

Huayllas Pinto Kevin

```
//
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DBConnection {
    private static final String DB_URL = "jdbc:postgresql://oregon-postgres.render.com:5432/sisinfo2db";
    private static final String DB_USERNAME = "kevin_123";
    private static final String DB_PASSWORD = "8ba06G0RmY0HjY2PqSrScxaR3p4aI3We";

    public static Connection getConnection() throws SQLException {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            return DriverManager.getConnection(DB_URL, DB_USERNAME, DB_PASSWORD);
        } catch (ClassNotFoundException e) {
            System.err.println("Error al cargar el controlador JDBC: " + e.getMessage());
            throw new SQLException("Error al cargar el controlador JDBC", e);
        }
    }
}
```

```
public FrutaTable() {
    // Crear la tabla con las columnas
    String[] columnNames = {"id", "fruta", "precio"};
    tableModel = new DefaultTableModel(columnNames, 0);
    frutaTable = new JTable(tableModel);
    JTextField nameTextField = new JTextField(20);
    JTextField priceTextField = new JTextField(10); //

    JLabel nameLabel = new JLabel("Nombre:");
    JLabel priceLabel = new JLabel("Precio:");

    // Agregar la tabla a un JScrollPane para permitir el desplazamiento
    JScrollPane scrollPane = new JScrollPane(frutaTable);
    add(scrollPane, BorderLayout.CENTER);

    // Botones para agregar, eliminar y editar frutas
    JButton agregarButton = new JButton("Agregar Fruta");
    agregarButton.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            // Lógica para agregar una nueva fruta
            agregarFruta(nameTextField.getText(), Double.parseDouble(priceTextField.getText()));
        }
    });
}
```

```

JButton eliminarButton = new JButton("Eliminar Fruta");
eliminarButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        // Lógica para eliminar una fruta seleccionada
        // Por ejemplo:
        int selectedRow = frutaTable.getSelectedRow();
        if (selectedRow != -1) {
            int id = (int) frutaTable.getValueAt(selectedRow, 0);
            eliminarFruta(id);
            tableModel.removeRow(selectedRow);
        }
    }
});

```

```

JButton editarButton = new JButton("Editar Fruta");
editarButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        // Lógica para editar una fruta seleccionada
        // Por ejemplo:
        int selectedRow = frutaTable.getSelectedRow();
        if (selectedRow != -1) {
            int id = (int) frutaTable.getValueAt(selectedRow, 0);

```

```

JPanel buttonPanel = new JPanel();
buttonPanel.add(nameLabel);
buttonPanel.add(nameTextField);
buttonPanel.add(priceLabel);
buttonPanel.add(priceTextField);
buttonPanel.add(agregarButton);
buttonPanel.add(eliminarButton);
buttonPanel.add(editarButton);
add(buttonPanel, BorderLayout.SOUTH);

cargarFrutas(); // Cargar las frutas desde la base de datos al iniciar la aplicación

pack();
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setLocationRelativeTo(null);
setVisible(true);
}

```

```
// Función para cargar las frutas desde la base de datos
private void cargarFrutas() {
    try (Connection conexion = DriverManager.getConnection(URL, USUARIO, CONTRASENA)) {
        String sql = "SELECT * FROM frutas";
        try (PreparedStatement statement = conexion.prepareStatement(sql)) {
            try (ResultSet resultSet = statement.executeQuery()) {
                while (resultSet.next()) {
                    int id = resultSet.getInt("id");
                    String nombre = resultSet.getString("fruta");
                    double precio = resultSet.getDouble("precio");
                    tableModel.addRow(new Object[]{id, nombre, precio});
                }
            }
        }
    } catch (SQLException e) {
        System.err.println("Error al cargar las frutas desde la base de datos: " + e.getMessage());
    }
}
```

```
private void agregarFruta(String nombreFruta, double precio) {
    try (Connection conexion = DriverManager.getConnection(URL, USUARIO, CONTRASENA)) {
        String sql = "INSERT INTO frutas (fruta, precio) VALUES (" + nombreFruta + "," + precio + ")";
        try (PreparedStatement statement = conexion.prepareStatement(sql)) {
            statement.setString(1, nombreFruta);
            statement.setDouble(2, precio);
            statement.executeUpdate();
            System.out.println("Fruta agregada correctamente.");
        }
    } catch (SQLException e) {
        System.err.println("Error al agregar la fruta: " + e.getMessage());
    }
}
```

```
private void eliminarFruta(int idFruta) {
    try (Connection conexion = DriverManager.getConnection(URL, USUARIO, CONTRASENA)) {
        String sql = "DELETE FROM frutas WHERE id = ?";
        try (PreparedStatement statement = conexion.prepareStatement(sql)) {
            statement.setInt(1, idFruta);
            statement.executeUpdate();
            System.out.println("Fruta eliminada correctamente.");
        }
    } catch (SQLException e) {
        System.err.println("Error al eliminar la fruta: " + e.getMessage());
    }
}
```

```
// Función para editar el nombre y precio de una fruta por su ID
private void editarFruta(int idFruta, String nuevoNombre, double nuevoPrecio) {
    try (Connection conexion = DriverManager.getConnection(URL, USUARIO, CONTRASENA)) {
        String sql = "UPDATE frutas SET fruta = ?, precio = ? WHERE id = ?";
        try (PreparedStatement statement = conexion.prepareStatement(sql)) {
            statement.setString(1, nuevoNombre);
            statement.setDouble(2, nuevoPrecio);
            statement.setInt(3, idFruta);
            statement.executeUpdate();
            System.out.println("Fruta editada correctamente.");
        }
    } catch (SQLException e) {
        System.err.println("Error al editar la fruta: " + e.getMessage());
    }
}
```