

Acceso a datos

Tarea Unidad 6

Contenido

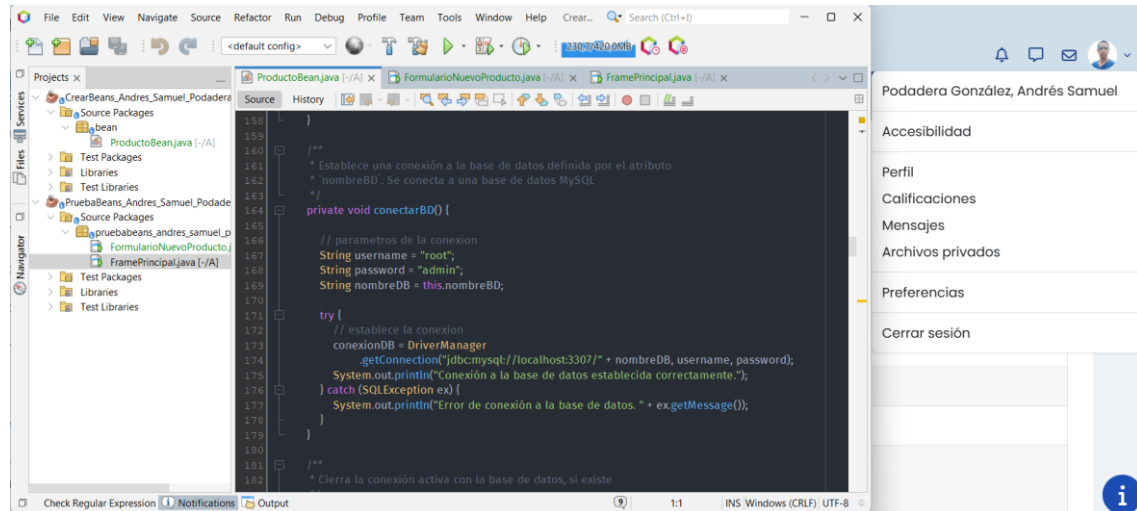
Estructura y partes del proyecto	2
EJERCICIO 1.	6
EJERCICIO 2.	9
Notas Aclaratorias	15

Estructura y partes del proyecto

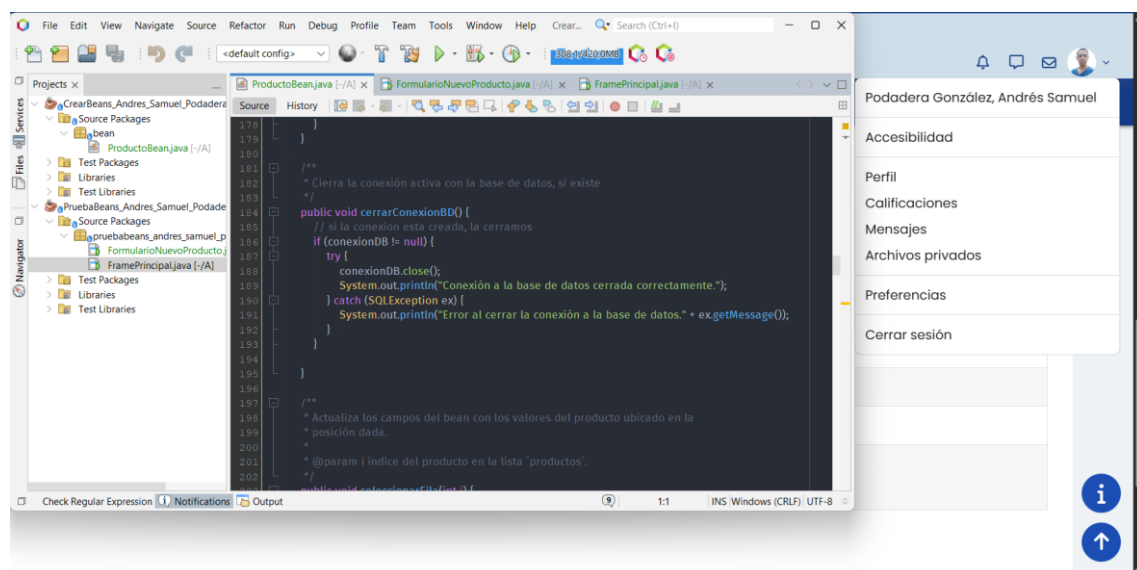
La carpeta de entrega contiene dos proyectos de Netbeans.

Dentro de **CrearBeans_Andres_Samuel_Podadera_Gonzalez** tenemos las clases pertenecientes al bean **ProductoBean.java** que contiene las siguientes partes:

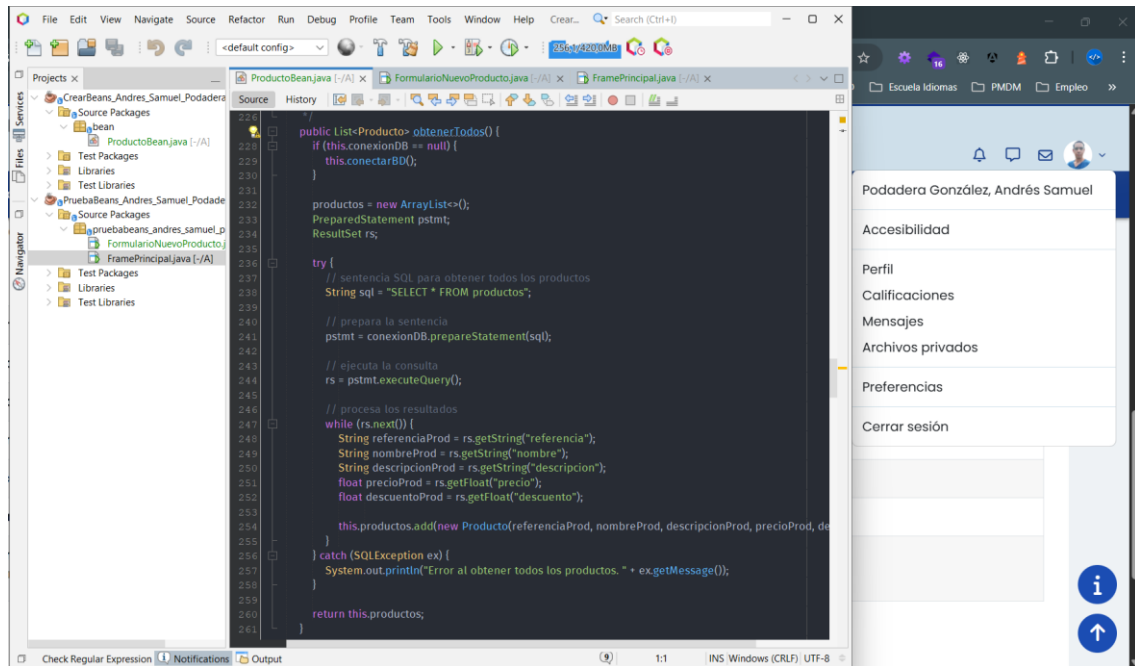
- **Método para conectar a la base de datos**



