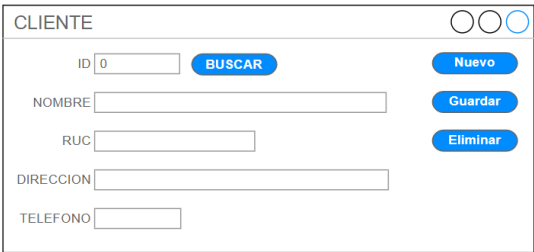


Trabajo Final – Entregable 2

Carrera:	Ingeniería de Software con IA	Semestre:	III
Curso:	Programación para desarrollo de software with oracle		
Alumno:	Principe Lino Jose Luis	ID:	1245487

CASO:

- Se tiene una base de datos con la tabla Cliente y los siguientes campos:
 - id*
 - nombre*
 - numruc*
 - direccion*
 - teléfono*
- Se desea implementar una aplicación desktop, que permita gestionar la tabla (CRUD):

Interface Gráfica del Usuario	
	<p>Buscar: Permite cargar los datos del cliente de acuerdo al ID ingresado.</p> <p>Nuevo: Limpia los controles y establece el ID en 0.</p> <p>Guardar: - Si el ID es 0 entonces y se existen datos digitados, entonces se agrega el cliente a la tabla y muestra el nuevo ID generado. - De lo contrario, si hay datos cargados de un cliente, estos se actualizan.</p> <p>Eliminar: Elimina el cliente cuyos datos se muestran cargados.</p>

- Se debe aplicar Programación Orientada a Objetos, es decir, crear una clase que contenga todas las funcionalidades para con la base de datos.
- Se debe utilizar de la librería SWING el JFrame para la interface grafica del usuario (GUI)
- Solo debe existir una clase principal con el método main.
- Utilizar package para organizar las clases.

Código de la clase Principal

```
package program;

import views.FrmCliente;

public class Principal {
    public static void main(String[] args) {
        //Instanciamos la Clase FrmCleinte del paquete views
        FrmCliente frm = new FrmCliente();
        //Llamamos al metodo para dar una localizacion a la ventana
        frm.setLocationRelativeTo(null);
        //Llamamos al metodo para que la ventana se inicialize
        frm.setVisible(true);
    }
}
```

Código de la Clase Conexión:

```
package models;

import java.sql.DriverManager;
import java.sql.Connection;

public class Conexion {

    public static Connection getConexion(){
        //Declaramos la variable para la conexion
        Connection conexion;

        //Controlamos la expecion try catch
        try{
            //Conexion a JDBC
            //Cargar el controlador
            Class.forName("com.mysql.jdbc.Driver");

            //Declaramos las variable de parametro de conexion
            String cadena = "jdbc:mysql://127.0.0.1:3306/bdpiad303";
            String usuario = "root";
            String clave = "";

            //instanaciamos y abrir la conexion
            conexion = DriverManager.getConnection(cadena, usuario, clave);

        }catch(Exception e){
            conexion = null;
            System.out.println("-- Error! --");
            System.out.println(e.getMessage());
        }

        return conexion;
    }
}
```

```
}  
}
```

