

JAVA STANDARD EDITION

Día 4

Oscar Chávez-Bosquez

<http://chavezbosquez.github.io/diplomado/java.html>

Hoy veremos...

1. *Refresh*
2. Modelo-Vista-Controlador (MVC)
3. Aplicación CRUD

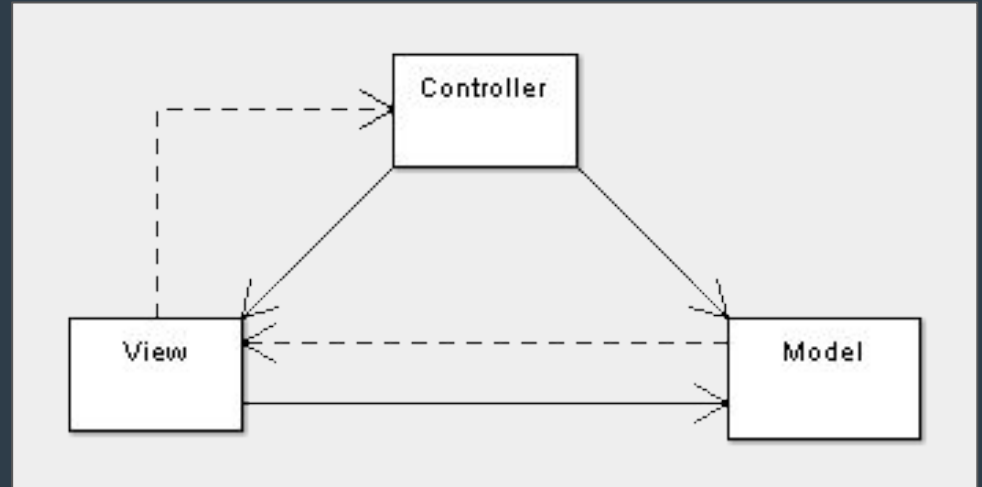
Refresh...

Daypo: [Java](#)

Universidad Carlos III: [Java Básico](#)

Pregunta2: [Test de Programacion en Java](#)

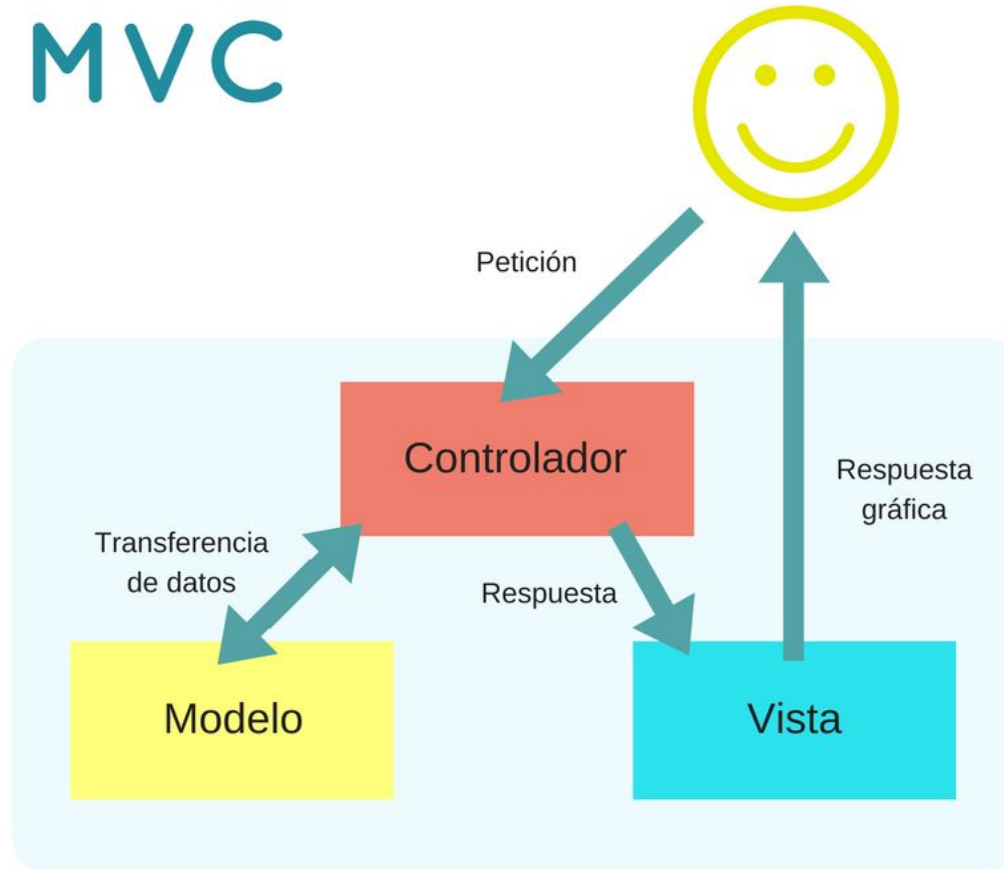
<https://chavezbosquez.github.io/diplomado/java.html>



