



Actividad [#2] - [Interfaz del Sistema]

[Lenguajes de Programación IV]

Ingeniería en Desarrollo de Software

Tutor: Juan Miguel Martínez Maldonado

Alumno: Alan David López Rojas

Fecha: 19/07/2022

Índice

Investigación.....	pág. 3
Interfaz.....	pág. 6
Codificación.....	pág. 8
Conclusión.....	pág. 13
Bibliografía.....	pág. 14

Investigación

JAVA 8

Como ya se había visto anteriormente en la unidad 1, Java es una plataforma informática de lenguaje de programación creada por Sun Microsystems en 1995, Java 8 es la versión más reciente de Java tiene características, mejoras y correcciones de bugs para mejorar la eficacia en el desarrollo y la ejecución de programas Java, esta nueva versión está muy orientada a la movilidad y al Internet de las cosas. Uno de los beneficios de contar con una plataforma Java 8 embebido en dispositivos, escritorios, centro de datos y en la nube, es que los usuarios podrán desplegar aplicaciones de forma más rápida, así como procesar y analizar datos en tiempo real y actuar en circunstancias tan pronto como estas ocurran.

Algunas mejoras que se incluyen en Java 8:

- Mejora la productividad del desarrollador y provoca importantes incrementos del rendimiento de apps mediante la reducción de código repetitivo, la mejora de las colecciones y anotaciones y la simplificación de modelos de programación.
- Métodos de extensión virtual y expresión Lambda
- Una de las funciones destacables de Java SE 8 es la implantación de expresiones Lambda y funciones adyacentes a la plataforma y el lenguaje de programación Java.
- API de fecha y hora
- Esta nueva API permitirá a los administradores gestionar datos de fecha y hora de forma mucho más natural y fácil de comprender.
- Motor de JavaScript Nashhorn
- Nueva implantación ligera de alto rendimiento del motor de JavaScript integrada en JDK y disponible en las aplicaciones Java mediante las API existentes.
- Seguridad mejorada
- Sustitución de la lista de métodos sensibles al emisor mantenida a mano existente por un mecanismo que identifica con mayor precisión dichos métodos y permite detectar a los emisores de forma fiable.
- Con respecto a la compatibilidad de Java SE 8 con previas versiones, esta preserva el set de desarrollo actual de Java y ayuda a proteger la inversión en la tecnología.

MySQL

Es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca. MySQL está disponible para múltiples plataformas, algunas de sus características son

Seguridad

- Maneja un sistema de privilegios muy seguro, la verificación se hace basado en host.

Escalabilidad y Límites

- Maneja base de datos grandes. Su uso se extiende a más de 50 millones de registros. Se tiene conocimiento de algunos usuarios que usan el servidor MySQL con más de 60,000 tablas y cerca de

5.000.000.000 de filas.

- Hasta 64 índices por tabla son permitidos. Cada índice puede consistir de 1 a 16 columnas.

Conectividad

- Los clientes pueden conectarse al servidor MySQL usando TCP/IP sobre cualquier plataforma.
- El conector/ODBC (MyODBC) provee soporte a programas cliente que usen ODBC (Open Database Connectivity).
- La interfase conectora/J provee soporte para programas cliente java que usan JDBC.

Inicialmente, MySQL carecía de algunos elementos esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de esto, atrajo a los desarrolladores de páginas web con contenido dinámico, debido a su simplicidad, de tal manera que los elementos faltantes fueron complementados por la vía de las aplicaciones que la utilizan. Poco a poco estos elementos faltantes, están siendo incorporados tanto por desarrolladores internos, como por desarrolladores de software libre

¿Cómo se realiza una conexión de Java 8 a MySQL?

Para conectar la aplicación Java a la base de datos Mysql, debemos seguir algunos pasos que se enumeran a continuación:

- Descargue e instale MySQL
- Crea una base de datos en MySQL
- Descargue el controlador JDBC y colóquelo en el classpath
- Escribe el código Java para la conectividad.
- Prueba la conexión

Descargar e instalar la base de datos MySQL

MySQL es un sistema de administración de bases de datos. Pasemos al siguiente paso.

Crear una base de datos en MySQL

Ahora, crearemos una base de datos para que podamos probar la conexión. Para crear una base de datos en MySQL, use la consulta SQL a continuación:

```
create database BancoppelDB;
```

usará en la parte de conectividad, después de crear una base de datos

Se descarga el controlador JDBC e inclúye en Classpath

El controlador JDBC es un archivo JAR proporcionado por MySQL; es un conector que actúa como puente entre MySQL y las aplicaciones Java. Para descargar el archivo JAR se obtiene en el sitio oficial de MySQL y coloque los archivos en la carpeta lib del proyecto java. Ahora, se continúa con el siguiente paso.

Conectarse con MySQL

Después de completar los procedimientos anteriores, se escribe el código Java para la conectividad. Aquí, usamos el método `class.forName()` para cargar el JDBC Driver, que descargamos del sitio oficial de MySQL.

El método `getConnection()` se utiliza para pasar la cadena de conexión: `MySQL:Port/Database/,username,dbpassword`. Esta cadena se utiliza para autenticar al usuario y proporcionar acceso solo a usuarios autorizados. Después de eso, usamos el método `createStatement()` para crear una instancia que se usará para ejecutar consultas SQL usando el código. Vea el ejemplo a continuación:

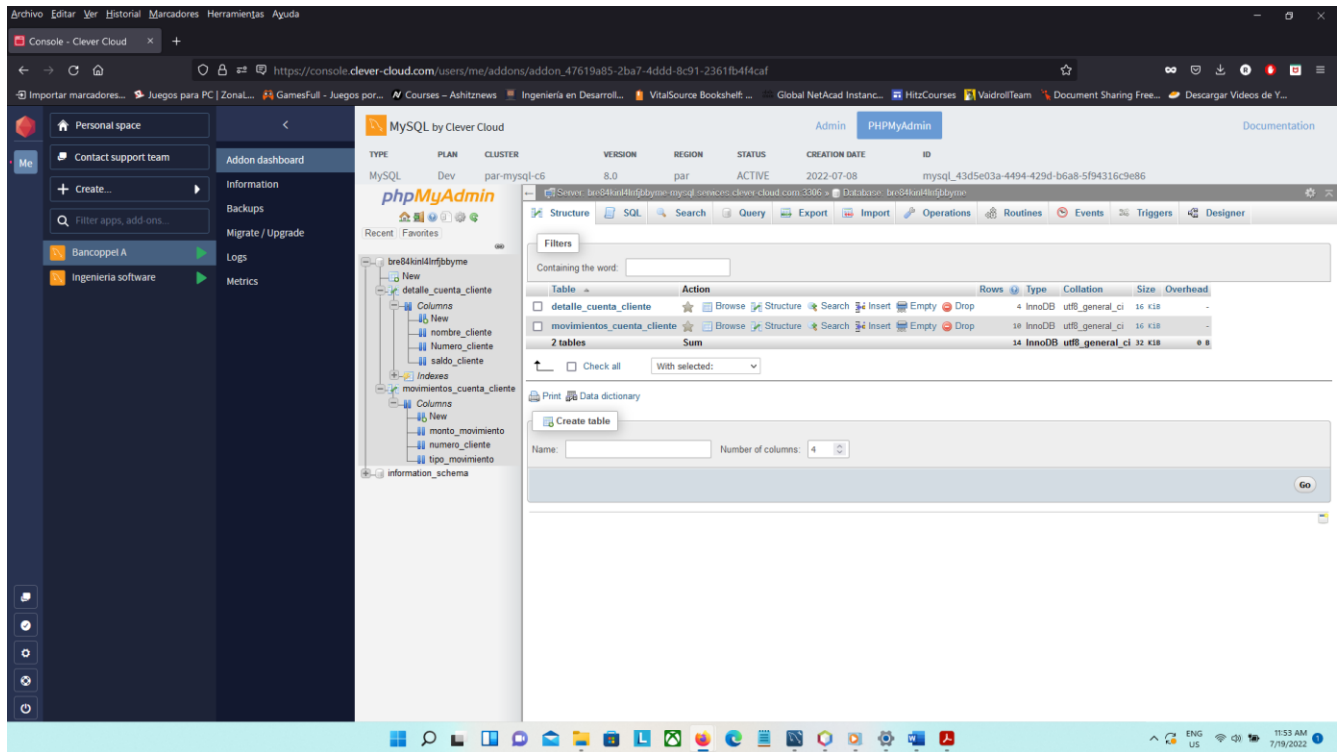
```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
public class SimpleTesting{
    public static void main(String[] args) {
        try
        {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con=DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/delftstackDB","username","dbPassword");
            Statement stmt=con.createStatement();
            ResultSet rs=stmt.executeQuery("show databases;");
            System.out.println("Connected");
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}
```

Probar la conexión JDBC

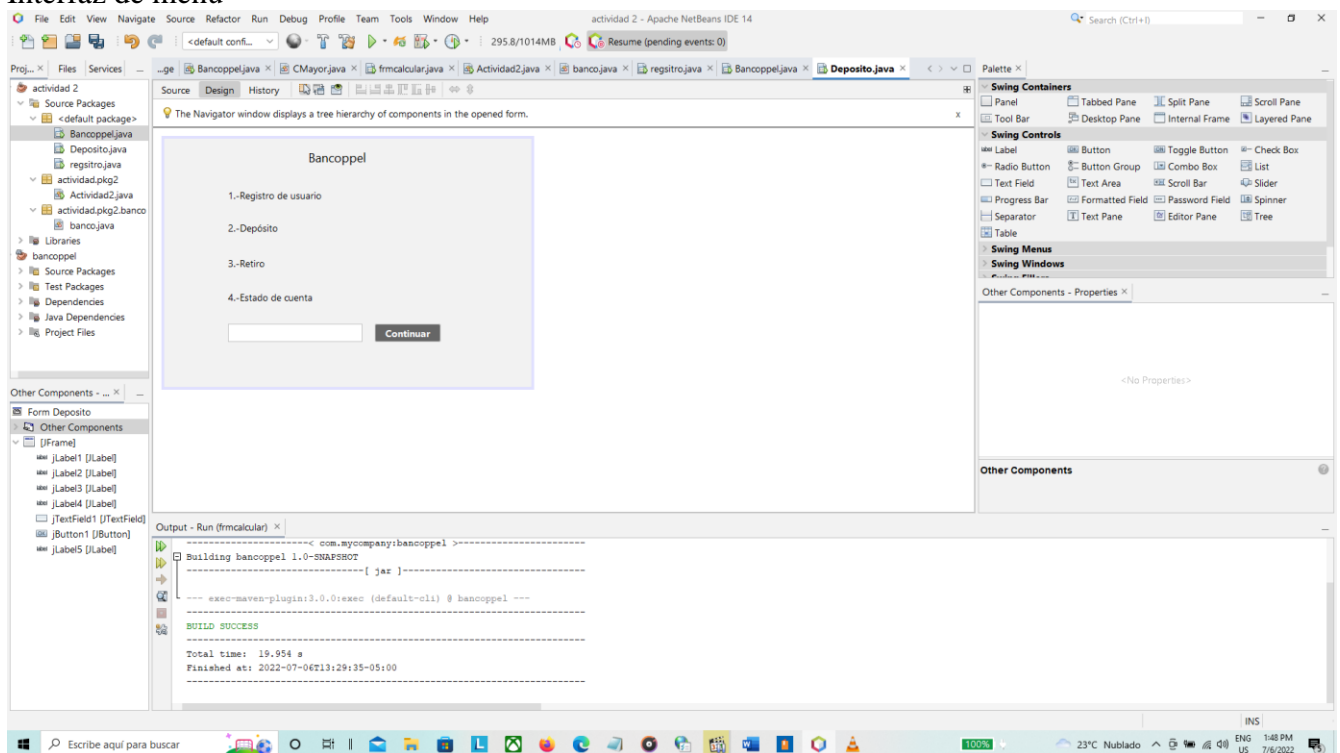
Después de escribir el código, solo se debe ejecutar. Si el código funciona bien, se obtendrá el siguiente resultado en la consola: `Connected`