Código de la clase Cliente

```
package models;  
  
import models.Conexion;  
import java.sql.Connection;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.util.Set;  
  
public class Cliente {  
    //Atributos  
    private int id;  
    private String nombre;  
    private String numruc;  
    private String direccion;  
    private String telefono;  
  
    //Encapsulamiento  
    public int getId() {  
        return id;  
    }  
    public void setId(int id) {  
        this.id = id;  
    }  
    public String getNombre() {  
        return nombre;  
    }  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
    public String getNumruc() {  
        return numruc;  
    }  
    public void setNumruc(String numruc) {  
        this.numruc = numruc;  
    }  
    public String getDireccion() {  
        return direccion;  
    }  
    public void setDireccion(String direccion) {  
        this.direccion = direccion;  
    }  
    public String getTelefono() {  
        return telefono;  
    }  
    public void setTelefono(String telefono) {  
        this.telefono = telefono;  
    }  
  
    //Constructor
```

```
public Cliente() {  
  
}  
  
public Cliente(int id, String nombre, String numruc, String direccion, String telefono) {  
    this.id = id;  
    this.nombre = nombre;  
    this.numruc = numruc;  
    this.direccion = direccion;  
    this.telefono = telefono;  
}  
  
//Metodos  
  
public Cliente getBuscarById(int idBuscar){  
    //Instanciamos la clase cliente  
    Cliente cliente = new Cliente();  
  
    try{  
        //Establecemos conexion  
        Connection cnx = Conexion.getConnection();  
        //Preparamos la instruccion sql  
        PreparedStatement ps = cnx.prepareStatement("select * from cliente where id=?");  
        //pasar los valores de los parametros sql  
        ps.setInt(1, idBuscar);  
        //Ejecutar SQL  
        ResultSet rs = ps.executeQuery();  
  
        if(rs.next()){  
            //Leer los valores de la fila  
            cliente.setId(rs.getInt("id"));  
            cliente.setNombre(rs.getString("nombre"));  
            cliente.setNumruc(rs.getString("numruc"));  
            cliente.setDireccion(rs.getString("direccion"));  
            cliente.setTelefono(rs.getString("telefono"));  
        }else{  
            //Asignamos valores por defecto  
            cliente.setId(0);  
            cliente.setNombre("");  
            cliente.setNumruc("");  
            cliente.setDireccion("");  
            cliente.setTelefono("");  
        }  
  
    }catch(Exception e){  
        //Contraolamos los errores  
        cliente.setId(-1);  
        cliente.setNombre("-- ERROR DE CONEXION --");  
        cliente.setNumruc("");  
        cliente.setDireccion("");  
        cliente.setTelefono("");  
    }  
}
```

```
return cliente;
}

public int setInsertar(Cliente cliente){
    //Instanciamos la clase cliente
    int nuevold;

    try{
        //Establecemos conexion
        Connection cnx = Conexion.getConnection();
        //Preparamos la instruccion sql
        PreparedStatement ps = cnx.prepareStatement("insert into cliente(nombre, numruc,
direccion, telefono) values(?,?,?,?);");
        //pasar los valores de los parametros sql
        ps.setString(1, cliente.getNombre());
        ps.setString(2, cliente.getNumruc());
        ps.setString(3, cliente.getDireccion());
        ps.setString(4, cliente.getTelefono());
        //Ejecutar la instruccion SQL
        ps.executeUpdate(); //INSERT-UPDATE-DELETE

        //Preparamos la instruccion sql
        ps = cnx.prepareStatement("select max(id) as nuevold from cliente");
        //Ejecutar SQL
        ResultSet rs = ps.executeQuery();//SQL

        if(rs.next()){
            //Leer nuevo id
            nuevold = rs.getInt("nuevold");
        }else{
            //Asignamos valores por defecto
            nuevold = 0;
        }

    }catch(Exception e){
        //Contraolamos los errores
        nuevold = -1;
        System.out.println("Error!");
        System.out.println(e.getMessage());
    }

    return nuevold;
}

public void setActualizar(Cliente cliente){

    try{
        //Establecemos conexion
        Connection cnx = Conexion.getConnection();
        //Preparamos la instruccion sql
        PreparedStatement ps = cnx.prepareStatement("update cliente set nombre=?,
numruc=?, direccion=?, telefono=? where id=?;");
        //pasar los valores de los parametros sql
```

```
ps.setString(1, cliente.getNombre());
ps.setString(2, cliente.getNumruc());
ps.setString(3, cliente.getDireccion());
ps.setString(4, cliente.getTelefono());
ps.setInt(5, cliente.getId());
//Ejecutar la instruccion SQL
ps.executeUpdate(); //INSERT-UPDATE-DELETE

}catch(Exception e){
    //Contraolamos los errores
    System.out.println("Error!");
    System.out.println(e.getMessage());
}

}

public void setEliminar(int idEliminar){
    try{
        //Establecemos conexion
        Connection cnx = Conexion.getConnection();
        //Preparamos la instruccion sql
        PreparedStatement ps = cnx.prepareStatement("delete from cliente where id=?;");
        //Pasar los valores de los parametros SQL
        ps.setInt(1, idEliminar);
        //Ejecutar la instruccion SQL
        ps.executeUpdate(); //INSERT-UPDATE-DELETE
    }catch(Exception e){
        //Contraolamos los errores
        System.out.println("Error!");
        System.out.println(e.getMessage());
    }
}
}
```

Código de la clase MyButton

```
package views;

import java.awt.Color;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.RenderingHints;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import javax.swing.JButton;

public class MyButton extends JButton {
```

```
public boolean isOver() {
    return over;
}

public void setOver(boolean over) {
    this.over = over;
}

public Color getColor() {
    return color;
}

public void setColor(Color color) {
    this.color = color;
    setBackground(color);
}

public Color getColorOver() {
    return colorOver;
}

public void setColorOver(Color colorOver) {
    this.colorOver = colorOver;
}

public Color getColorClick() {
    return colorClick;
}

public void setColorClick(Color colorClick) {
    this.colorClick = colorClick;
}

public Color getBorderColor() {
    return borderColor;
}

public void setBorderColor(Color borderColor) {
    this.borderColor = borderColor;
}

public int getRadius() {
    return radius;
}

public void setRadius(int radius) {
    this.radius = radius;
}

public MyButton() {
    // Init Color

    setColor(Color.WHITE);
    colorOver = new Color(179, 250, 160);
}
```

```
colorClick = new Color(152, 184, 144);
borderColor = new Color(30, 136, 56);
setContentAreaFilled(false);
// Add event mouse
addMouseListener(new MouseAdapter() {
    @Override
    public void mouseEntered(MouseEvent me) {
        setBackground(colorOver);
        over = true;
    }

    @Override
    public void mouseExited(MouseEvent me) {
        setBackground(color);
        over = false;
    }

    @Override
    public void mousePressed(MouseEvent me) {
        setBackground(colorClick);
    }

    @Override
    public void mouseReleased(MouseEvent me) {
        if (over) {
            setBackground(colorOver);
        } else {
            setBackground(color);
        }
    }
});
}

private boolean over;
private Color color;
private Color colorOver;
private Color colorClick;
private Color borderColor;
private int radius = 0;

@Override
protected void paintComponent(Graphics grphcs) {
    Graphics2D g2 = (Graphics2D) grphcs;
    g2.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
RenderingHints.VALUE_ANTIALIAS_ON);
    // Paint Border
    g2.setColor(borderColor);
    g2.fillRoundRect(0, 0, getWidth(), getHeight(), radius, radius);
    g2.setColor(getBackground());
    // Border set 2 Pix
    g2.fillRoundRect(2, 2, getWidth() - 4, getHeight() - 4, radius, radius);
    super.paintComponent(grphcs);
}
```



```
}
```

