



UEA: SISTEMAS DISTRIBUIDOS

Profesor: Netz Romero

NOMBRE: Aldo Enrique Hernández Flores

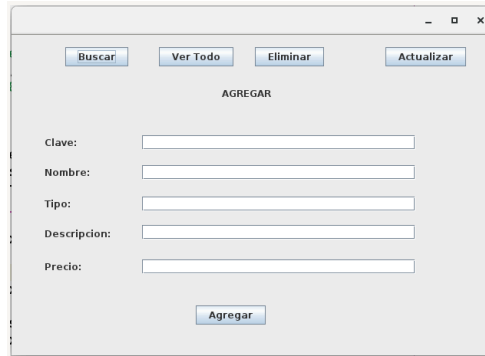
MATRICULA: 2193035908

PROYECTO Sistemas Distribuidos

Operación Inserción

La operación insertar, realizará la inserción de un registro proporcionado por la interfaz gráfica hacia Tuxedo y éste realizará la operación para un registro en una tabla de la BD.

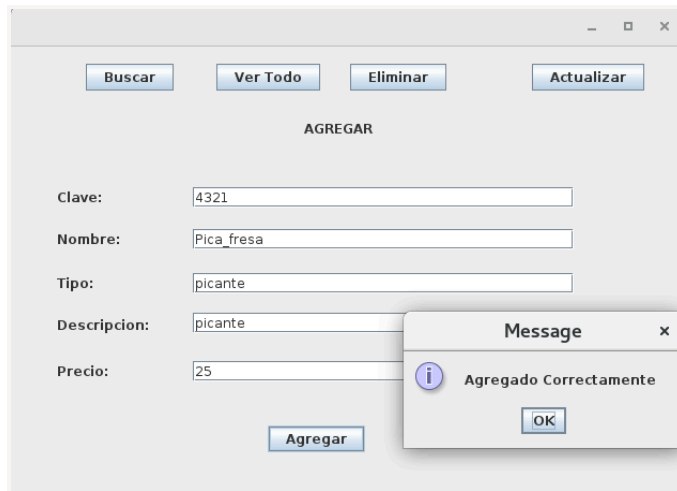
La interfaz puede ser como la que se muestra a continuación.



Código de la interfaz:

```
211 private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {  
212     //Generamos el archivo ud32  
213     String nombreArchivo = "Ud32_Agregar";  
214     String carpeta = System.getProperty("user.dir");  
215     String direccionCompleta = "/home/oracle/tuxedo3/ud/" + nombreArchivo + ".ud";  
216     FileWriter Ubicacion = null;  
217     try {  
218         Ubicacion = new FileWriter(direccionCompleta);  
219     } catch (IOException ex) {  
220         System.out.println("Error Carpeta:" + ex);  
221     }  
222     try {  
223         BufferedWriter escritor = new BufferedWriter(Ubicacion);  
224         String Cadena = "SRVCNM\t" + "INSERT_FML32\n"  
225             + "CLAVE\t" + txt_clave.getText() + "\n"  
226             + "NOMBRE\t" + txt_nombre.getText() + "\n"  
227             + "TIPO\t" + txt_tipo.getText() + "\n"  
228             + "DESC\t" + txt_descripcion.getText() + "\n"  
229             + "PRECIO\t" + txt_precio.getText() + "\n\n";  
230         escritor.write(Cadena);  
231         JOptionPane.showMessageDialog(null, "Agregado Correctamente");  
232         escritor.close();  
233     } catch (Exception ex) {  
234         System.out.println("Error Contenido:" + ex);  
235     }  
236     Process proc = null;  
237     try {  
238         proc = Runtime.getRuntime().exec("/home/oracle/tuxedo3/ud/" + "scrip1.sh");  
239     } catch (Exception ex) {  
240         System.out.println("Error" + ex);  
241     }  
242     BufferedReader buf = new BufferedReader(new InputStreamReader(proc.getInputStream()));  
243     String s;  
244     try {  
245         while ((s = buf.readLine()) != null) {  
246             System.out.println(s);  
247         }  
248     } catch (IOException ex) {  
249         System.out.println(ex.toString());  
250     }  
251 }  
252 }  
253 }  
254 }  
255 }
```