Interfaz

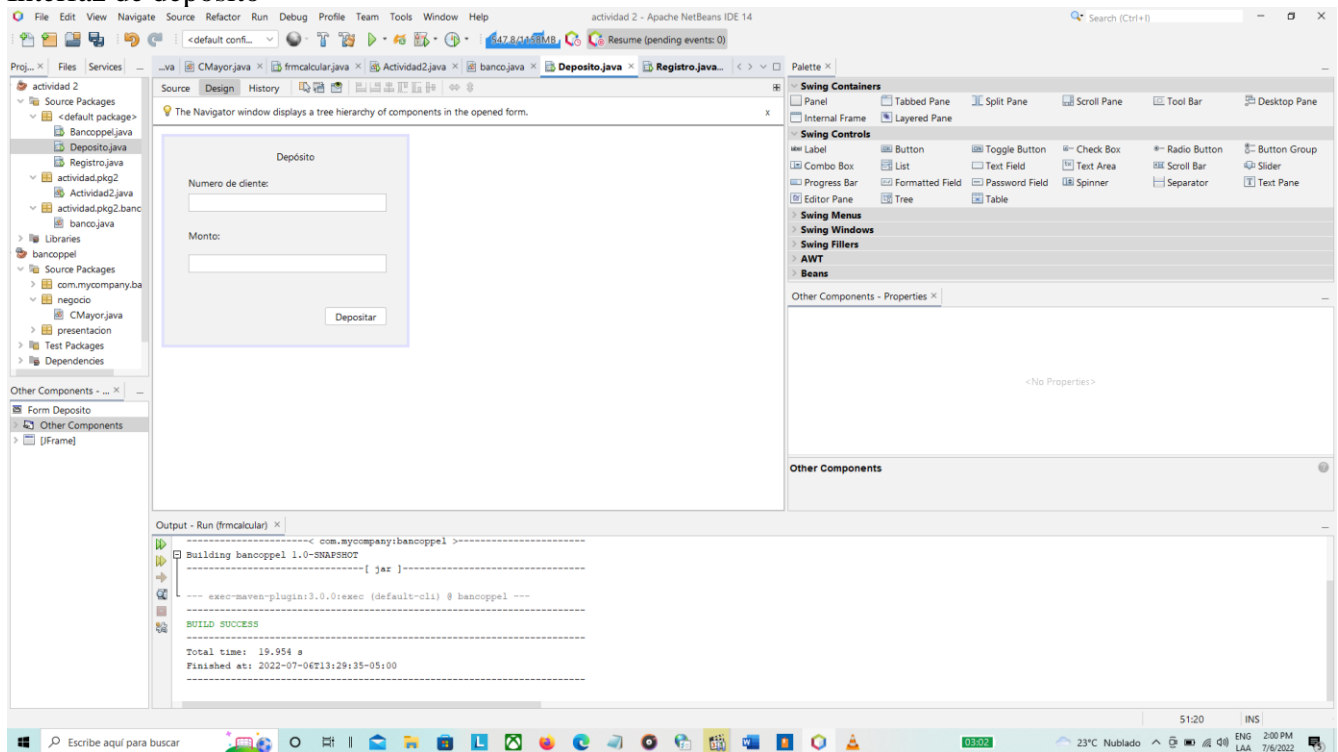
Base de datos creada en la nube



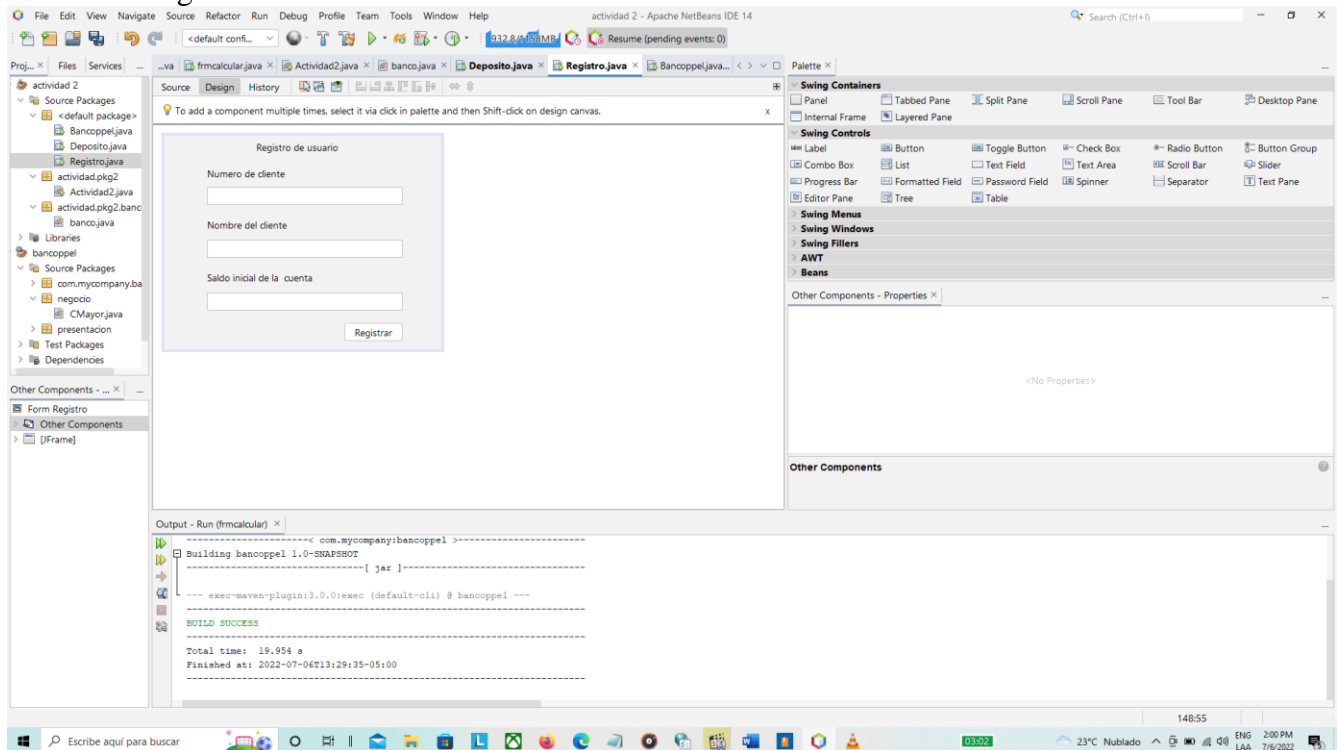
Interfaz de menú



Interfaz de deposito

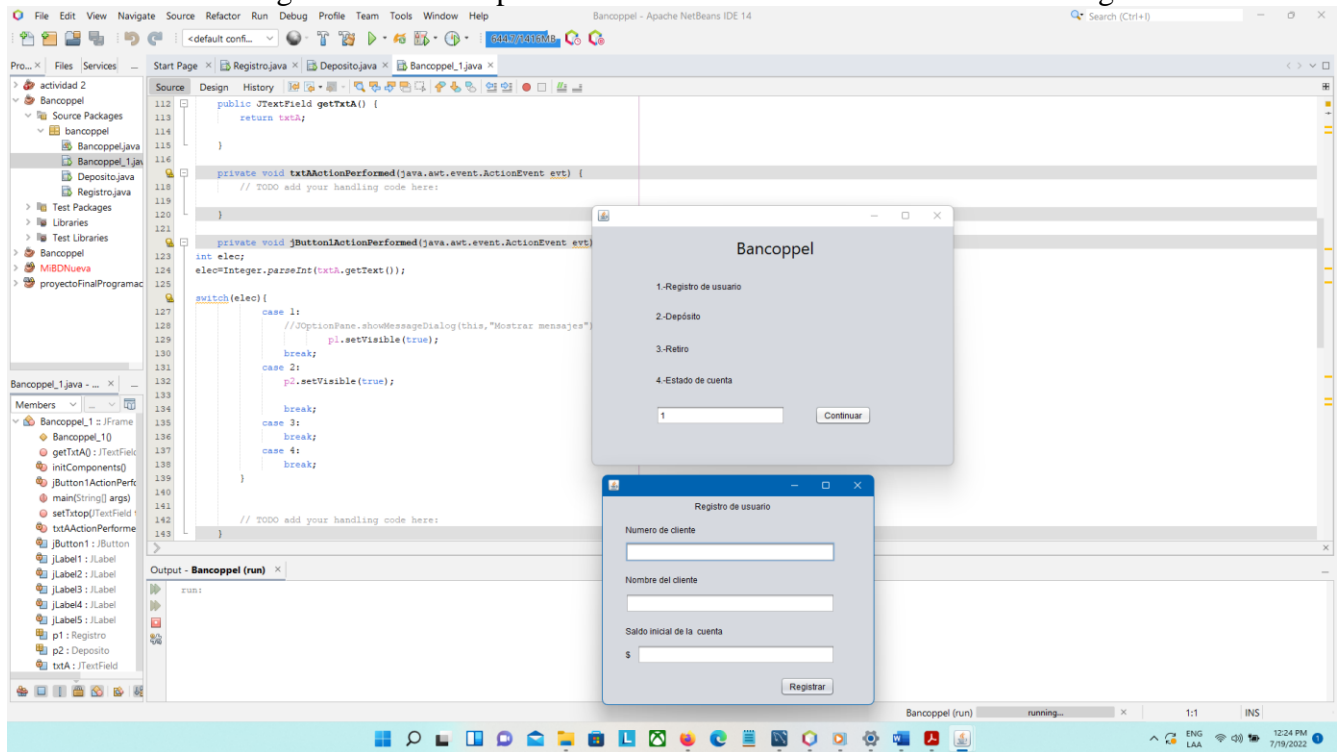