Código de la clase FrmCliente:

```
package views;

import views.MyButton;
import models.Cliente;
import models.Conexion;

public class FrmCliente extends javax.swing.JFrame {

    public FrmCliente() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jSeparator1 = new javax.swing.JSeparator();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        txtId = new javax.swing.JTextField();
        txtNombre = new javax.swing.JTextField();
        txtRuc = new javax.swing.JTextField();
        txtDireccion = new javax.swing.JTextField();
        txtTelefono = new javax.swing.JTextField();
        myButton6 = new views.MyButton();
        myButton7 = new views.MyButton();
        myButton8 = new views.MyButton();
        btnBuscar = new javax.swing.JButton();
        btnNuevo = new javax.swing.JButton();
        btnGuardar = new javax.swing.JButton();
        btnEliminar = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

        jLabel1.setFont(new java.awt.Font("Segoe UI", 0, 16)); // NOI18N
        jLabel1.setText("CLIENTE");

        jLabel2.setText("ID");

        jLabel3.setText("NOMBRE");

        myButton6.setText("BUSCAR");
        myButton7.setText("NUEVO");
        myButton8.setText("GUARDAR");
        btnBuscar.setText("BUSCAR");
        btnNuevo.setText("NUEVO");
        btnGuardar.setText("GUARDAR");
        btnEliminar.setText("ELIMINAR");

        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, 200, Short.MAX_VALUE)
                    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(jLabel5, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(jLabel6, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(txtId, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(txtNombre, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(txtRuc, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(txtDireccion, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(txtTelefono, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(myButton6, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(myButton7, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(myButton8, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(btnBuscar, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(btnNuevo, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(btnGuardar, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                    .addComponent(btnEliminar, javax.swing.GroupLayout.DEFAULT_SIZE, 100, Short.MAX_VALUE)
                )
                .addContainerGap()
            )
        );
        jPanel1Layout.setVerticalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jLabel1)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel2)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel3)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel4)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel5)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel6)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(txtId)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(txtNombre)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(txtRuc)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(txtDireccion)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(txtTelefono)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(myButton6)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(myButton7)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(myButton8)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(btnBuscar)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(btnNuevo)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(btnGuardar)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(btnEliminar)
                .addContainerGap(100, Short.MAX_VALUE)
            )
        );
    }
}
```

```
jLabel4.setText("RUC");

jLabel5.setText("DIRECCION");

jLabel6.setText("TELEFONO");

txtId.setText("0");

txtNombre.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtNombreActionPerformed(evt);
    }
});

myButton6.setBorderColor(new java.awt.Color(204, 204, 204));
myButton6.setRadius(50);
myButton6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        myButton6ActionPerformed(evt);
    }
});

myButton7.setBorderColor(new java.awt.Color(102, 204, 255));
myButton7.setRadius(50);

myButton8.setBorderColor(new java.awt.Color(204, 204, 204));
myButton8.setRadius(50);

btnBuscar.setBackground(new java.awt.Color(102, 204, 255));
btnBuscar.setForeground(new java.awt.Color(255, 255, 255));
btnBuscar.setText("BUSCAR");
btnBuscar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnBuscarActionPerformed(evt);
    }
});

btnNuevo.setBackground(new java.awt.Color(102, 204, 255));
btnNuevo.setForeground(new java.awt.Color(255, 255, 255));
btnNuevo.setText("Nuevo");
btnNuevo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnNuevoActionPerformed(evt);
    }
});

btnGuardar.setBackground(new java.awt.Color(102, 204, 255));
btnGuardar.setForeground(new java.awt.Color(255, 255, 255));
btnGuardar.setText("Guardar");
btnGuardar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnGuardarActionPerformed(evt);
    }
});
```

```

btnEliminar.setBackground(new java.awt.Color(102, 204, 255));
btnEliminar.setForeground(new java.awt.Color(255, 255, 255));
btnEliminar.setText("Eliminar");
btnEliminar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnEliminarActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jSeparator1)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(14, 14, 14)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                        .addGroup(jPanel1Layout.createSequentialGroup()
                            .addComponent(jLabel3)
                            .addComponent(jLabel2)
                            .addGap(18, 18, 18))
                        .addGroup(jPanel1Layout.createSequentialGroup()
                            .addComponent(jLabel6)
                            .addComponent(jLabel5)
                            .addComponent(jLabel4)
                            .addGap(18, 18, 18)))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(txtNombre)
                        .addGap(14, 14, 14))
                    .addComponent(txtRuc,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 201,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(14, 14, 14))
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addComponent(txtNombre)
                    .addGap(14, 14, 14))
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addComponent(txtRuc,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 201,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(14, 14, 14))
            )
        )
    );