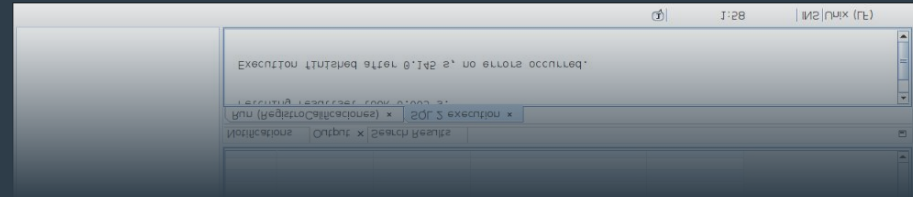
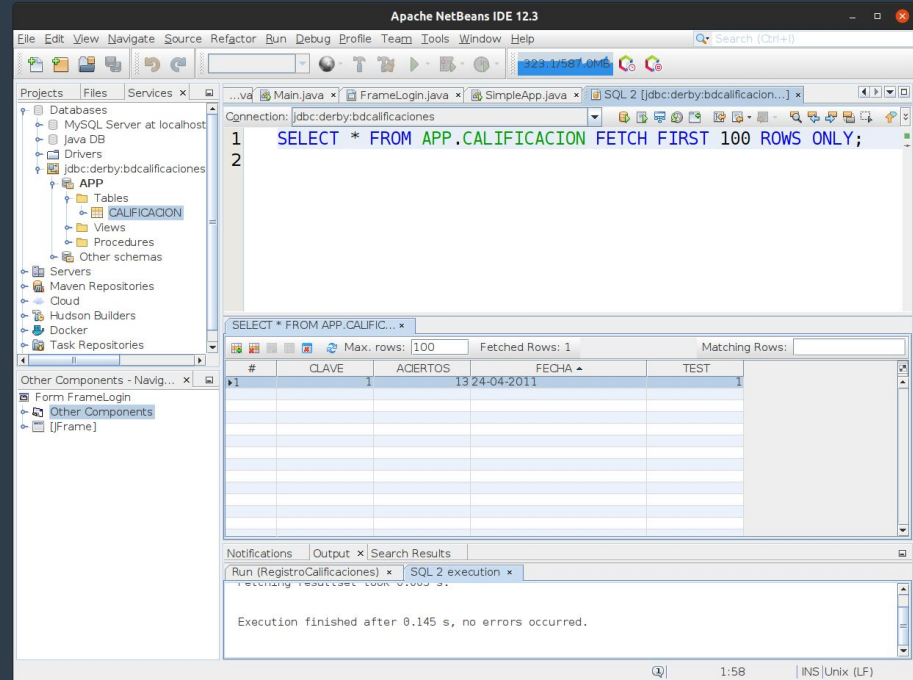
Modelo-Vista-Controlador
(MVC)

Patrón de diseño **Modelo-Vista-Controlador**

MVC



Aplicaciones CRUD



CRUD applications

- **CRUD** = **C**reate, **R**etrieve, **U**ppdate, **D**elete.
- **Aplicaciones ABCC** = **A**ltas, **B**ajas, **C**onsultas y **C**ambios.

Copiar proyecto AppCalificaciones

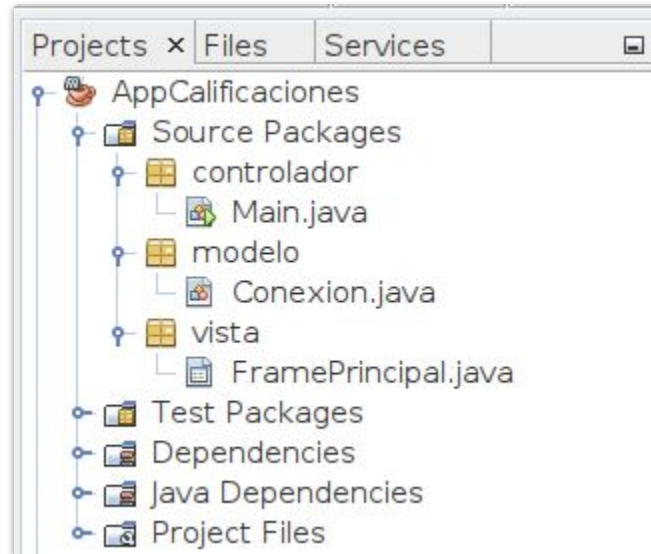
1. Crear copia de AppCalificaciones como AppCalificacionesCRUD.
2. Copiar contenido del método `main()` dentro `FrameCalificaciones.java` dentro en `Main.java`.
3. Eliminar método `main()` dentro `FrameCalificaciones.java`.
4. Eliminar código relacionado con archivos.