Interfaz de registro de clientes

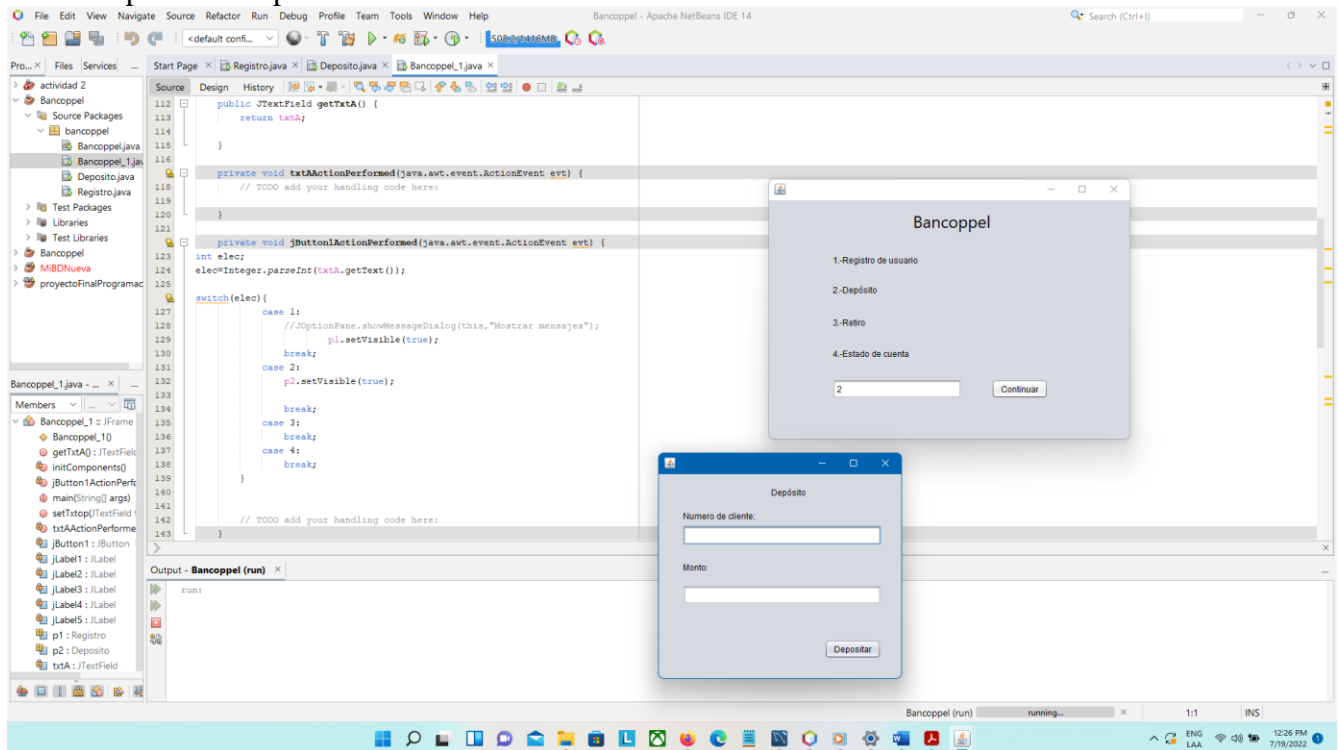


Codificación

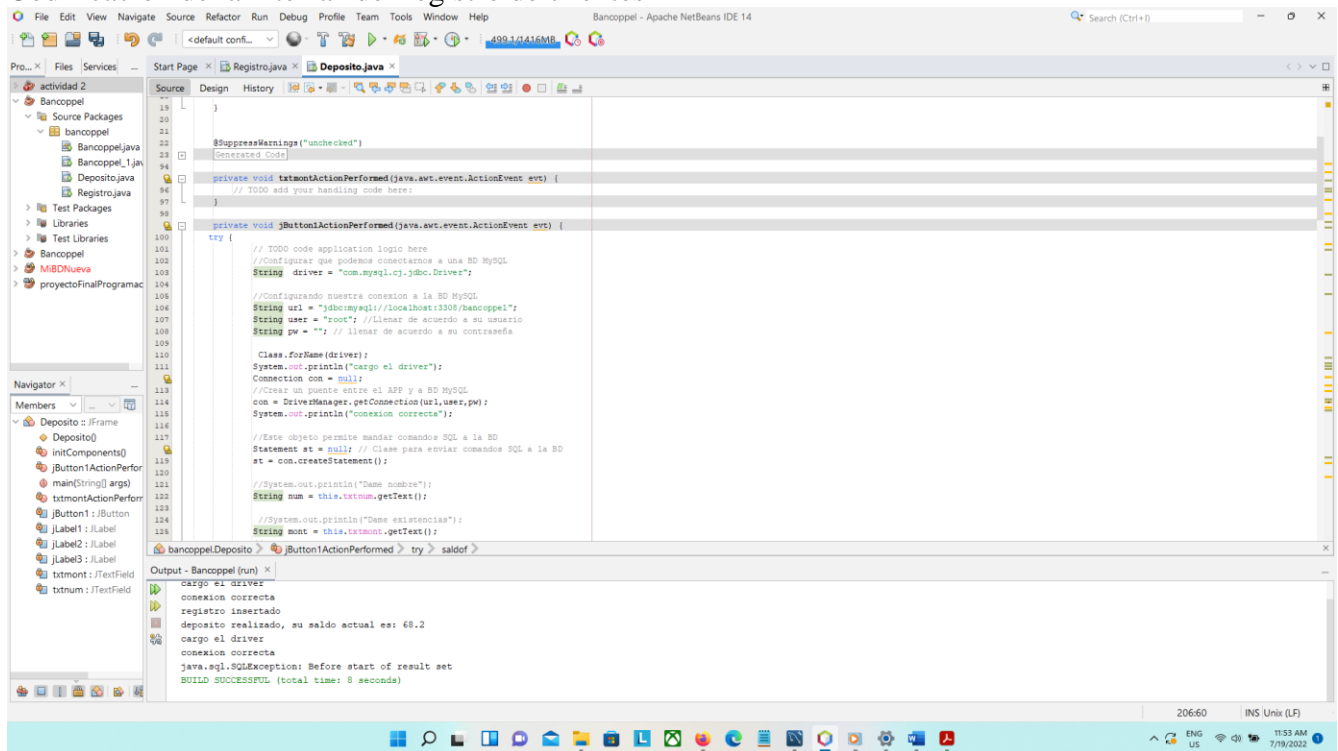
Menú de inicio al elegir una de las 4 opciones abre la ventana deseada en este caso registro de clientes



