

Práctica XIV - JTable y MySql

Al momento de insertar una tabla es importante visualizar los datos extraídos de una consulta, una forma de hacerlo es utilizando `JTable` el cual funciona como las lista mediante un modelo `DefaultTableModel` el cual posee métodos para agregar columnas y filas.

Crear una tabla `JTable`

El siguiente código crea una tabla con tres columnas y 10 filas ficticias:

```
public class MiVentana extends javax.swing.JFrame {

    DefaultTableModel tableModel = new DefaultTableModel();

    public MiVentana() {
        initComponents();

        this.tableModel.addColumn("A");
        this.tableModel.addColumn("B");
        this.tableModel.addColumn("C");

        for (int i = 0; i < 10; i++) {
            Object[] row = new Object[3];
            row[0] = String.format("a%d", i);
            row[1] = String.format("b%d", i);
            row[2] = String.format("c%d", i);
            this.tableModel.addRow(row);
        }

        this.jTable1.setModel(tableModel);
    }

    ...
}
```

Conectar los resultados de un query `MySQL` con `JTable`

Cuando ejecutamos un query en `MySQL` obtenemos un `ResultSet` el cual puede ser iterado en un *while*. En cada iteración del *while* podemos ir agregando una nueva fila en nuestro modelo. El siguiente código primero ajusta el modelo y agrega las columnas extraídas del `ResultSet` para después agregar cada fila:

```
Statement st = conn.createStatement();

ResultSet rs = st.executeQuery("select * from productos;");

// Creamos el modelo de la tabla
DefaultTableModel tableModel = new DefaultTableModel();

// Extraemos los metadatos para obtener los nombres de las columnas
// de la base de datos
ResultSetMetaData metaData = rs.getMetaData();

// Obtenemos el número de columnas
int columnCount = metaData.getColumnCount();

// Obtenemos los nombres de las columnas (su label)
// Observa que el índice empieza en 1
for (int columnIndex = 1; columnIndex <= columnCount; columnIndex++){
    // Agregamos al modelo la columna
    tableModel.addColumn(metaData.getColumnLabel(columnIndex));
}

// Iteramos el ResultSet que trae los datos del query
while (rs.next()){
    // Creamos una fila vacía con tantas columnas como haya en la tabla
    Object[] row = new Object[columnCount];
    // Obtenemos el valor del registro en cada columna
    // Observa que nuevamente empieza en 1, pero el índice de la fila
    // debe ser uno menos
    for (int i = 1; i <= columnCount; i++){
        row[i - 1] = rs.getObject(i);
    }

    // Agregamos la fila al modelo
    tableModel.addRow(row);
}

// Debemos ajustar el nuevo modelo a la tabla
this.jTable1.setModel(tableModel);
```

Problemas

- Crear la tabla productos de la práctica 12 y llenarla con al menos 4 registros.
- Crear una ventana que muestre un `JTable` y un botón que diga actualizar.
- En el evento de control del botón extrae los datos de la consulta `SELECT * FROM productos;`, almacénalos en un modelo como en el código anterior para mostrar los registros en la tabla.
- Cambia la consulta para mostrar los productos con costo mayor a `1000`.

////////////////////////////////////

Diplomado de Java - Alan Badillo Salas (badillo.soft@hotmail.com)

Instituto Politécnico Nacional - Centro de Investigación en Computo