

18-4-2023

# Swing Java

SystemMain@DMP

Cliente

GESTION DE CLIENTES

Dato a Buscar

Exportar

Excel

PDF

Nuevo Registrar Eliminar

DNI/RUC:

Nombre:

T.Doc: Seleccione

#	DNI/RUC	Nombres	Tipo Doc
---	---------	---------	----------

David Mamani Pari  
SYSCENTERLIFE@

# Swing Java

## 1. Base de Datos – ArrayList

cliente	
dniruc: VARCHAR(12)	NOT NULL [ PK ]
nombres: VARCHAR(60)	NOT NULL
documento: VARCHAR(12)	NOT NULL



## 2. Modelo de la Base de datos a nivel de objetos

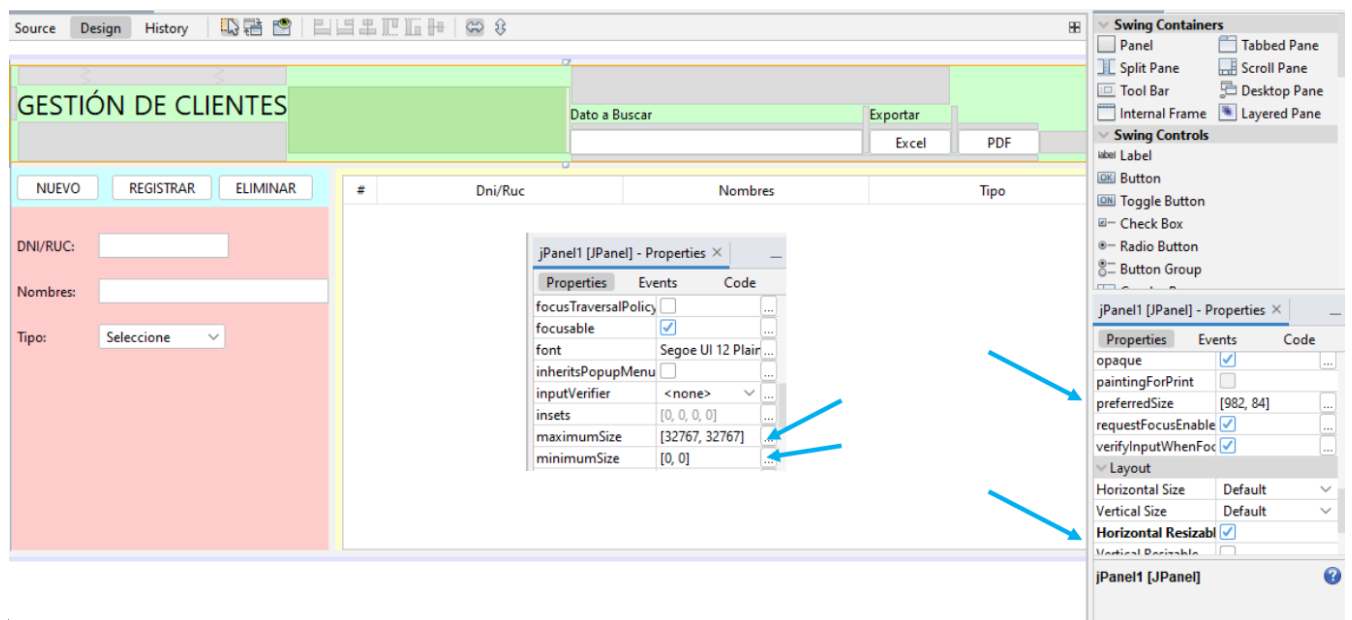
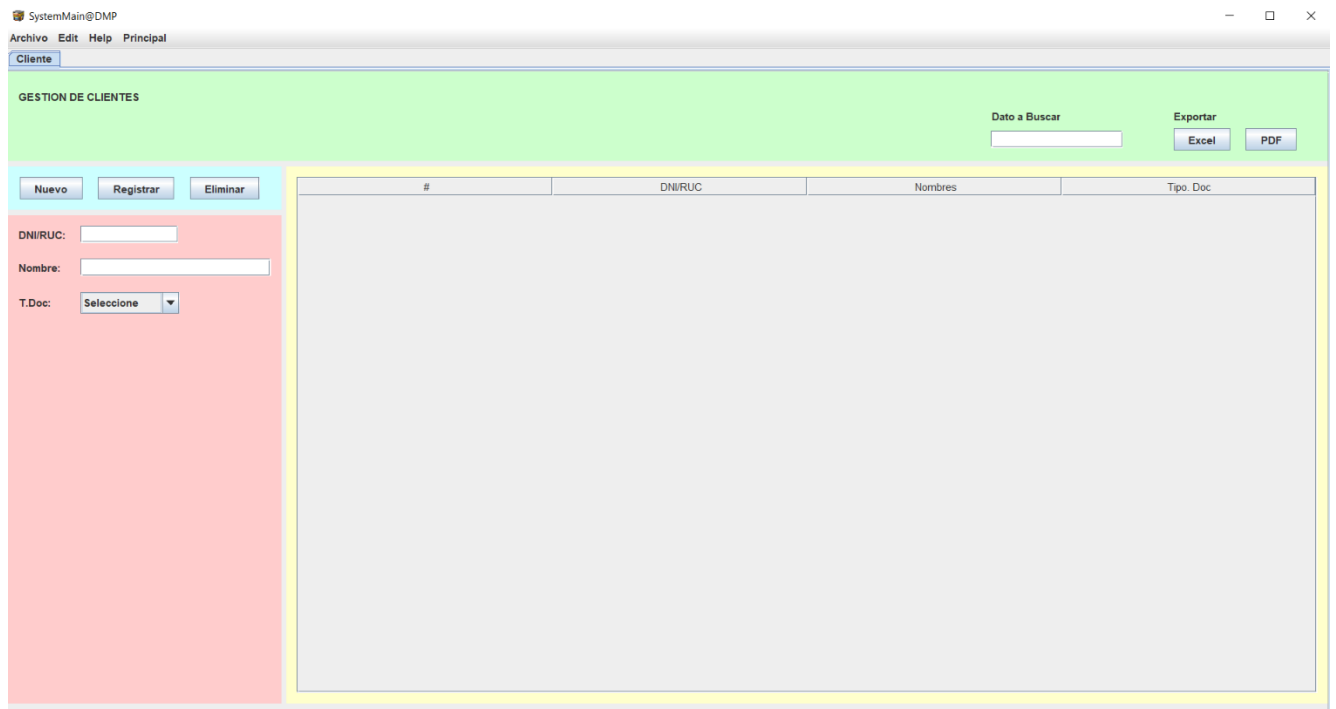
```
import lombok.Data;
@Data
public class Cliente {
    private String dniruc;
    private String nombres;
    private String documento;
}
```