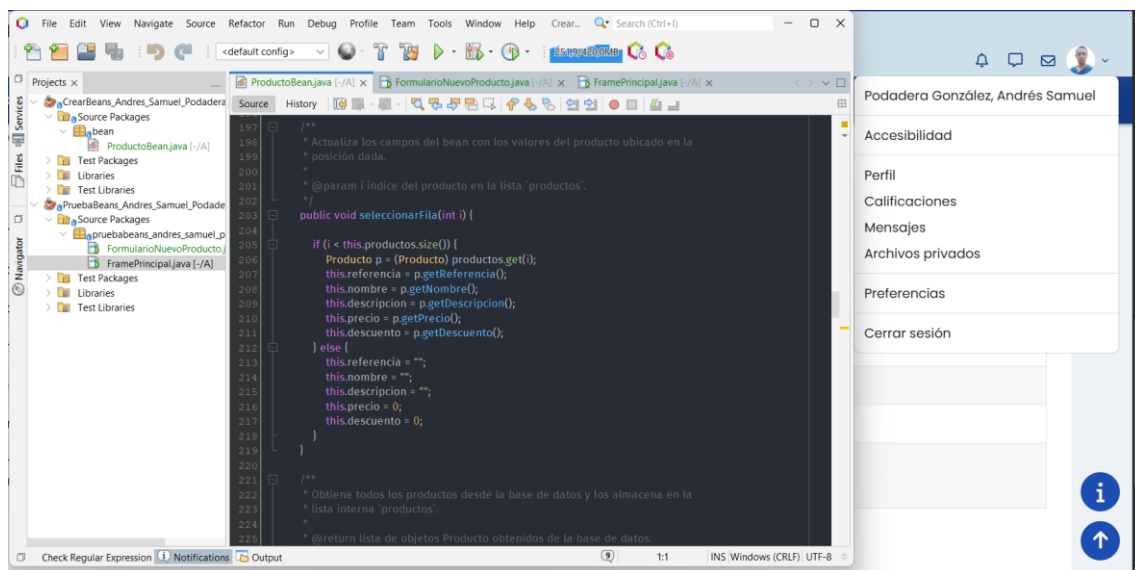
- **Método para desconectar de la base de datos**



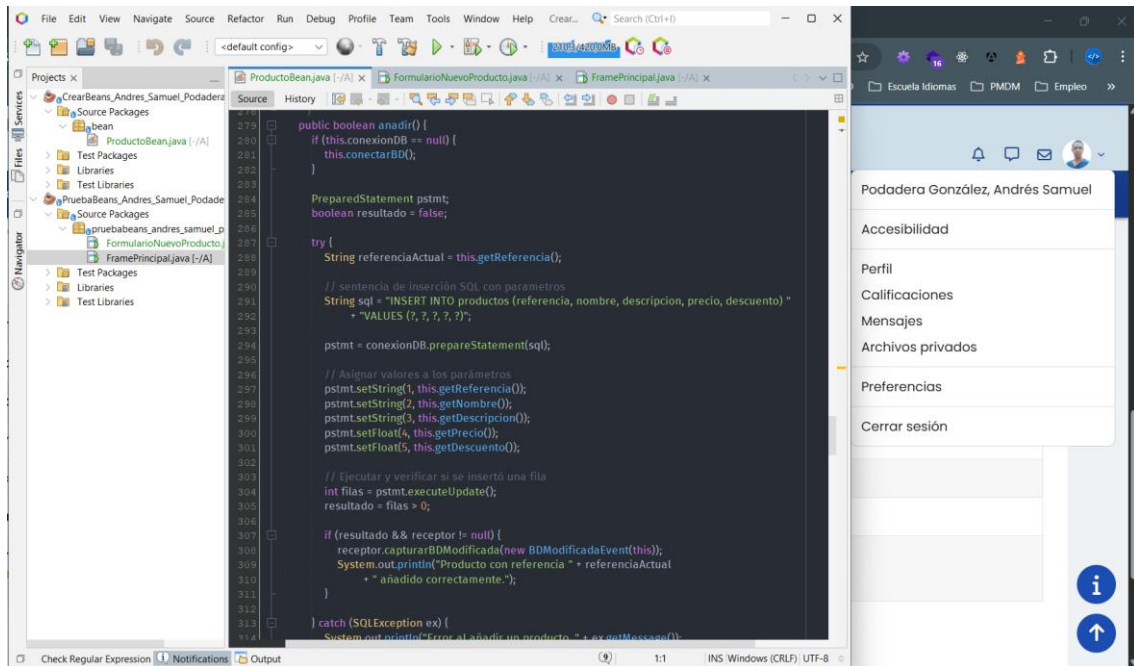
- **Método para listar los productos de la base de datos**



