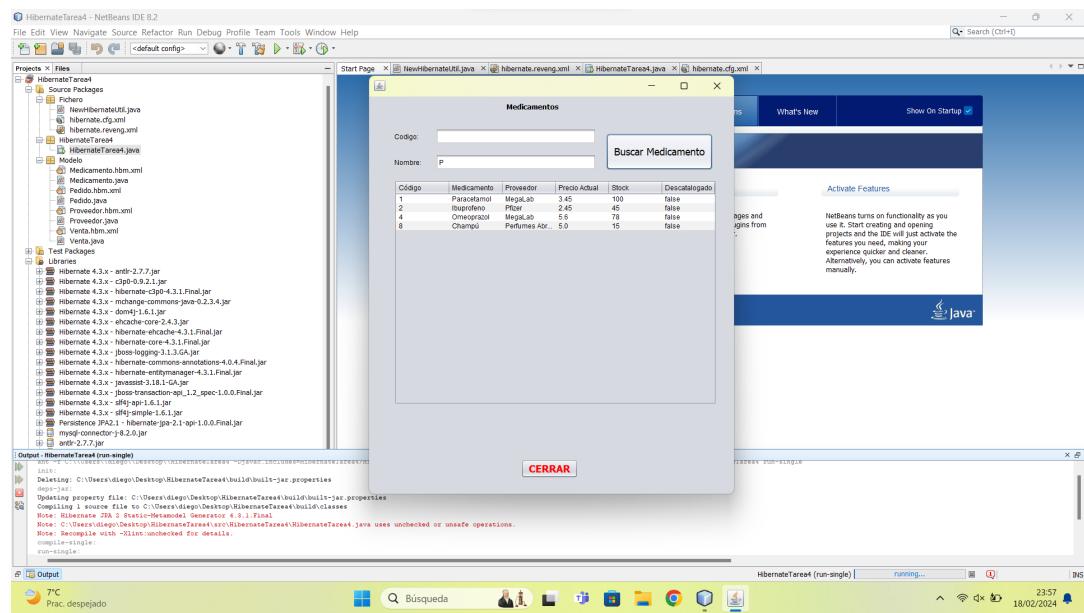
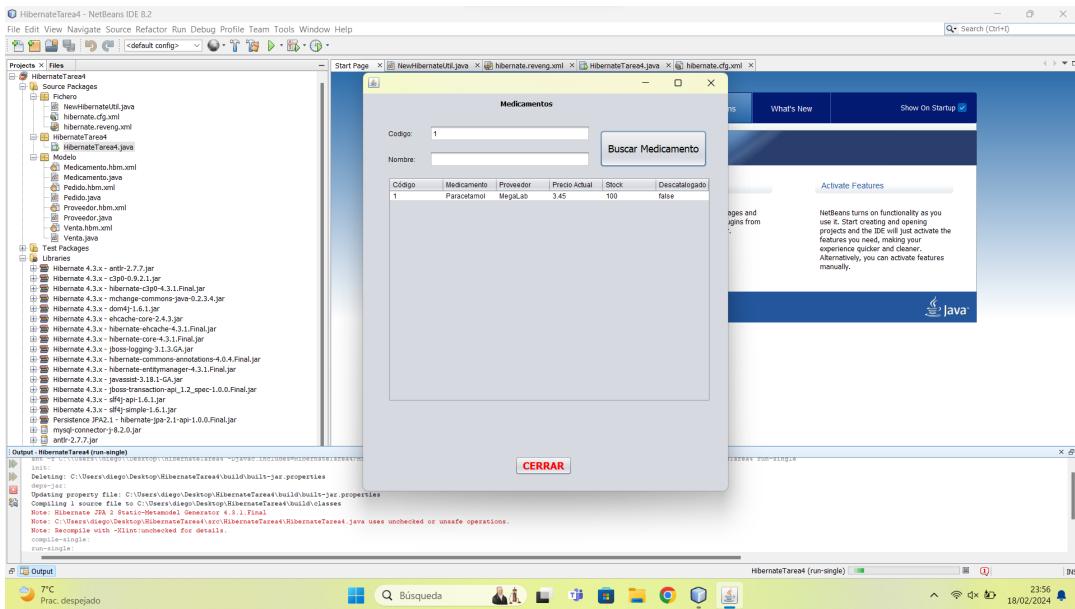


Tarea 4 para Acceso a Datos



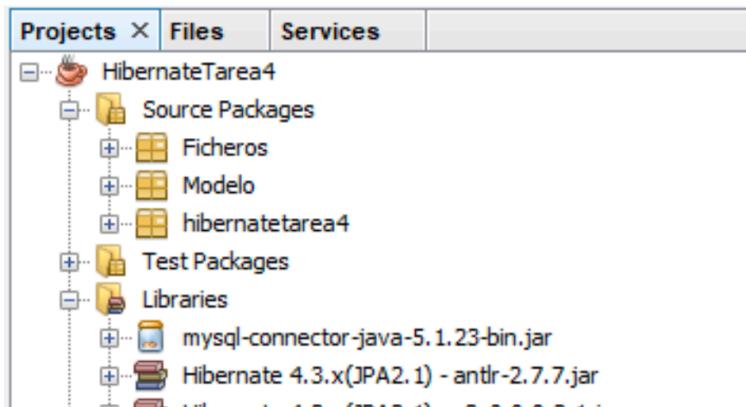
Diego Manuel Carrasco Castañares

TAREA 4

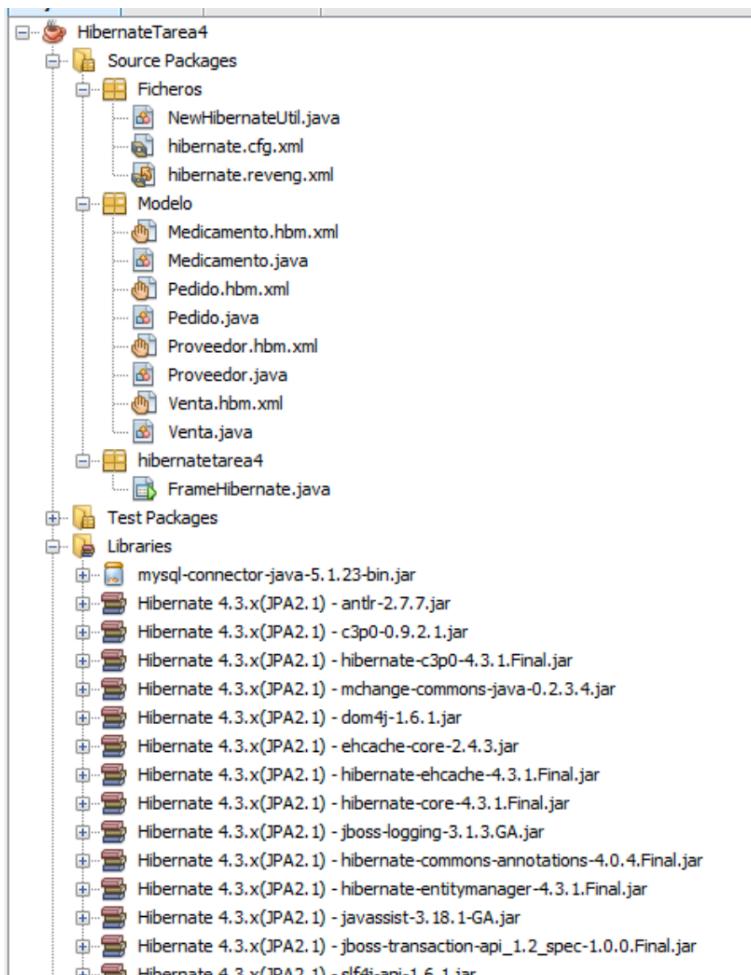
Elabora un documento .pdf con la explicación del desarrollo de la tarea y su ejecución. Si no lo entregas la puntuación de la tarea se verá reducida en 2 puntos.

Con el servidor xampp y librerías Hibernate.

Hay que crear una aplicación o proyecto en java con Netbeans, que tendrá la siguiente estructura (respeta los nombres)y debe hacer lo siguiente:



Debe quedar al final así.

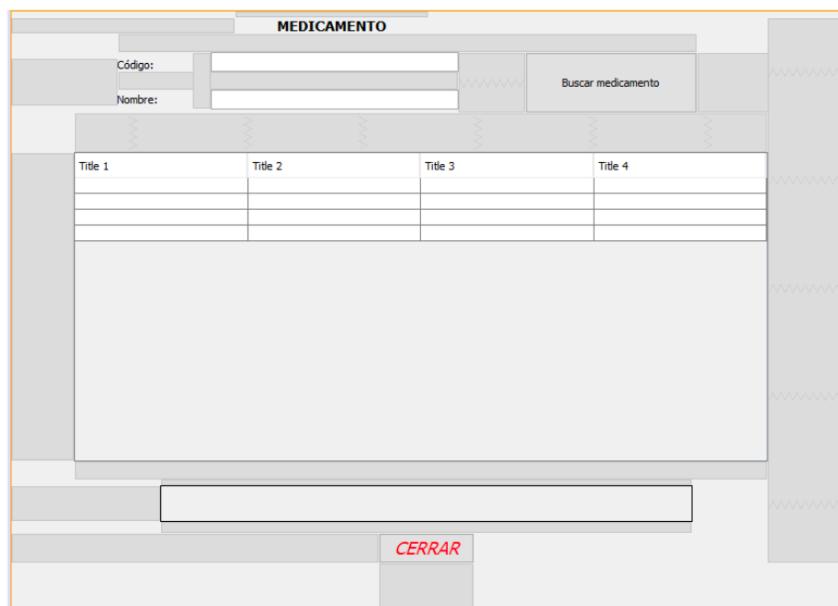


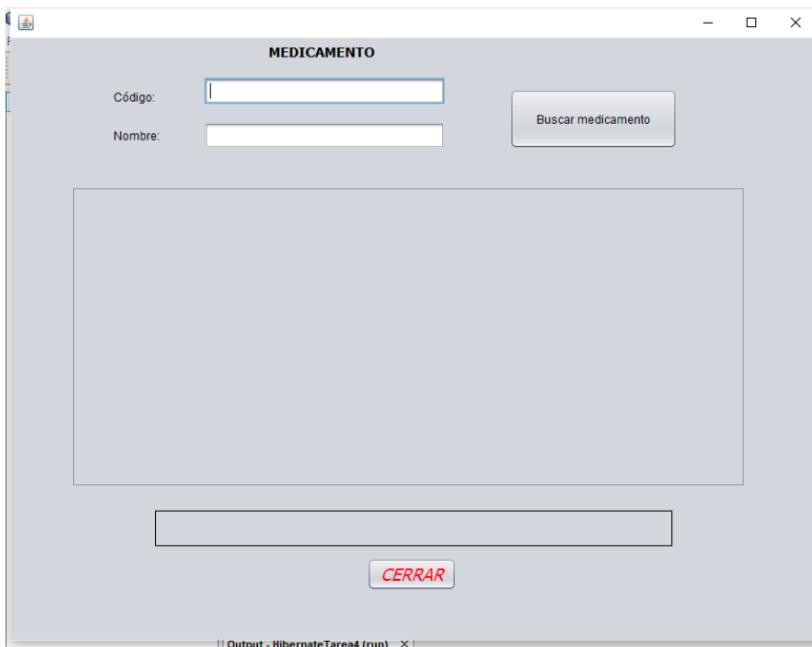
- 1.- Instala (puedes utilizar por ejemplo phpmyAdmin o workbench) la base de datos farmacia utilizando el sql que se adjunta. 0,25 puntos.
- 2.- Una vez creado el proyecto crea el fichero de configuración de Hibernate, con acceso a la base de datos farmacia. 1 punto
- 3.- Utiliza el asistente de archivos de ingeniería inversa de Hibernate para poder acceder a todas las tablas de la base de datos. Crea también el NewHibernateUtil.java 1 punto
- 4.- Genera el fichero el fichero de mapeo POJOS. Unir los ficheros de configuración y el de inversa . Deberán aparecer los .xml y los .java. por cada tabla que han sido elegidas en el apartado anterior. 1 punto
- 5.- Guarda y cierra todos los ficheros creados para poder probar el editor de Netbeans de HQLQuery (fichero de configuración con botón derecho) 3 consultas
- 5.1.- Mostrar el contenido de Venta en orden ascendente por código. . 0,5 puntos

"FROM Venta ORDER BY código ASC"

Foto 16

- 5.2.- Mostrar el contenido de Venta del código 1. 0,5 puntos
- 5.3. Mostrar el contenido de Venta por orden ascendente por código pero no debe aparecer el código del medicamento sino el nombre. . 0,75 puntos
- 6.- JFrame debe ser así cuando se ejecute debe quedar como la segunda imagen. 1 punto





7.- Lo que debe hacer la aplicación lo podéis ver en el vídeo que se adjunta y en el fichero con los pantallazos. 4 puntos

Todo lo de código

Todo lo de Nombre

Todos los Mensajes

CERRAR y al seleccionar la X de la pantalla.