Estructura de directorios

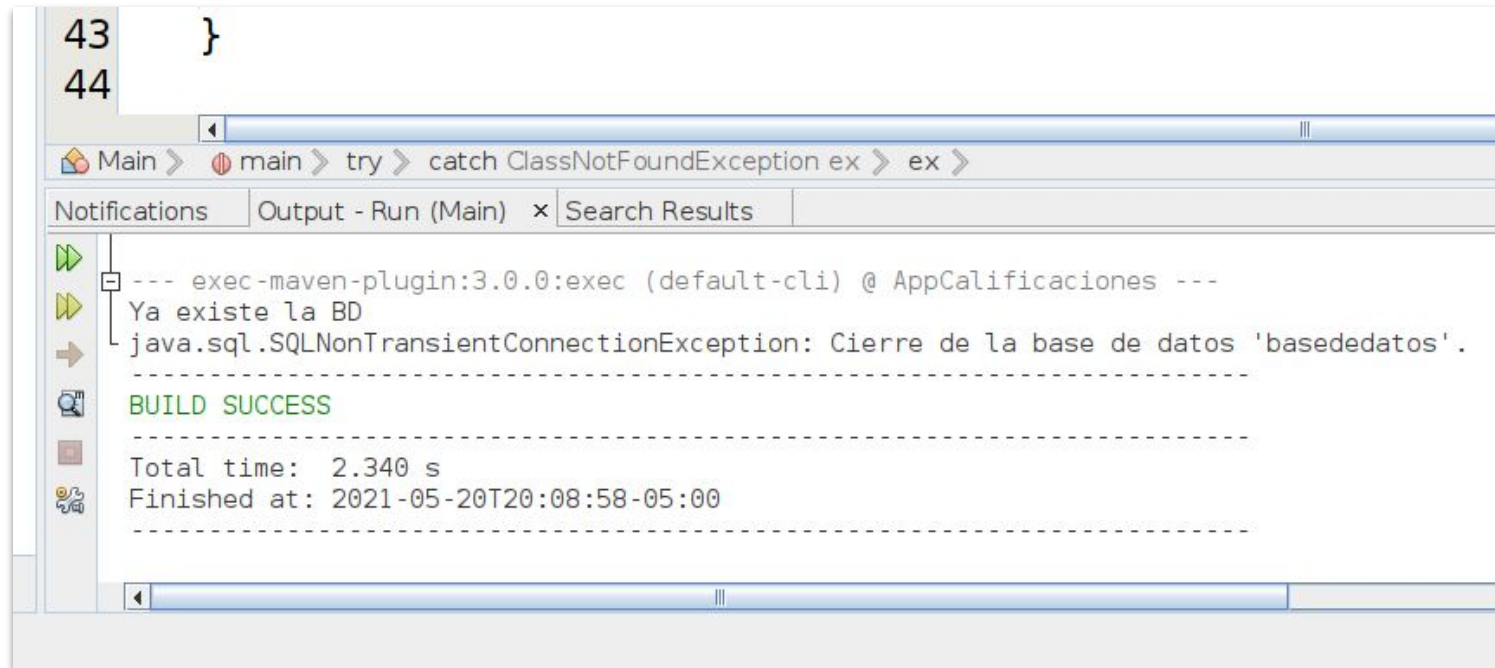
Un proyecto MVC debe contener los siguientes paquetes:

- **modelo**: Conexión a la base de datos.
- **vista**: Interfaces de usuario (UIs).
- **controlador**: Interacción con el usuario, lógica de negocio.

Estructura del proyecto



Excepción de la Base de datos



The screenshot shows an IDE window with a Java code editor at the top and a console output window at the bottom. The code editor shows lines 43 and 44, with line 43 containing a closing curly brace '}'. The console output window has tabs for 'Notifications', 'Output - Run (Main)', and 'Search Results'. The 'Output - Run (Main)' tab is active, displaying the following text:

```
--- exec-maven-plugin:3.0.0:exec (default-cli) @ AppCalificaciones ---  
Ya existe la BD  
java.sql.SQLException: Cierre de la base de datos 'basededatos'.  
-----  
BUILD SUCCESS  
-----  
Total time: 2.340 s  
Finished at: 2021-05-20T20:08:58-05:00  
-----
```

Crear base de datos. Main.java

```
...  
} catch (SQLException ex) {  
    if ( ex.getErrorCode() == 45000 && ex.getSQLState().equals("08006") ) {  
        System.out.println("Base de datos cerrada correctamente");  
    } else {  
        System.err.println(ex);  
    }  
}  
...  
}
```

