


Actividad 3 -Programa Banco Mexicano Pt. 2

Lenguajes de Programación IV Ingeniería en Desarrollo de Software

Tutor: Aaron Iván Salazar Macias

Alumno: Homero Ramirez Hurtado

Fecha: 25 de Julio del 2023



Índice.

. Introducción.

. Descripción.

. Justificación.

. Desarrollo.

- Interfaz.
- Codificación.

. Conclusión.

. Referencias.



Introducción.

A continuación, retomaremos parte de la actividad 2 para poder finalizar la interfaz del cajero del banco mexicano con ello realizaremos una base de datos para poder hacer la conexión con MySQL en la cual podremos consultar saldo, depositar e incluso retirar dinero con ello debemos realizar la conexión entre ambos programas y codificarlo para que este funcione de acuerdo a lo que se pide en esta última actividad.

Descripción.

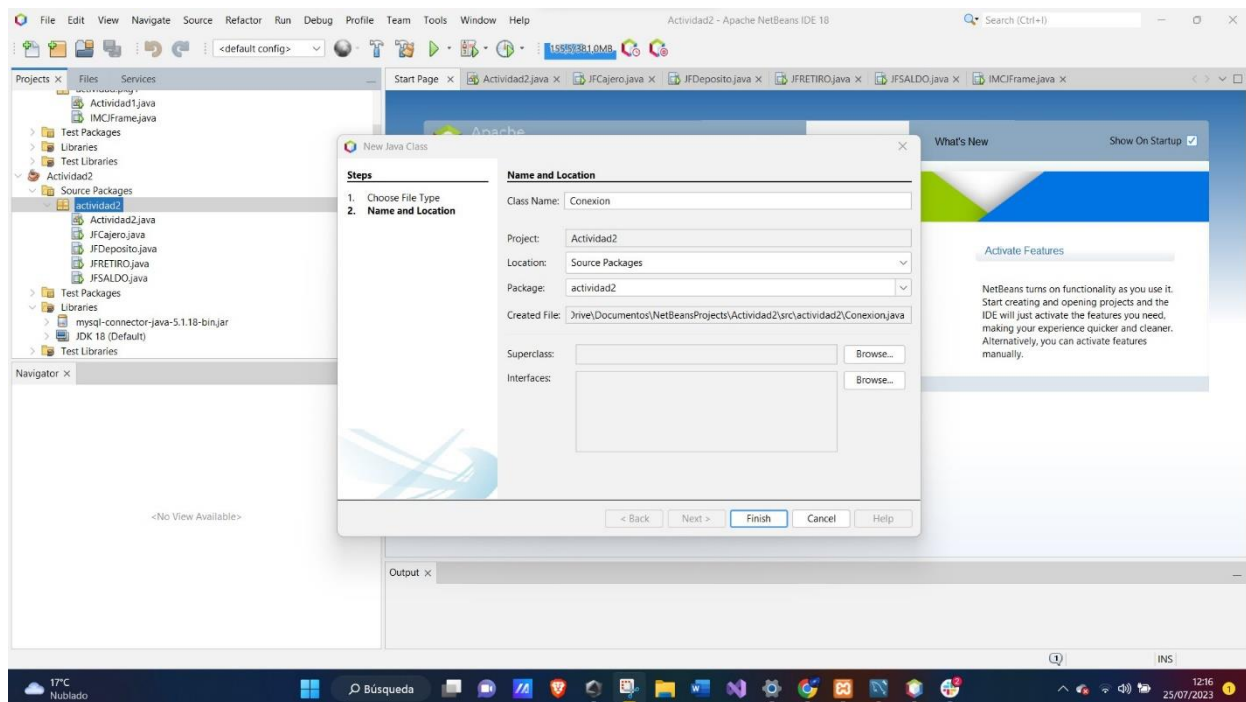
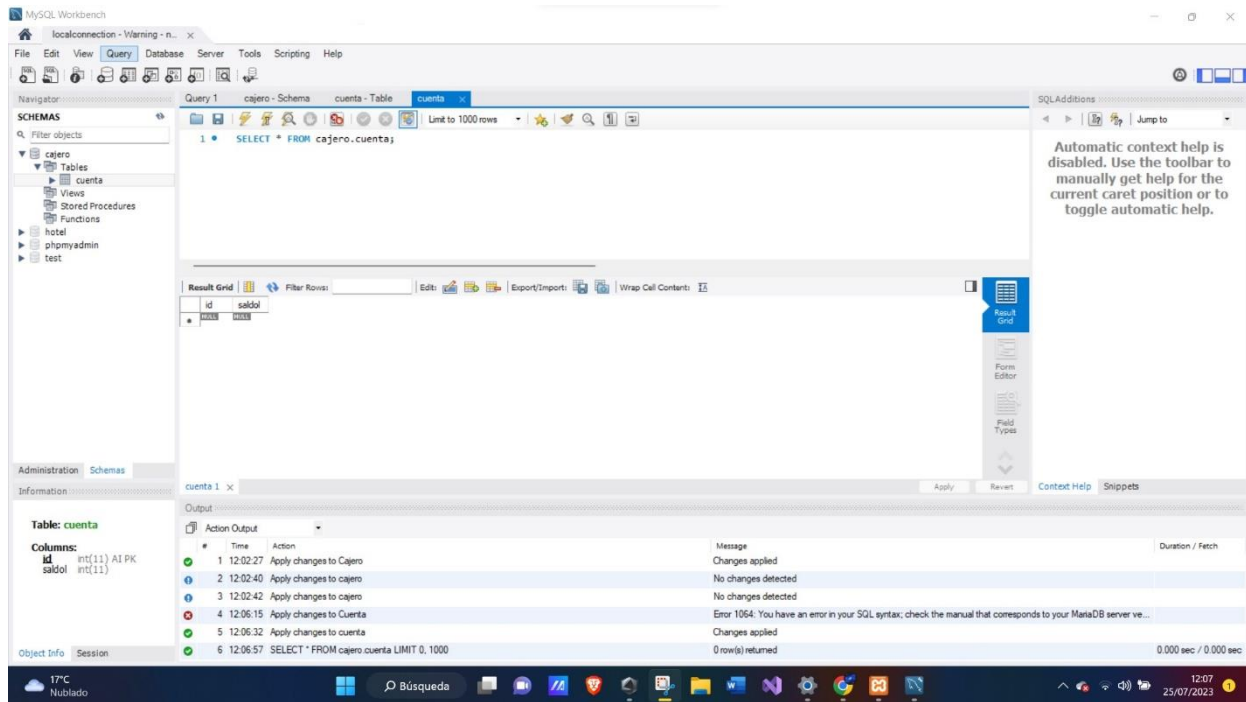
Los clientes de Banco Mexicano necesitan un programa que les permita a sus clientes el realizar depósitos, retiros y consultas de su saldo. Por lo que necesitan que un ingeniero en desarrollo de software genere una base de datos que atienda a esta necesidad. Con la información anterior se concluirá la interfaz de la actividad 2 para de esta forma poder entregar el trabajo o sea el producto final al banco mexicano para que pueda instalarlo en los cajeros y de esta manera poder satisfacer las necesidades de sus clientes.