Criterios de puntuación. Total 10 puntos.

La tarea consta de 10 actividades a desarrollar que, en conjunto, sumarán 10 puntos.

Indicaciones de entrega.

Una vez realizada la tarea entregarás un fichero comprimido con el proyecto y con el documento donde explicas el desarrollo de la tarea (pantallazos de ejecución y código de la ejecución de cada botón (en formato pdf) donde figuren las respuestas correspondientes. El envío se realizará a través de la plataforma de la forma establecida para ello, y el archivo se nombrará siguiendo las siguientes pautas:

apellido1_apellido2_nombre_ADxx_Tarea

Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna Begoña Sánchez Mañas para la primera unidad del MP de AD, debería nombrar esta tarea como...

sanchez_manas_begona_AD02_Tarea

Diego Manuel Carrasco Castañares

Índice

Instalación base de datos farmacia.....	06
Creación del fichero de configuración de Hibernate.....	09
Creación Hibernate.revenge y NewHibernateUtil.java.....	12
Generación del fichero el fichero de mapeo POJOS.....	15
Realización de 3 consultas.....	16
1.- Mostrar el contenido de Venta en orden ascendente por código.....	16
2.- Mostrar el contenido de Venta del código 1.....	17
3.- Mostrar el contenido de Venta por orden ascendente por código pero no debe aparecer el código del medicamento sino el nombre.....	17
Diseño del JFrame.....	18
Lo que debe hacer la aplicación.....	20

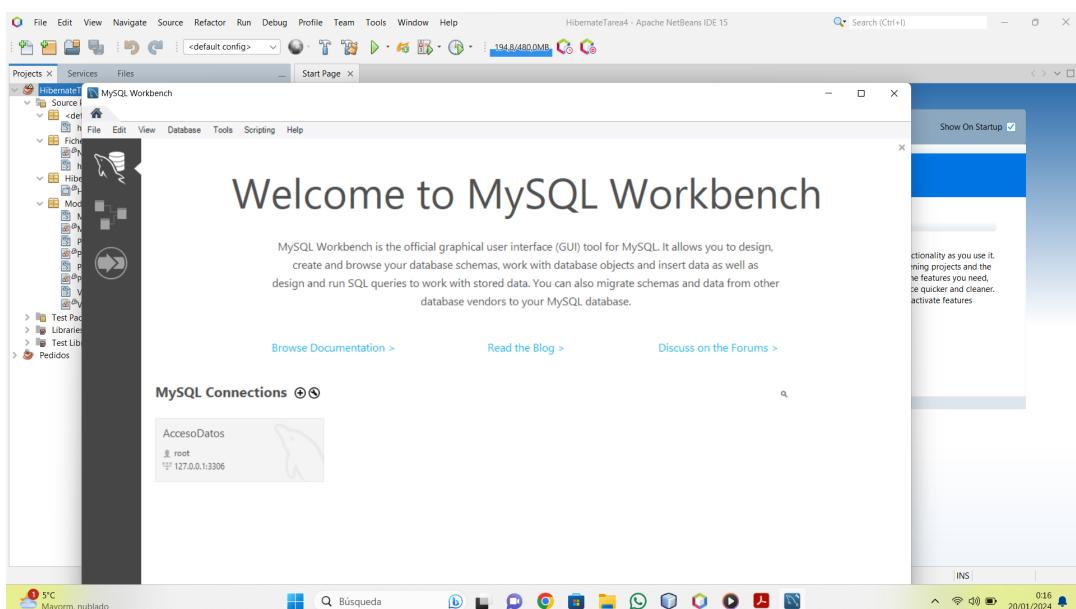
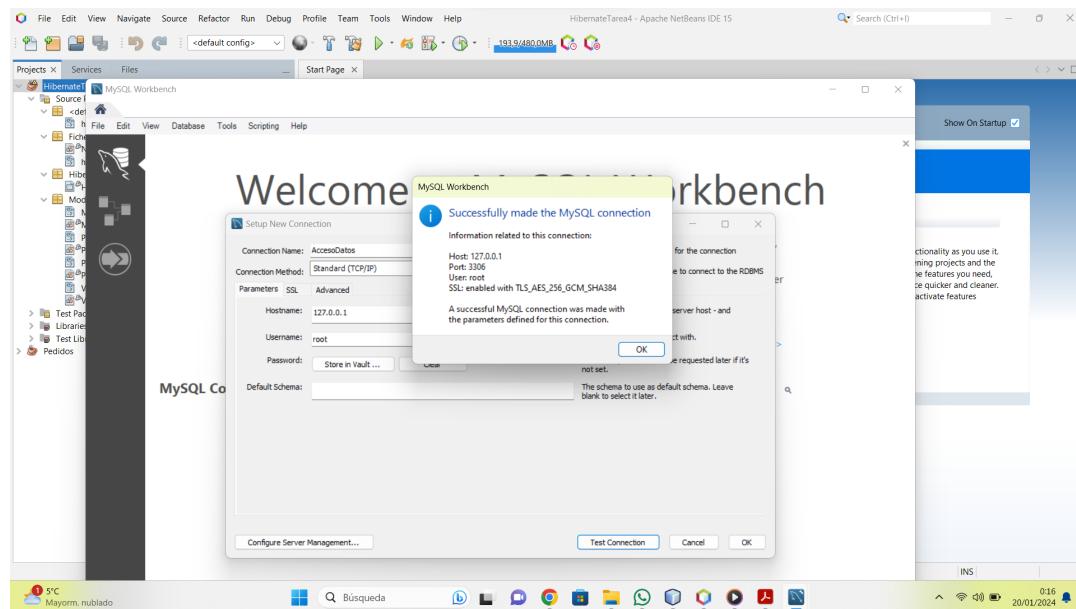
1.- Instala (puedes utilizar por ejemplo phpmyAdmin o workbench) la base de datos farmacia utilizando el sql que se adjunta. 0,25 puntos.

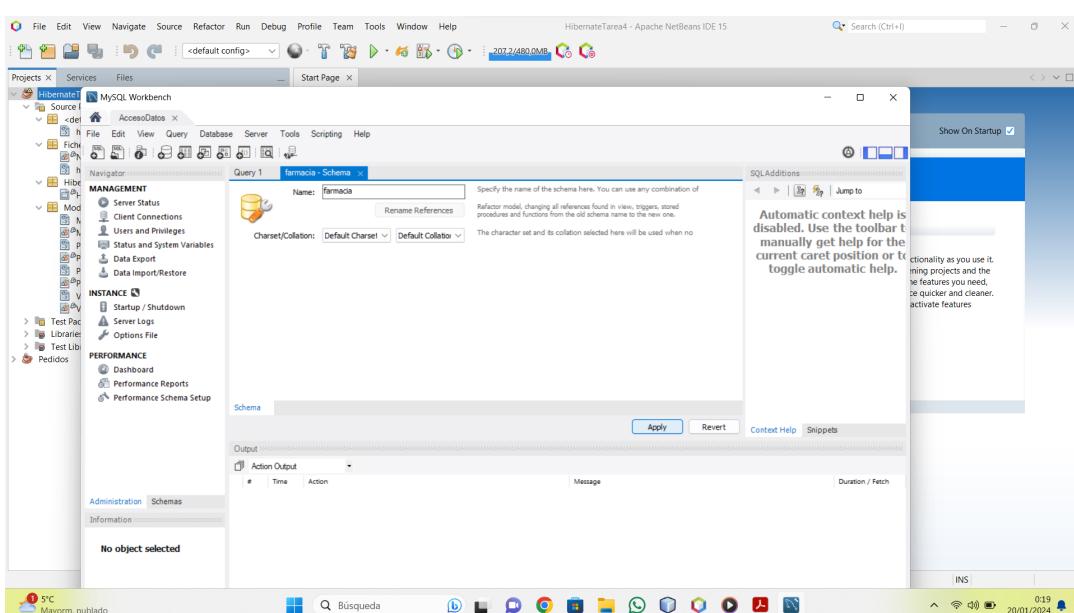
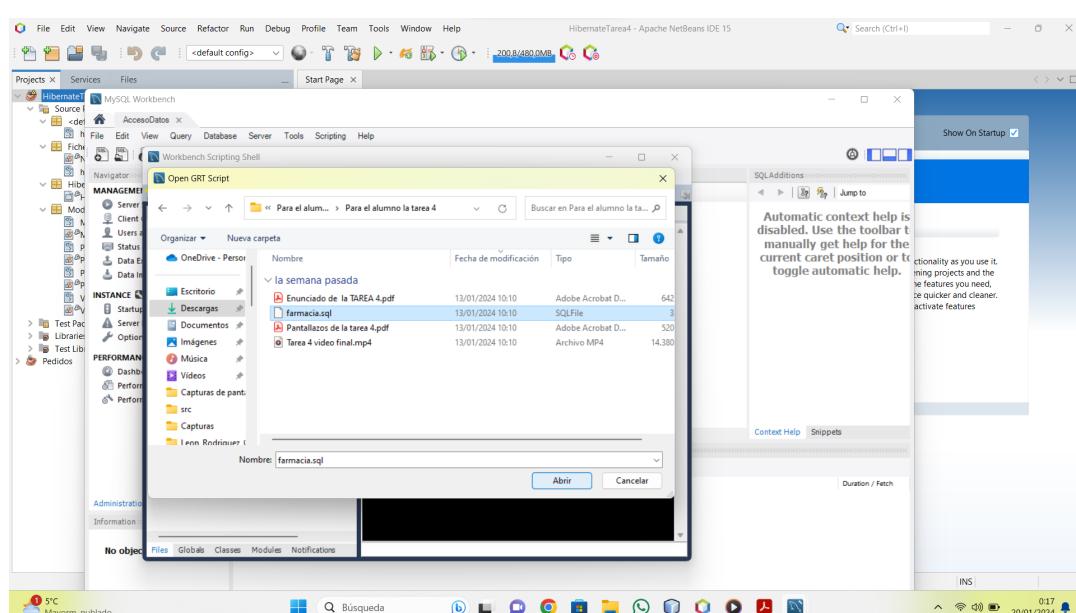
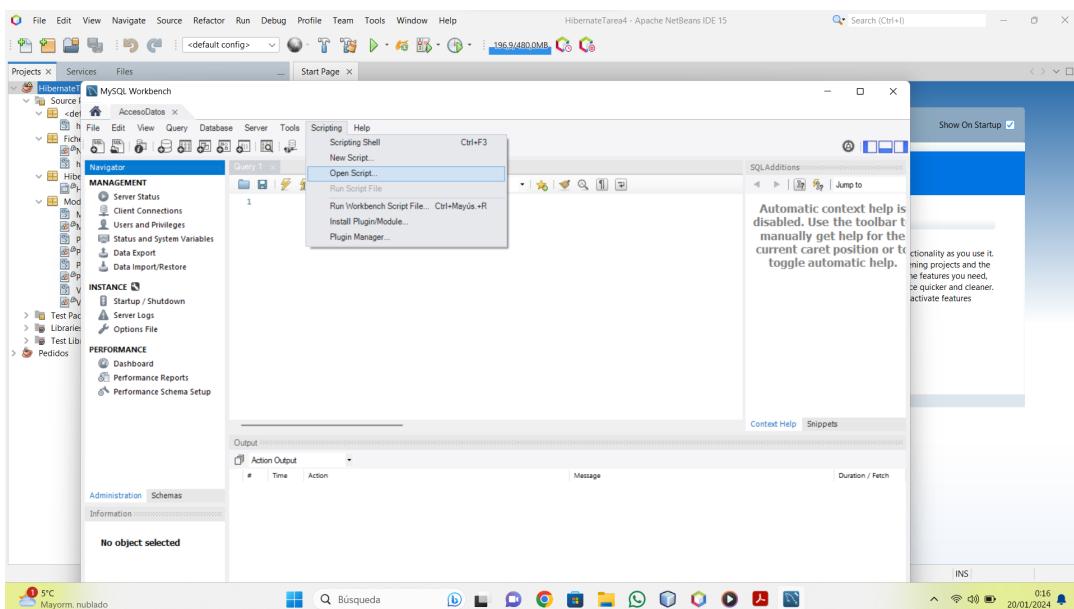
Para instalar la base de datos de farmacia abriremos el MySQL Workbench, crearemos, en mi caso, una nueva conexión a la que llamaremos "AccesoDatos".