```

```
.addGroup(jPanel1Layout.createSequentialGroup()  
    .addComponent(txtId,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
    .addGap(35, 35, 35)  
    .addComponent(btnBuscar,  
javax.swing.GroupLayout.PREFERRED_SIZE, 83,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
    .addComponent(txtDireccion,  
javax.swing.GroupLayout.PREFERRED_SIZE, 240,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
    .addGap(0, 10, Short.MAX_VALUE)))  
.addGap(40, 40, 40)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING  
, false)  
    .addComponent(btnGuardar,  
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,  
Short.MAX_VALUE)  
    .addComponent(btnNuevo, javax.swing.GroupLayout.DEFAULT_SIZE,  
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)  
    .addComponent(btnEliminar,  
javax.swing.GroupLayout.DEFAULT_SIZE, 80, Short.MAX_VALUE)))  
.addGroup(jPanel1Layout.createSequentialGroup()  
    .addComponent(txtTelefono,  
javax.swing.GroupLayout.PREFERRED_SIZE, 100,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
    .addGap(0, 0, Short.MAX_VALUE)))  
.addGroup(jPanel1Layout.createSequentialGroup()  
    .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 87,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,  
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)  
    .addComponent(myButton8, javax.swing.GroupLayout.PREFERRED_SIZE,  
22, javax.swing.GroupLayout.PREFERRED_SIZE)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
    .addComponent(myButton6, javax.swing.GroupLayout.PREFERRED_SIZE,  
22, javax.swing.GroupLayout.PREFERRED_SIZE)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
    .addComponent(myButton7, javax.swing.GroupLayout.PREFERRED_SIZE,  
22, javax.swing.GroupLayout.PREFERRED_SIZE)))  
.addContainerGap()  
);  
jPanel1Layout.setVerticalGroup(  
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(jPanel1Layout.createSequentialGroup()  
        .addContainerGap()  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING  
)  
    .addComponent(myButton7, javax.swing.GroupLayout.PREFERRED_SIZE, 22,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addComponent(jLabel1)
.addComponent(myButton6, javax.swing.GroupLayout.PREFERRED_SIZE, 22,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(myButton8, javax.swing.GroupLayout.PREFERRED_SIZE, 22,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(14, 14, 14)
.addComponent(jSeparator1, javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel2)
.addComponent(txtId, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(btnBuscar)
.addComponent(btnNuevo))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel3)
.addComponent(txtNombre, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(btnGuardar))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(txtRuc, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(btnEliminar)