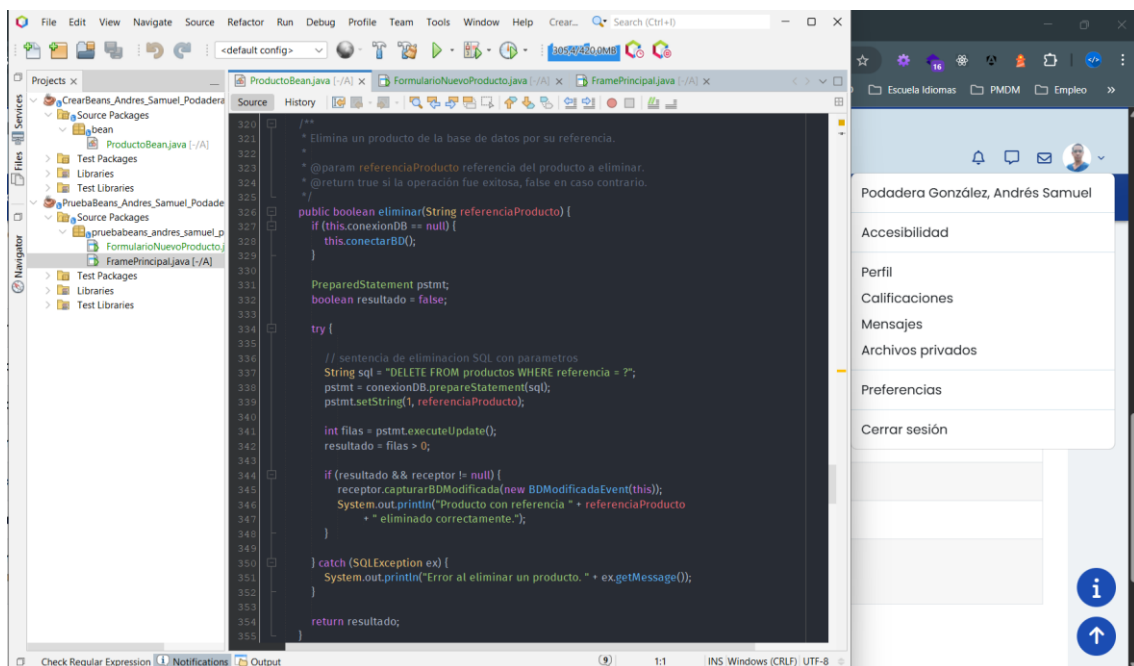
- **Método para obtener la información de cada producto de la lista**



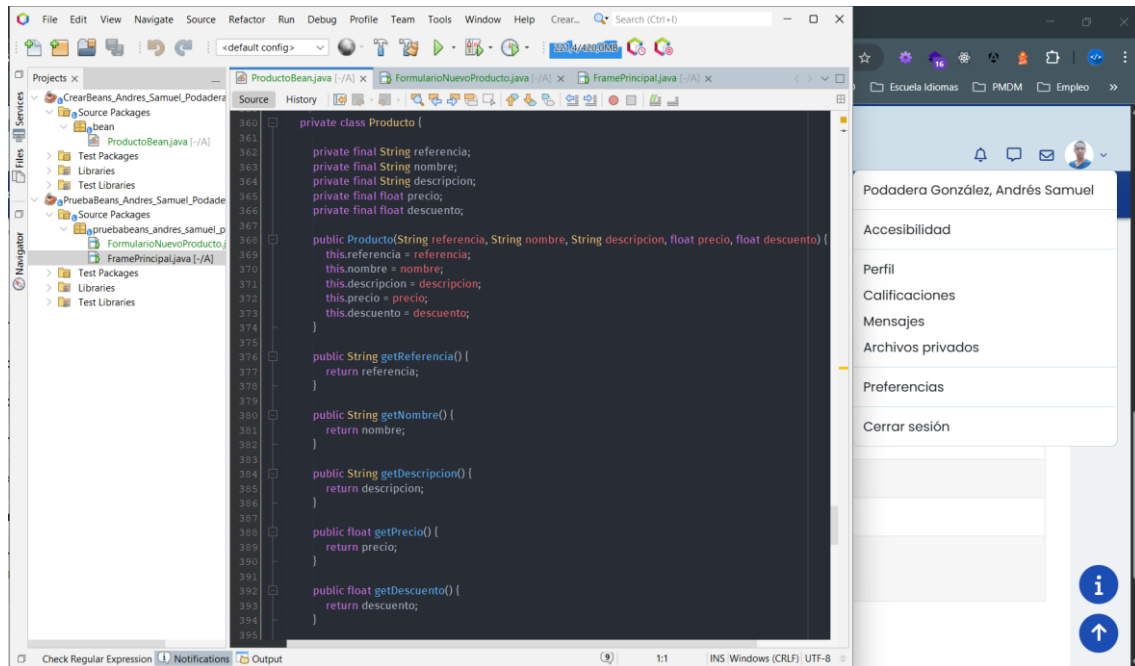
- **Método para añadir un nuevo producto en la base de datos**



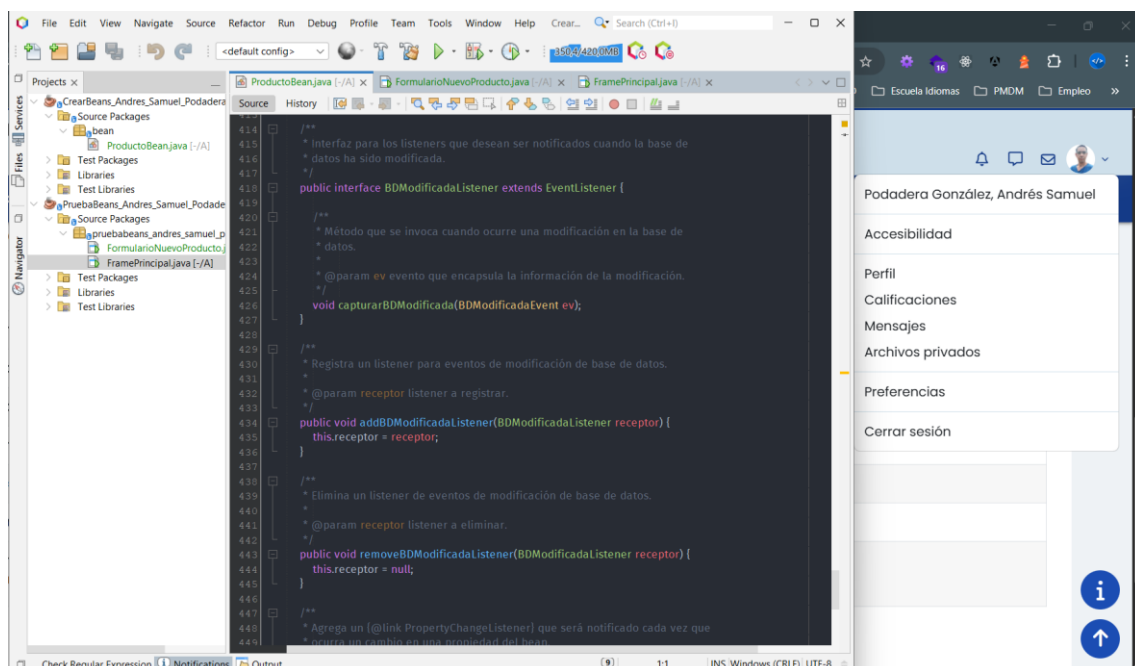
- **Método para eliminar un producto de la base de datos**