Archivo Conexion.java

```
public Connection getConexion() {  
    try {  
        Class.forName("org.apache.derby.jdbc.EmbeddedDriver");  
        Connection conexion =  
            DriverManager.getConnection("jdbc:derby:basededatos;create=true");  
        return conexion;  
    } catch (ClassNotFoundException ex) {  
        System.err.println(ex);  
        return null;  
    } catch (SQLException ex) {  
        System.err.println(ex);  
        return null;  
    }  
}
```

Guardar los datos: Conexion.java

```
public boolean guardarDatos(int aciertos, String fecha, int test) {
    Connection conexion = this.getConexion();
    try {
        PreparedStatement psInsert = conexion.prepareStatement(
            "INSERT INTO calificacion(aciertos, fecha, test) VALUES(?,?,?)");
        psInsert.setInt(1, aciertos);
        psInsert.setString(2, fecha);
        psInsert.setInt(3, test);
        psInsert.executeUpdate();
        psInsert.close();
        conexion.close();
        return true;
    } catch (SQLException ex) {
        System.err.println(ex);
        return false;
    }
}
```

Guardar los datos: FrameCalificaciones.java

```
private void btnGuardarActionPerformed(java.awt.event.ActionEvent evt) {  
    int test = cboTest.getSelectedIndex();  
    int aciertos = Integer.parseInt(spnAciertos.getText());  
    DateFormat formatter = new SimpleDateFormat("dd-MM-yyyy");  
    String fecha = formatter.format(calFecha.getDate());  
    Conexion conexion = new Conexion();  
    if (conexion.guardarDatos(aciertos, fecha, test)) {  
        JOptionPane.showMessageDialog(this, "Datos guardados");  
    } else {  
        JOptionPane.showMessageDialog(this, "Error al guardar los datos", "Error",  
                                       JOptionPane.ERROR_MESSAGE);  
    }  
    conexion.cerrarConexion();  
}
```

Leer los datos: Conexion.java

```
public String getDatos() {
    Connection conexion = this.getConexion();
    try {
        Statement sentencia = conexion.createStatement();
        ResultSet resultado = sentencia.executeQuery(
            "SELECT clave,aciertos,fecha,test FROM calificacion");
        String texto = "";
        while (resultado.next()) {
            int id = resultado.getInt(1);
            int aciertos = resultado.getInt(2);
            String fecha = resultado.getString(3);
            int test = resultado.getInt(4);
            texto = texto + aciertos + " " + fecha + " " + test + "\n";
        }
        resultado.close();
        return texto;
    } catch (SQLException ex) {
        System.err.println(ex);
        return null;
    }
}
```

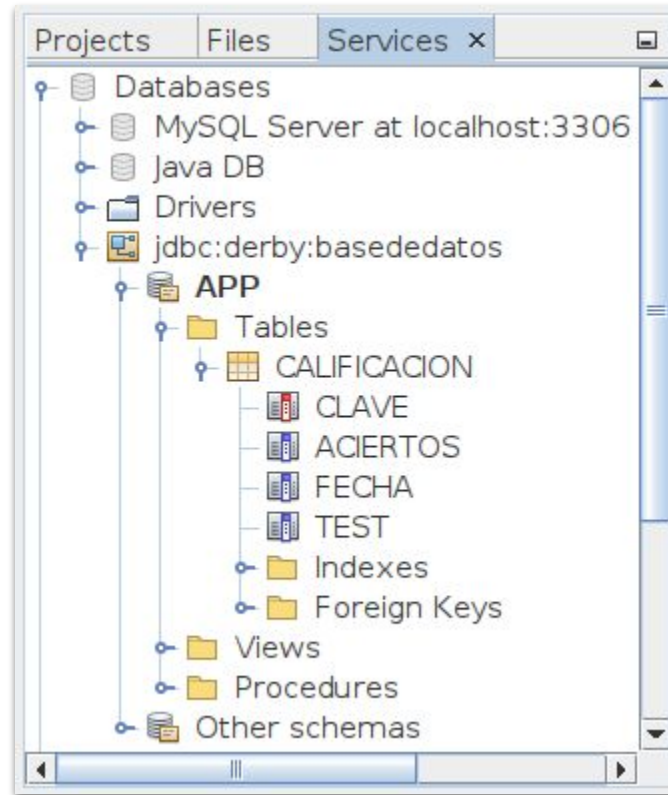

Leer los datos: `FrameCalificaciones.java`

```
private void btnLeerActionPerformed(java.awt.event.ActionEvent evt) {  
    Conexion conexion = new Conexion();  
    String datos = conexion.getDatos();  
    JOptionPane.showMessageDialog(this, datos);  
    conexion.cerrarConexion();  
}
```