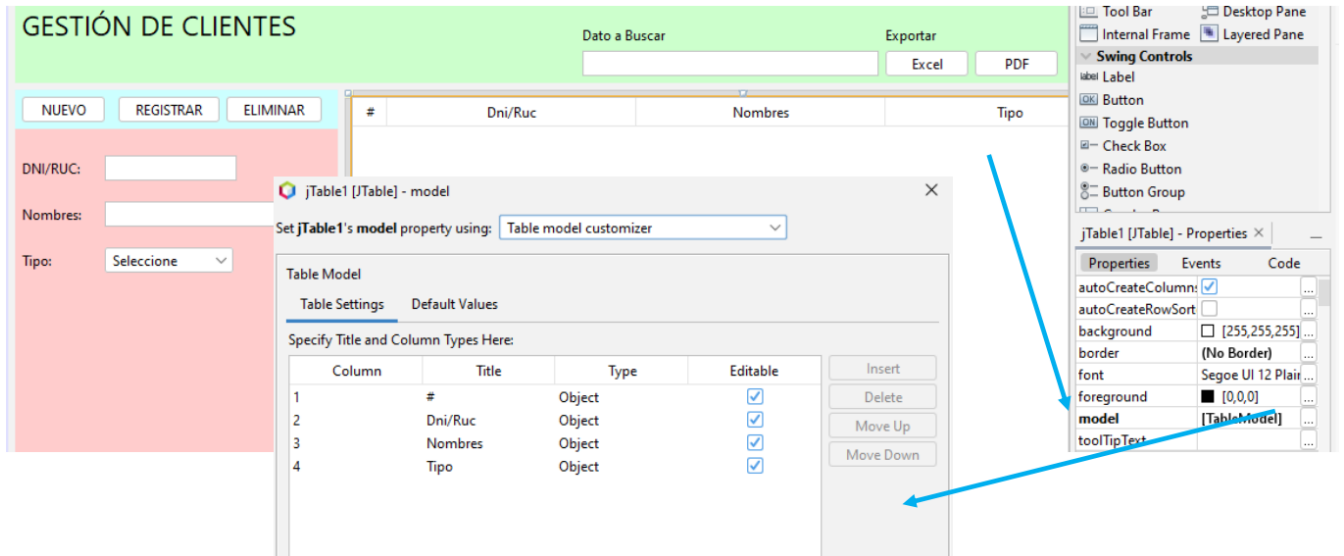
## 3. CRUD Capa de Service

```
@Service
public class ClienteService {

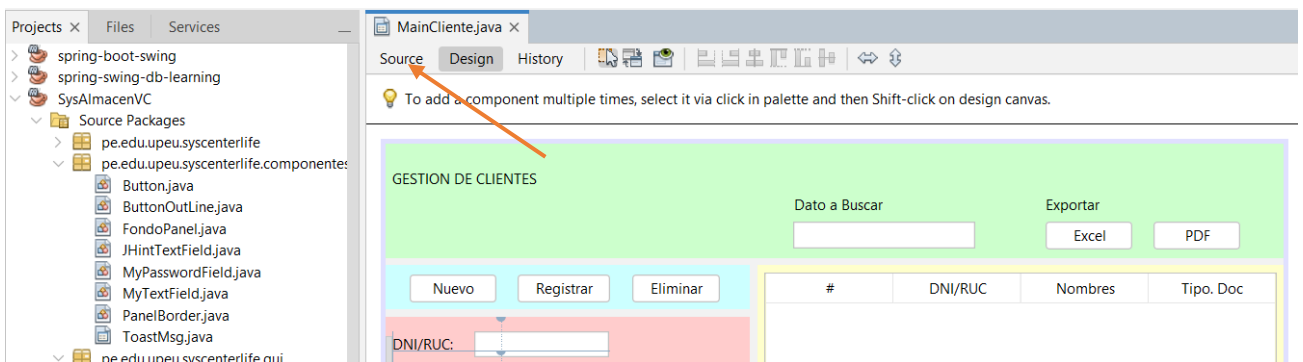
    List<Cliente> listaCli = new ArrayList<>();
    public boolean saveEntidad(Cliente categoria) {
        return this.listaCli.add(categoria);
    }
    public List<Cliente> listarEntidad() {
        return listaCli;
    }
    public Cliente buscarCliente(String dnirucx) {
        return listaCli.stream()
            .filter(cliente -> cliente.getDniruc().equals(dnirucx))
            .findFirst()
            .orElse(null); // Devuelve null si no se encuentra ningún
cliente
    }
    public Cliente updateEntidad(Cliente clientex) {
        listaCli.stream()
            .filter(cliente ->
cliente.getDniruc().equals(clientex.getDniruc())) // Filtrar por DNI
            .findFirst() // Obtener el primer cliente que cumpla con el
filtro
            .ifPresent(cliente ->
cliente.setNombres(clientex.getNombres()));
        //return this.listaCli.set(index, cliente);
        return buscarCliente(clientex.getDniruc());
    }
    public void deleteRegistEntdad(String dato) {
        this.listaCli.remove(listaCli.stream()
            .filter(cliente -> cliente.getDniruc().equals(dato))
            .findFirst().orElse(null));
    }
}
```