- **Clase Producto**



- **Interfaz BDModificada** que permite generar, capturar y gestionar eventos a todas las clases que la implementen

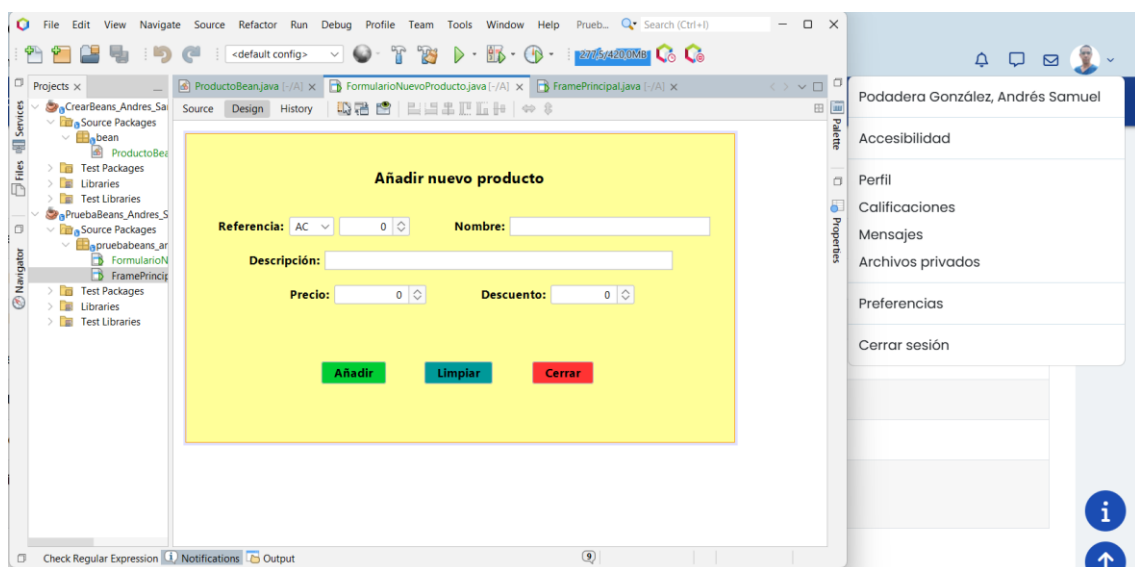


Y dentro de **PruebaBeans_Andres_Samuel_Podadera_Gonzalez** tenemos dos frames:

- El **FramePrincipal** contiene un título, una tabla donde se mostrarán los productos y dos botones, “Añadir Producto” nos permitirá desplegar un `JDialog` para introducir la información del nuevo producto y “Cerrar Aplicación” nos permitirá cerrar la aplicación



- **FormularioNuevoProducto** que contendrá varios textfield para introducir la información del nuevo producto y 3 botones, “Añadir” que se utiliza para añadir el producto, “Limpiar” que se utiliza para limpiar todos los campos del formulario y “Cerrar” que se utiliza para cerrar el formulario.



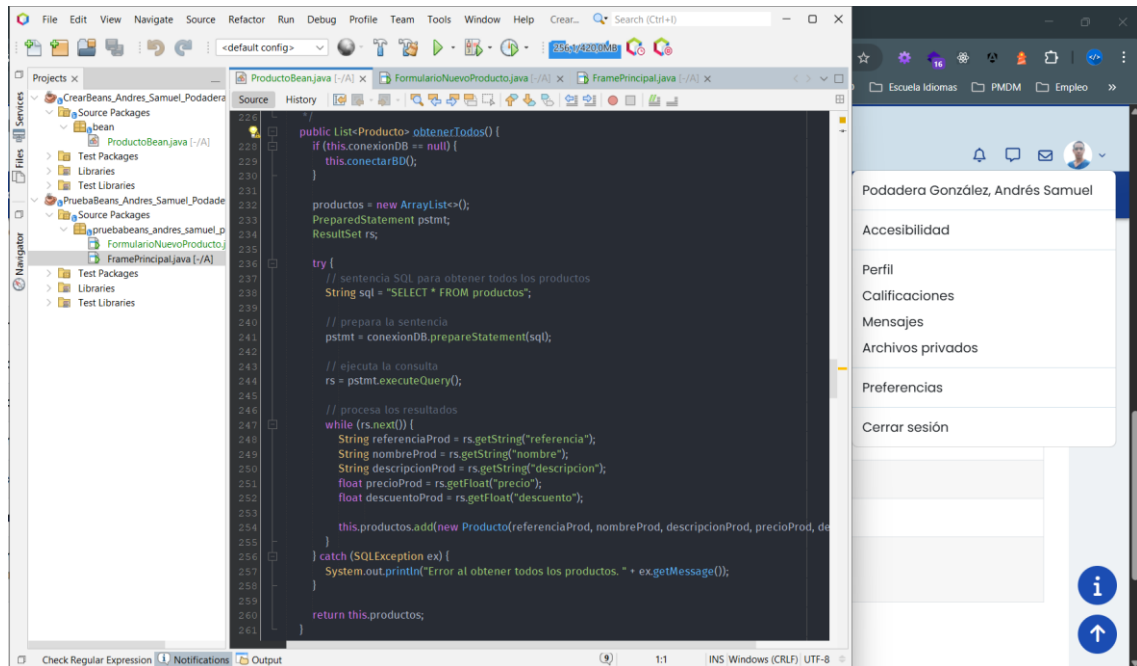
EJERCICIO 1.

Crear un proyecto java llamado CrearBeans_Nombre_Apellidos para crear un componente llamado ProductoBean. El componente actuará como componente de acceso a datos para manejar la base de datos facilitada y deberá permitir mostrar el listado de los productos, añadir nuevos productos y eliminar productos.

