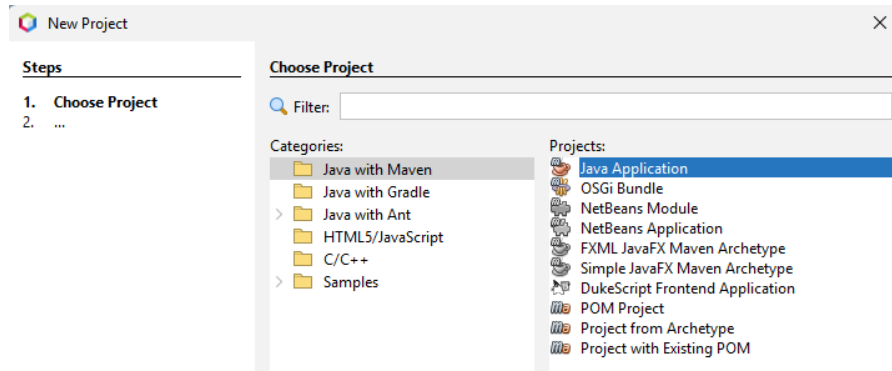


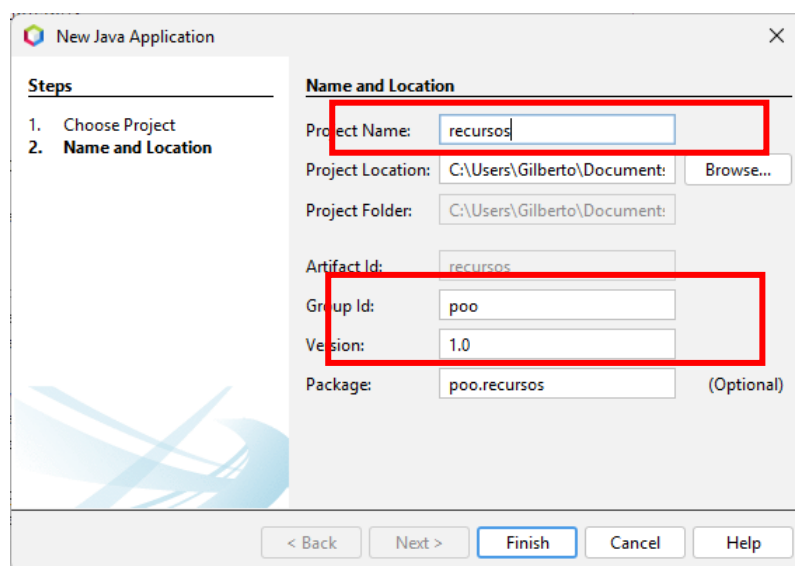
EJERCICIO EXTRA: 3. Creando biblioteca de recursos propia

Objetivo. Aprender a crear nuestra propia biblioteca de recursos que podamos reutilizar en diferentes proyectos.

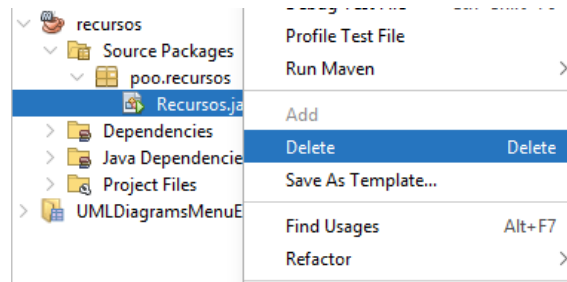
1. Abre Netbeans y elige un proyecto Java Application with Maven y damos clic en el botón **Next**.



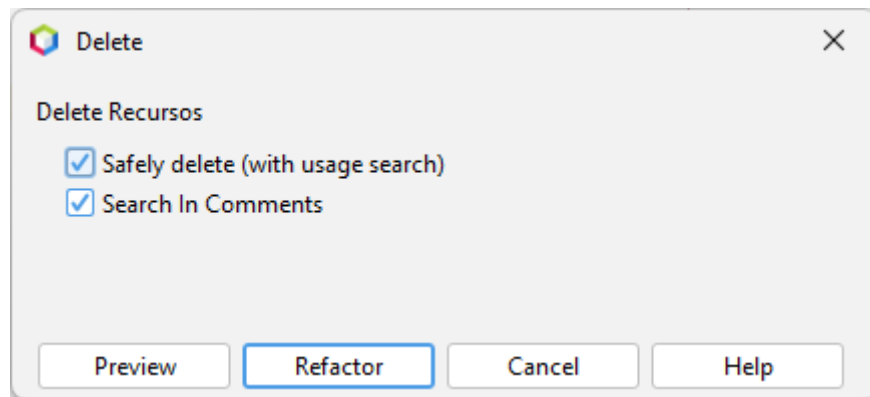
2. El nombre del proyecto será **MensajImagen**, Group Id: **poo** y versión: **1.0**, tal como se muestra en la siguiente imagen y damos clic en el botón **Finish**.