.addComponent(jLabel4))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel5)
.addComponent(txtDireccion, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel6)
.addComponent(txtTelefono, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addContainerGap(14, Short.MAX_VALUE))
);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);

pack();
} // </editor-fold>

private void myButton6ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void txtNombreActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void btnBuscarActionPerformed(java.awt.event.ActionEvent evt) {
    //Entrada
    int idBuscar = Integer.parseInt(txtId.getText());
    //Proceso
    //Instanciamos la clase cliente
    Cliente cli = new Cliente();
    //Ejecutar el metodo de busqueda
    cli = cli.getBuscarById(idBuscar);
    //Salida
    txtId.setText(String.valueOf(cli.getId()));
    txtNombre.setText(cli.getNombre());
    txtRuc.setText(cli.getNumruc());
    txtDireccion.setText(cli.getDireccion());
    txtTelefono.setText(cli.getTelefono());
}

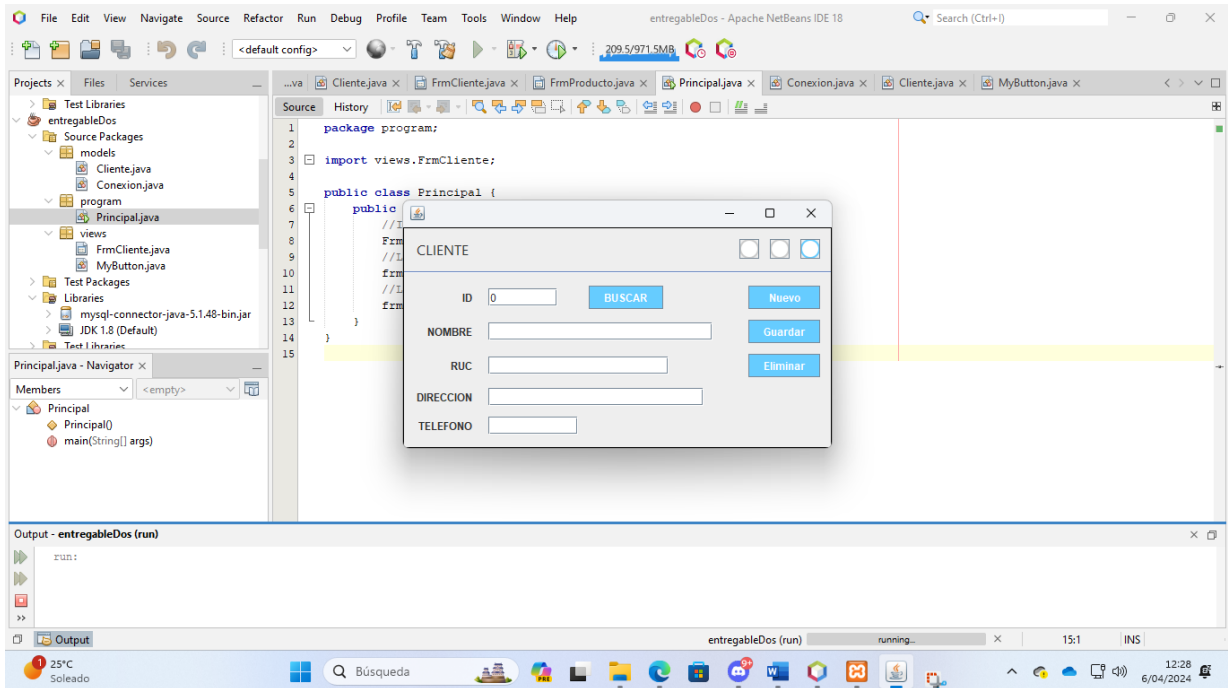
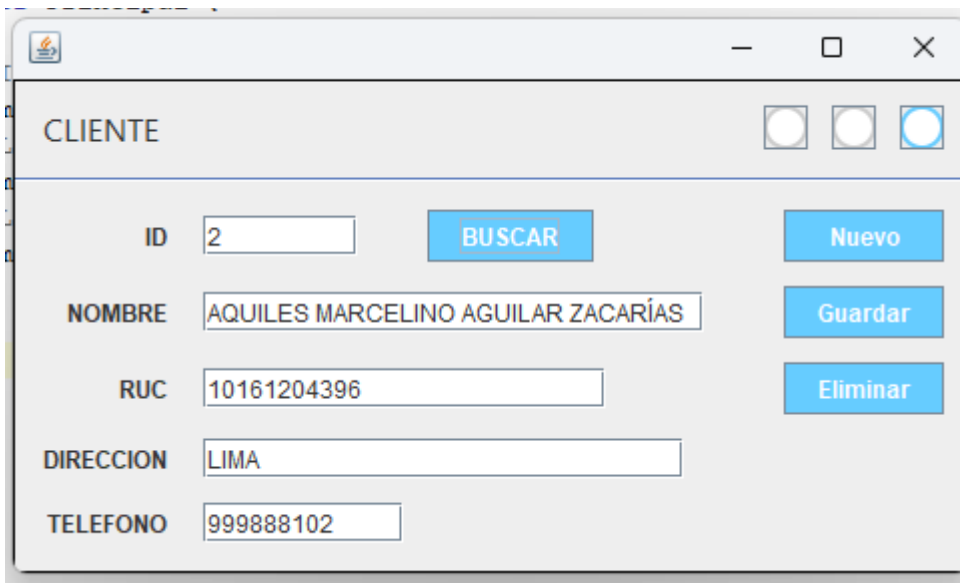
private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {
    //Valores por defeto
    txtId.setText("0");
    txtNombre.setText("");
    txtRuc.setText("");
    txtDireccion.setText("");
    txtTelefono.setText("");
}

private void btnGuardarActionPerformed(java.awt.event.ActionEvent evt) {
    //Entrada
    int id = Integer.parseInt(txtId.getText());
    String nombre = txtNombre.getText();
    String numruc = txtRuc.getText();
    String direccion = txtDireccion.getText();
    String telefono = txtTelefono.getText();
    //Instanciamos la clase Ciente
```

```
Cliente cli = new Cliente(id, nombre, numruc, direccion, telefono);
//Proceso
if(txtId.getText().equals("0")){
    //Insertar nuevo registro
    int nuevold = cli.setInsertar(cli);
    //Mostrar el nuevo id
    txtId.setText(String.valueOf(nuevold));
}else{
    //Actualizar el registro actual
    cli.setActualizar(cli);
}
//Salida
}

private void btnEliminarActionPerformed(java.awt.event.ActionEvent evt) {
    //Entrada
    int id = Integer.parseInt(txtId.getText());
    Cliente cli = new Cliente();
    //Proceso
    cli.setEliminar(id);
    //Salida
    btnNuevoActionPerformed(null);
}

// Variables declaration - do not modify
private javax.swing.JButton btnBuscar;
private javax.swing.JButton btnEliminar;
private javax.swing.JButton btnGuardar;
private javax.swing.JButton btnNuevo;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JPanel jPanel1;
private javax.swing.JSeparator jSeparator1;
private views.MyButton myButton6;
private views.MyButton myButton7;
private views.MyButton myButton8;
private javax.swing.JTextField txtDireccion;
private javax.swing.JTextField txtId;
private javax.swing.JTextField txtNombre;
private javax.swing.JTextField txtRuc;
private javax.swing.JTextField txtTelefono;
// End of variables declaration
}
```