Vista la estructura de nuestro ProductoBean en el apartado anterior, aquí explicaré el código de los métodos más importantes que se usan en este apartado:

- **Mostrar el listado de productos:**

El método **obtenerTodos()** presenta la siguiente estructura:



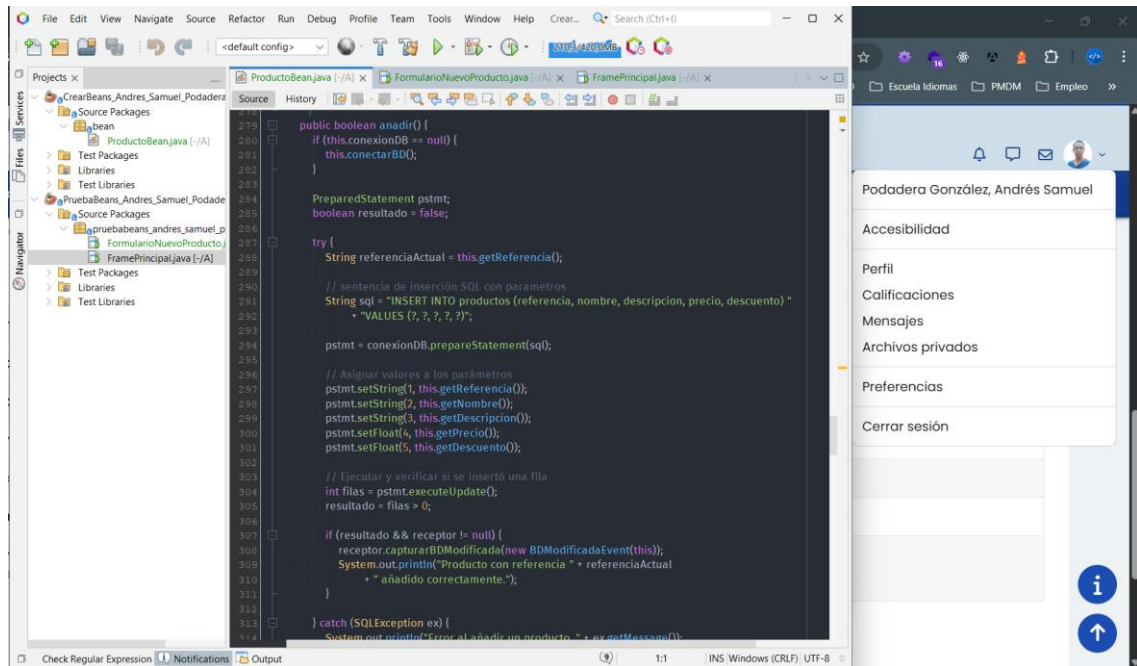
En la que se puede ver como comprueba primero si el programa ya esta conectado a la base de datos y caso de que no, se conecta.

Seguidamente declara las variables necesarias y realiza una consulta tipo Select sobre la tabla de productos.

Mediante un bucle while va obteniendo los datos de los productos y almacenándolos en una lista para, en último momento, devolver el listado de productos.

- **Añadir un producto:**

En este caso el método **anadir()** presenta el siguiente código:



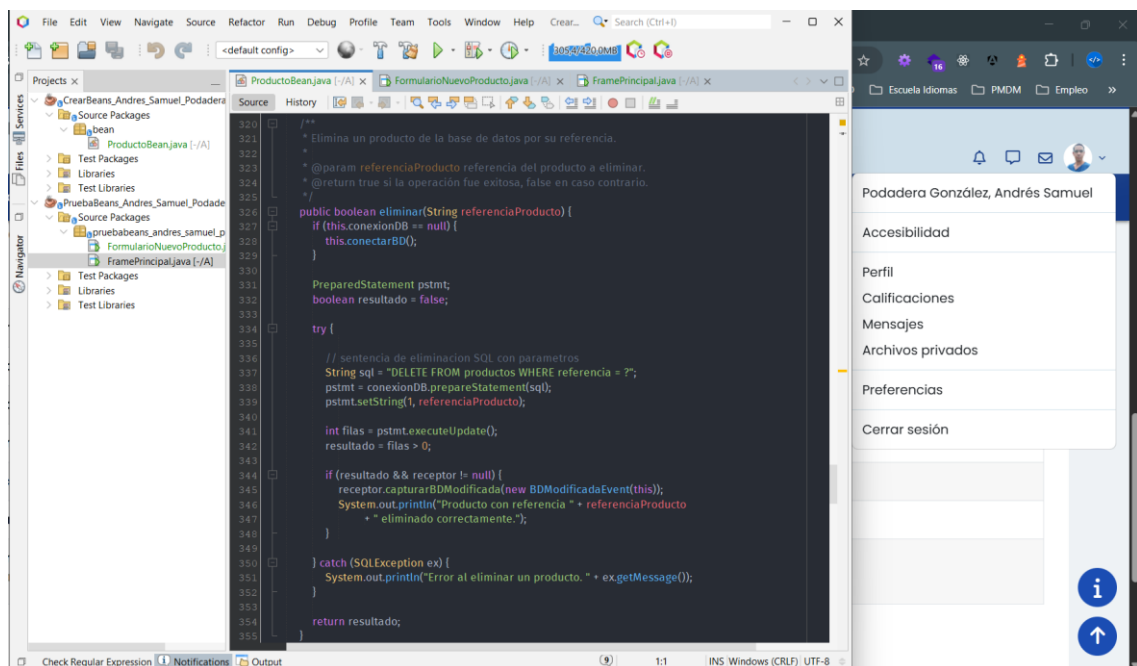
En el que primeramente comprobamos si el programa está conectado a la base de datos y si no se conecta.

Seguidamente declara las variables necesarias y ejecuta una sentencia tipo Insert parametrizada con la información del producto sobre la tabla productos.

Recoge el resultado de la sentencia ya sea que todo ha salido bien o que se han producido errores y devuelve el resultado.

- **Eliminar un producto:**

En este caso el método `eliminar()` presenta el siguiente código:



En el que se puede observar que en primer lugar se comprueba si se ha realizado una conexión a la base de datos y caso de que no, se conecta.

Seguidamente crea las variables necesarias y ejecuta una sentencia Delete con la referencia del producto sobre la tabla productos.