Insertión de datos a través de la interfaz:



Buscar Ver Todo Eliminar Actualizar

AGREGAR

Clave: 4321

Nombre: Pica_fresa

Tipo: picante

Descripción: picante

Precio: 25

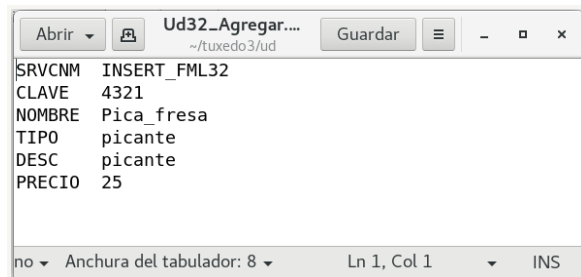
Agregar

Message x

Agregado Correctamente

OK

Archivo ud32:



Abrir Ud32_Agregar... Guardar

~/tuxedo3/ud

SRVCNM INSERT_FML32

CLAVE 4321

NOMBRE Pica_fresa

TIPO picante

DESC picante

PRECIO 25

no Anchura del tabulador: 8 Ln 1, Col 1 INS

Código de Insert:

src > srcTuxedoBD01.pc

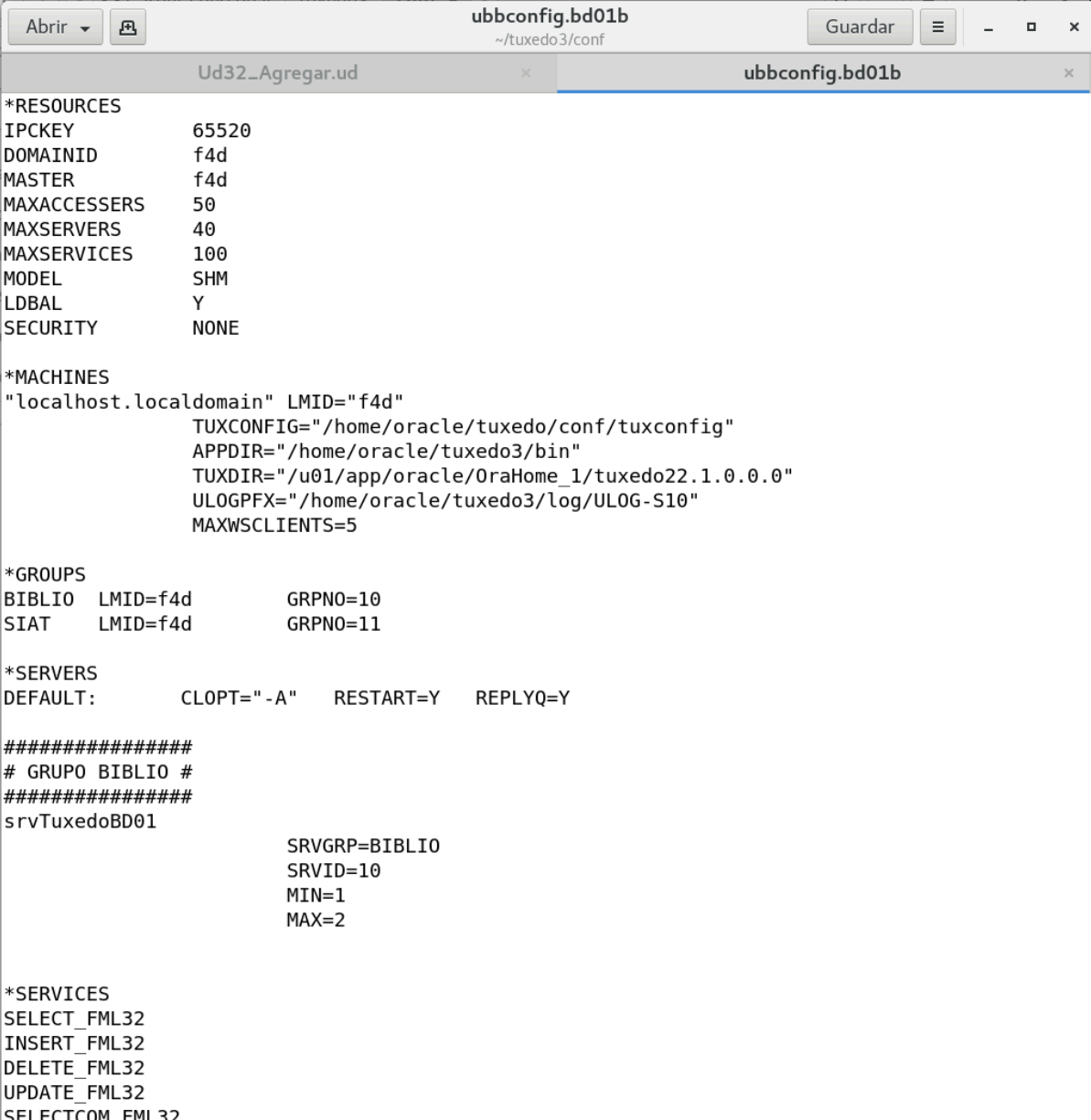
```
93 void INSERT_FML32(TPSVCINFO *rqst)
94 {
95     FLDLEN32      flen;
96     paramConexion svL_paramCon;
97     char          msgbuf[256];
98
99     EXEC SQL BEGIN DECLARE SECTION;
100     int t_idClave;
101     char t_nombre[51];
102     char t_tipo[151];
103     char t_desc[50];
104     long t_precio;//Nuevo
105     EXEC SQL END DECLARE SECTION;
106
107     FBFR32* fbfr = (FBFR32*)rqst->data ;
108
109     strcpy(svL_paramCon.usuario_DB, "aldoh");
110     strcpy(svL_paramCon.password_DB, "2193035908");
111     strcpy(svL_paramCon.name_DB, "orcl21");
112
113     memset(&t_idClave, 0x00, sizeof(t_idClave));
114     memset(t_nombre, 0x00, sizeof(t_nombre));
115     memset(t_tipo, 0x00, sizeof(t_tipo));
116     memset(t_desc, 0x00, sizeof(t_desc));
117     memset(&t_precio, 0x00, sizeof(t_precio));//Nuevo
118
119
120
121     userlog ("Servicio INSERT_FML32 Invocado") ;
122
123     if(Fget32 (fbfr, CLAVE, 0, (char*)&t_idClave, 0) < 0) {
124         userlog ("Error Fget32(CLAVE): %s", Fstrerror32(Ferror32)) ;
125         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
126     }
127     userlog (" CLAVE: [%d]", t_idClave) ;
128
129     if(Fget32 (fbfr, NOMBRE, 0, t_nombre, 0) < 0) {