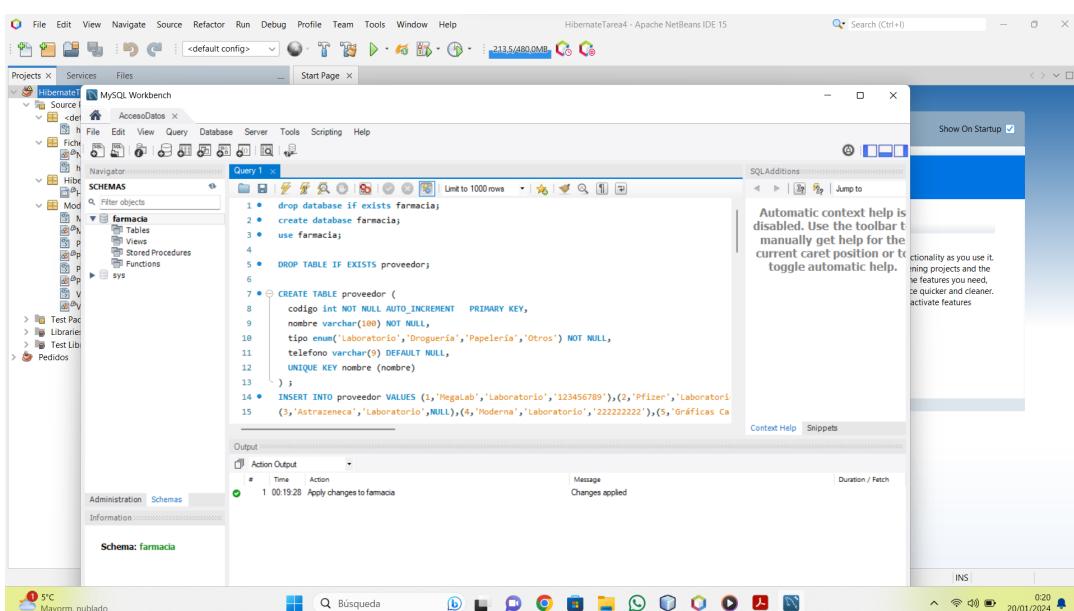
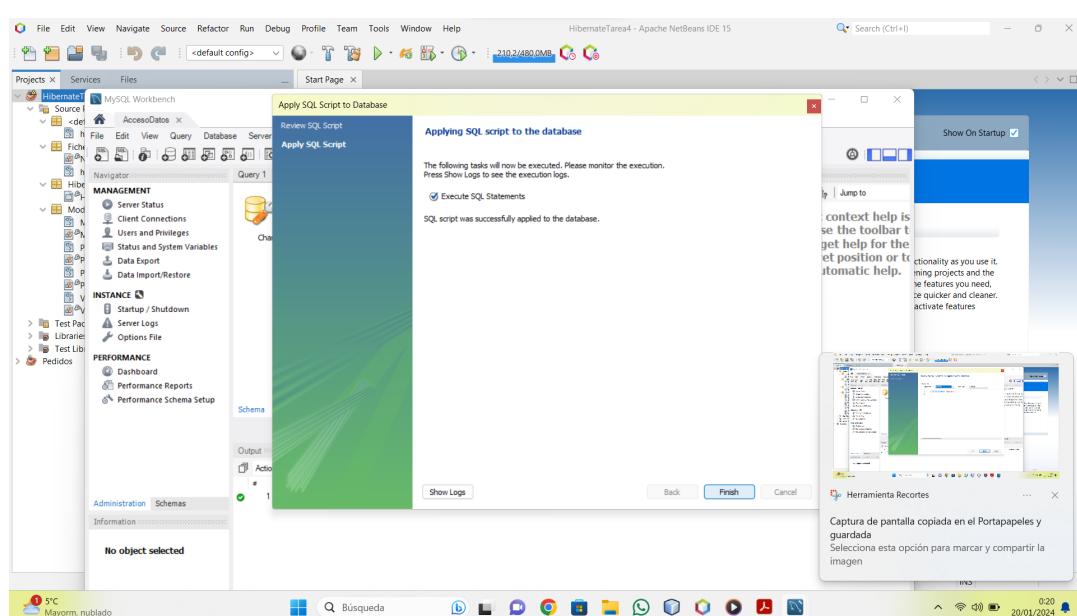
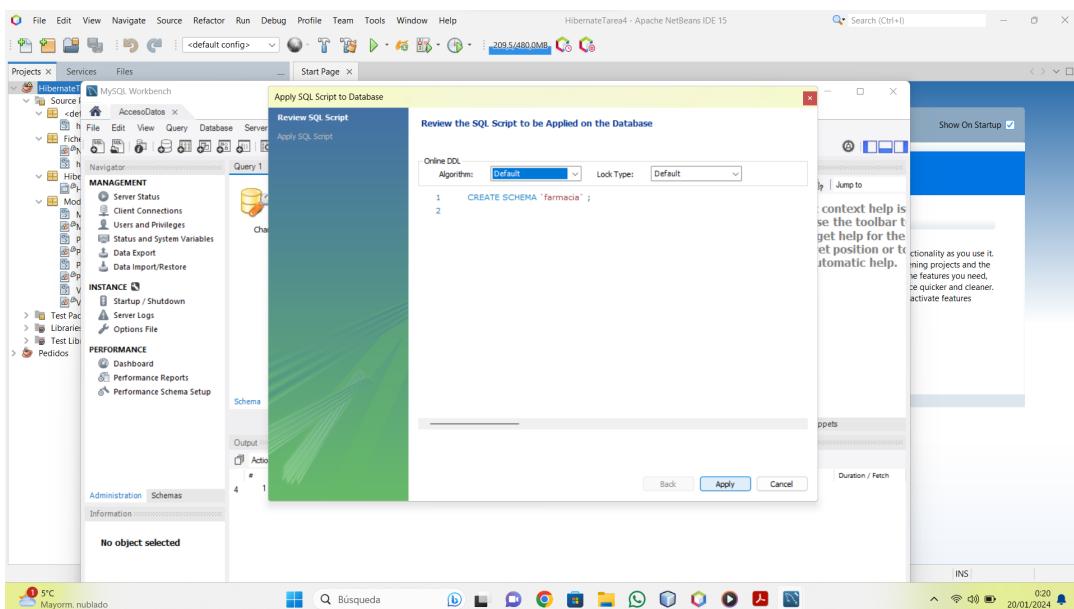
Una vez creada, accederemos a ella, haremos clic en "script", "open script", buscamos el script de nuestra tarea, clicamos en abrir y seguimos los pasos que nos van indicando las diferentes ventanas que se nos van abriendo (podemos verlo en las imágenes).

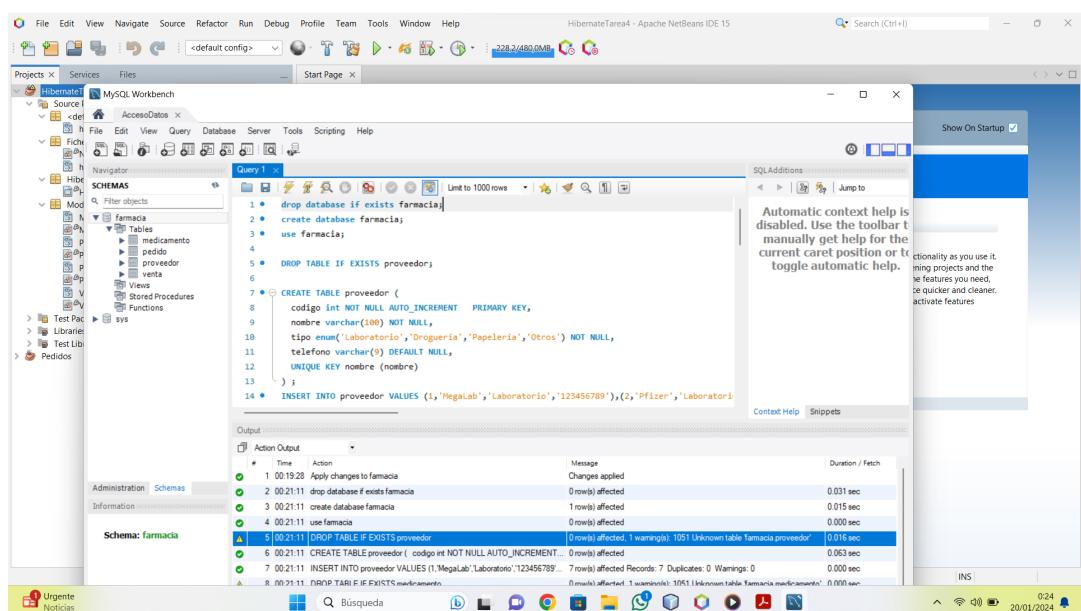
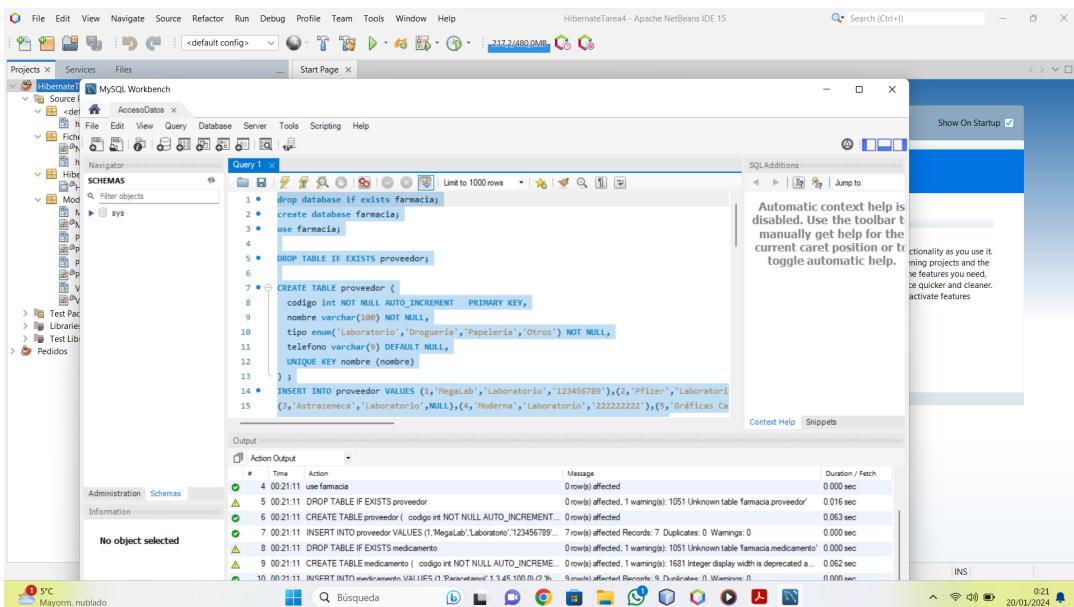
Tras ello se nos abrirá la ventana que contiene el script, seleccionamos todo el contenido como se muestra en las imágenes y le damos al icono de ejecutar script.

En las siguientes imágenes podemos ver el proceso.









2.- Una vez creado el proyecto crea el fichero de configuración de Hibernate, con acceso a la base de datos farmacia. 1 punto

En primer lugar empezaremos creando el proyecto en NetBeans al cual llamaremos "HibernateTarea04".

A continuación crearemos un nuevo paquete llamado "Fichero" dentro del paquete por defecto.

Tras ello crearemos un nuevo archivo de hibernante del tipo "Hibernate Configuration Wizard" llamado "hibernate.cfg" dentro del paquete "Fichero".

En la siguiente ventana seleccionaremos el driver que vamos a usar (MySQL conector j driver), pondremos el nombre del host "localhost" en nuestro caso, el puerto de conexión que será el 3306, el nombre de la base de datos que tenemos creada "farmacia"

para esta tarea, el nombre de usuario y al contraseña de acceso a la base de datos.