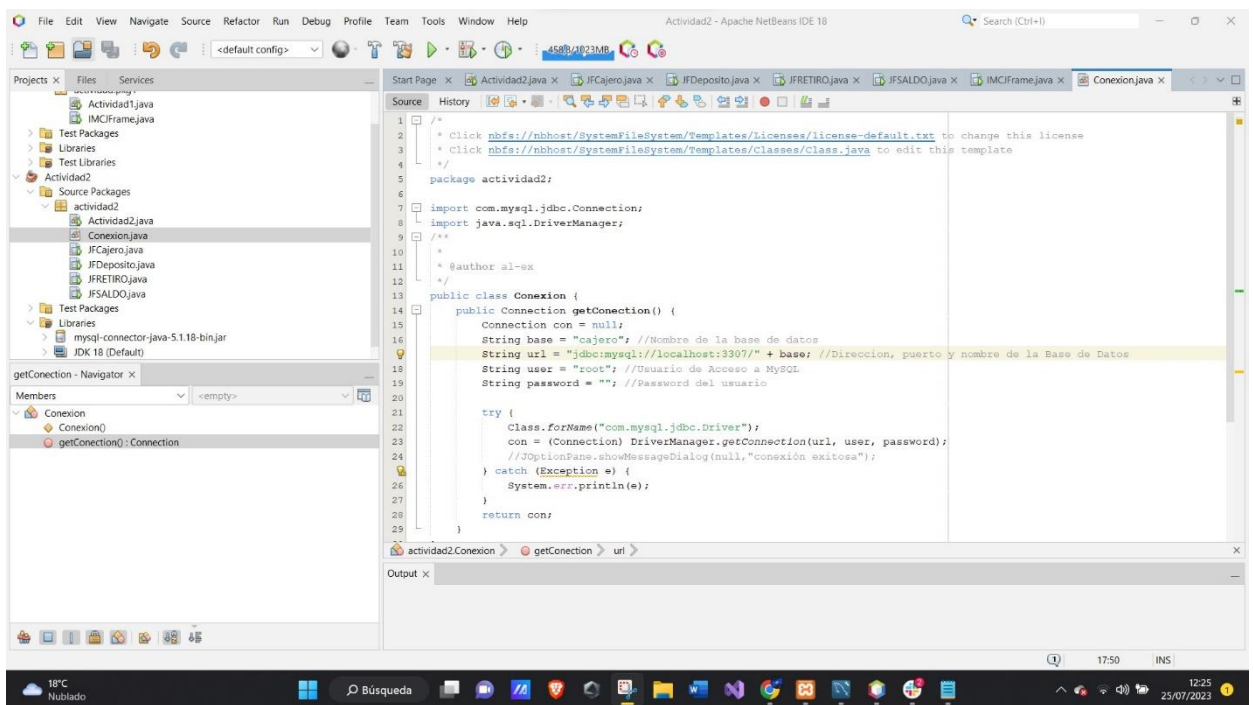
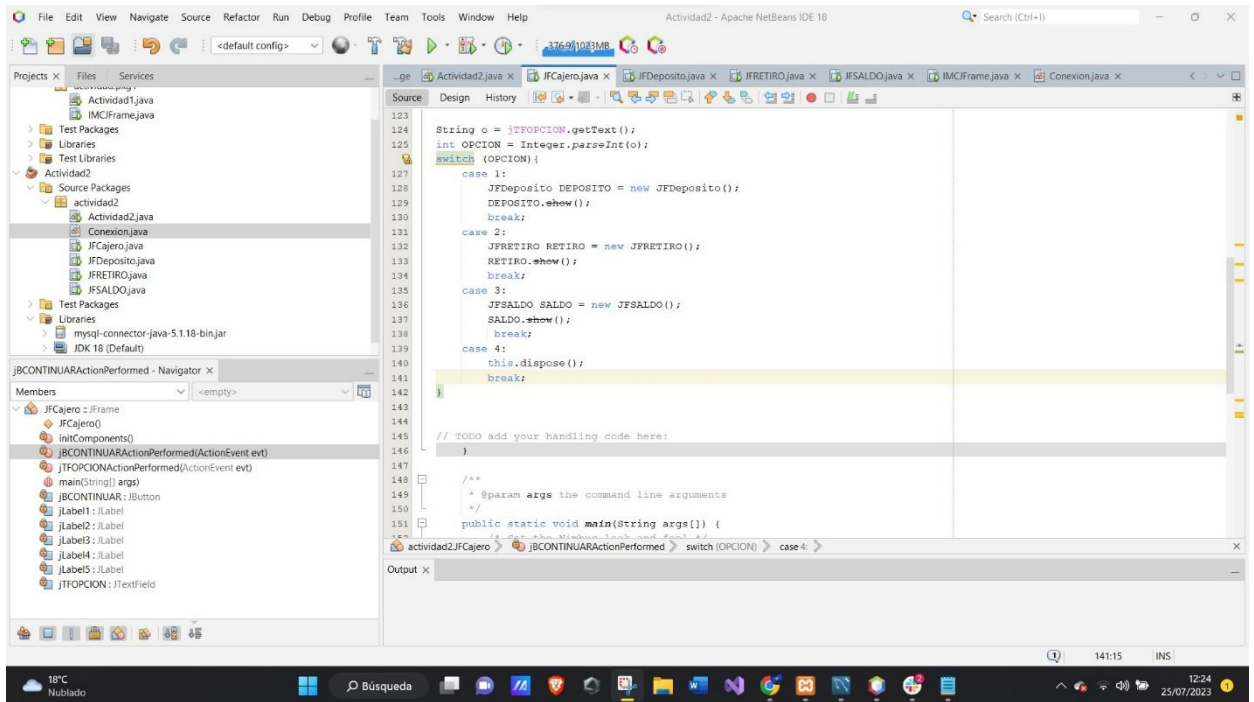
Justificación.