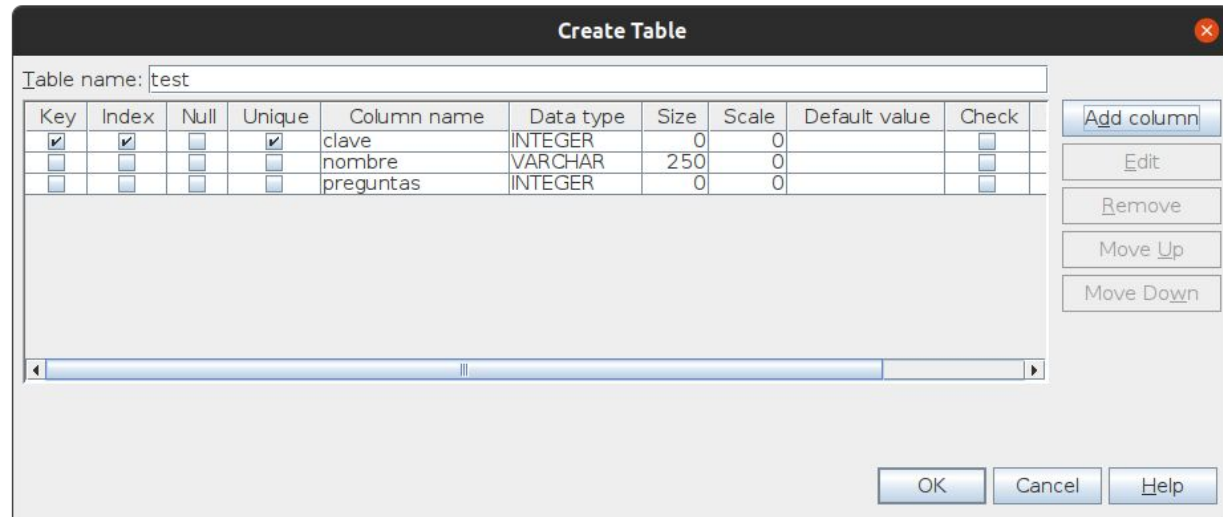
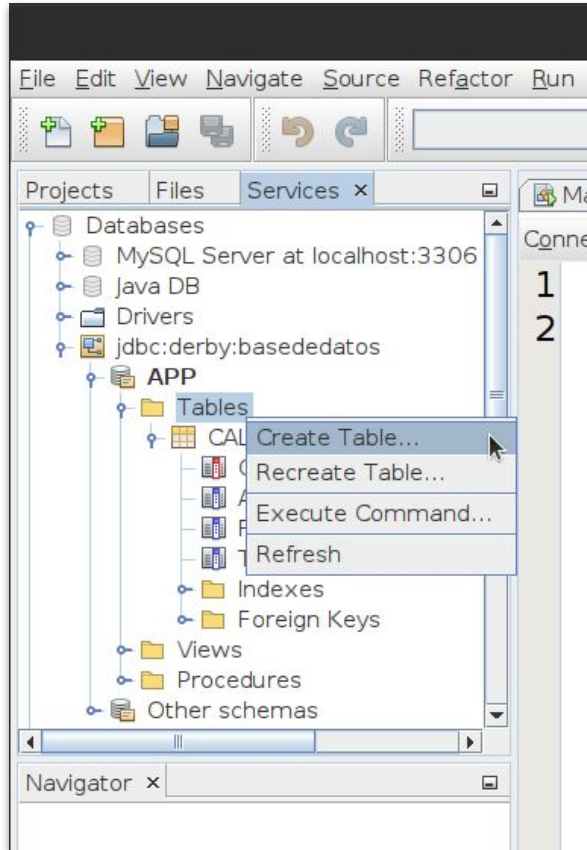
No olviden cerrar la conexión: `Conexion.java`

```
public void cerrarConexion() {  
    try {  
        DriverManager.getConnection("jdbc:derby:basededatos;shutdown=true");  
    } catch (SQLException ex) {  
        if ( ex.getErrorCode() == 45000 && ex.getSQLState().equals("08006") ) {  
            System.out.println("Base de datos cerrada correctamente");  
        } else {  
            System.err.println(ex);  
        }  
    }  
}
```

