

## 5.3. Empaquetado de componentes



# Índice

---

Objetivos .....	3
Creación de librerías personalizadas .....	4
Crear una librería Java .....	4
Usar una librería Java .....	7
Despedida .....	9
Resumen.....	9

# Objetivos

El contenido de esta lección ya lo estudiaste en la asignatura de Programación, pero te recomendamos que vuelvas a estudiarlo como repaso, ya que te ayudará a continuar con éxito el resto de contenidos y a realizar las actividades propuestas sin dificultad.

**Adelante, te costará poco esfuerzo recordar estos conceptos y volverlos a poner en la práctica.**

Con esta lección perseguimos los siguientes objetivos:

- Construir librerías de clases Java empaquetadas en archivos .jar.
- Utilizar las librerías de clases en múltiples proyectos.

# Creación de librerías personalizadas

## Crear una librería Java

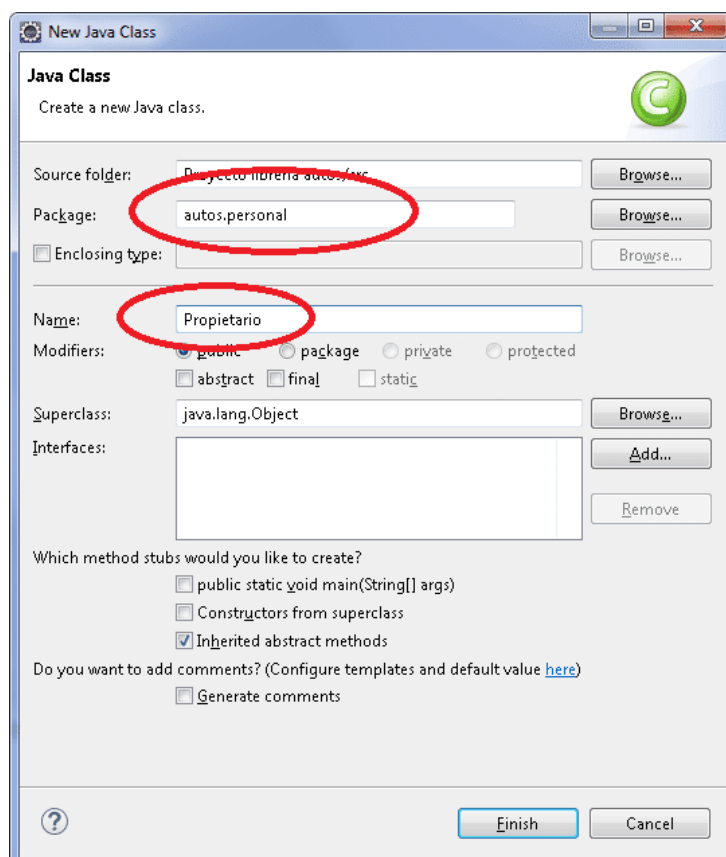
En esta lección recordarás cómo crear una librería de clases Java empaquetada en un archivo .jar para utilizarla después en múltiples proyectos.

En esta ocasión implementaremos las clases *Propietario* y *Coche*, y las empaquetaremos en una librería de clases situada en un archivo llamado *autos.jar*. Recuerda que las clases deberán estar organizadas en paquetes por una cuestión de orden.

Sigue estos pasos para crear la librería:

1. Crea un proyecto Java estándar llamado "Proyecto librería autos" (**File / New / Java Project**).
2. Ahora crea la clase *Propietario* dentro del paquete *autos.personal*. Para ello, haz  **clic derecho sobre el nombre del proyecto** y selecciona en el menú contextual **New / Class**.

En el cuadro de diálogo "New Java Class" puedes especificar el nombre del paquete y el nombre de la clase, como ves en la imagen:



### 5.3. Empaquetado de componentes

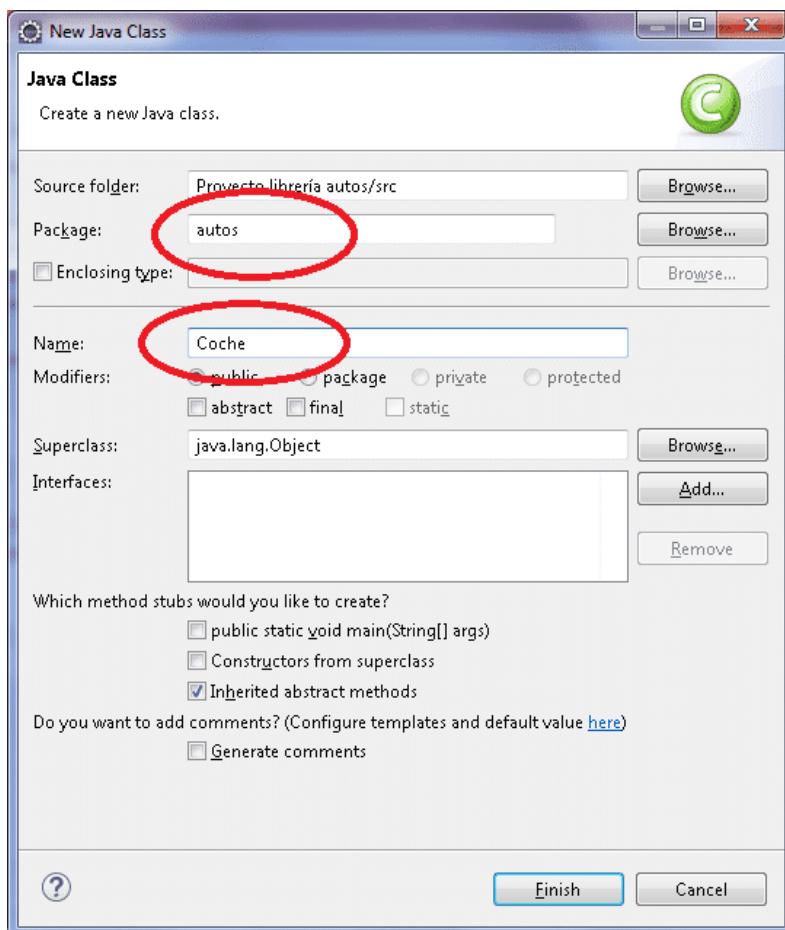
De manera automática se ha creado este código:

```
package autos.personal;  
  
public class Propietario {  
  
}
```