Ejecución del código:**BUSCAR:**

CLIENTE

ID: 2 **BUSCAR** **Nuevo**


NOMBRE: AQUILES MARCELINO AGUILAR ZACARÍAS **Guardar**




RUC: 10161204396 **Eliminar**

DIRECCION: LIMA

TELEFONO: 999888102

GUARDAR:

 — □ ×

CLIENTE   

ID

BUSCAR

Nuevo

NOMBRE


Guardar




RUC

Eliminar

DIRECCION

TELEFONO

 — □ ×

CLIENTE   

ID

BUSCAR

Nuevo

NOMBRE

Guardar

RUC

Eliminar

DIRECCION

TELEFONO

ELIMINAR:

CLIENTE

ID

4

BUSCAR

Nuevo

NOMBRE

CARROCERIAS INTEGRADAS S.A. - CAISA

Guardar

RUC

20331634281

Eliminar

DIRECCION

LIMA

TELEFONO

999888104

CLIENTE

ID

0

BUSCAR

Nuevo

NOMBRE

Guardar

RUC

Eliminar

DIRECCION

TELEFONO

OPCIONAL:

Esta tarea es opcional y representa puntaje adicional.

Implementar una aplicación de consola utilizando Programación Orientada a Objetos que realice la gestión de la base de datos:

Menu Principal =====	Buscar: Permite ingresar un ID y mostrar los datos del cliente. Después de una pausa retorna al MENU.
1 – Buscar	Editar:
2 – Editar	Permite ingresar el ID de un cliente, se muestra los datos del cliente y luego se debe poder digitar los nuevos datos. Al finalizar, los nuevos datos se actualizan en la base de datos y retorna al MENU.
3 – Nuevo	Nuevo:
4 – Eliminar	Se solicita el ingreso de los DATOS de un nuevo cliente. Al finalizar los datos se agregan a la base de datos y se muestra el NUEVO ID generado. Después de una pausa retorna al MENU.
0 – Salir	Eliminar:
Opcion: _	Permite el ID del Cliente que se desea Eliminar. Se ejecuta la eliminación y se retorna al MENU.

Entregable:

Presentar en un archivo de Ms Word/PDF:

- El código fuente de cada clase.
- Pantallazos de la ejecución de la aplicación.

Código de la Clase Principal:

```
package program;  
  
import models.InterfazConsola;  
  
public class Principal {  
    public static void main(String[] args) {  
  
        //Instanciamos la clase Interfaz  
        InterfazConsola interfazC = new InterfazConsola();  
        //Llamamos al metodo para que inicie el objeto  
        interfazC.opciones();  
    }  
}
```

Código de la clase Conexión:

```
package models;

import java.sql.DriverManager;
import java.sql.Connection;

public class Conexion {

    public static Connection getConexion(){
        //Declaramos la variable para la conexion
        Connection conexion;

        //Controlamos la excepcion try catch
        try{
            //Conexion a JDBC
            //Cargar el controlador
            Class.forName("com.mysql.jdbc.Driver");

            //Declaramos las variable de parametro de conexion
            String cadena = "jdbc:mysql://127.0.0.1:3306/bdpiad303";
            String usuario = "root";
            String clave = "";

            //instanciamos y abrir la conexion
            conexion = DriverManager.getConnection(cadena, usuario, clave);

        }catch(Exception e){
            conexion = null;
            System.out.println("-- Error! --");
            System.out.println(e.getMessage());
        }

        return conexion;
    }
}
```

Código de la clase Cliente:

```
package models;

import models.Conexion;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.Set;
import java.util.Scanner;

public class Cliente {
```

```
//Atributos
private int id;
private String nombre;
private String numruc;
private String direccion;
private String telefono;

//Encapsulamiento
public int getId() {
    return id;
}
public void setId(int id) {
    this.id = id;
}
public String getNombre() {
    return nombre;
}
public void setNombre(String nombre) {
    this.nombre = nombre;
}
public String getNumruc() {
    return numruc;
}
public void setNumruc(String numruc) {
    this.numruc = numruc;
}
public String getDireccion() {
    return direccion;
}
public void setDireccion(String direccion) {
    this.direccion = direccion;
}
public String getTelefono() {
    return telefono;
}
public void setTelefono(String telefono) {
    this.telefono = telefono;
}

//Constructor
public Cliente() {

}

public Cliente(int id, String nombre, String numruc, String direccion, String telefono) {
    this.id = id;
    this.nombre = nombre;
    this.numruc = numruc;
    this.direccion = direccion;
    this.telefono = telefono;
}
```

```
}

//Metodos

public Cliente getBuscarById(int idBuscar){
    //Instanciamos la clase cliente
    Cliente cliente = new Cliente();

    try{
        //Establecemos conexion
        Connection cnx = Conexion.getConnection();
        //Preparamos la instruccion sql
        PreparedStatement ps = cnx.prepareStatement("select * from cliente where id=?");
        //pasar los valores de los parametros sql
        ps.setInt(1, idBuscar);
        //Ejecutar SQL
        ResultSet rs = ps.executeQuery();

        if(rs.next()){
            //Leer los valores de la fila
            cliente.setId(rs.getInt("id"));
            cliente.setNombre(rs.getString("nombre"));
            cliente.setNumruc(rs.getString("numruc"));
            cliente.setDireccion(rs.getString("direccion"));
            cliente.setTelefono(rs.getString("telefono"));
        }else{
            //Asignamos valores por defecto
            cliente.setId(0);
            cliente.setNombre("");
            cliente.setNumruc("");
            cliente.setDireccion("");
            cliente.setTelefono("");
        }

    }

    catch(Exception e){
        //Contraolamos los errores
        cliente.setId(-1);
        cliente.setNombre("-- ERROR DE CONEXION --");
        cliente.setNumruc("");
        cliente.setDireccion("");
        cliente.setTelefono("");
    }

    return cliente;
}
```

```
public int setInsertar(Cliente cliente){
    //Instanciamos la clase cliente
    int nuevold;

    try{
        //Establecemos conexion
        Connection cnx = Conexion.getConnection();
        //Preparamos la instruccion sql
        PreparedStatement ps = cnx.prepareStatement("insert into cliente(nombre, numruc,
direccion, telefono) values(?,?,?,?);");
        //pasar los valores de los parametros sql
        ps.setString(1, cliente.getNombre());
        ps.setString(2, cliente.getNumruc());
        ps.setString(3, cliente.getDireccion());
        ps.setString(4, cliente.getTelefono());
        //Ejecutar la instruccion SQL
        ps.executeUpdate(); //INSERT-UPDATE-DELETE

        //Preparamos la instruccion sql
        ps = cnx.prepareStatement("select max(id) as nuevold from cliente");
        //Ejecutar SQL
        ResultSet rs = ps.executeQuery();//SQL

        if(rs.next()){
            //Leer nuevo id
            nuevold = rs.getInt("nuevold");
        }else{
            //Asignamos valores por defecto
            nuevold = 0;
        }

    }

    }catch(Exception e){
        //Contraolamos los errores
        nuevold = -1;
        System.out.println("Error!");
        System.out.println(e.getMessage());
    }

    return nuevold;
}

public void setAtcualizar(Cliente cliente){

    try{
        //Establecemos conexion
        Connection cnx = Conexion.getConnection();
        //Preparamos la instruccion sql
```

```
PreparedStatement ps = cnx.prepareStatement("update cliente set nombre=?,  
numruc=?, direccion=?, telefono=? where id=?;");
```

```
//pasar los valores de los parametros sql
```

```
ps.setString(1, cliente.getNombre());
```

```
ps.setString(2, cliente.getNumruc());
```

```
ps.setString(3, cliente.getDireccion());
```

```
ps.setString(4, cliente.getTelefono());
```

```
ps.setInt(5, cliente.getId());
```

```
//Ejecutar la instruccion SQL
```

```
ps.executeUpdate(); //INSERT-UPDATE-DELETE
```

```
}catch(Exception e){
```

```
//Contraolamos los errores
```

```
System.out.println("Error!");
```

```
System.out.println(e.getMessage());
```

```
}
```

```
}
```

```
public void setEliminar(int idEliminar){
```

```
try{
```

```
//Establecemos conexion
```

```
Connection cnx = Conexion.getConnection();
```

```
//Preparamos la instruccion sql
```

```
PreparedStatement ps = cnx.prepareStatement("delete from cliente where id=?;");
```

```
//Pasar los valores de los parametros SQL
```

```
ps.setInt(1, idEliminar);
```

```
//Ejecutar la instruccion SQL
```

```
ps.executeUpdate(); //INSERT-UPDATE-DELETE
```

```
}catch(Exception e){
```

```
//Contraolamos los errores
```

```
System.out.println("Error!");
```

```
System.out.println(e.getMessage());
```

```
}
```

```
}
```

```
}
```