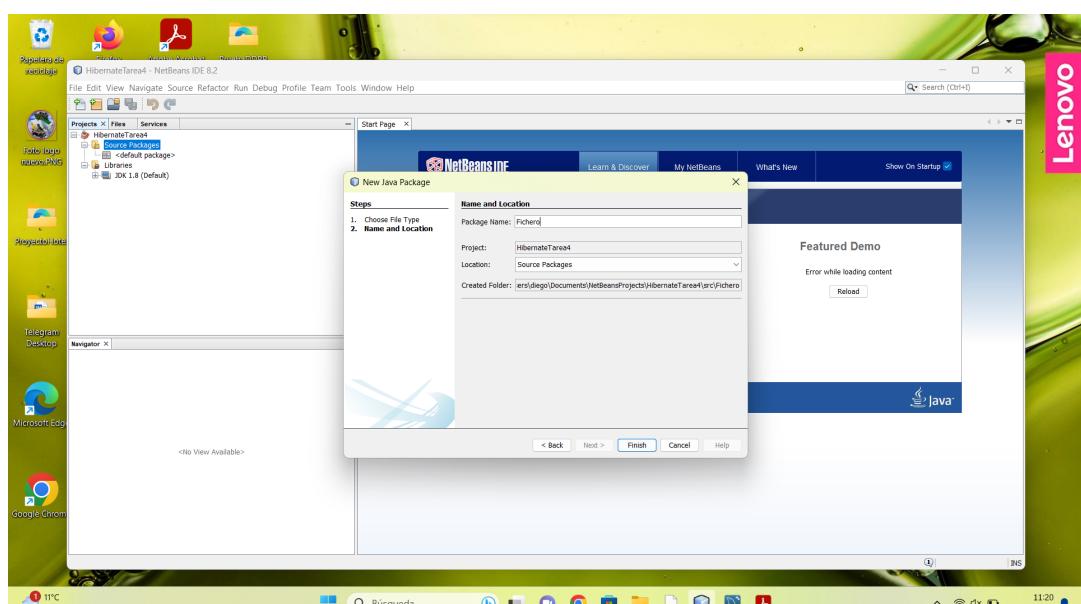
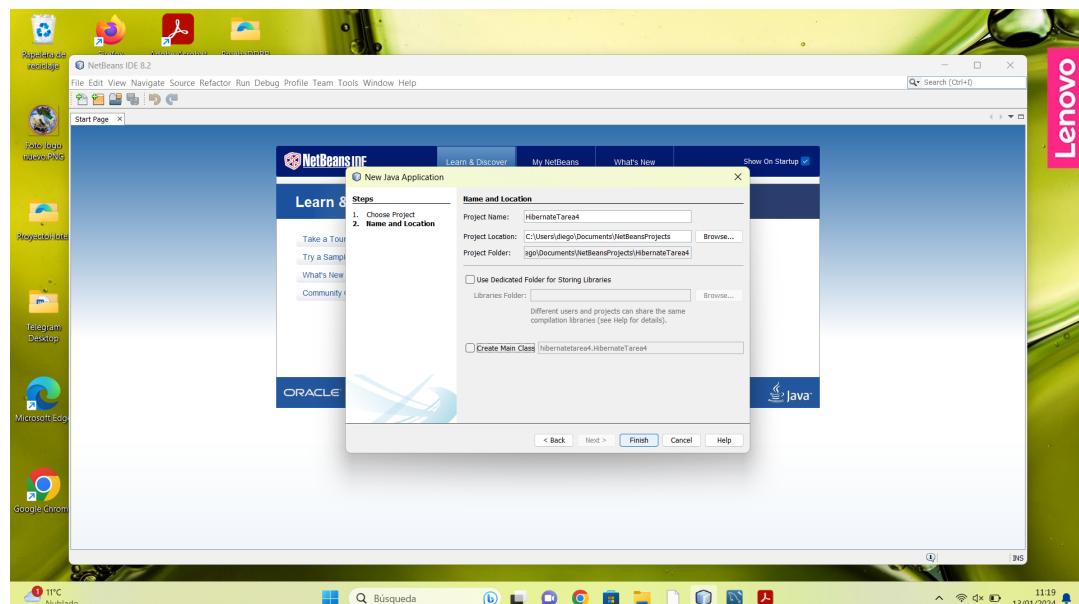
Si todos los datos son correctos, al hacer el test de conexión debe de ser satisfactorio.

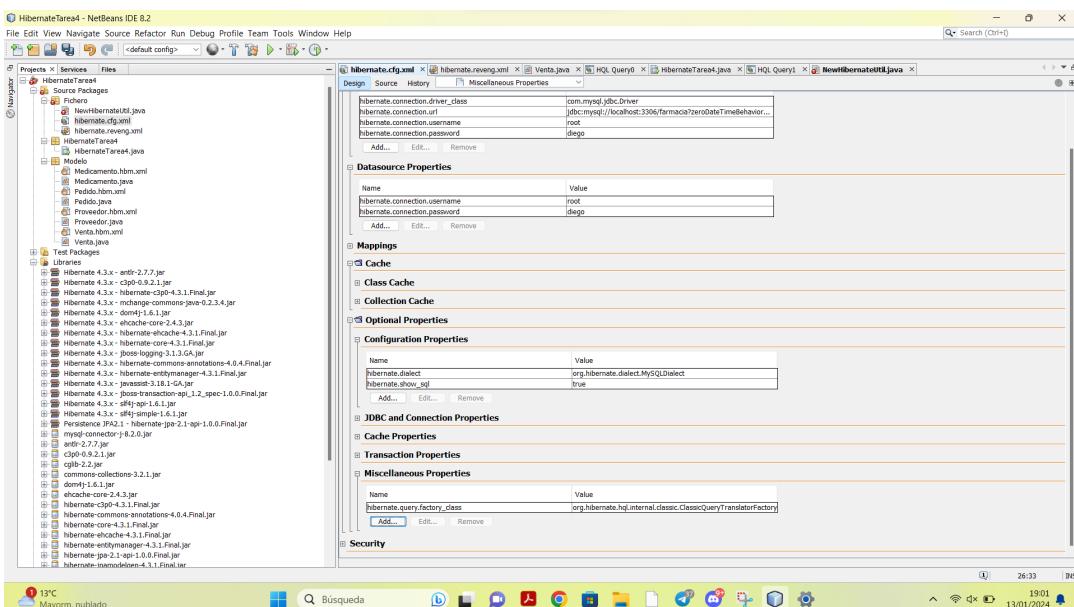
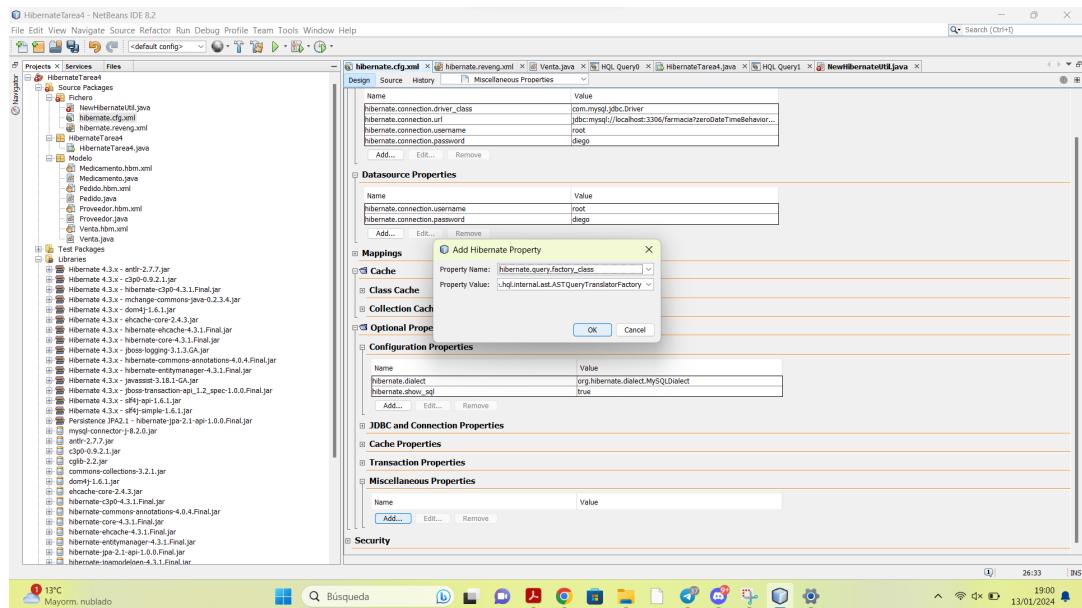
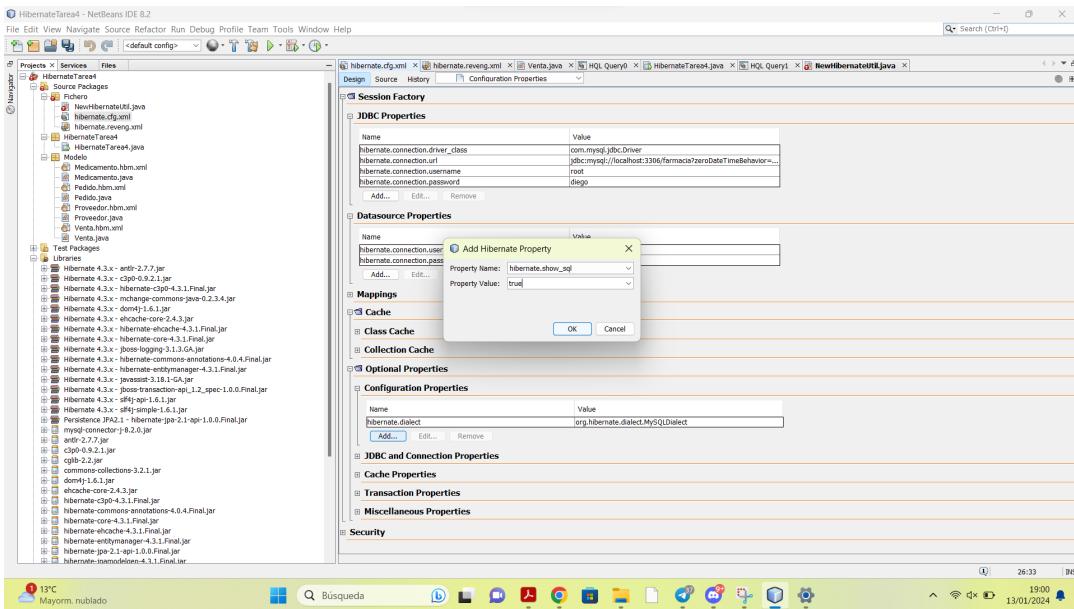
Al clicar en siguiente nos abrirá el “select data source” donde comprobaremos que los datos son correctos y clicaremos en finalizar.

Además, deberemos añadir la propiedad hibernate.show_sql e instanciarla a true en el apartado de configuration properties.

Al igual, en el mismo apartado, añadiremos el hibernate.query.factory_class con la propiedad hibernate.hql.internal.ast.ASTQueryTranslatorFactory.

En las siguientes imágenes podemos ver el proceso.



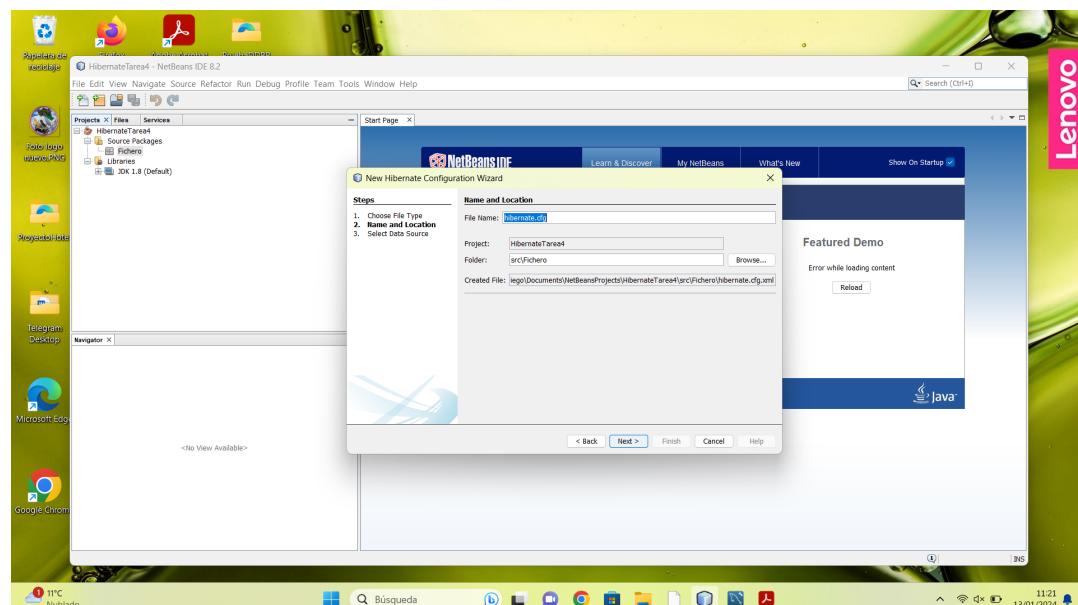
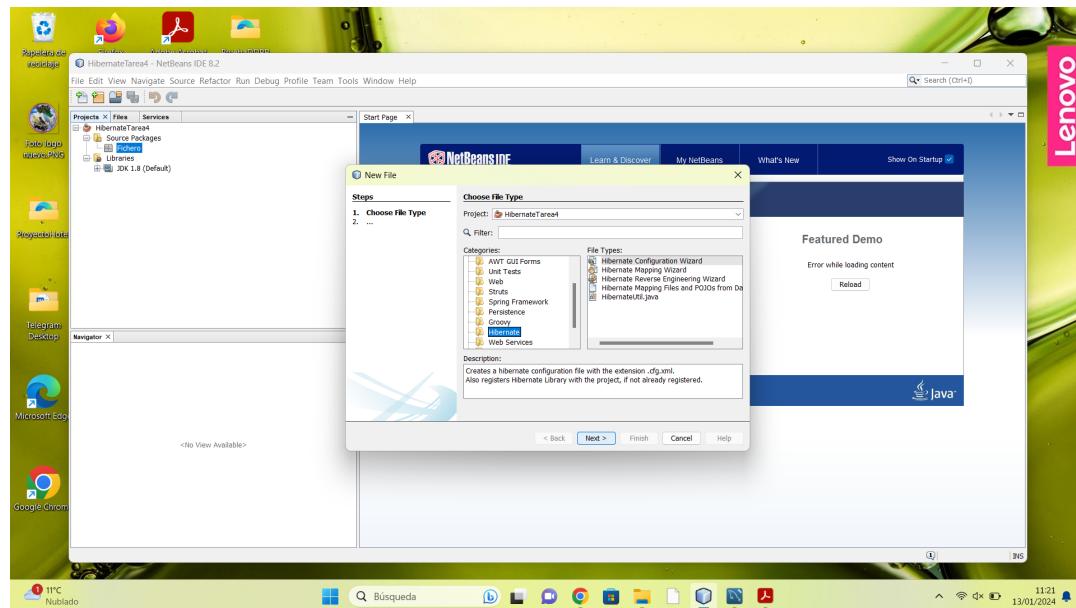