130         userlog ("Error Fget32(NOMBRE): %s", Fstrerror32(Ferror32)) ;
131         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
132     }
133     userlog (" NOMBRE : [%s]", t_nombre) ;
134
```

```

135     if(Fget32 (fbfr, TIPO, 0, t_tipo, 0) < 0) {
136         userlog ("Error Fget32(TIPO): %s",
137             Fstrerror32(Ferror32)) ;
138         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
139     }
140     userlog (" TIPO: [%s]", t_tipo) ;
141
142     if(Fget32 (fbfr, DESC, 0, t_desc, 0) < 0) {
143         userlog ("Error Fget32(DESC): %s", Fstrerror32(Ferror32)) ;
144         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
145     }
146     userlog (" DESC  : [%s]", t_desc) ;
147
148     //Nuevo
149     if(Fget32 (fbfr, PRECIO, 0, (char*)&t_precio, 0) < 0) {
150         userlog ("Error Fget32(PRECIO): %s", Fstrerror32(Ferror32)) ;
151         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
152     }
153     userlog (" PRECIO: [%d]", t_precio) ;
154
155
156     userlog ("Conectando a la BD");
157     conectando_aBD(&svL_paramCon);
158
159     EXEC SQL AT ORACLE2
160     INSERT INTO Dulceria (id_clave, nombre_dulce, tipo_dulce, descripcion, precio)
161     VALUES (:t_idClave, :t_nombre, :t_tipo, :t_desc, :t_precio) ;
162
163     if
164     (sqlca.sqlcode != 0){
165         sprintf(msgbuf, "Fallo insert: sqlcode = %d(%s)\n",
166             sqlca.sqlcode, sqlca.sqlerrm.sqlerrmc);
167         sqlError("Error al realizar Insert");
168         Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
169         cierraConexion();
170         tpreturn(TPFAIL, 0, rqst->data, 0L, 0);
171     }
172
173     strcpy(msgbuf, "Insert exitoso!");
174     userlog ("Insert exitoso");
175     Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
176     cierraConexion();

