problema del conflicto entre los nombres de las clases. Al crecer el número de clases hay probabilidad de designar con el mismo nombre a dos clases diferentes.

Las clases, atributos y métodos tienen ciertos privilegios de acceso y los paquetes nos ayudan con esto.



Modificadores de Acceso

Los modificadores de acceso permiten dar un nivel de seguridad mayor a nuestras aplicaciones restringiendo el acceso a diferentes clases, atributos, métodos, constructores asegurándonos que el usuario deba seguir una "ruta" especificada por nosotros para acceder a la información.

Es muy posible que nuestras aplicaciones vayan a ser usadas por otros programadores o usuarios con cierto nivel de experiencia; haciendo uso de los modificadores de acceso podremos asegurarnos de que un valor no será modificado incorrectamente por parte de otro programador o usuario,

Se colocan de primero al crear una clase, atributo o método.



default

Java nos da la opción de no usar un modificador de acceso y al no hacerlo, el elemento tendrá un acceso conocido como **default** o acceso por defecto que permite que tanto la propia clase como las clases que se encuentren en el mismo paquete accedan a dichos componentes (de aquí la importancia de declararle siempre un paquete a nuestras clases)

Las clases, atributos y métodos incluyendo los constructores pueden tener este tipo de acceso.

```
class Auto {
    String color;
    String marca;
    String patente;
    boolean encendido;

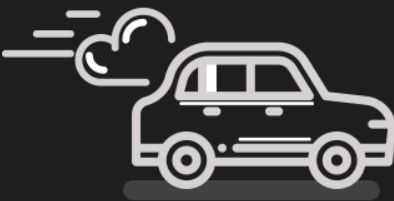
    // Constructor
    Auto(){

    }

    // devuelve una cadena de caracteres con las características que posea el objeto
    String mostrarDatos() {
        String mensaje = "El Auto es de color " + color + ", marca " + marca
            + ", patente " + patente + " y se encuentra "
            + ((encendido) ? "encendido" : "apagado");

        // la palabra reservada return le indica al metodo que finalizo su ejecucion
        // y que devuelva el objeto mensaje
        return mensaje;
    }

    //cambia el estado del atributo encendido sin devolver ningun dato
    void cambiarEstado(boolean encendido){
        this.encendido = encendido;
    }
}
```

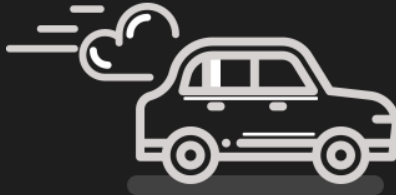


private

Es el más restrictivo de todos, básicamente cualquier elemento de una clase que sea privado puede ser accedido únicamente por la misma clase por nada más.

Este modificador de acceso no puede ser asignado a las clases, ya que no tiene sentido que crees una para que nadie puede acceder a ella.

```
class Auto {  
    private String color;  
    private String marca;  
    private String patente;  
    private boolean encendido;  
  
    // Constructor  
    private Auto(){  
  
    }  
  
    // devuelve una cadena de caracteres con las características que posea el objeto  
    private String mostrarDatos() {  
        String mensaje = "El Auto es de color " + color + ", marca " + marca  
            + ", patente " + patente + " y se encuentra "  
            + ((encendido) ? "encendido" : "apagado");  
  
        // la palabra reservada return le indica al metodo que finalizo su ejecucion  
        // y que devuelva el objeto mensaje  
        return mensaje;  
    }  
  
    //cambia el estado del atributo encendido sin devolver ningun dato  
    private void cambiarEstado(boolean encendido){  
        this.encendido = encendido;  
    }  
}
```

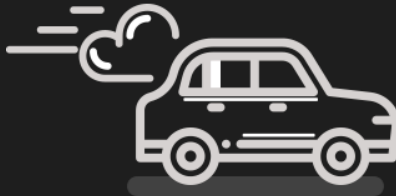


protected

Nos permite acceso a los componentes con dicho modificador desde la misma clase, clases del mismo paquete y clases que hereden de ella (incluso en diferentes paquetes)

Como nos pasa con el modificador de acceso **private**, una clase no puede estar **protected**.

```
class Auto {  
    protected String color;  
    protected String marca;  
    protected String patente;  
    protected boolean encendido;  
  
    // Constructor  
    protected Auto(){  
  
    }  
  
    // devuelve una cadena de caracteres con las características que posea el objeto  
    protected String mostrarDatos() {  
        String mensaje = "El Auto es de color " + color + ", marca " + marca  
            + ", patente " + patente + " y se encuentra "  
            + ((encendido) ? "encendido" : "apagado");  
  
        // la palabra reservada return le indica al metodo que finalizo su ejecucion  
        // y que devuelva el objeto mensaje  
        return mensaje;  
    }  
  
    //cambia el estado del atributo encendido sin devolver ningun dato  
    protected void cambiarEstado(boolean encendido){  
        this.encendido = encendido;  
    }  
}
```

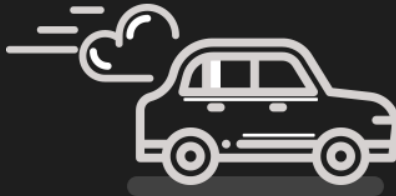


public

Es el más permisivo de todos, básicamente **public** es lo contrario a **private** en todos los aspectos (lógicamente), esto quiere decir que si un componente de una clase es public, tendremos acceso a él desde cualquier clase o instancia sin importar el paquete o procedencia de ésta.

Las clases, atributos y métodos incluyendo los constructores pueden tener este tipo de acceso.

```
public class Auto {  
    public String color;  
    public String marca;  
    public String patente;  
    public boolean encendido;  
  
    // Constructor  
    public Auto(){  
  
    }  
  
    // devuelve una cadena de caracteres con las características que posea el objeto  
    public String mostrarDatos() {  
        String mensaje = "El Auto es de color " + color + ", marca " + marca  
            + ", patente " + patente + " y se encuentra "  
            + ((encendido) ? "encendido" : "apagado");  
  
        // la palabra reservada return le indica al metodo que finalizo su ejecucion  
        // y que devuelva el objeto mensaje  
        return mensaje;  
    }  
  
    //cambia el estado del atributo encendido sin devolver ningun dato  
    public void cambiarEstado(boolean encendido){  
        this.encendido = encendido;  
    }  
}
```



Cuadro comparativo de Acceso

Modificador	La misma clase	Mismo paquete	Subclase	Otro paquete
private	✓	x	x	x
default	✓	✓	x	x
protected	✓	✓	✓	x
public	✓	✓	✓	✓