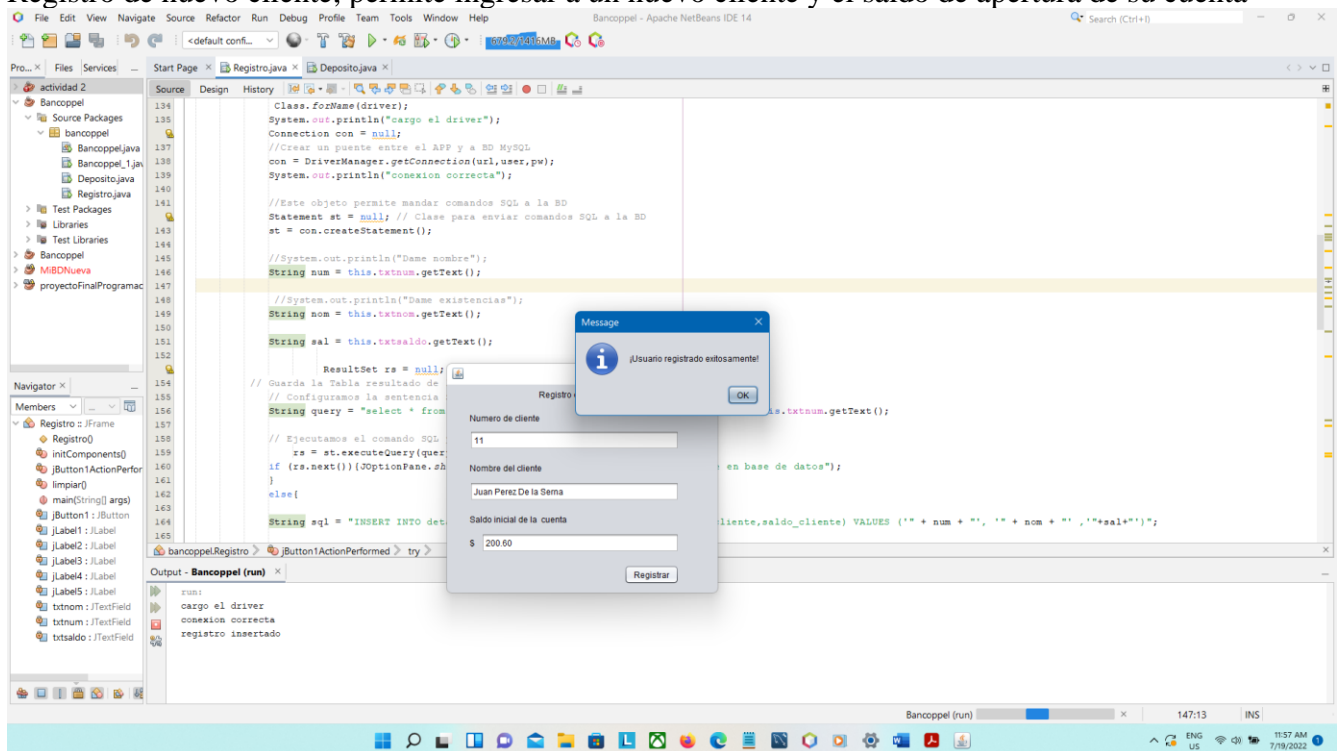
Abre la opción de deposito



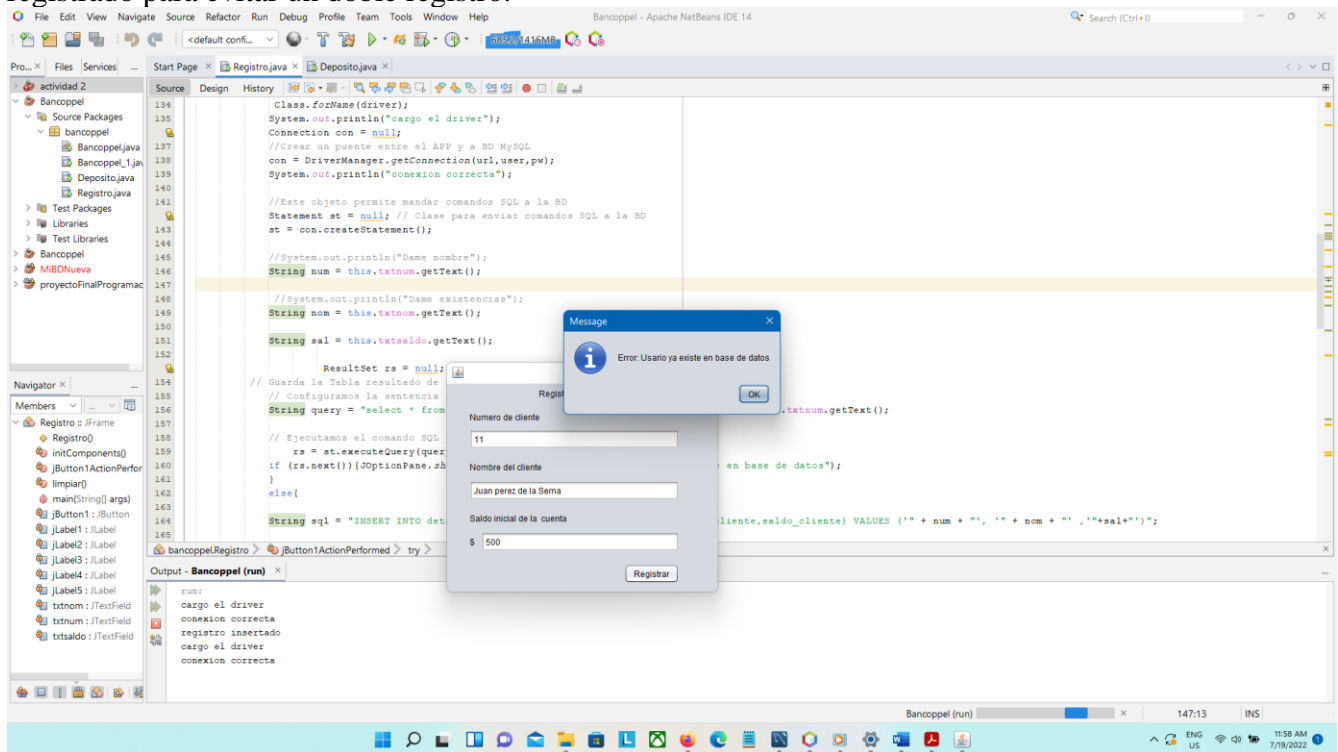
Codificación de la interfaz de Registro de clientes