**3.** Completa el código de la clase *Propietario* dejándolo así:

```
package autos.personal;  
  
public class Propietario {  
    private String dni;  
    private String nombre;  
    private String apellidos;  
  
    public Propietario  
        (String dni, String nombre, String apellidos) {  
        this.dni = dni;  
        this.nombre = nombre;  
        this.apellidos = apellidos;  
    }  
  
    @Override  
    public String toString() {  
        return this.nombre + " " + this.apellidos +  
            " con DNI " + this.dni;  
    }  
}
```

**4.** Ahora crea la clase *Coche* dentro del paquete *autos*. Recuerda que debes hacer clic derecho sobre el nombre del proyecto y seleccionar en el menú contextual **New / Class**. Completa el cuadro de diálogo "New Java Class" de la siguiente manera:



De forma automática has conseguido este código:

```
package autos;

public class Coche {

}
```

**5.** Ahora completa el código de la clase *Coche* para dejarlo así:

```
package autos;

import autos.personal.Propietario;

public class Coche {
    private Propietario propietario;
    private String matricula;
    private String marca;
    private String modelo;
    private int velocidad;

    public Coche(String matricula, String marca,
        String modelo, Propietario propietario) {
        this.matricula = matricula;
        this.marca = marca;
        this.modelo = modelo;
    }
}
```

### 5.3. Empaquetado de componentes

```

        this.propietario = propietario;
        this.velocidad = 0;
    }

    public void acelerar(int cuanto) {
        this.velocidad = this.velocidad + cuanto;
    }

    public void frenar(int cuanto) {
        this.velocidad = this.velocidad - cuanto;
    }

    @Override
    public String toString() {
        return "El " + this.marca + " " + this.modelo +
            " con matricula " + this.matricula +
            " propiedad de " +
            this.propietario.toString() +
            " va a " + this.velocidad + " km/hora";
    }
}

```

**6.** Has llegado al punto clave, **crear el archivo *.jar*** que empaquetará el modelo de clases. El siguiente vídeo te muestra los pasos que debes seguir desde Eclipse para crear el archivo *.jar* en la ubicación deseada. <https://vimeo.com/telefonicaed/review/258245213/21ad3a7de1>

## Usar una librería Java

Una vez que tengas creada tu librería de clases *autos.jar*, podrás utilizarla en múltiples proyectos.

Sigue estos pasos para ponerlo en práctica:

- 1.** Crea un nuevo proyecto estándar Java llamado “Uso librería autos” (***File / New / Java Project***).
- 2.** Entra en las propiedades del proyecto y **agrega la librería *autos.jar***. El siguiente vídeo te muestra los pasos necesarios. <https://vimeo.com/telefonicaed/review/258245220/77e1e91fb6>
- 3.** Ahora que has incluido la librería en tu proyecto, puedes actuar igual que si las clases *Propietario* y *Coche* estuvieran creadas directamente en este proyecto. Crea la clase *Principal* con el siguiente código:

```

import autos.Coche;
import autos.personal.Propietario;

public class Principal {
    public static void main(String[] args) {
        Propietario yo = new
            Propietario("666666666K", "Perico", "De los Palotes");
    }
}

```



### 5.3. Empaquetado de componentes

```
        Coche miCoche = new Coche("5577FJK", "Suzuki", "Ignis", yo);  
        miCoche.acelerar(100);  
        miCoche.frenar(10);  
        System.out.println(miCoche.toString());  
    }  
}
```

Recuerda que se puede utilizar el carácter de asterisco como comodín para importar todas las clases situadas en un paquete.

```
import autos.*;  
import autos.personal.*;
```

Así importamos todas las clases incluidas en *autos* y todas las clases incluidas en *personal*. Como en nuestro ejemplo sólo tenemos una clase en cada paquete, no hay diferencia entre este sistema y el usado en el ejemplo.



# Despedida

## Resumen

Has terminado la lección, vamos a ver los puntos más importantes que hemos tratado.

- Una **librería** contiene un conjunto de clases que pueden servir a múltiples proyectos.
- **Las librerías de clases Java están organizadas en paquetes**, siendo un paquete una carpeta en el sistema de archivos.
- **Las librerías están empaquetadas en archivos .jar** que cada proyecto que la requiera deberá importar.