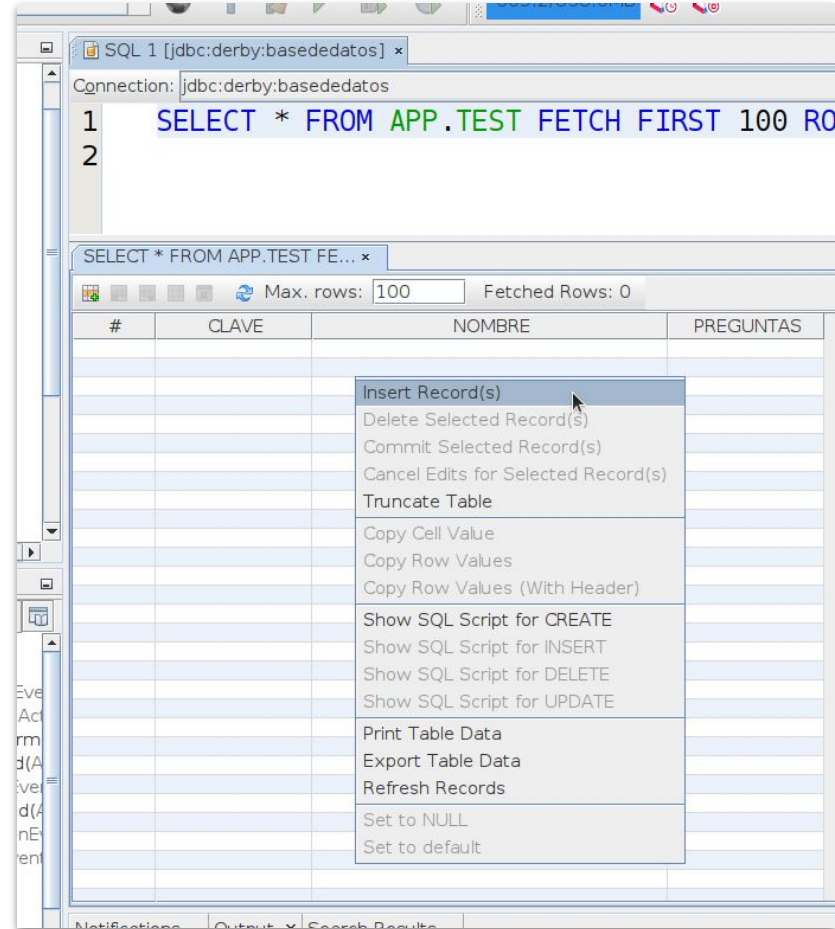
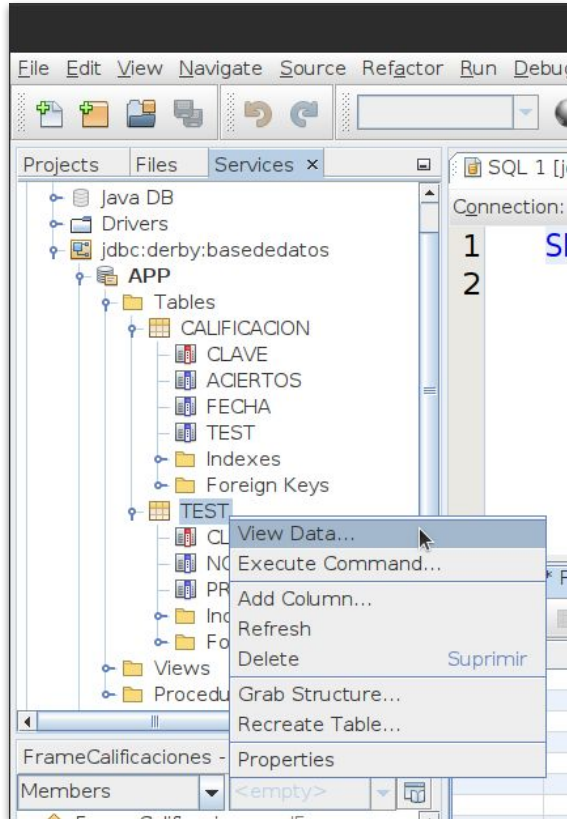
Acceso a la Base de datos