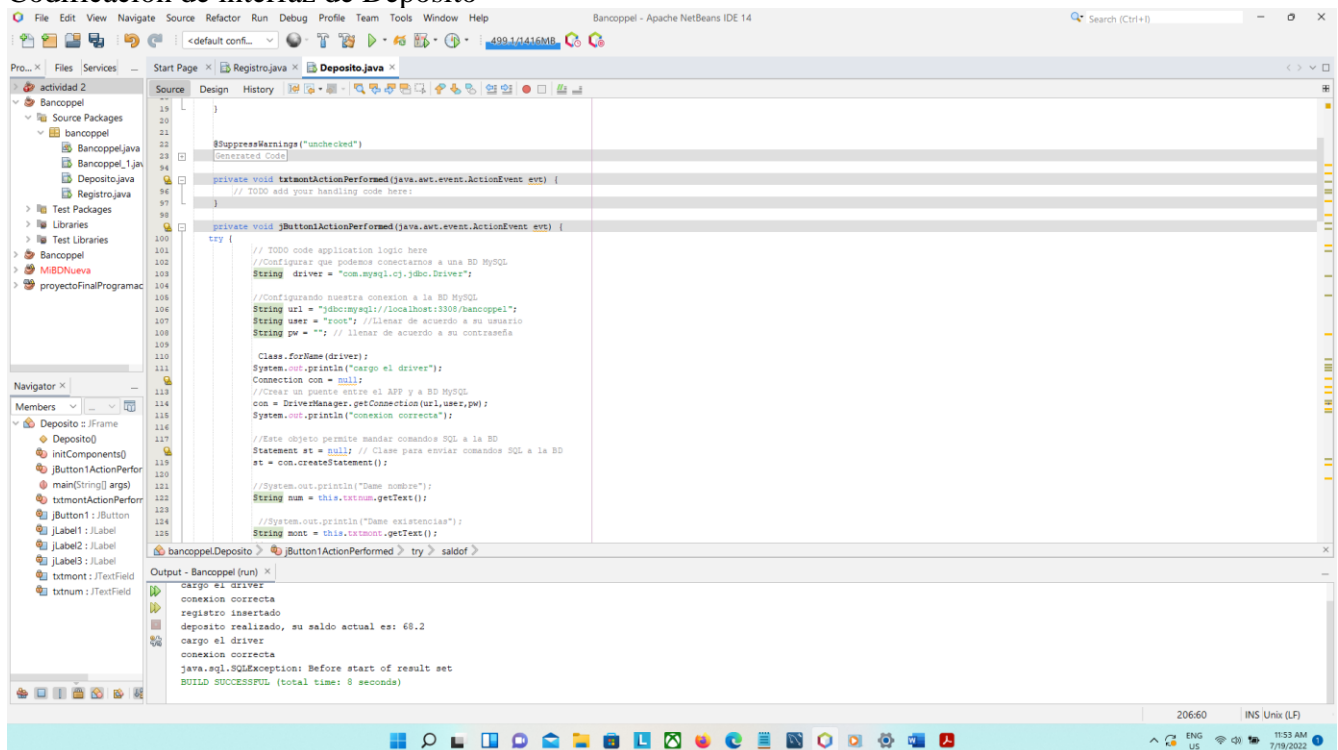
Registro de nuevo cliente, permite ingresar a un nuevo cliente y el saldo de apertura de su cuenta



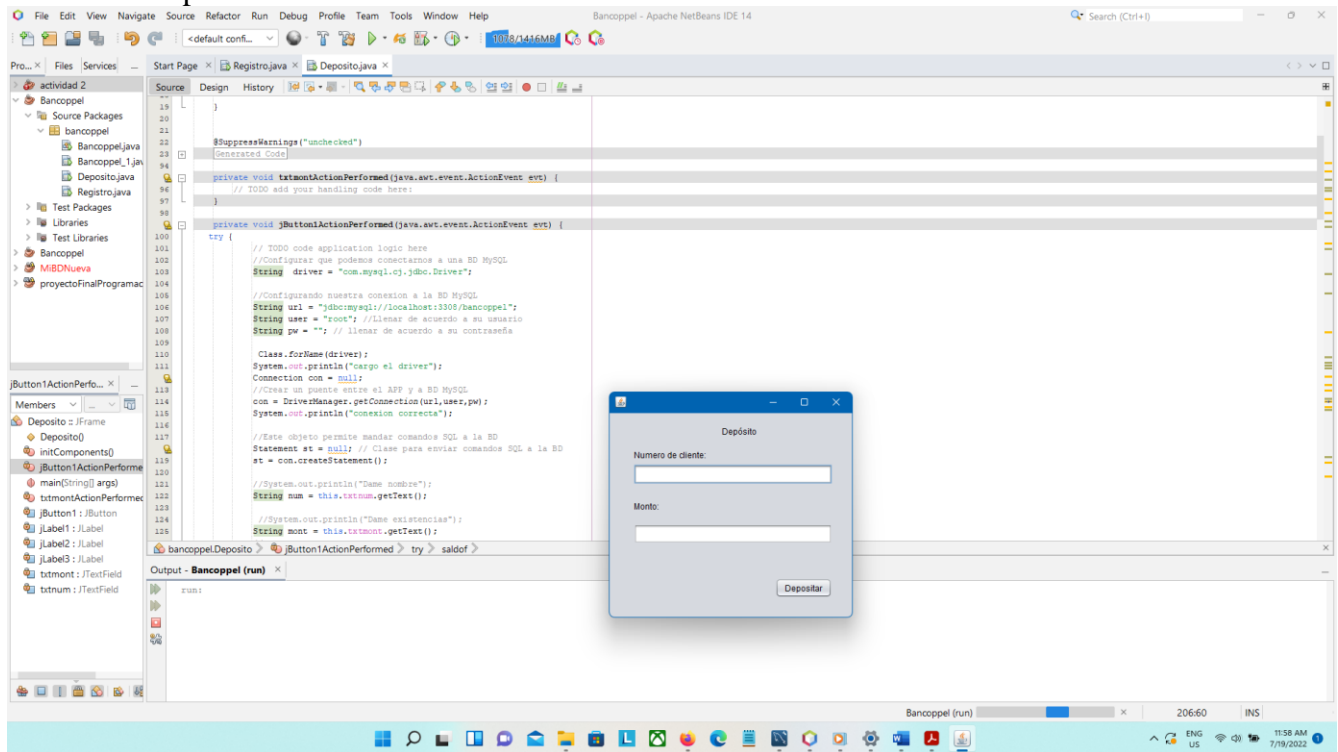
En caso de que el cliente ya este registrado en la base de datos nos arroja el mensaje de error usuario ya registrado para evitar un doble registro.