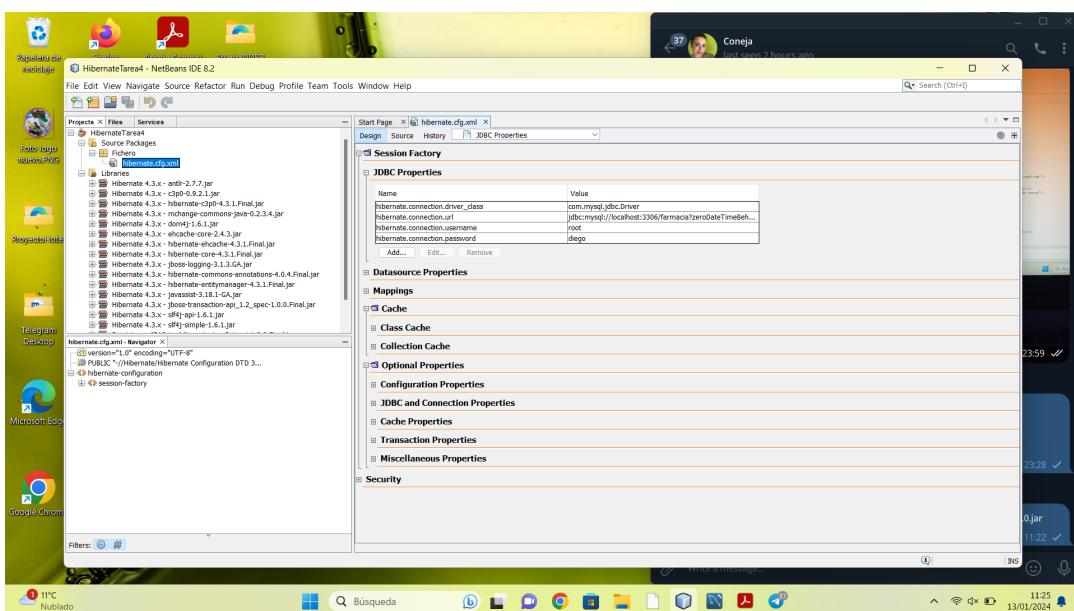
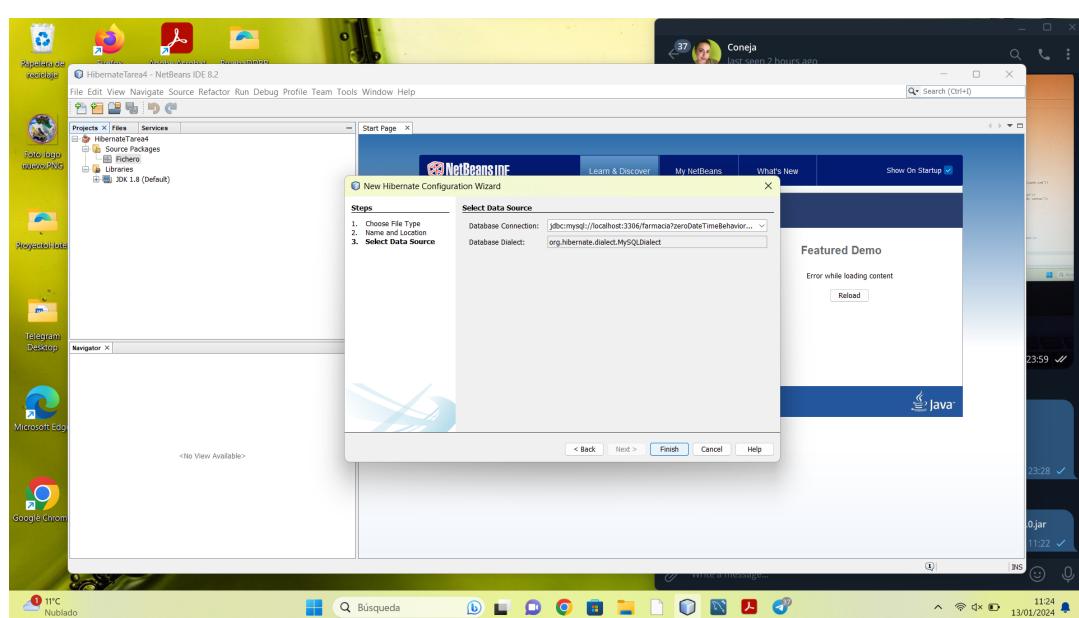
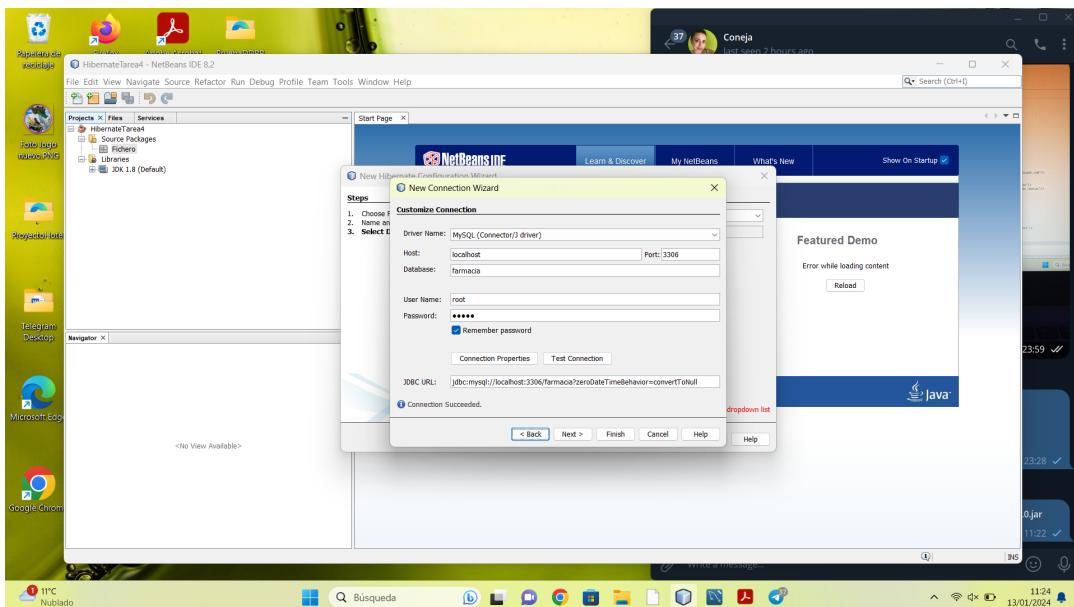
3.- Utiliza el asistente de archivos de ingeniería inversa de Hibernate para poder acceder a todas las tablas de la base de datos. Crea también el NewHibernateUtil.java 1 punto

Para crear estos dos archivos seguiremos los mismos pasos que en el apartado anterior pero esta vez seleccionaremos como tipo de archivo "HibernateUtil.java" al cual llamaremos "Modelo" y lo incluiremos en el paquete por defecto.

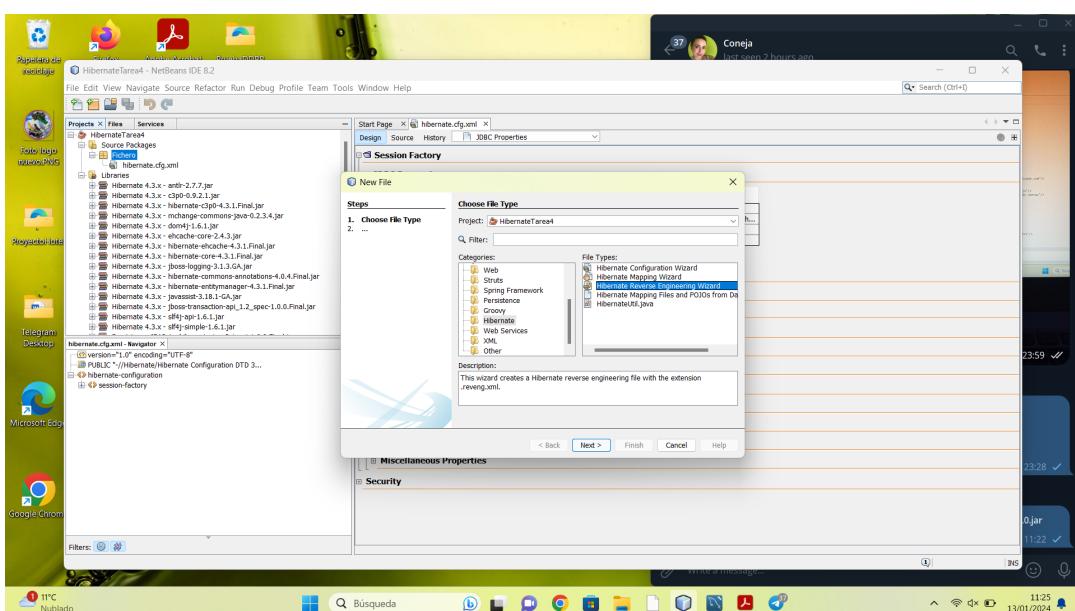
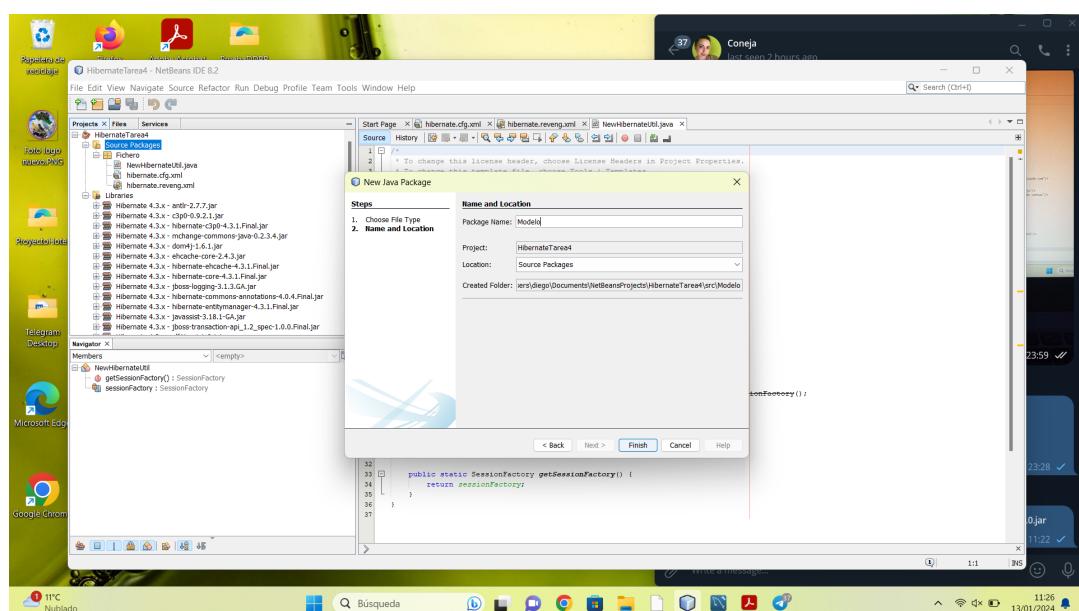
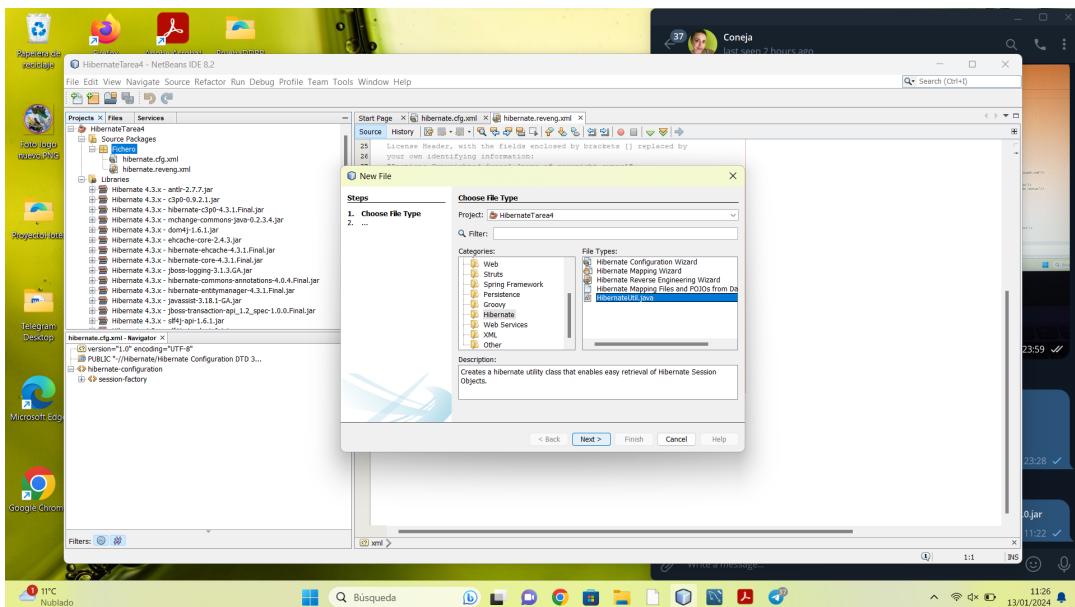
A continuación también seguiremos los mismos pasos seguidos en los dos apartados anteriores pero seleccionado en el tipo de archivo "Hibernate Reverse Engineering Wizard" al cual llamaremos "hibernate.reveng" y lo incluiremos en el paquete Fichero.