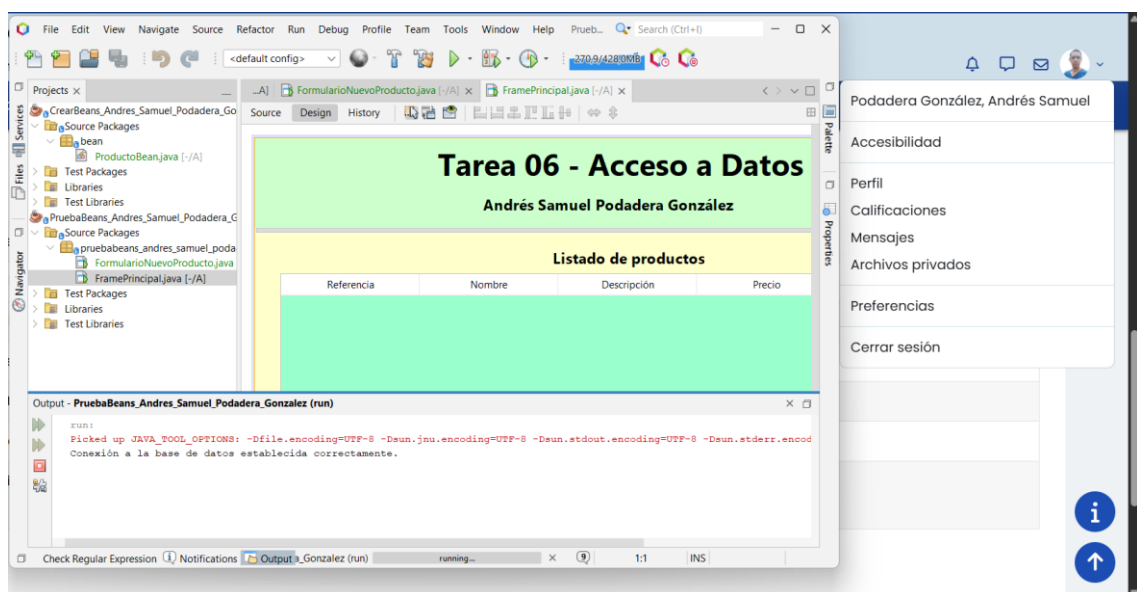
Recoge el resultado de la operación, ya sea que el producto se ha eliminado correctamente o ha ocurrido algún error y lo retorna.

EJERCICIO 2.

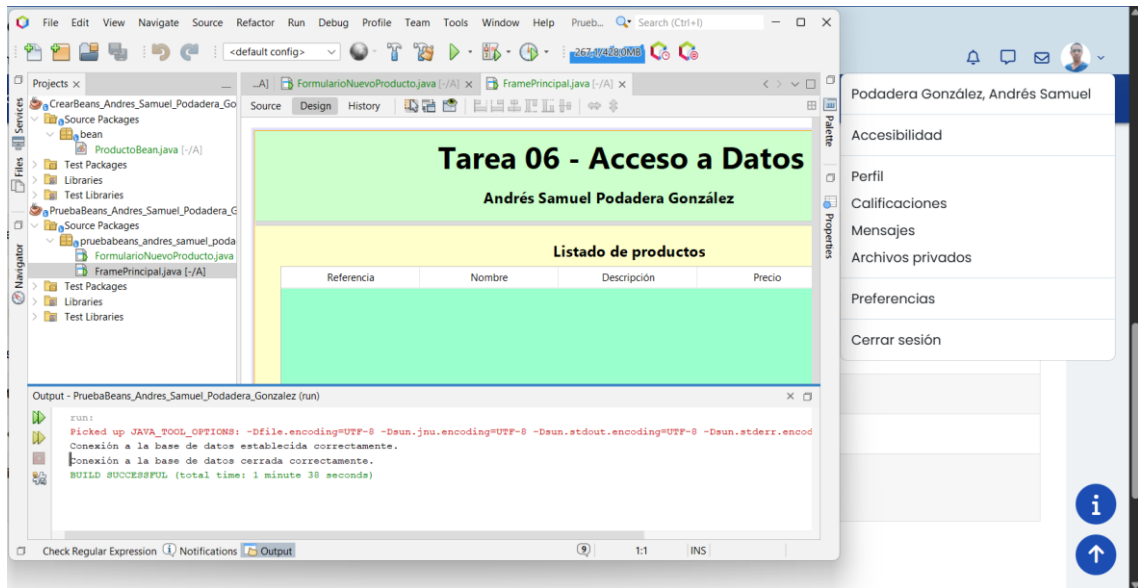
Crear un proyecto Java con swing, llamado *PruebaBeans_Nombre_Apellidos*, para probar el componente desarrollado en el ejercicio anterior (por tanto, en esta aplicación es necesario usar el fichero .jar creado en el ejercicio 1). El programa deberá permitir realizar las siguientes acciones:

- Deberá conectar con la base de datos Comercio.

Cada vez que arranca la aplicación se muestra en la consola un mensaje de conexión a la base de datos:



Cada vez que se cierra la aplicación aparece en la consola un mensaje de desconexión de la base de datos:



- *Mostrar un listado numerado con los registros de la tabla productos.*

Al arrancar la aplicación se muestra la tabla con el listado numerado de todos los productos:



Esta tabla es actualizada cada vez que se dispara un evento de añadir o eliminar un producto de la base de datos. Todo esto queda gestionado por la implementación de la interfaz BDModificada.

- *Añadir un nuevo producto a la base de datos. Se debe informar por pantalla del resultado de la acción y se debe mostrar el listado de productos actualizado.*

Para añadir un producto, haremos clic en el botón “Añadir Producto” del FramePrincipal y nos abrirá un JDialog con el formulario de nuevo producto. Allí rellenaremos todos los campos y haremos clic en “Añadir”.

