

## SQL

Clase "Conexión.java"	
ResultSet rs = conexion.consultar(sql);	<pre> public ResultSet consultar(String sql) {     try {         Statement statement = conexion.createStatement();         return statement.executeQuery(sql);     } catch (SQLException e) {         e.printStackTrace();         return null;     } } </pre>
conexion. actualizar(sql);	<pre> public void actualizar(String sql) {     try {         Statement statement = conexion.createStatement();         statement.executeUpdate(sql);     } catch (SQLException e) {         e.printStackTrace();     } } </pre>

Clase "Gestion.java"	
Escribir en lenguaje SQL	<pre> String nombre = nomField.getText(); //JFrame  String sql = "DELETE FROM clientes WHERE nombre = '" + nombre + "'";  String sql = "SELECT * FROM clientes";  String sql = "INSERT INTO clientes (nombre, email, telefono) VALUES ('" + nombre + "', '" + email + "', '" + telefono + "')"; </pre>
Conectarse + Pasar String a int & Leer booleano de RadioButton	<pre> Conexion conexion = new Conexion("localhost", "tpfinalmarcos", "root", "");  int cantidad = Integer.parseInt(cantidadField.getText()); int cant = (Integer) cantidad.getValue(); if (radioButton1.isSelected() == 1) </pre>
Mostrar rs	<pre> private JTable tabla; private DefaultTableModel modeloTabla; (...) ResultSet rs = conexion.consultar(sql); ResultSetMetaData metaData = rs.getMetaData(); int columnCount = metaData.getColumnCount(); // Limpiar la tabla modeloTabla.setRowCount(0); modeloTabla.setColumnCount(0); // Agregar nombres de columnas for (int i = 1; i &lt;= columnCount; i++) {     modeloTabla.addColumn(metaData.getColumnName(i)); } // Agregar filas while (rs.next()) {     Object[] row = new Object[columnCount]; </pre>

	<pre> for (int i = 1; i &lt;= columnCount; i++) {     row[i - 1] = rs.getObject(i); } modeloTabla.addRow(row); } </pre>
Insertar en bdd	<pre> Conexion cc = new Conexion("localhost", "segundoparcial", "root", "");  String sql = "INSERT INTO contingencia (problema, resultado, montoTotal, mes, probabilidad) VALUES ('" + tipoProblema + "', '" + resultado + "', " + montoTotalFinal + ", '" + mes + "', " + probabilidad + ")";  cc.insertar(sql); cc.cerrarConexion(); </pre>
Crear tabla en JFrame y, desde una tabla sql mostrar el resultado dividido en filas y columnas	<pre> private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  DefaultTableModel modeloTabla = new DefaultTableModel();  modeloTabla.addColumn("ID Contingencia"); modeloTabla.addColumn("Problema"); modeloTabla.addColumn("Resultado"); modeloTabla.addColumn("Monto Total"); modeloTabla.addColumn("Mes"); modeloTabla.addColumn("Probabilidad");  Conexion cc = new Conexion("localhost", "segundoparcial", "root", """); String sql = "SELECT * FROM contingencia";  try {     ResultSet rs = cc.consultar(sql);     while (rs.next()) {         Object[] fila = new Object[6];         fila[0] = rs.getInt("idContingencia");         fila[1] = rs.getString("problema");         fila[2] = rs.getString("resultado");         fila[3] = rs.getDouble("montoTotal");         fila[4] = rs.getString("mes");         fila[5] = rs.getDouble("probabilidad");         modeloTabla.addRow(fila);     }     tablaContingencia.setModel(modeloTabla); } catch (SQLException ex) {     System.out.println("Error al mostrar los datos en la tabla: " + ex.getMessage()); } finally {     cc.cerrarConexion(); } } </pre>

## Colecciones

ArrayList & LinkedList	<pre> List&lt;integer&gt; arrayList1 = new ArrayList&lt;integer&gt;(); arrayList1.add(1); //private ArrayList&lt;Cartucho&gt; cartuchos;  LinkedList&lt;object&gt; linkedList1 = new LinkedList&lt;object&gt;(arrayList); </pre>
------------------------	--

	linkedList1.addFirst("green");
Mostrar	Println(set) o <i>fore</i> con .get(index:int)
Stack (PILA) & Queue (COLA)	Añado elementos con .push():E Añado elementos con .offer(element:E):boolean
Set & LinkedHashSet	Set<String> set1 = new LinkedHashSet<String>();