Crear tabla test



Insertar datos en la tabla test



Insertar datos en la tabla test

Insert Record(s) ✕

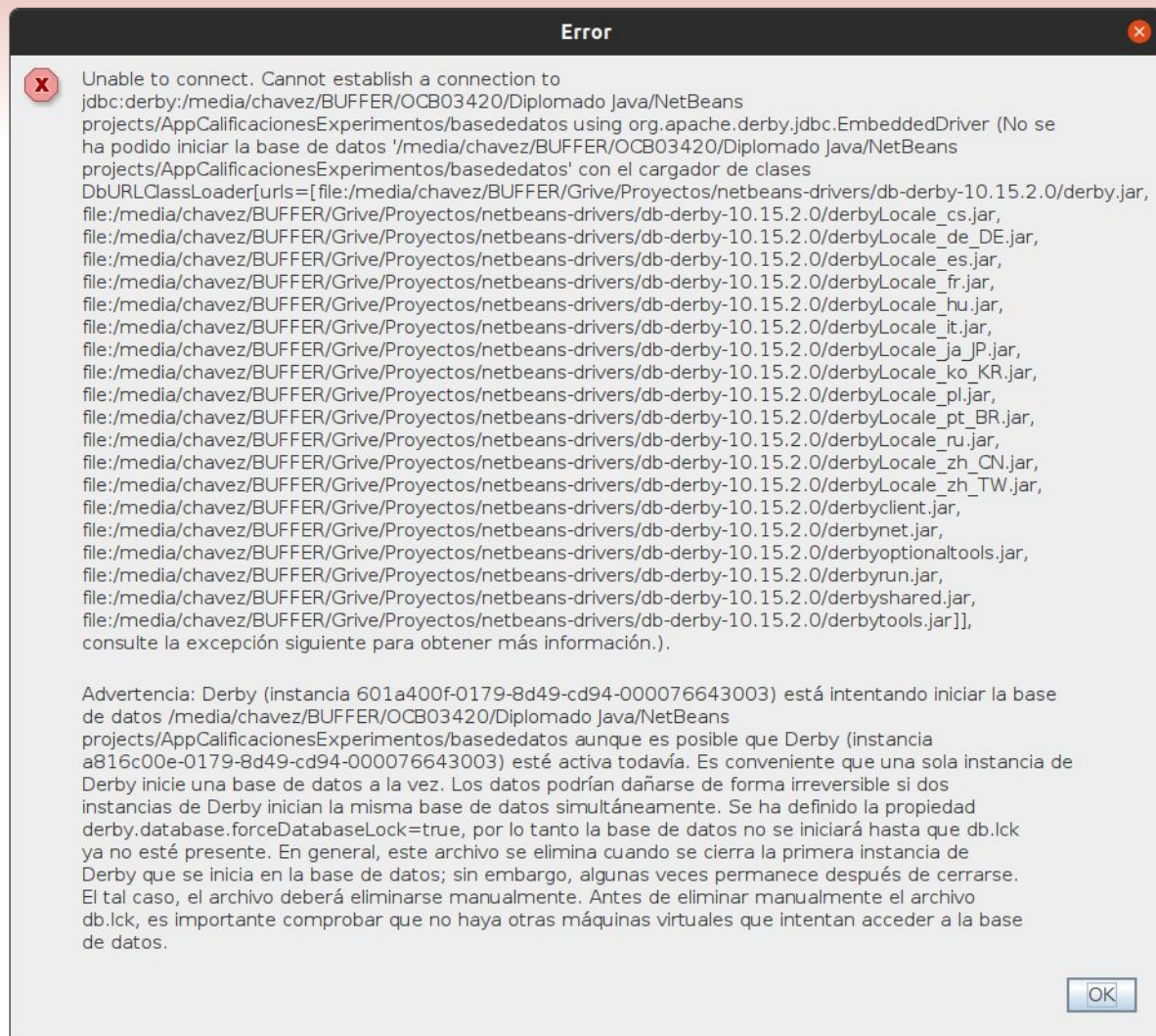
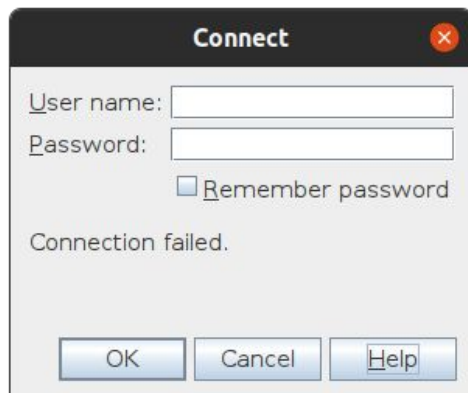
Press CTRL+Tab to exit data entry mode from the table. Press CTRL+0 to set NULL value and CTRL+1 to set DEFAULT value for a given column.

#	CLAVE	NOMBRE	PREGUNTAS
1		1 Digital Learning SL: Test de Java y POO	10
2		2 Cibertest: Programación en Java Básico	20
3		3 Daypo: Test de Java, nivel básico	15
4		4 Daypo: Java	5
5		5 Universidad Carlos III: Java Básico	12
6		6 Pregunta2: Test de Programacion en Java	10