```

Archivo de configuración:



```
*RESOURCES
IPCKEY          65520
DOMAINID        f4d
MASTER          f4d
MAXACCESSERS    50
MAXSERVERS      40
MAXSERVICES     100
MODEL           SHM
LDBAL           Y
SECURITY        NONE

*MACHINES
"localhost.localdomain" LMID="f4d"
                        TUXCONFIG="/home/oracle/tuxedo/conf/tuxconfig"
                        APPDIR="/home/oracle/tuxedo3/bin"
                        TUXDIR="/u01/app/oracle/OraHome_1/tuxedo22.1.0.0.0"
                        ULGPFX="/home/oracle/tuxedo3/log/ULOG-S10"
                        MAXWSCLIENTS=5

*GROUPS
BIBLIO  LMID=f4d      GRPN0=10
SIAT    LMID=f4d      GRPN0=11

*SERVERS
DEFAULT:      CLOPT="- A"   RESTART=Y   REPLYQ=Y

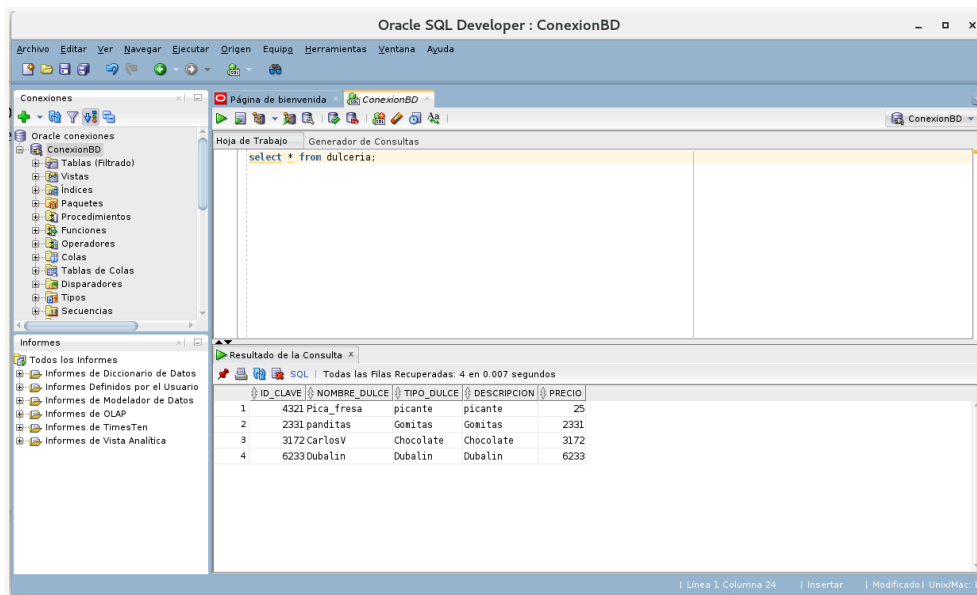
#####
# GRUPO BIBLIO #
#####
srvTuxedoBD01

                        SRVGRP=BIBLIO
                        SRVID=10
                        MIN=1
                        MAX=2

*SERVICES
SELECT_FML32
INSERT_FML32
DELETE_FML32
UPDATE_FML32
SELECTCOM_FML32
```

Descripción de la tabla Creada:

```
CREATE TABLE Dulceria( id_clave NUMBER(20),nombre_dulce VARCHAR2(20),tipo_dulce
VARCHAR2(20),descripcion VARCHAR2(20),precio NUMBER(20));
```



Como podemos observar con la interfaz podemos realizar la inserción de los datos y este se muestra visualiza en la tabla.