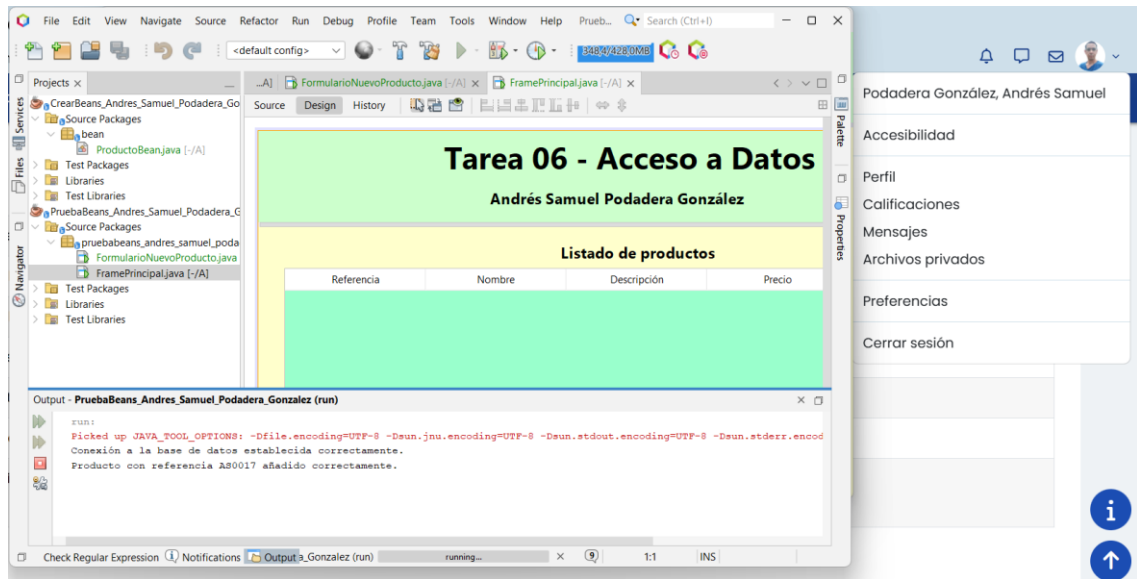
The screenshot shows a web application window titled 'Tarea 06 - Acceso a Datos - Componente ProductoBean'. The main content area is a yellow form titled 'Añadir nuevo producto'. On the left, there is a table with a 'Referencia' column and a list of product codes: AC0001, AC0002, AS0001, AS0002, AS0003, CC0001, CC0002, CC0003, PC0001, PC0002, and PS0001. The form fields include: 'Referencia' (a dropdown menu showing 'AS' and a text input with '0017'), 'Nombre' (a text input with 'Prueba'), 'Descripción' (a text input with 'Prueba'), 'Precio' (a text input with '4,00' and a spinner), and 'Descuento' (a text input with '1d' and a spinner). At the bottom of the form are three buttons: 'Añadir' (green), 'Limpiar' (blue), and 'Cerrar' (red). Below the form, there are two more buttons: 'Añadir Producto' (green) and 'Cerrar Aplicación' (red). On the right side of the application, there is a user profile section for 'Podadera González, Andrés Samuel' with a dropdown menu containing options: 'Accesibilidad', 'Perfil', 'Calificaciones', 'Mensajes', 'Archivos privados', 'Preferencias', and 'Cerrar sesión'. A blue information icon is visible in the bottom right corner.

El programa mostrará un aviso si todo se ha realizado correctamente y podremos consultar la tabla para ver si efectivamente se ha agregado el nuevo producto.

This screenshot shows the same 'Añadir nuevo producto' form as the previous one, but with a small modal dialog box overlaid in the center. The dialog box is titled 'Éxito' and contains an information icon and the text 'Producto añadido correctamente'. It has an 'OK' button. The form fields and buttons remain the same as in the previous screenshot. The user profile section on the right is also visible.

Además, en consola, aparecerá un mensaje con la referencia del producto y si se ha añadido correctamente o ha ocurrido algún error.

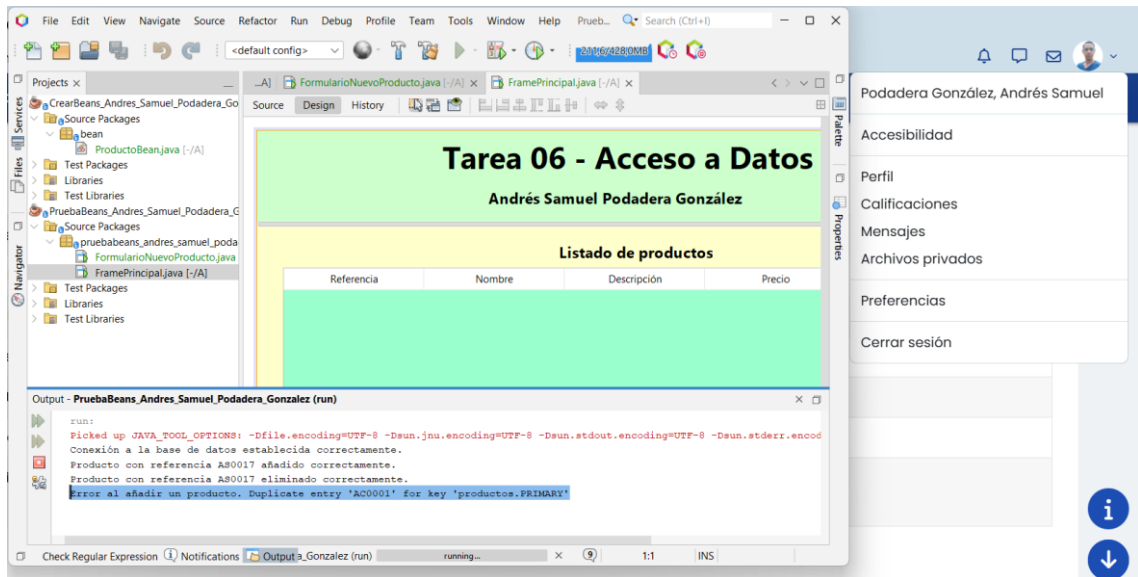
Nombre: Andrés Samuel Podadera González
Ciclo: CFGS Desarrollo de aplicaciones multiplataforma
Curso: 2024/2025



En caso de intentar ingresar una referencia de un producto ya existente, el programa nos avisará con un mensaje de error en pantalla:



Y nos especificará el problema en la consola:



- *Borrar un producto de la base de datos. Se debe informar por pantalla del resultado de la acción y se debe mostrar el listado de productos actualizado.*

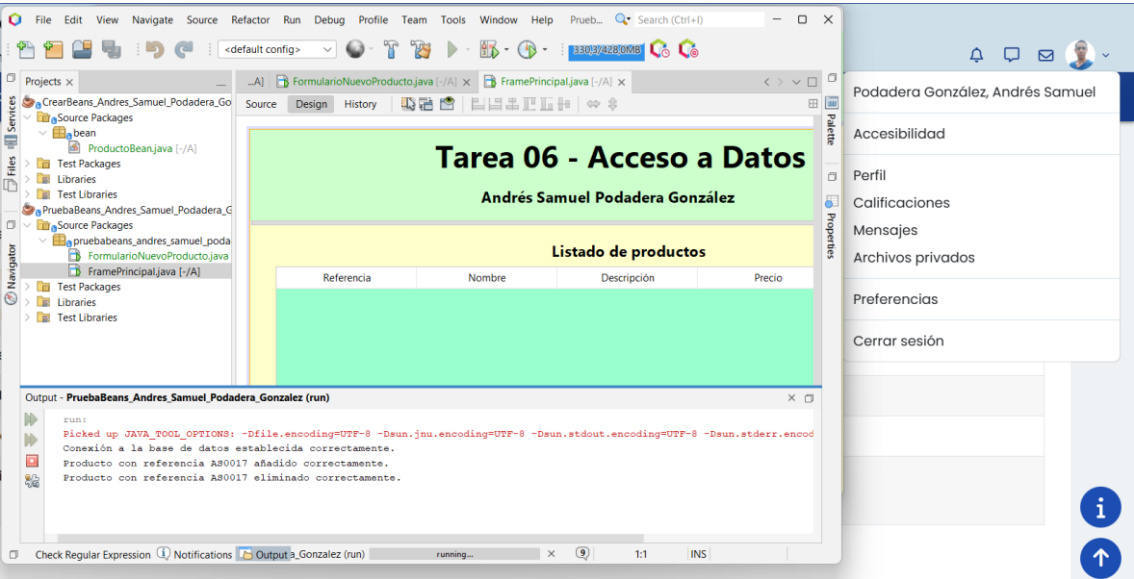
Para eliminar un producto, primero seleccionaremos la fila del producto que deseamos eliminar en la tabla de productos. Seguidamente haremos clic con el botón derecho del ratón y haremos clic sobre eliminar.