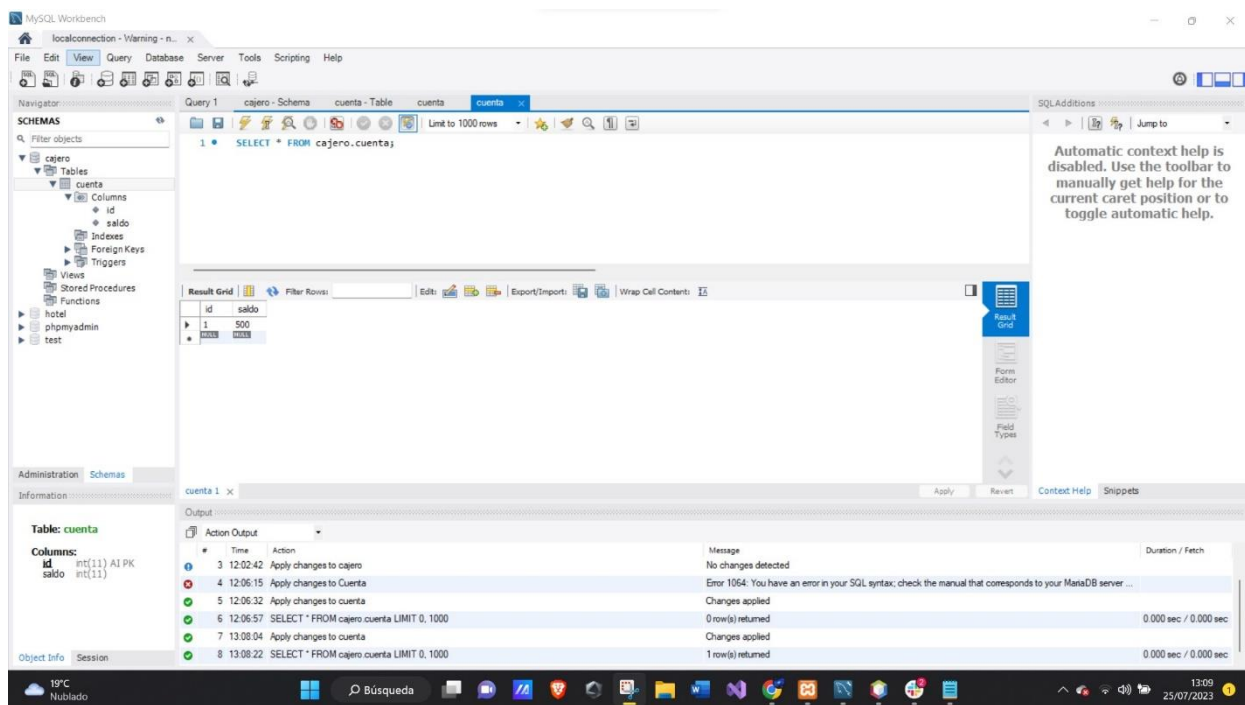
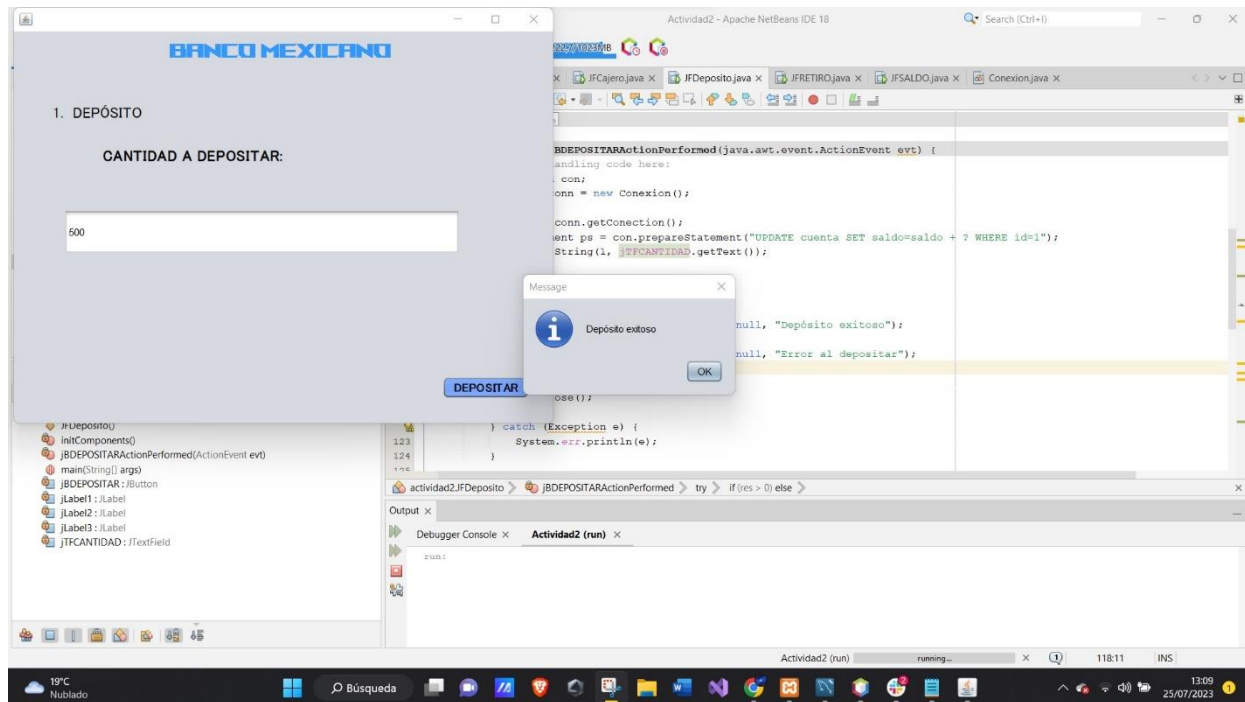
Ya una vez visto la situación anterior para esta última actividad utilizaremos los siguientes programas NetBeans, Xampp y Workbench Mysql con la finalidad de poder enlazar los programas y de esta manera poder realizar la interfaz ya que se necesita hacer la conexión entre estos programas para que el resultado de la interfaz sea el adecuado con ello realizaremos la conexión y haremos la codificación para que este nos dé el resultado final.