Operación Buscar:

Es importante que los datos que representan a un producto en el negocio, uno de ellos sea único y representa a la llave primaria en la tabla. En nuestro caso del negocio de la librería, representan libro como artículo en el negocio será el ID clave el dato para identificar un dulce. A través de este valor se realizará la búsqueda en la tabla contenida en la BD.

Interfaz select:

Para ello se debe desarrollar una interfaz que solicite el dato primario y regrese la información del producto en la misma ventana, como se muestra en la figura a continuación.

BUSCAR

Clave

Nombre:

Tipo:

Descripcion:

Precio:

Código de la Interfaz select:

```
private void BuscarActionPerformed(java.awt.event.ActionEvent evt) {  
230  
231     String nombreArchivo = "Ud32_Buscar";  
232     String carpeta = System.getProperty("user.dir");  
233     String direccionCompleta = "/home/oracle/tuxedo3/ud/" + nombreArchivo + ".ud";  
234     FileWriter Ubicacion = null;  
235     try {  
236         Ubicacion = new FileWriter(direccionCompleta);  
237     } catch (IOException ex) {  
238         System.out.println("Error Carpeta:" + ex);  
239     }  
240  
241     try {  
242         BufferedWriter escritor = new BufferedWriter(Ubicacion);  
243         String Cadena = "SRVCNM\t" + "SELECT_FML32\n"  
244             + "CLAVE\t" + txt_clave.getText() + "\n\n";  
245         escritor.write(Cadena);  
246         JOptionPane.showMessageDialog(null, "Buscar");  
247         escritor.close();  
248  
249     } catch (Exception ex) {  
250         System.out.println("Error Contenido:" + ex);  
251     }  
252  
253     Process proc = null;  
254     try {  
255         proc = Runtime.getRuntime().exec("/home/oracle/tuxedo3/ud/" + "scrip2.sh");  
256     } catch (Exception ex) {  
257         System.out.println("Error" + ex);  
258     }  
259  
260     BufferedReader buf = new BufferedReader(new InputStreamReader(proc.getInputStream()));  
261     String s;  
262     try {  
263         while ((s = buf.readLine()) != null) {  
264             System.out.println(s);  
265             txt_nombre.setText(s);  
266         }  
267     }  
268     } catch (IOException ex) {  
269         System.out.println(ex.toString());  
270     }  
271     String url = "jdbc:oracle:thin:@localhost:1521:orcl21";  
272     String usuario = "aldoh";  
273     String pass = "2193035908";  
  
274     try {  
275  
276         Connection cn = DriverManager.getConnection(url, usuario, pass);  
277         int id = Integer.parseInt(txt_clave.getText());  
278         PreparedStatement pst = cn.prepareStatement("select * from dulceria where ID_CLAVE = ?");  
279         pst.setString(1, txt_clave.getText().trim());  
280  
281         ResultSet rs = pst.executeQuery();  
282  
283         if (rs.next()) {  
284             txt_nombre.setText(rs.getString("nombre_dulce"));  
285             txt_tipo.setText(rs.getString("tipo_dulce"));  
286             txt_descripcion.setText(rs.getString("descripcion"));  
287             txt_precio.setText(rs.getString("precio"));  
288         } else {  
289             JOptionPane.showMessageDialog(null, "Registro no encontrado");  
290         }  
291     }  
292  
293     } catch (HeadlessException | SQLException e) {  
294         System.out.println("Error :" + e);  
295     }  
296  
297 }
```


Buscar de datos a través de la interfaz:

Insertar Ver Todo Eliminar Actualizar

BUSCAR

Clave

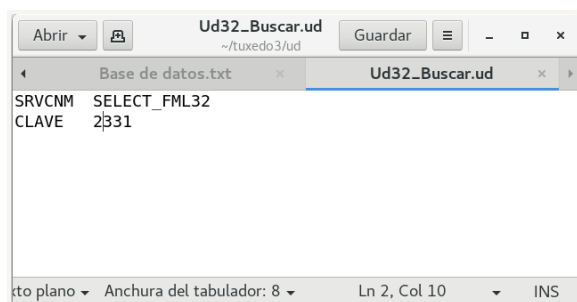
Nombre:

Tipo:

Descripcion:

Precio:

Archivo ud32:



```
SENT pkt(1) is :  
CLAVE 2331  
SRVCNM SELECT_FML32
```

```
RTN pkt(1) is :  
CLAVE 2331  
TIPO 2331  
NOMBRE panditas  
TIPO Gomitas  
TIPO Gomitas
```

Código de select:

```

181 void SELECT_FML32(TPSVCINFO *rqst)
182 {
183     FLDLEN32      flen;
184     paramConexion svL_paramCon;
185     char          msgbuf[256];
186
187     EXEC SQL BEGIN DECLARE SECTION;
188     int t_idClave;
189     char t_nombre[51];
190     char t_tipo[151];
191     char t_desc[20];
192     int t_precio[20]; //Nuevo
193     EXEC SQL END DECLARE SECTION;
194
195     FBFR32* fbfr = (FBFR32*)rqst->data ;
196
197     strcpy(svL_paramCon.usuario_DB, "aldoh");
198     strcpy(svL_paramCon.password_DB, "2193035908");
199     strcpy(svL_paramCon.name_DB, "orcl21");
200     t_idClave = 0;
201     memset(t_nombre, 0x00, sizeof(t_nombre));
202     memset(t_tipo, 0x00, sizeof(t_tipo));
203     memset(t_desc, 0x00, sizeof(t_desc));
204     memset(t_precio, 0x00, sizeof(t_precio));
205
206
207     userlog ("Servicio SELECT_FML32 Invocado") ;
208
209     /* Obtenemos los campos FML del buffer de entrada */
210     if(Fget32 (fbfr, CLAVE, 0, (char*)&t_idClave, 0) < 0) {
211         userlog ("Error Fget32(CLAVE): %s", Fstrerror32(Error32)) ;
212         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
213     }
214
215     userlog (" CLAVE: [%d]", t_idClave) ;
216
217     Finit32 (fbfr, Fsizeof32 (fbfr));
218
219     userlog ("Conectando a la BD");
220     conectando_aBD(&svL_paramCon);
221
222     /* Declaramos y Abrimos Cursor */
223
224     sprintf(msgbuf, "Fallo insert: sqlcode = %d(%s)\n",
225             sqlca.sqlcode, sqlca.sqlerrm.sqlerrmc);
226     sqlError("Error al realizar Insert");
227     Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
228     cierraConexion();
229     tpreturn(TPFAIL, 0, rqst->data, 0L, 0);
230 }
231
232     strcpy(msgbuf, "Insert exitoso!");
233     userlog ("Insert exitoso");
234     Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
235     cierraConexion();
236     tpreturn(TPSUCCESS, 0, rqst->data, strlen(rqst->data), 0);
237 }
238

```

Descripción de la tabla select:

```
CREATE TABLE Dulceria( id_clave NUMBER(20),nombre_dulce VARCHAR2(20),tipo_dulce VARCHAR2(20),descripcion VARCHAR2(20),precio NUMBER(20));
```

The interface features a search form with the following elements:

- Buttons at the top: Insertar, Ver Todo, Eliminar, Actualizar.
- A section titled "BUSCAR".
- A "Clave" label followed by a text input field containing "2331".
- A "Buscar" button.
- Below the search section, four labels with corresponding text input fields:
 - "Nombre:" followed by a field containing "panditas".
 - "Tipo:" followed by a field containing "Gomitas".
 - "Descripcion:" followed by a field containing "Gomitas".
 - "Precio:" followed by a field containing "2331".