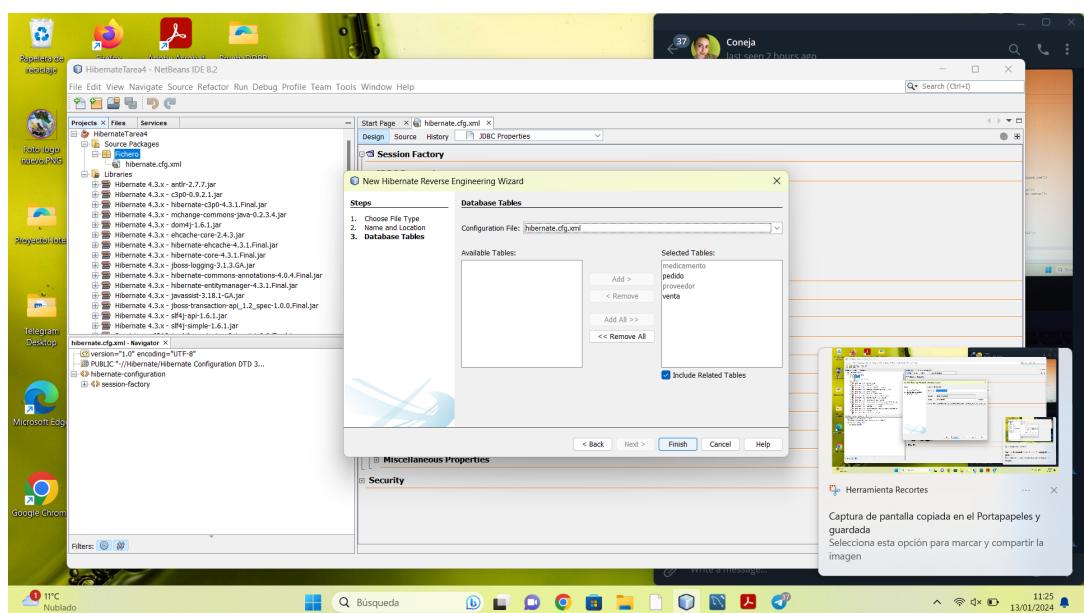
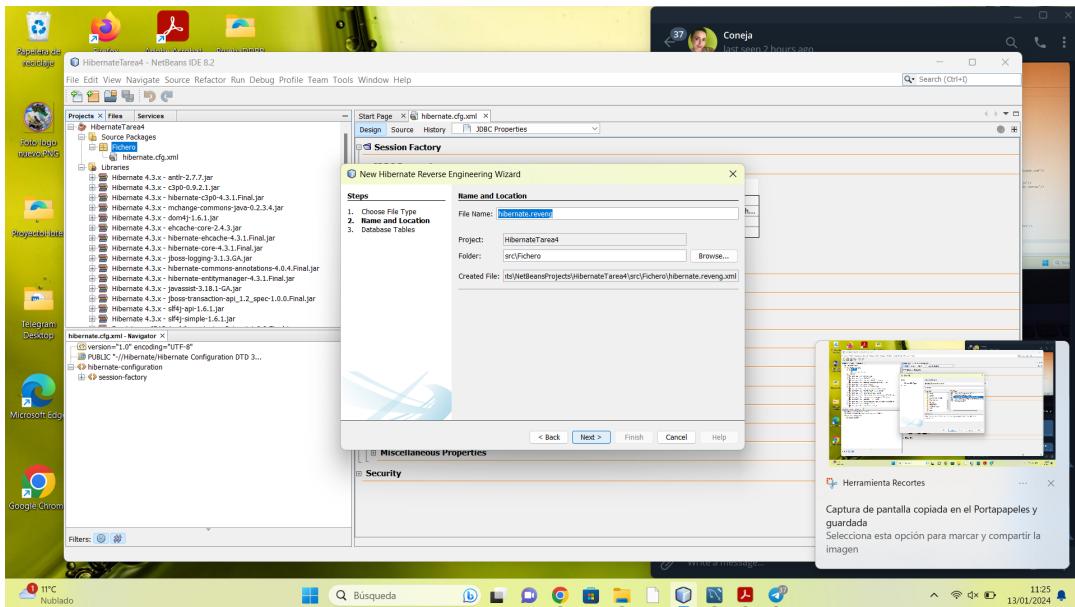
En la pantalla siguiente seleccionaremos todas las tablas que nos muestra en "Available Tables" y las pasaremos al cuadro de "Selected Tables".





Acceso a Datos





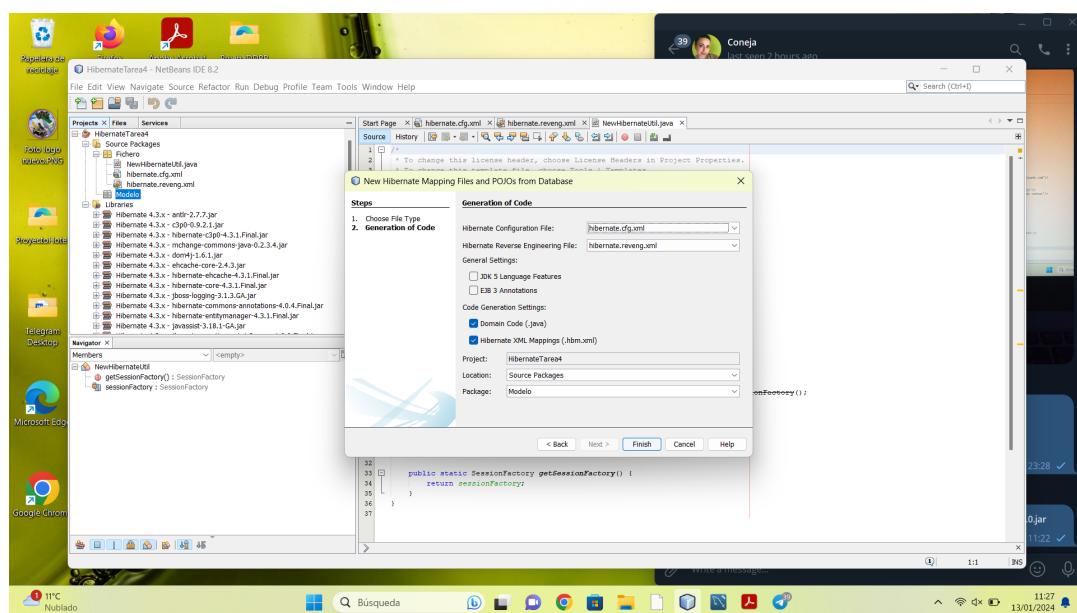
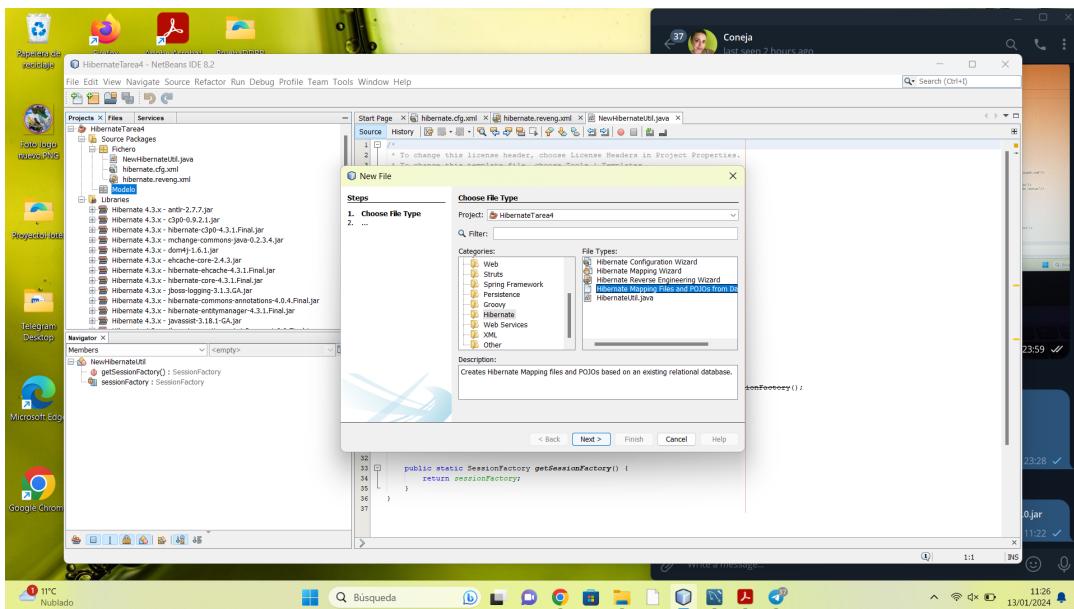
4.- Genera el fichero el fichero de mapeo POJOS. Unir los ficheros de configuración y el de inversa . Deberán aparecer los .xml y los .java. por cada tabla que han sido elegidas en el apartado anterior. 1 punto

Para esta pregunta, siguiendo los mismos pasos de los apartados anteriores, seleccionaremos en tipo de archivo "Hibernate Mapping Files and POJOs from Database".

En la siguiente ventana nos aseguraremos que en "Hibernate Configuration File" tengamos seleccionado "hibernate.cfg.xml", en "Hibernate Reserve Engineering File" debe estar seleccionado "hibernate.reveng.xml" y en "Code Generation Settings" marcados los dos check box de "Domain Code (.java)" y "Hibernate XML Mappings (.hbm.xml)".

Lo guardaremos en el paquete "Modelo".

Diego Manuel Carrasco Castañares



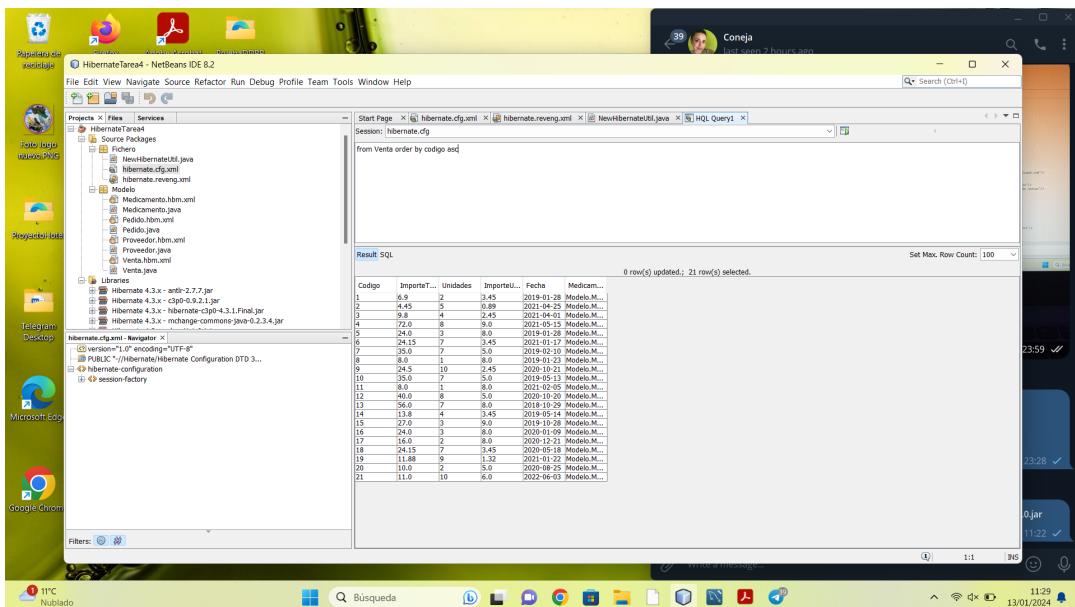
5.- Guarda y cierra todos los ficheros creados para poder probar el editor de Netbeans de HQLQuery (fichero de configuración con botón derecho) 3 consultas

5.1.- Mostrar el contenido de Venta en orden ascendente por código. . 0,5 puntos

La sentencia para la consulta sería la siguiente:

"FROM Venta ORDER BY código ASC"

En la siguiente imagen podemos ver el resultado.