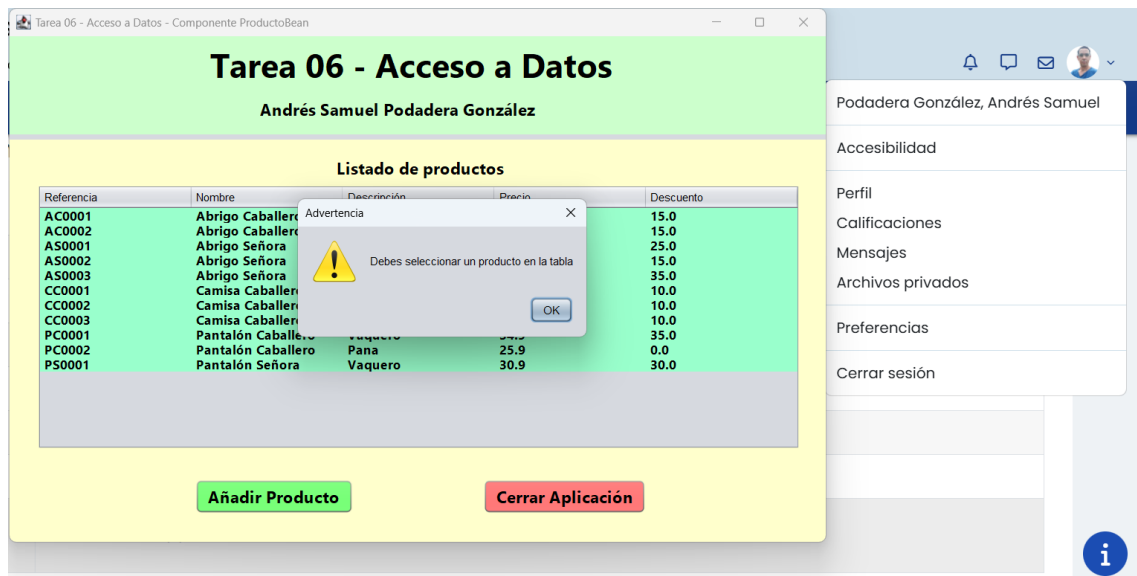
Esta acción nos mostrará un mensaje de si el producto se ha eliminado correctamente o, por el contrario, ha ocurrido algún error. Automáticamente la tabla de productos se actualizará, desapareciendo el producto eliminado.



A través de la consola también se mostrará si la operación se ha llevado a cabo correctamente o ha ocurrido algún error:



En caso intentar eliminar un producto sin tener marcada una fila de la tabla, el programa nos avisará con un mensaje de advertencia:



Notas Aclaratorias

Doy por supuesto que nuestro `ProductoBean` va a ser utilizado en el contexto de Netbeans y que por tanto vamos a tener la opción de mostrar mensajes por consola. Si nuestro bean se utilizara en otro contexto como el de un servidor real, habría que realizarle modificaciones a esos **`System.out.println()`**, y utilizar otra forma de logging de mensajes.

Como se puede observar en el proyecto de pruebas del bean, nuestro jar **`ProductoBeanFatJar.jar`** ya encapsula toda la lógica necesaria para realizar las operaciones necesarias, incluido el código del conector JDBC para la base de datos MySQL. Con estos simplificamos la tarea de compartir, reutilizar y gestionar la única dependencia del proyecto en un sólo archivo .jar. Para realizar la construcción del jar he utilizado esta configuración en el archivo `build.xml`:


```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project name="CrearBeans_Andres_Samuel_Podadera_Gonzalez" default="default" basedir=".">
3   <description>Builds, tests, and runs the project CrearBeans_Andres_Samuel_Podadera_Gonzalez.</description>
4
5   <!-- Importa la configuración y targets de NetBeans -->
6   <import file="nbproject/build-impl.xml"/>
7
8   <!-- Tareas personalizadas para generar fat JAR -->
9
10  <!-- Propiedades -->
11  <property name="src.dir" location="src"/>
12  <property name="build.classes.dir" location="build/classes"/>
13  <property name="dist.dir" location="dist"/>
14  <property name="fatjar.dir" location="build/fatjar"/>
15  <property name="driver.jar" location="mysql-connector-j-9.1.0.jar"/>
16
17  <!-- Crear fat JAR con driver incluido -->
18  <target name="jar-todo-en-uno" depends="jar">
19
20    <!-- 1. Extraer driver -->
21    <mkdir dir="temp-driver"/>
22    <unzip src="${driver.jar}" dest="temp-driver"/>
23
24    <!-- 2. Combinar clases compiladas + driver -->
25    <mkdir dir="${fatjar.dir}" />
26    <copy todir="${fatjar.dir}">
27      <fileset dir="${build.classes.dir}" />
28      <fileset dir="temp-driver" />
29    </copy>
30
31    <!-- 3. Crear JAR final -->
32    <mkdir dir="${dist.dir}" />
33    <jar destfile="${dist.dir}/ProductoBeanFatJar.jar" basedir="${fatjar.dir}">
34      <!-- Main-Class opcional -->
35      <manifest>
36        <attribute name="Main-Class" value="bean.ProductoBean" />
37      </manifest>
38    </jar>
39  </target>
40</project>
```

La numeración de las filas de la tabla ha sido una modificación de última hora es por eso por lo que no aparece en algunas de las capturas.