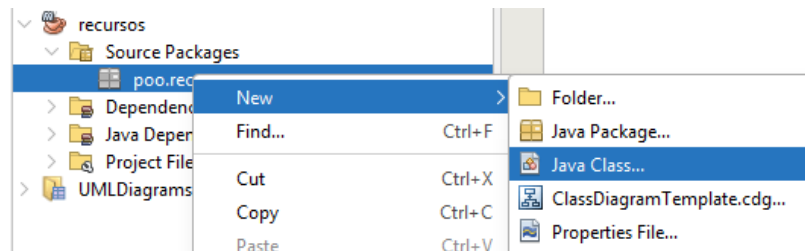
3. Eliminamos la clase que contiene al método main, esto debido a que una biblioteca de recursos siempre se utiliza de forma externa, es decir, es usada por otra aplicación (que contiene un método main). Damos clic con el botón derecho del mouse sobre **Recursos** y elegimos la opción **Delete**.



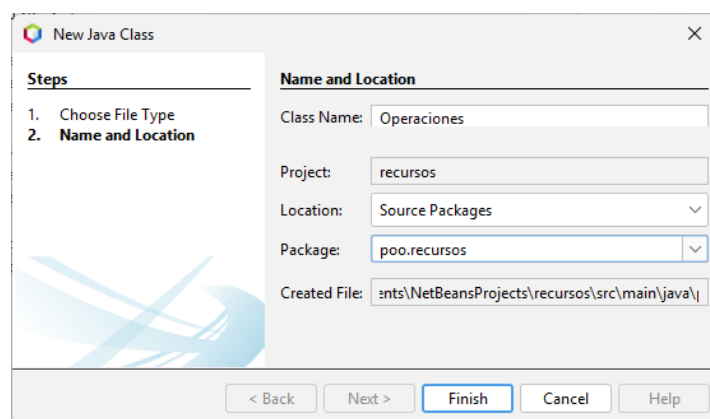
4. En la ventana que nos aparezca nos aseguramos de marcar las dos casillas que se muestran a continuación y damos clic en el botón **Refactor**.



5. Agregamos una nueva clase a nuestro paquete.



6. Le ponemos como nombre a nuestra clase: **Operaciones** y damos clic en el botón **Finish**.



7. Eliminamos los comentarios de nuestra clase y nos aseguramos de agregar el siguiente **import** debajo del nombre de nuestro paquete.

```
package poo.recursos;  
import javax.swing.JOptionPane;
```

8. Agregamos los siguientes métodos a nuestra clase.

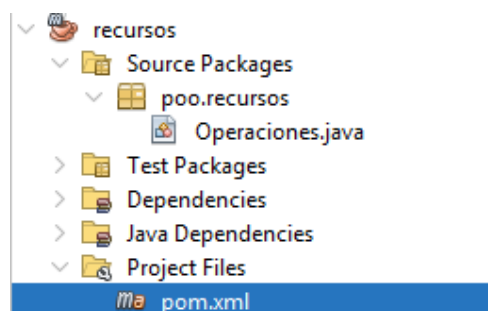
```
public class Operaciones {  
    public static void sumar(int num1, int num2) {  
        int res = num1 + num2;  
        String mensaje = num1+" + "+num2+" = "+res;  
        JOptionPane.showMessageDialog(null,mensaje );  
    }  
  
    public static void restar(int num1, int num2) {  
        int res = num1 - num2;  
        String mensaje = num1+" - "+num2+" = "+res;  
        JOptionPane.showMessageDialog(null,mensaje );  
    }  
}
```

9. Damos clic en el botón **clean and build project (recursos)** 

Si todo salió bien se ha empaquetado nuestra clase en un jar y se agregó al repositorio local de maven, por lo que debemos ver una ventana como la siguiente.

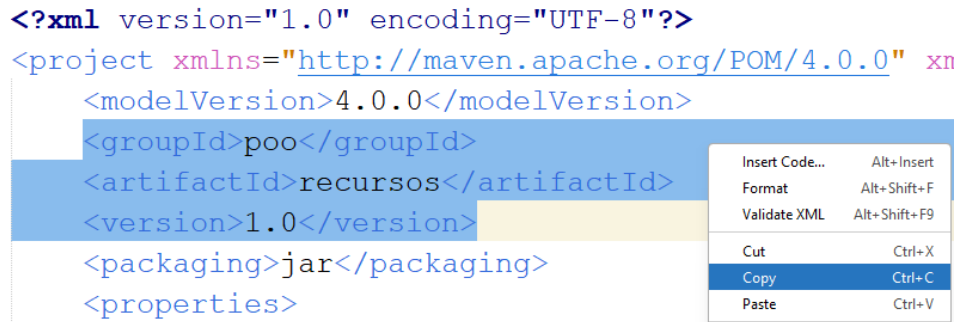
```
-----  
BUILD SUCCESS  
-----  
Total time: 1.268 s  
Finished at: 2025-04-17T07:06:45-06:00  
-----
```