## 4. Diseño de Interfaz gráfica





## 5. Implementamos la lógica en **MainCliente**



Hacer clic en la opción de **Source** para ir al código de **MainCliente**

## 6. Colocar el siguiente código en la clase de **MainCliente**

```
DefaultTableModel modelo;
ConfigurableApplicationContext ctx;

ClienteService clienteService=new ClienteService();

TableRowSorter<TableModel> trsfiltro;

enum TIPOCLXIENTE {
    Natural, General, Juridico
};
```

## 7. Colocar los siguientes métodos la clase **MainCliente** después del constructor

```
public void listarClientes() {
    List<Cliente> listarCleintes = clienteService.listarEntidad();
    if (listarCleintes != null) {
        jTable1.setAutoCreateRowSorter(true);
        modelo = (DefaultTableModel) jTable1.getModel();
        modelo.setNumRows(0);
        Object[] ob = new Object[4];
        for (int i = 0; i < listarCleintes.size(); i++) {
            ob[0] = i + 1;
            ob[1] = listarCleintes.get(i).getDniruc();
            ob[2] = listarCleintes.get(i).getNombres();
            ob[3] = listarCleintes.get(i).getDocumento();
            modelo.addRow(ob);
        }
        jTable1.setModel(modelo);
    }
}
```

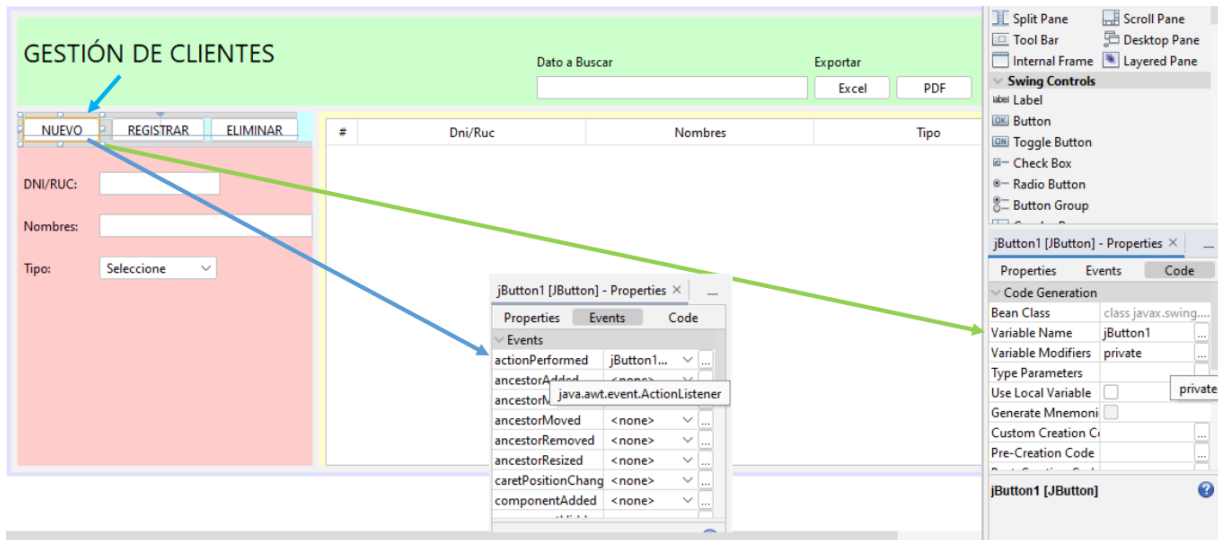
```
public void resetForm() {
    txtDNI.setText("");
    txtNombre.setText("");
    cbxTipo.setSelectedIndex(0);
    txtDNI.requestFocus();
}
```

```
private void paintForm() {
    if (jTable1.getSelectedRow() != -1) {
        modelo = (DefaultTableModel) jTable1.getModel();
        int rowx = jTable1.getSelectedRow();
        Object valor = jTable1.getValueAt(rowx, 1);
        //ClienteTO filax = (ClienteTO)
        modelo.getRow(jTable1.getSelectedRow());
        Cliente d = clienteService.buscarCliente(valor.toString());
        txtDNI.setText(d.getDniruc());
        txtNombre.setText(d.getNombres());
        cbxTipo.setSelectedItem(d.getDocumento());
        txtDNI.setEditable(false);
        btnRegistrar.setText("Modificar");
        //guardarButton.setToolTipText("MODIFICAR");
    } else {
        txtDNI.setEditable(true);
    }
}
```