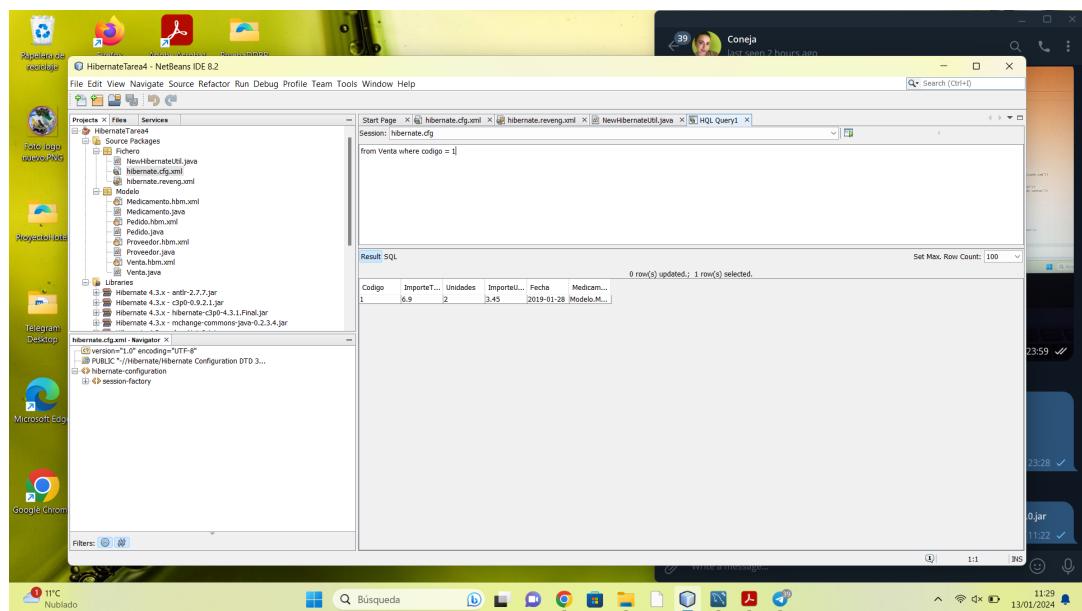


5.2.- Mostrar el contenido de Venta del código 1. 0,5 puntos

La sentencia para la consulta sería la siguiente:

Consulta de la tabla venta con código 1 "FROM Venta WHERE código = 1"

En la siguiente imagen podemos apreciar el resultado.



5.3. Mostrar el contenido de Venta por orden ascendente por código pero no debe aparecer el código del medicamento sino el nombre. . . 0,75 puntos

La sentencia para la consulta sería la siguiente:

```
SELECT m.nombre AS nombreMedicamento, DATE_FORMAT(v.fecha, '%Y-%m-%d') AS fecha, v.unidades, v.importeUnidad, v.importeTotal FROM Venta as v JOIN V.medicamento m ORDER BY v.codigo ASC.
```

En la siguiente imagen podemos apreciar el resultado.

Column 1	Column 2	Column 3	Column 4	Column 5
Paracetamol	2019-01-28	2	3.45	6.9
Vitaminas	2019-01-28	1	5.00	4.45
Ibuprofeno	2021-04-01	4	2.45	9.8
Gel de baño	2021-05-15	8	9.0	72.0
Colonia inf.	2020-10-21	2	5.0	10.0
Paracetamol	2021-01-17	7	3.45	24.15
Champú	2019-02-10	7	5.0	35.0
Colonia inf.	2020-09-21	1	6.00	6.0
Ibuprofeno	2020-10-21	10	2.45	24.5
Colonia inf.	2021-02-05	1	8.0	8.0
Champú	2020-10-20	8	5.0	40.0
Colonia inf.	2020-12-21	2	5.0	10.0
Paracetamol	2019-05-14	4	3.45	13.8
Gel de baño	2019-10-28	3	9.0	27.0
Colonia inf.	2020-09-21	2	8.0	16.0
Paracetamol	2020-05-18	7	3.45	24.15
Nutri	2020-08-25	2	1.35	11.25
Champú	2020-08-25	2	5.0	10.0
Nutri	2022-06-03	10	6.0	60.0

6.- JFrame debe ser así cuando se ejecute debe quedar como la segunda imagen. 1 punto

En el JFrame vamos a crear nuestra interfaz de usuario. Contara con los siguientes elementos:

Un label que contendrá el texto "Medicamentos" a modo de título.

Un label que contendrá el texto "Código:"

Un label que contendrá el texto "Nombre:"

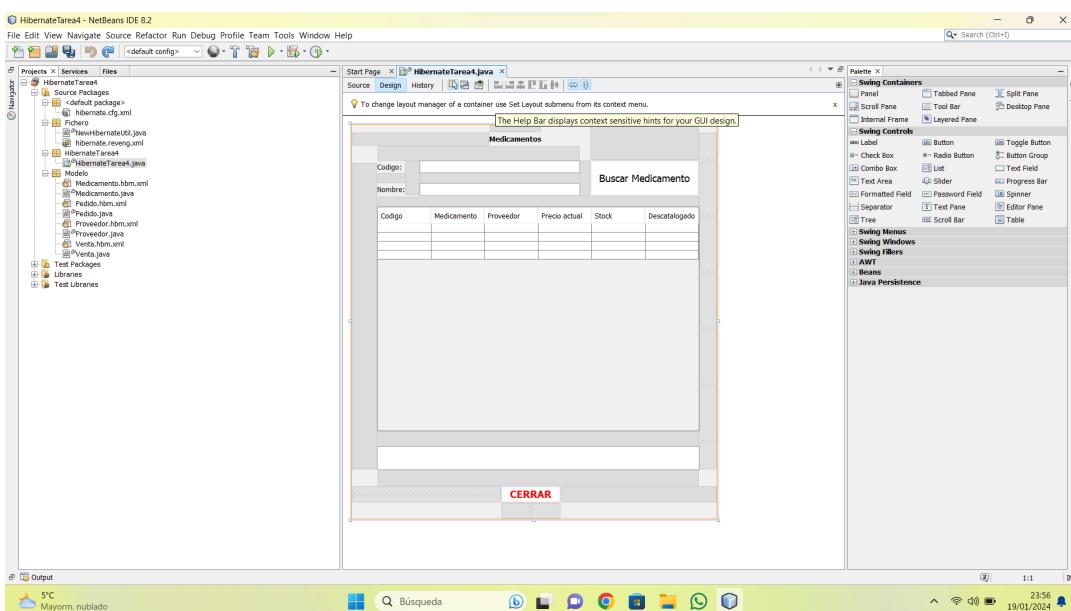
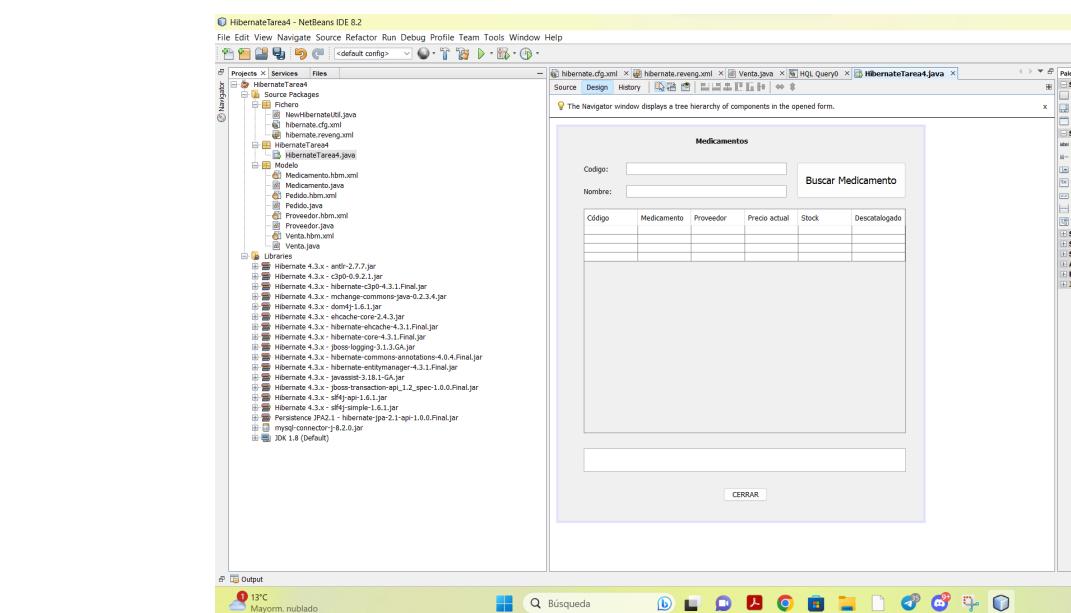
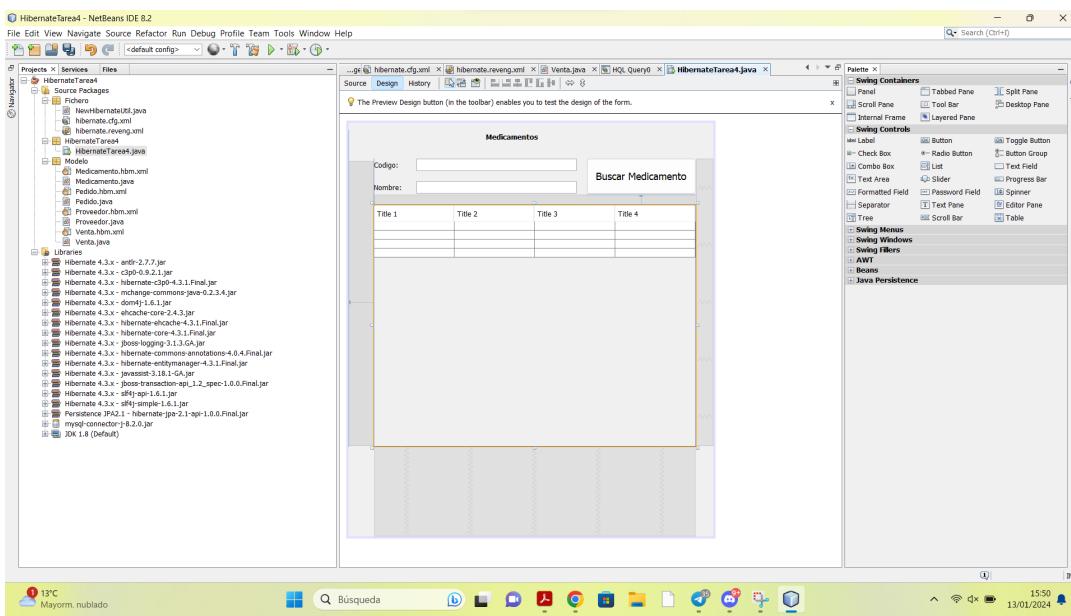
Dos text field, uno que ira al lado del label código para que el usuario introduzca el código del medicamento y otro al lado del label nombre para que introduzca el nombre del medicamento.

Un Button que tendrá la función de buscar en la base de datos y mostrar en una tabla los medicamentos que coincidan con los criterios de búsqueda del usuario.

Una tabla donde se mostraran los datos de los medicamentos, que estará compuesta por seis columnas llamadas "Código, Medicamento, Proveedor, Precio actual, Stock y Descatalogado".

Un text fiel que mostrara los mensajes necesarios de nuestra aplicación

Un botón que tendrá la función de cerrar la conexión y la aplicación. Contendrá el texto "CERRAR".