10. Damos doble clic en el archivo pom.xml de nuestro proyecto.



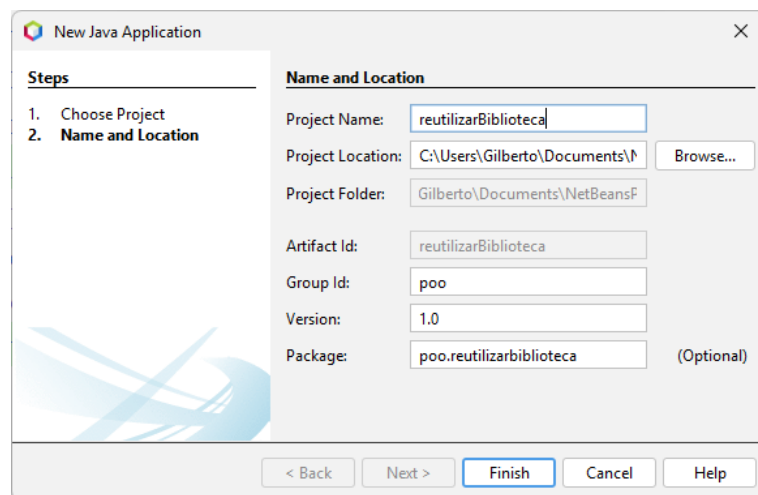
11. Seleccionamos y copiamos las líneas sombreadas.

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>poo</groupId>
    <artifactId>recursos</artifactId>
    <version>1.0</version>
    <packaging>jar</packaging>
    <properties>
```

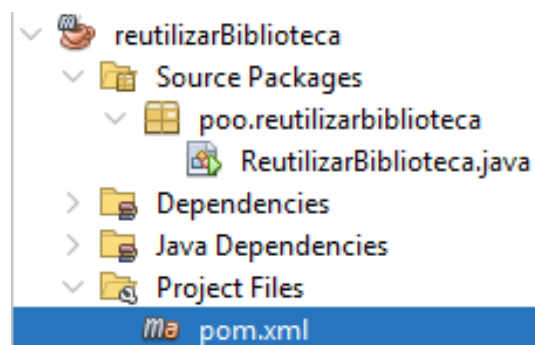


REUTILIZANDO NUESTRA BIBLIOTECA DE RECURSOS

12. Creamos un nuevo proyecto con el nombre **reutilizarBiblioteca** y damos clic en el botón **Finish**.



13. Damos doble clic sobre el archivo pom.xml de nuestro proyecto.



14. Debajo de `</properties>` agregamos las siguientes etiquetas.

```
</properties>
<dependencies>
  <dependency>
  </dependency>
</dependencies>
</project>
```

15. Debajo de la etiqueta `<dependency>` pegamos las etiquetas copiadas de nuestro proyecto recursos. El código debe tener la siguiente apariencia.

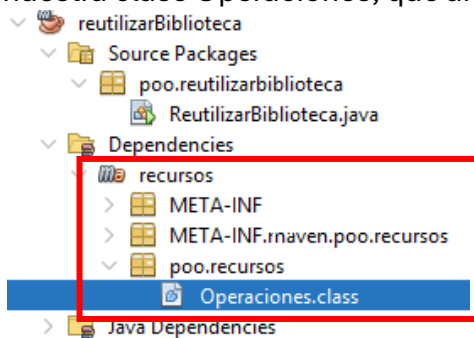
```
</properties>
<dependencies>
  <dependency>
    <groupId>poo</groupId>
    <artifactId>recursos</artifactId>
    <version>1.0</version>
  </dependency>
</dependencies>
</project>
```

Lo que acabamos de hacer es indicarle a nuestro proyecto actual que vamos a utilizar una dependencia (recurso) con los valores especificados.

16. Damos clic en el botón **clean and build project (recursos)**



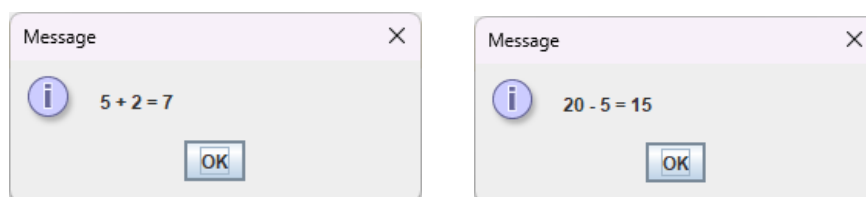
NOTA. Ahora podemos observar que dentro de la carpeta Dependencies dentro del cual encontraremos nuestra clase Operaciones, que ahora podemos reutilizar.



17. Agregamos al método main de nuestra aplicación las siguientes líneas de código.


```
public static void main(String[] args) {
    Operaciones.sumar(5, 2);
    Operaciones.restar(20, 5);
}
```

18. Corremos nuestra aplicación y podremos observar los siguientes resultados.



Actividad para realizar:

Agrega los métodos de multiplicar y dividir a la clase **Operaciones** del proyecto recursos y mándalos a llamar desde la clase **ReutilizarBiblioteca**.

NOTA: una vez creados los nuevos métodos (multiplicar y dividir) asegúrate de dar clic en el botón **clean and build project**  para actualizar los cambios en el repositorio local de Maven.