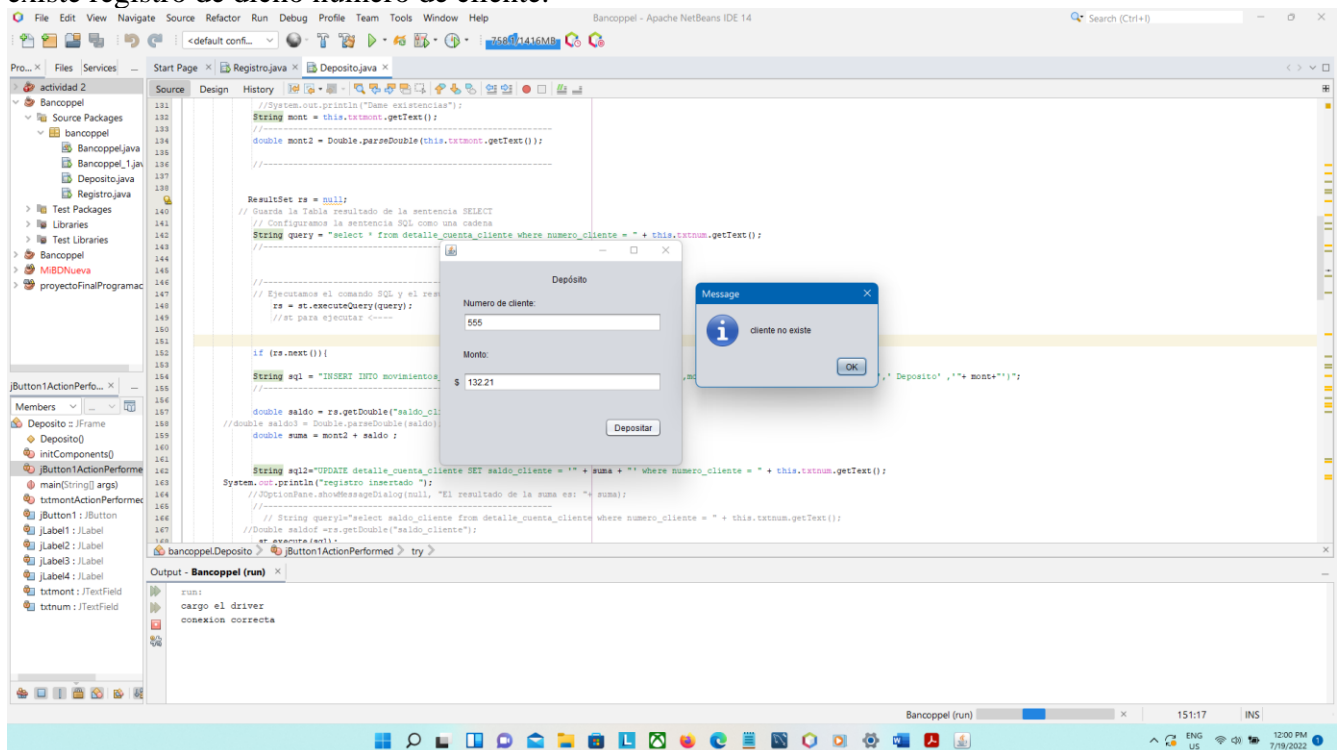
Codificación de interfaz de Deposito



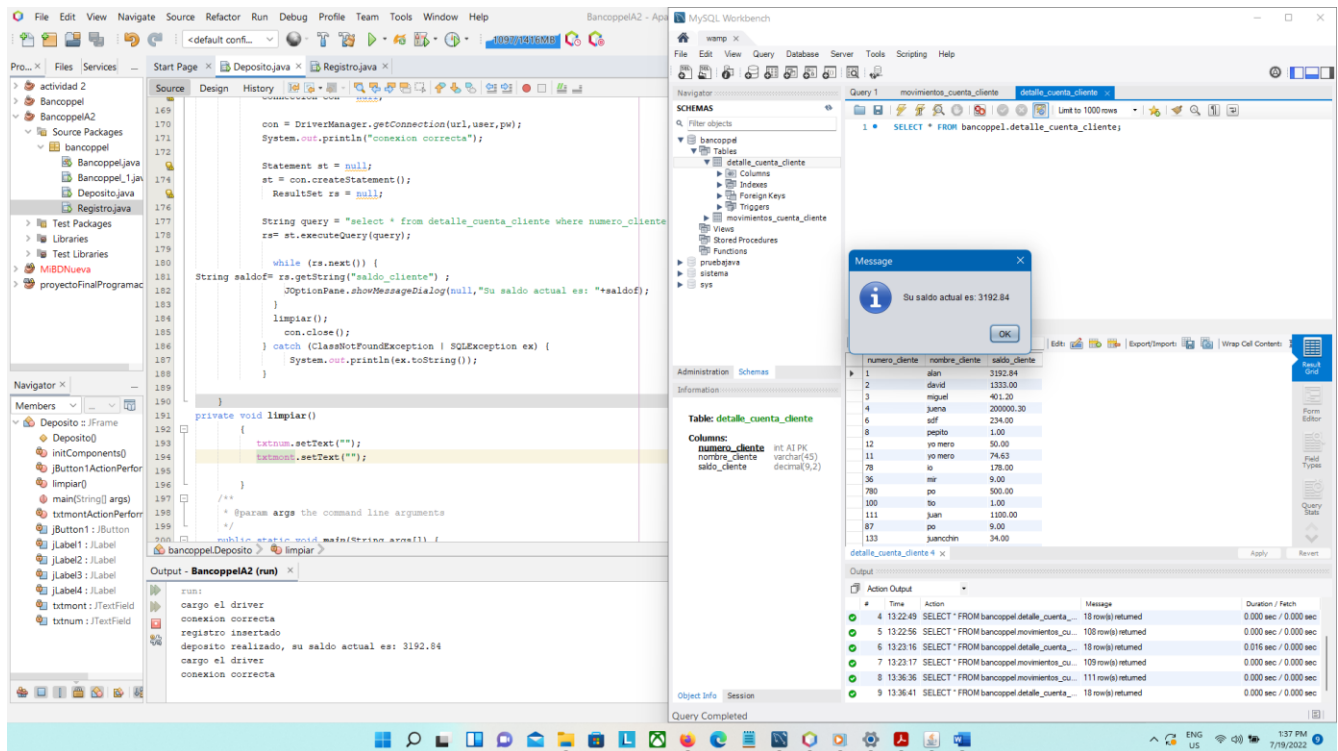
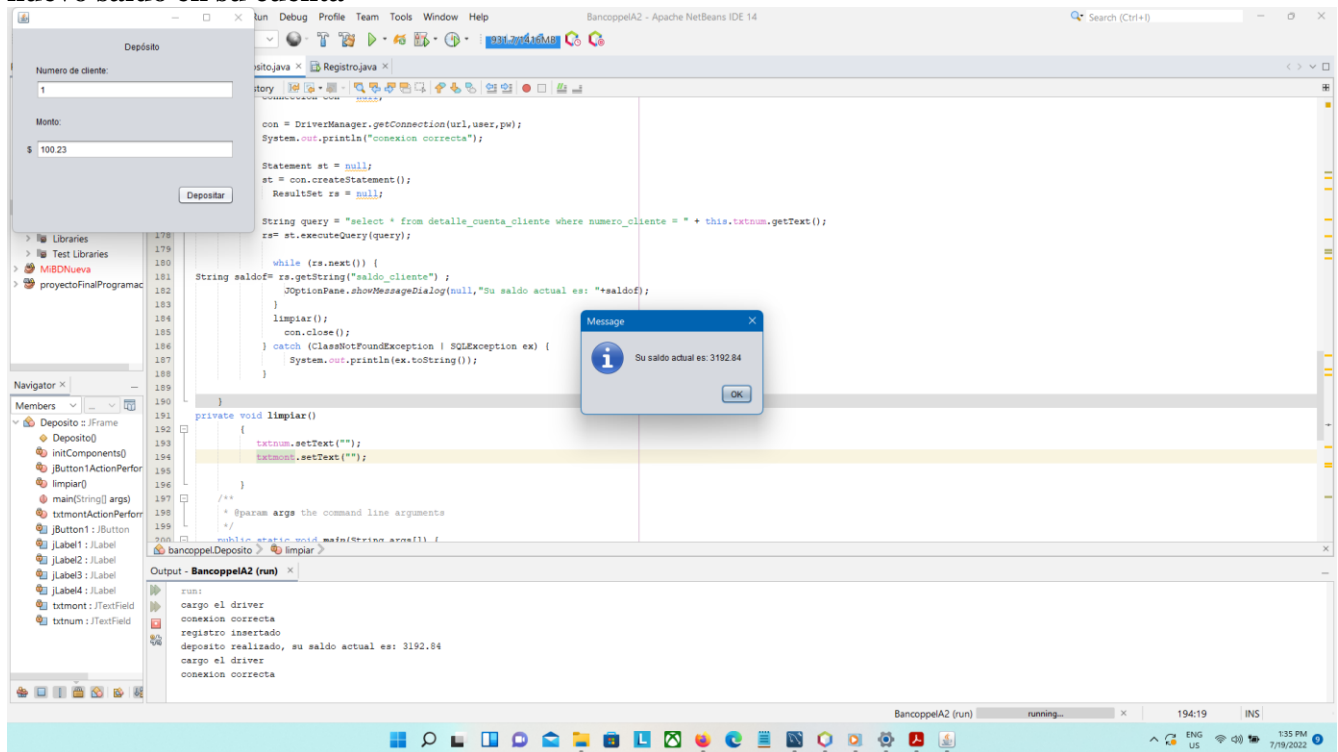
Interfaz de deposito a cuenta de cliente



En esta interfaz de deposito primero verifica que el numero de cliente exista en la base de datos para poder hacer el deposito a dicha cuenta, en caso de no existir manda un mensaje advirtiendole que no existe registro de dicho número de cliente.



Si encuentra el registro del cliente hace el deposito a el saldo que tenia y arroja un mensaje con el nuevo saldo en su cuenta



Conclusión

En esta segunda parte de la actividad pude conocer mas de la forma de programar con el lenguaje de JAVA y como usarlo para trabajar en conjunto con la base de datos de MySQL, si bien es un poco complicado al inicio solo es cuestión de aplicar lo aprendido en las anteriores materias de lenguajes de programación para aplicarlo, solo cambian algunos aspectos de la sintaxis.

En cuanto a NetBeans es una herramienta IDE que facilita mucho el trabajo de la programación ya que ayuda a ver muchos errores en las sentencias o sintaxis que no podríamos percatarnos hasta hacer compilar el código, de igual manera facilita hacer las sentencias en casos como los getters y setters. En el mundo profesional las ventajas que pude investigar son el desarrollo de cualquier tipo de software y en sus distintas fases. A los programadores les