## 8. Colocar el siguiente código en el constructor de la clase **MainCliente**

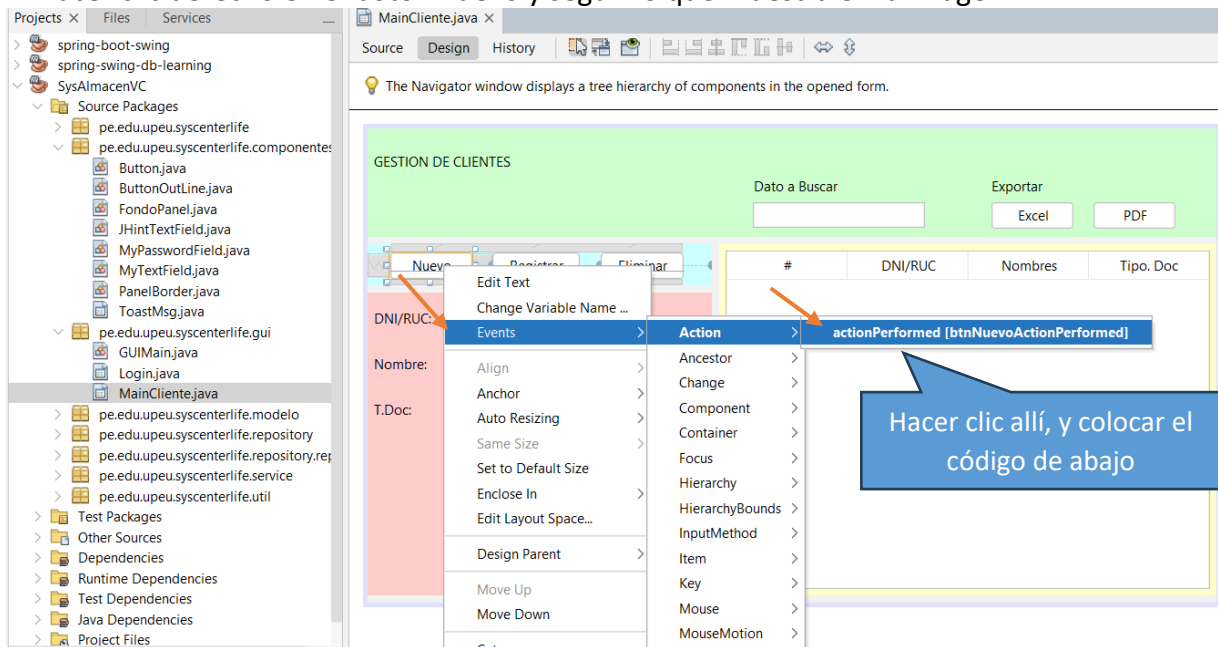
```
for (TIPOCLXIENTE myVar : TIPOCLXIENTE.values()) {
    cbxTipo.addItem(myVar.toString());
}
listarClientes();
```

## 9. Trabajar con eventos en los componentes de la interfaz.



## 10. En el evento de **actionPerformed** del botón Nuevo

Al hacer clic derecho en el botón nuevo y seguir lo que muestra en la imagen.



```
resetForm();  
btnRegistrar.setText("Registrar");  
txtDNI.setEditable(true);  
jTable1.getSelectionModel().clearSelection();
```

## 11.En el evento de **actionPerformed** del botón Eliminar

```
if (jTable1.getSelectedRowCount() > 0) {
    try {
        modelo = (DefaultTableModel) jTable1.getModel();
        int rowx = jTable1.getSelectedRow();
        Object valor = jTable1.getValueAt(rowx, 1);

        if (JOptionPane.showConfirmDialog(this, "Esta seguro de eliminar este registrtro DNI: " + valor + "?", "Mensaje de confirmación",JOptionPane.OK_CANCEL_OPTION,
JOptionPane.INFORMATION_MESSAGE) == 0) {
            modelo.removeRow(rowx);
            clienteService.deleteRegistEntdad(valor.toString());
            resetForm();
        }

    } catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
    }
} else {
    JOptionPane.showMessageDialog(this, "Seleccione un item");
}
```