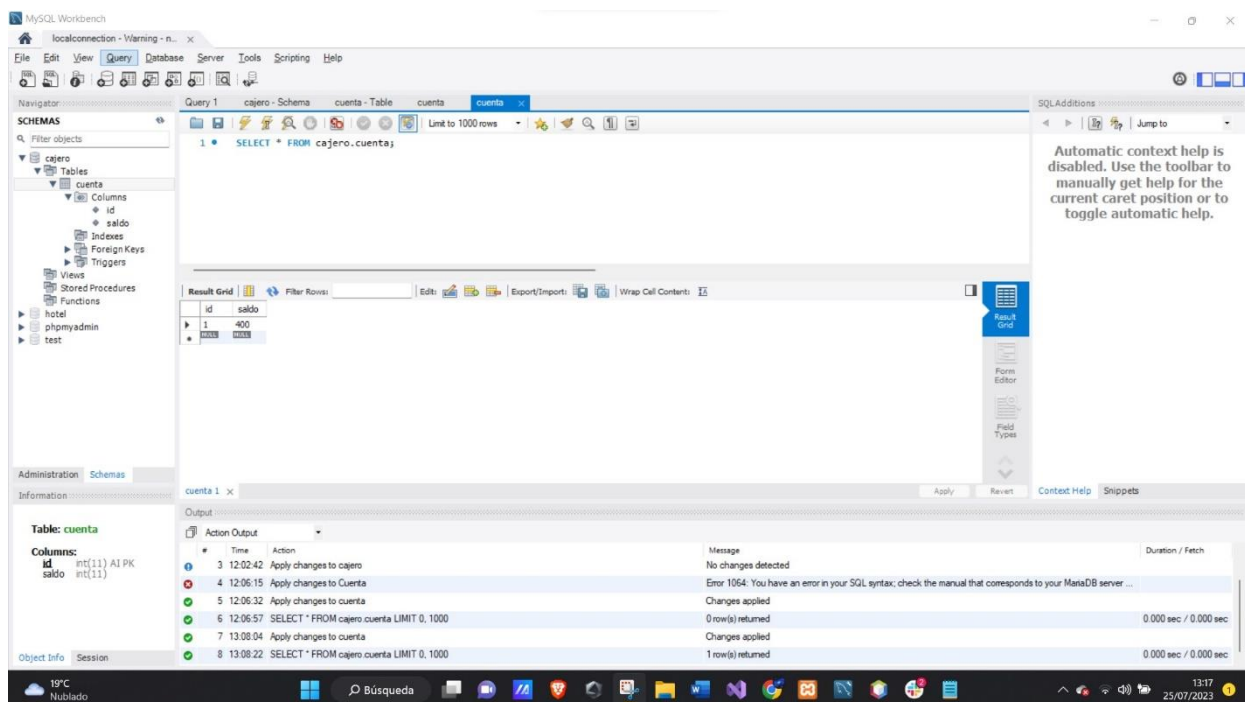
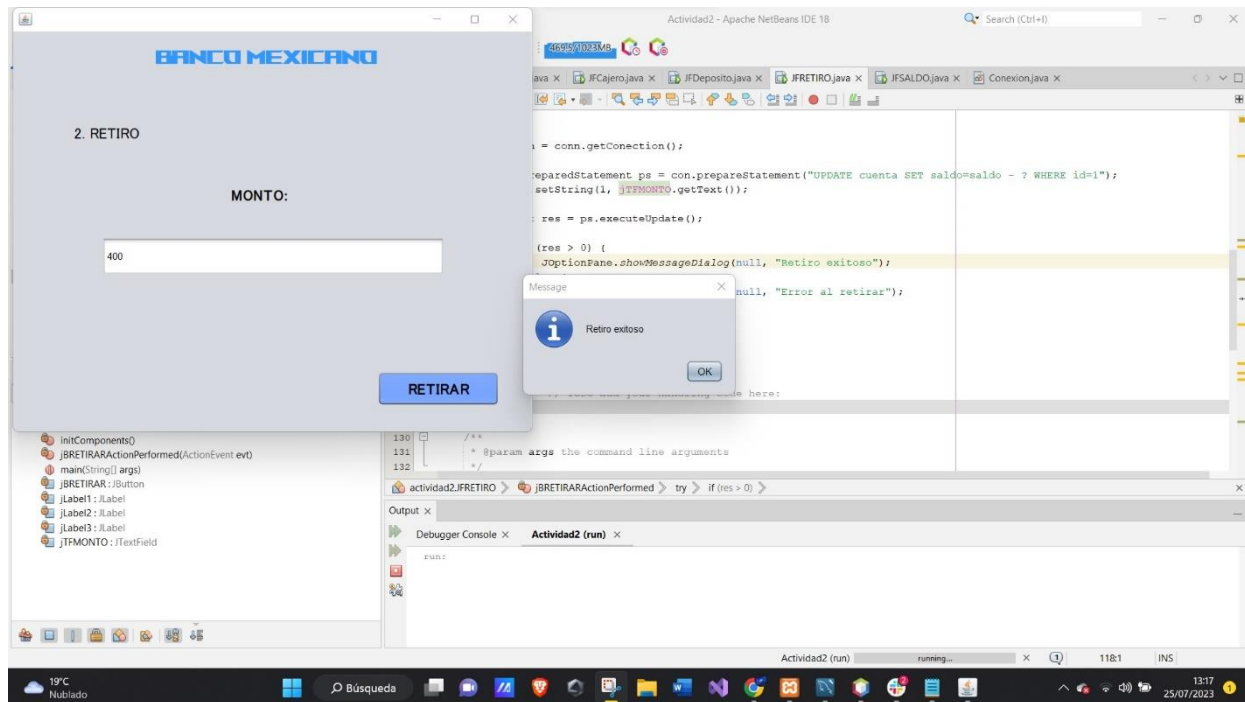
Desarrollo.