sirve como medio para la escritura, compilación o, simplemente, para hallar errores en programas basados en los anteriores lenguajes. Es útil para ejecutar programas en diferentes dispositivos electrónicos, tanto móviles como PC. Asimismo, NetBeans permite a través de Bookmarking marcar las líneas de código que a posteriori se modificarán.

Por otro lado, proporciona una librería visual donde conseguir los diferentes widgets y permite gestionar el almacenamiento de los datos de una forma óptima.

Estas tres herramientas permiten el desarrollo de muchas aplicaciones y son de fácil acceso, en general solo pude ver una pequeña parte superficial de lo que es trabajar con estas aplicaciones, pero me ha ayudado a ver un poco el panorama de trabajo y a que área me puedo enfocar en un futuro como desarrollador software.

Nota (Dejo enlace de Google drive con el proyecto para descargar y evaluar, como respaldo de la actividad, las capturas y pruebas en este archivo se hicieron con MySQL workbench pero el archivo enviado esta programado para trabajar con la base de datos en la nube por eso no envió el respaldo de la base de datos.)

<https://drive.google.com/file/d/1RbTy4ivXm6eSvORFj2dGdZPgWNeyiScR/view?usp=sharing>

Bibliografía

(Barnes, 2007)

Barnes, D. J. (2007).

Programación Orientada a Objetos con Java. Una Introducción Práctica Usando BlueJ . [[VitalSource Bookshelf version]]. Retrieved from vbk://9788483223505