Como la tabla debe de estar desactivada al iniciar la aplicación, en el método constructor deshabilitamos la tabla y el header de la misma.

```

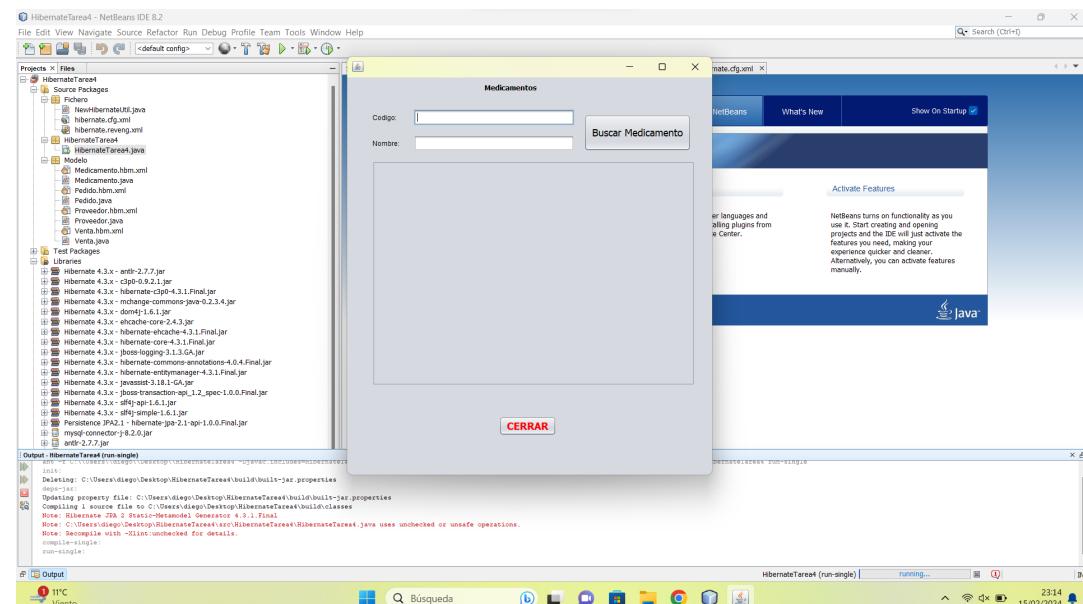
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package HibernateTarea4;

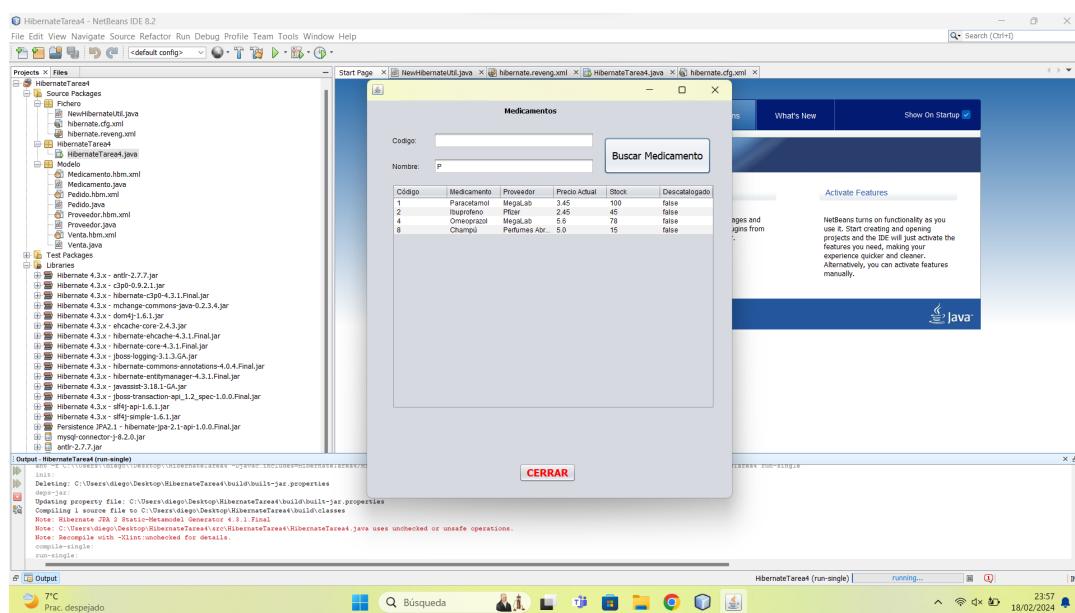
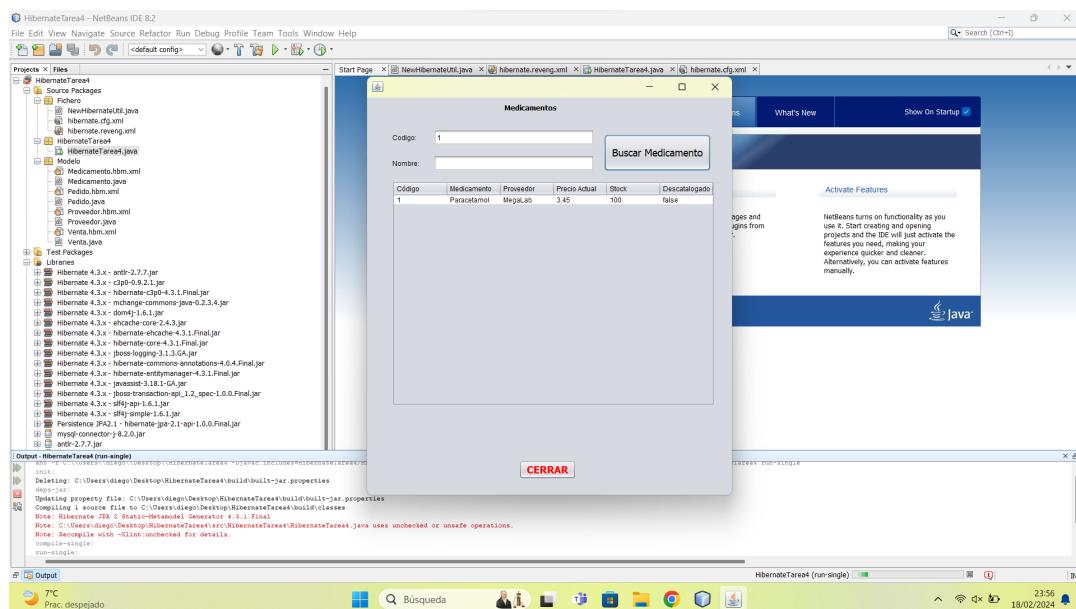
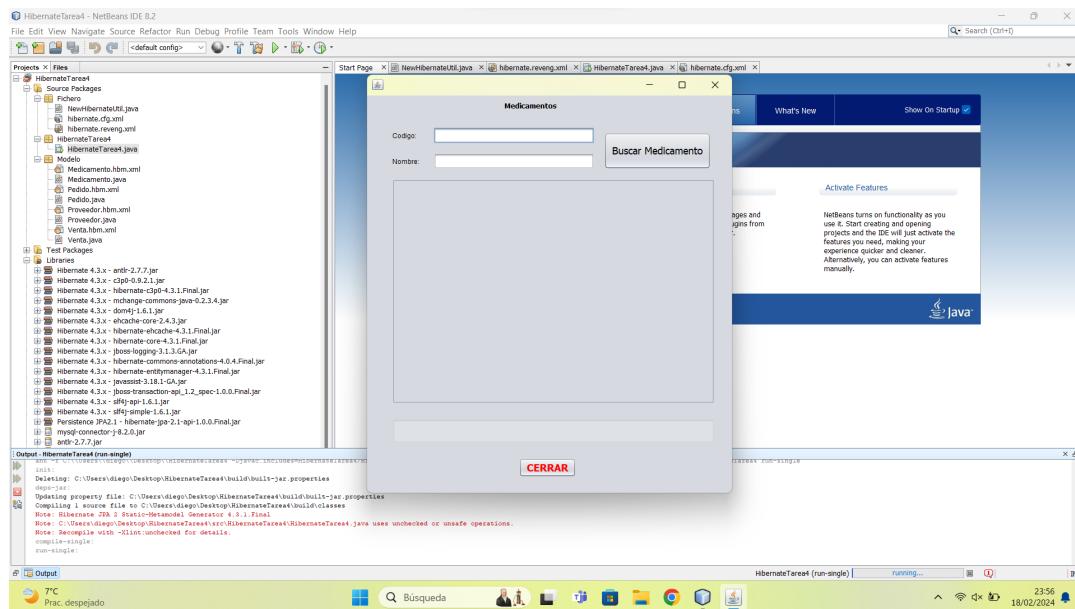
/**
 * Section diego
 */
public class HibernateTarea4 extends javax.swing.JFrame {

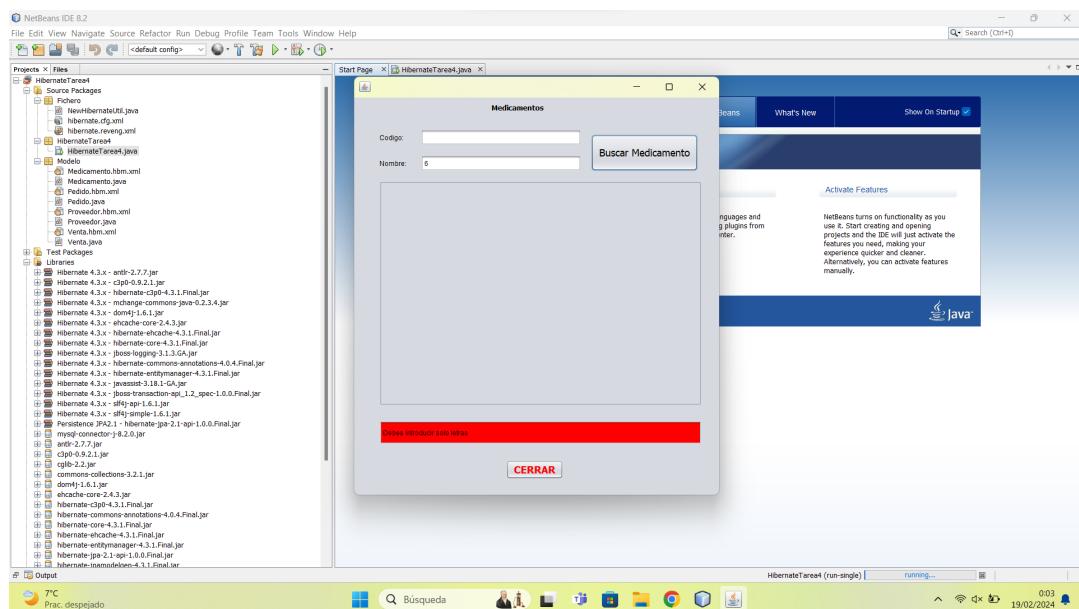
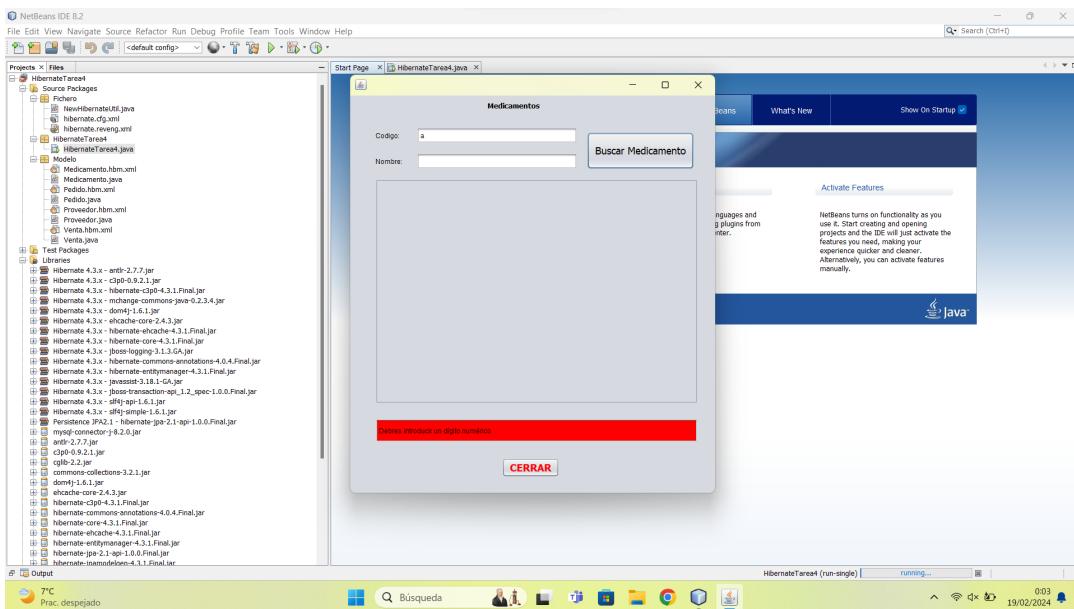
    /**
     * Creates new form HibernateTarea4
     */
    public HibernateTarea4() {
        initComponents();
        jTableTable.setVisible(false);
        jTableTable.getTableHeader().setVisible(false);
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // Generated Code
    //
```

En las siguientes imágenes podemos ver el funcionamiento, como muestra los resultados al buscar por código o por nombre y muestra en el jTextField las excepciones en caso de no encontrar medicamentos o de introducir algún dato erróneo a la hora de usar el buscador.







Y con esto damos por finalizada la tarea.