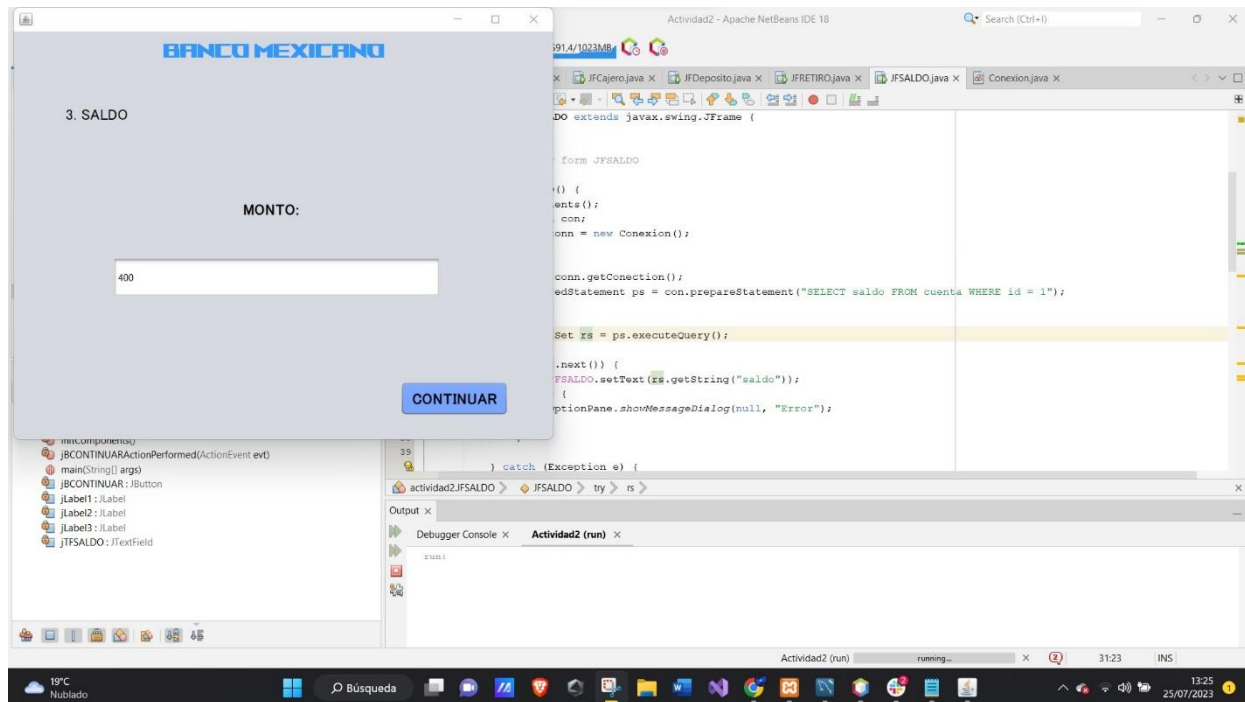
A continuación, se mostrarán pasos que seguimos para la realización de esta actividad.