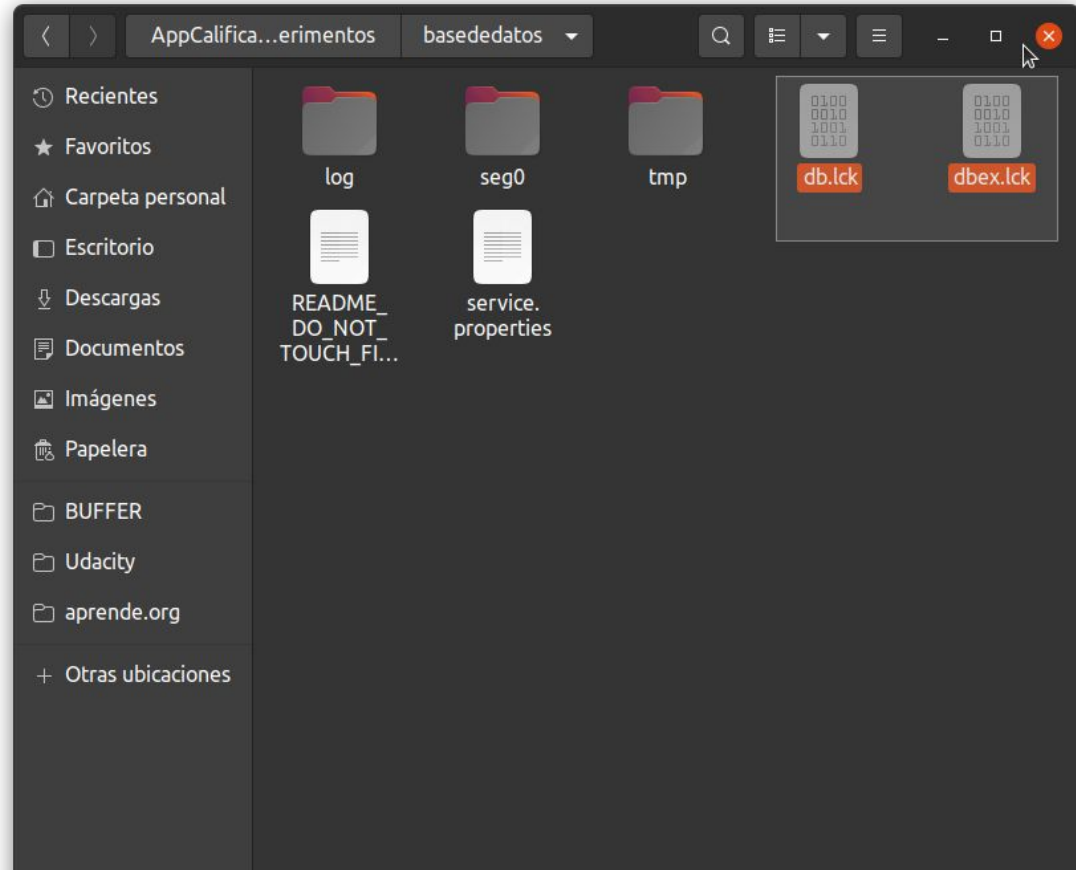
Show SQL
Add Row
Remove
OK
Cancel

[illegible]

¿Error?

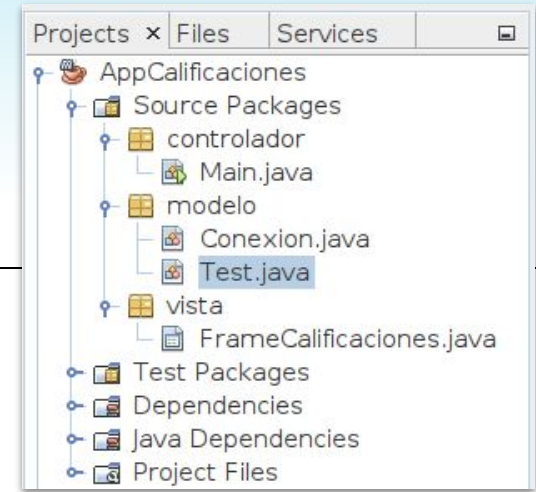


¿Error?



Clase Test.java

```
public class Test {  
    int clave;  
    String nombre;  
    public int preguntas;  
  
    public Test(int clave, String nombre, int preguntas) {  
        this.clave = clave;  
        this.nombre = nombre;  
        this.preguntas = preguntas;  
    }  
  
    @Override  
    public String toString() {  
        return nombre;  
    }  
}
```



Obtener la lista de tests desde Conexion.java

```
public List<Test> getTests() {
    Connection conexion = this.getConexion();
    try {
        Statement sentencia = conexion.createStatement();
        ResultSet resultado =
            sentencia.executeQuery("SELECT clave,nombre,preguntas FROM test");
        List<Test> lista = new ArrayList<Test>();
        while (resultado.next()) {
            int clave = resultado.getInt(1);
            String nombre = resultado.getString(2);
            int preguntas = resultado.getInt(3);
            Test test = new Test(clave, nombre, preguntas);
            lista.add(test);
        }
        resultado.close();
        return lista;
    } catch (SQLException ex) {
        System.err.println(ex);
        return null;
    }
}
```

FrameCalificaciones.java

Cargar datos en el constructor `FrameCalificaciones()`:

```
Conexion conexion = new Conexion();  
List lista = conexion.getTests();  
conexion.cerrarConexion();  
DefaultComboBoxModel modelo = new  
                                DefaultComboBoxModel(lista.toArray());
```

Base de datos