Código de la clase InterfazConsola:

```
package models;
```

```
import java.util.Scanner;
```

```
import models.Cliente;
```

```
public class InterfazConsola {
```

```
    public void opciones(){
```

```
        //Instanciamos la clase escaner
```



```
Scanner tec = new Scanner(System.in);
//Instanciamos la clase Cliente
Cliente cliente = new Cliente();
//Instanciamos nuestra misma clase
InterfazConsola interfazC = new InterfazConsola();

//Menu a mostrar en pantalla
System.out.println("Menu Principal");
System.out.println("=====");
System.out.println("1 - Buscar");
System.out.println("2 - Editar");
System.out.println("3 - Nuevo");
System.out.println("4 - Eliminar");
System.out.println("0 - Salir");
System.out.print("Opcion : ");
//Declaramos y asignamos un valor desde teclado
int opcion = tec.nextInt();

//Condicionales
if(opcion == 1){
    //En caso se 1 el programa debe buscar
    System.out.println("=====");
    System.out.println("----- BUSCAR -----");
    //Creamos y asignamos valor al id que usaremos para buscar el registro
    System.out.print("ID:");
    int buscarId = tec.nextInt();
    //Llamamos al metodo getBuscarById y le pasamos el parametro
    //Recogemos el valor que este caso es del tipo Cliente y lo asignamos a la variable ya
creada
    cliente = cliente.getBuscarById(buscarId);
    //Recogemos los datos y los mostramos en pantalla
    System.out.println("Nombre: " + cliente.getNombre());
    System.out.println("Numruc: " + cliente.getNumruc());
    System.out.println("Direccion: " + cliente.getDireccion());
    System.out.println("Telefono: " + cliente.getTelefono());
    System.out.println("=====");

    //Llamamos al metodo de opciones()
    interfazC.opciones();

}else if(opcion == 2){
    //En caso opcion sea 2 se debera editar los valores
    System.out.println("=====");
    System.out.println("----- EDITAR VALORES -----");
    System.out.println("Dejar el espacio en blanco si no se desea Editar el valor");
    //Creamos y asignamos valor al id que usaremos para buscar el registro
    System.out.print("Ingrese el ID del cliente que desees editar: ");
    int buscarId = tec.nextInt();
    cliente = cliente.getBuscarById(buscarId);
```

```
//Mostrar el valor actual
System.out.println("Nombre: " + cliente.getNombre());
tec.nextLine();
//Pedir nuevo valor
System.out.print("Nuevo Nombre: ");
String nombre = tec.nextLine();
//Verificamos si el valor cambio para guardarlo
if (!nombre.isEmpty()) {
    cliente.setNombre(nombre);
}
System.out.println("Numruc: " + cliente.getNumruc());
System.out.print("Nuevo RUC : ");
String numruc = tec.nextLine();
if (!numruc.isEmpty()) {
    cliente.setNumruc(numruc);
}
System.out.println("Direccion: " + cliente.getDireccion());
System.out.print("Nueva Direccion: ");
String direccion = tec.nextLine();
if (!direccion.isEmpty()) {
    cliente.setDireccion(direccion);
}
System.out.println("Telefono: " + cliente.getTelefono());
System.out.print("Nuevo Telefono: ");
String telefono = tec.nextLine();
if (!telefono.isEmpty()) {
    cliente.setTelefono(telefono);
}

//Llamamos al metodo para actualizar la tabla
cliente.setActualizar(cliente);

//Mostramos los valores actualizados
System.out.println("=====");
System.out.println("----- VALORES ACTUALIZADOS -----");
System.out.println("ID: " + cliente.getId());
System.out.println("Nombre: " + cliente.getNombre());
System.out.println("Numruc: " + cliente.getNumruc());
System.out.println("Direccion: " + cliente.getDireccion());
System.out.println("Telefono: " + cliente.getTelefono());
System.out.println("=====");

//LLamamos al metodo de opciones()
interfazC.opciones();

}else if(opcion == 3){
    //La opcion 3 es para guardar un nuevo cliente
    System.out.println("=====");
    System.out.println("----- NUEVO CLIENTE -----");
```

```
tec.nextLine();
//Recogemos los datos
System.out.print("Nombre: ");
String nombre = tec.nextLine();
System.out.print("Numruc: ");
String numruc = tec.next();
System.out.print("Direccion: ");
String direccion = tec.next();
tec.nextLine();
System.out.print("Telefono: ");
String telefono = tec.next();

//Instanciamos el constructor y le pasamos los atributos
Cliente cli = new Cliente(0, nombre, numruc, direccion, telefono);
//Insertamos el nuevo registro
int nuevold = cli.setInsertar(cli);
//Guardamos el valor del nuevold
cli.setId(nuevold);

//Mostramos los valores guardados
System.out.println("=====");
System.out.println("----- VALORES GUARDADOS -----");
System.out.println("Nuevo ID: " + cli.getId());
System.out.println("Nombre: " + cli.getNombre());
System.out.println("Numruc: " + cli.getNumruc());
System.out.println("Direccion: " + cli.getDireccion());
System.out.println("Telefono: " + cli.getTelefono());
System.out.println("=====");

//LLamamos al metodo de opciones()
interfazC.opciones();

}else if(opcion == 4){
    //Opcion 4 eliminara a un cliente
    System.out.println("=====");
    System.out.println("----- ELIMINAR -----");
    //Recogemos el id
    System.out.print("Ingrese ID: ");
    int eliminarId = tec.nextInt();
    //Llamamos al metodo eliminar y pasamos el parametro
    cliente.setEliminar(eliminarId);
    System.out.println("=====");

    //LLamamos al metodo de opciones()
    interfazC.opciones();
}else if(opcion == 0){
    //Salir del sistema por medio de un return
    System.out.println("=====SALIO DEL SISTEMA=====");
    return;
}else{
```