Codificacion.

-----Clase Conexion-----

```
import com.mysql.jdbc.Connection;
```

```
import java.sql.DriverManager;
```

```
import javax.swing.JOptionPane;
```

```
public class Conexion {
```

```
    public Connection getConnection() {
```

```
        Connection con = null;
```

```
        String base = "cajerobd"; //Nombre de la base de datos
```

```
        String url = "jdbc:mysql://localhost:3306/" + base; //Direccion, puerto y nombre de la Base de Datos
```

```
        String user = "root"; //Usuario de Acceso a MySQL
```

```
        String password = ""; //Password del usuario
```



```
try {  
    Class.forName("com.mysql.jdbc.Driver");  
    con = (Connection) DriverManager.getConnection(url, user, password);  
    //JOptionPane.showMessageDialog(null,"conexión exitosa");  
} catch (Exception e) {  
    System.err.println(e);  
}  
return con;  
}  
}
```

Cajero principal

```
private void JBRealizarActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String o = jTFOpcion.getText();  
    int opcion = Integer.parseInt(o);  
  
    switch(opcion){  
        case 1:  
            JFDeposito deposito = new JFDeposito();  
            deposito.show();  
            break;
```

```
case 2:
    JFRetiro retiro = new JFRetiro();
    retiro.show();
    break;
case 3:
    JFSaldo saldo = new JFSaldo();
    saldo.show();
    break;
case 4:
    this.dispose();
    break;
}
}
```

```
private void jBDepositarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Connection con;
    Conexion conn = new Conexion();
    try {
        con = conn.getConexion();
        PreparedStatement ps = con.prepareStatement("UPDATE cuenta SET saldo=saldo + ?
WHERE id=1");
        ps.setString(1, jTFCantidad.getText());
```

```
int res = ps.executeUpdate();

if (res > 0) {
    JOptionPane.showMessageDialog(null, "Depósito exitoso");
} else {
    JOptionPane.showMessageDialog(null, "Error al depositar");
}

con.close();

} catch (Exception e) {
    System.err.println(e);
}
}
```

```
private void jBRetirarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
```

```
Connection con;
```

```
Conexion conn = new Conexion();
```

```
try {
```

```
    con = conn.getConexion();
```

```
        PreparedStatement ps = con.prepareStatement("UPDATE cuenta SET saldo=saldo - ?  
WHERE id=1");  
  
        ps.setString(1, jTFMonto.getText());  
  
        int res = ps.executeUpdate();  
  
        if (res > 0) {  
            JOptionPane.showMessageDialog(null, "Retiro exitoso");  
        } else {  
            JOptionPane.showMessageDialog(null, "Error al retirar");  
        }  
  
        con.close();  
  
    } catch (Exception e) {  
        System.err.println(e);  
    }  
}
```

//En el constructor

```

public JFSaldo() {
    initComponents();
    Connection con;
    Conexion conn = new Conexion();

    try {
        con = conn.getConnection();
        PreparedStatement ps = con.prepareStatement("SELECT saldo FROM cuenta WHERE id
= 1");

        ResultSet rs = ps.executeQuery();

        if (rs.next()) {
            jTFMonto.setText(rs.getString("saldo"));
        } else {
            JOptionPane.showMessageDialog(null, "Error");
        }
    } catch (Exception e) {
        System.err.println(e);
    }
}

```

```

private void jBContinuarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.dispose();
}

```

Conclusión.

En conclusión, aprendí a cómo utilizar 3 tipos de programas para la realización de esta actividad así mismo el como configurarlos para que haya una conexión entre si y hacer la codificación para que este me diera el resultado final con ello espero seguir aprendiendo mas y con ello ponerlo en práctica para poder mejorar mi conocimiento.

Referencias.

Video de la Tutoría 3.