## 12.En el evento de **actionPerformed** del botón Registrar

```
Cliente to = new Cliente();
to.setDniruc(txtDNI.getText());
to.setNombrrs(txtNombre.getText());
to.setDocumento(cbxTipo.getSelectedItem() == null ? "" :
cbxTipo.getSelectedItem().toString());
int fila = jTable1.getSelectedRow();
if (fila != -1) {
    try {
        Cliente resultado = clienteService.updateEntidad(to);
        if (resultado != null) {
            modelo = (DefaultTableModel) jTable1.getModel();
            Object nuevo[] = {fila + 1, to.getDniruc(), to.getNombrrs(),
to.getDocumento()};
            modelo.removeRow(fila);
            modelo.insertRow(fila, nuevo);
            resetForm();
            JOptionPane.showMessageDialog(this, "Re registro");
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
    }
} else {
    try {

        if (JOptionPane.showConfirmDialog(this, "Esta seguro de crear un nuevo cliente?", "Mensaje de confirmación",JOptionPane.OK_CANCEL_OPTION,
```



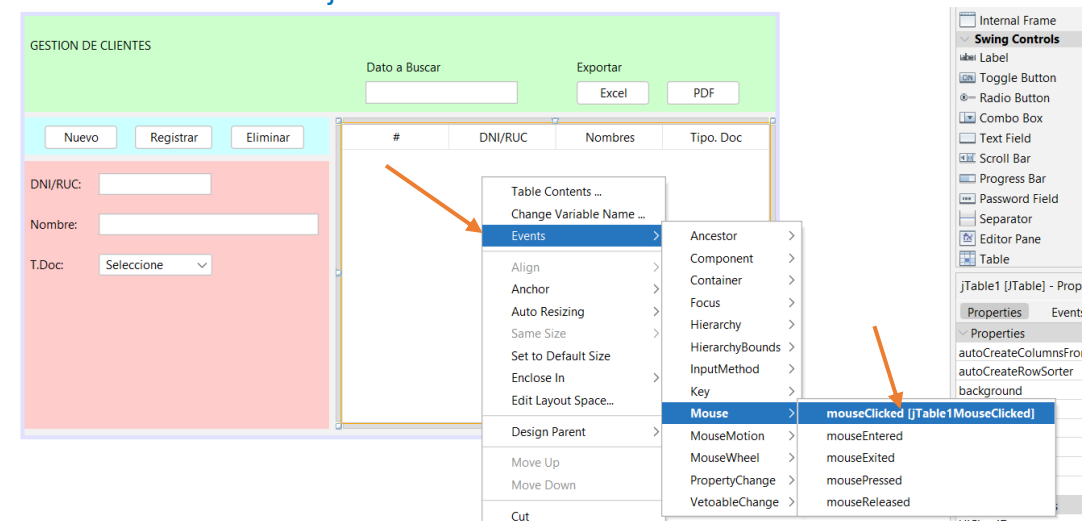
```

JOptionPane.INFORMATION_MESSAGE) == 0) {
    if (clienteService.saveEntidad(to) != false) {
        modelo = (DefaultTableModel) jTable1.getModel();
        Object nuevo[] = {modelo.getRowCount() + 1, to.getDniruc(),
to.getNombres(), to.getDocumento()};
        modelo.addRow(nuevo);
        resetForm();
    }
}
} catch (Exception e) {
    //JOptionPane.showMessageDialog(this, e.getMessage());
    System.err.println("Crear Cliente");
    //log.log(Level.SEVERE, "Crear Cliente", e);
}
}
}