//En caso la opcion no exista

```
System.out.println("=====");
System.out.println("Opcion no encontrada - Ingrese otra vez la opción");

System.out.println("=====");
interfazC.opciones();
    }
}
}
```

Ejecución del código:

Opción Buscar:

```
run:
Menu Principal
=====
1 - Buscar
2 - Editar
3 - Nuevo
4 - Eliminar
0 - Salir
Opcion : 1
=====
----- BUSCAR -----
ID:2
Nombre: AQUILES MARCELINO AGUILAR ZACARÍAS
Numruc: 10161204396
Direccion: LIMA
Telefono: 999888102
=====
Menu Principal
=====
```

Opción Editar:

```
Menu Principal
=====
1 - Buscar
2 - Editar
3 - Nuevo
4 - Eliminar
0 - Salir
Opcion : 2
=====
----- EDITAR VALORES -----
Dejar el espacio en blanco si no se dese Editar el valor
Ingrese el ID del cliente que deseas editar: 2
Nombre: AQUILES MARCELINO AGUILAR ZACARÍAS
Nuevo Nombre:
Numruc: 10161204396
Nuevo RUC : 10161204369
Direccion: LIMA
Nueva Direccion: CALLAO
Telefono: 999888102
Nuevo Telefono:
=====
----- VALORES ACTUALIZADOS -----
ID: 2
Nombre: AQUILES MARCELINO AGUILAR ZACARÍAS
Numruc: 10161204369
Direccion: CALLAO
Telefono: 999888102
=====
Menu Principal
=====
```

Opción Nuevo:

```
Menu Principal
=====
1 - Buscar
2 - Editar
3 - Nuevo
4 - Eliminar
0 - Salir
Opcion : 3
=====
----- NUEVO CLIENTE -----
Nombre: TIENDAS TAMBO
Numruc: 1023432131
Direccion: SANTA ANITA
Telefono: 987654312
=====
----- VALORES GUARDADOS -----
Nuevo ID: 121
Nombre: TIENDAS TAMBO
Numruc: 1023432131
Direccion: SANTA
Telefono: 987654312
=====
Menu Principal
=====
```

Opción Eliminar:

```
Menu Principal
=====
1 - Buscar
2 - Editar
3 - Nuevo
4 - Eliminar
0 - Salir
Opcion : 4
=====
----- ELIMINAR -----
Ingrese ID: 121
=====
Menu Principal
=====
1 - Buscar
2 - Editar
3 - Nuevo
4 - Eliminar
0 - Salir
Opcion : 1
=====
----- BUSCAR -----
ID:121
Nombre:
Numruc:
Direccion:
Telefono:
=====
Menu Principal
```

Opción Salir:

```
=====
Menu Principal
=====
1 - Buscar
2 - Editar
3 - Nuevo
4 - Eliminar
0 - Salir
Opcion : 0
=====SALIO DEL SISTEMA=====
BUILD SUCCESSFUL (total time: 2 minutes 56 seconds)
```