```

### 13. En el evento de **mouseClicked** del **jTable**

Hacer clic derecho en el **jTable**



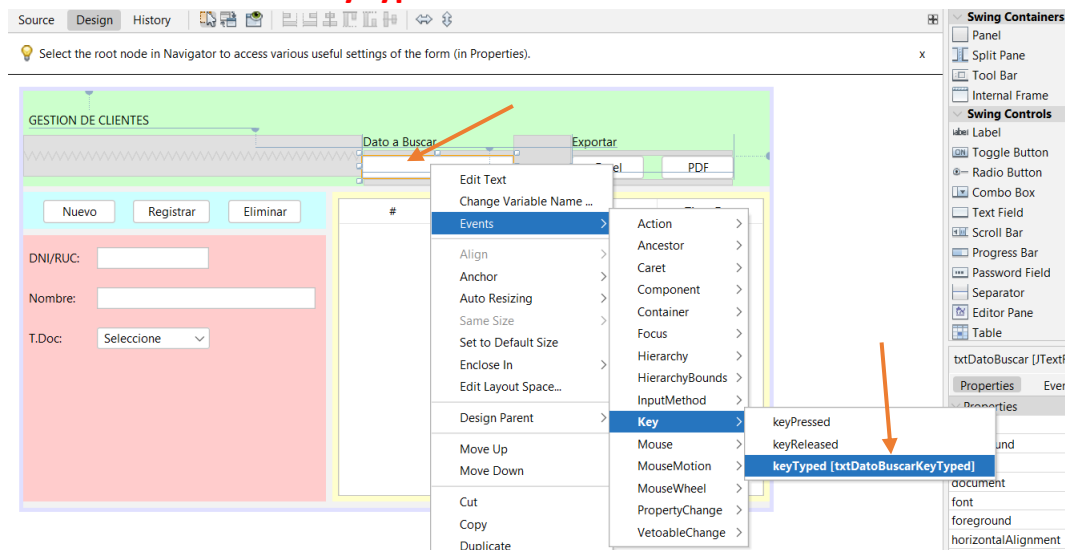
Al hacer clic en **mouseClicked** colocar el siguiente código:

```

paintForm();

```

## 14. En el evento de **keyTyped** del **TxtDatoBuscar**



Al hacer clic en **keyTyped** colocar el siguiente código:

```
txtDatoBuscar.addKeyListener(new KeyAdapter() {
    @Override
    public void keyReleased(KeyEvent e) {
        String cadena = (txtDatoBuscar.getText());
        System.out.println("v:" + cadena);
        txtDatoBuscar.setText(cadena);
        repaint();
    }
});

trsfiltro.setRowFilter(RowFilter.regexFilter(txtDatoBuscar.getText()));

System.out.println("llego");
trsfiltro = new TableRowSorter<>(jTable1.getModel());
jTable1.setRowSorter(trsfiltro);
```

## 15. Crear la Clase **GUIMain**

```
public class GUIMain extends JFrame {
    JTabbedPane jtpane;
    JScrollPane scrollPane;
    ConfigurableApplicationContext ctx;
    public GUIMain() {

        this.setTitle("SystemMain@DMP");
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
        this.setSize(new Dimension(screenSize.width, (screenSize.height
            - 36)));
        MainCliente mc = new MainCliente();
        //mc.setContexto(ctx);
        mc.setPreferredSize(new Dimension(1024, 600));
    }
}
```

```

jtpane = new JTabbedPane();
jtpane.setName("Cliente");
jtpane.add(mc);
scrollPane = new JScrollPane(mc);
scrollPane.setHorizontalScrollBarPolicy(
    JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);
scrollPane.setVerticalScrollBarPolicy(
    JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED);
jtpane.add(scrollPane, "Cliente");
this.add(BorderLayout.CENTER, jtpane);
this.validate();
this.repaint();
}

public void setContexto(ConfigurableApplicationContext ctx) {
    this.ctx = ctx;
}

public static void main(String[] args) {
    new GUIMain().setVisible(true);
}
}

```

Como resultado final debes tener un CRUD con interacción de Base de Datos.

#	DNI/RUC	Nombres	Tipo. Doc.
1	43631917	David Mamani Pari	Natural
3	46961417	Pedro Gomez Apaza	Natural
4	43631912	Pedro Peredo	General

**Felicitaciones**  
**Si llegaste hasta aquí**