The screenshot shows the Oracle SQL Developer interface with the following details:

- Window title: Oracle SQL Developer : ConexionBD
- Menu bar: Archivo, Editar, Ver, Navegar, Ejecutar, Origen, Equipo, Herramientas, Ventana, Ayuda.
- Left sidebar: Tree view showing database objects like Tablas (Filtrado), Vistas, Índices, Paquetes, Procedimientos, Funciones, Operadores, Colas, Tablas de Colas, Disparadores, Tipos, and Secuencias.
- Central workspace: A SQL script editor with the query `select * from dulceria;`.
- Bottom pane: "Resultado de la Consulta" showing the query results in a table format.

	ID_CLAVE	NOMBRE_DULCE	TIPO_DULCE	DESCRIPCION	PRECIO
1	4321	Pica_fresa	picante	picante	25
2	2331	panditas	Gomitas	Gomitas	2331
3	3172	CarlosV	Chocolate	Chocolate	3172
4	6233	Dubalin	Dubalin	Dubalin	6233

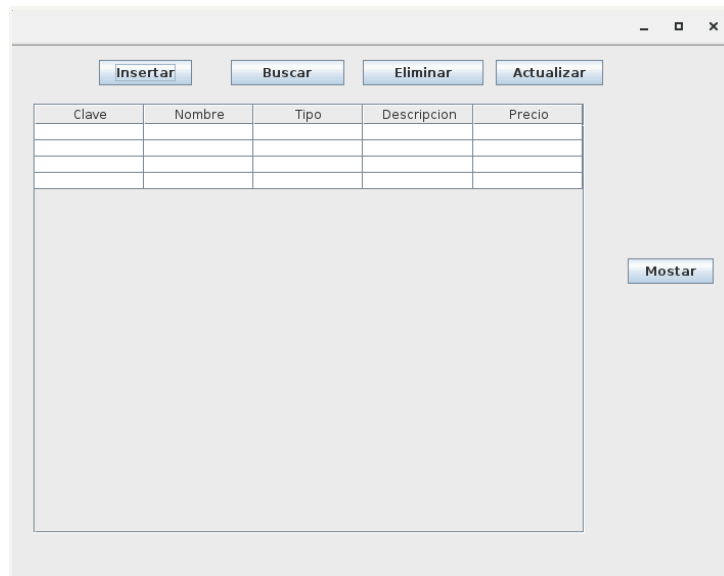
Como podemos observar en la imagen de la interfaz buscamos por clave dulce el producto en la base de datos y esta búsqueda se nos visualiza en los demás campos.

Operación Ver Todo:

En esta operación el usuario deberá visualizar todos los registros del negocio contenidos en una tabla en la BD, por lo que solamente con el clic del botón “Ver todo” mostrará en un textArea o una tabla todos los registros.

Interfaz Ver Todo:

Por lo que la interfaz puede ser representada de la siguiente forma.



Clave	Nombre	Tipo	Descripcion	Precio

Código de la Interfaz select todo:

```

206 private void MostrarActionPerformed(java.awt.event.ActionEvent evt) {
207     String nombreArchivo = "Ud32_Mostrar";
208     String direccionCompleta = "/home/oracle/tuxedo3/ud/" + nombreArchivo + ".ud";
209     FileWriter Ubicacion = null;
210     try {
211         Ubicacion = new FileWriter(direccionCompleta);
212     } catch (IOException ex) {
213         System.out.println("Error Carpeta:" + ex);
214     }
215
216     try {
217         BufferedWriter escritor = new BufferedWriter(Ubicacion);
218         String Cadena = "SRVCNM\t" + "SELECTCOM_FML32\n\n";
219         escritor.write(Cadena);
220         JOptionPane.showMessageDialog(null, "Buscar Todo exitoso");
221         escritor.close();
222     } catch (Exception ex) {
223         System.out.println("Error Contenido:" + ex);
224     }
225     Process proc = null;
226     try {
227         proc = Runtime.getRuntime().exec("/home/oracle/tuxedo3/ud/" + "scrip3.sh");
228     } catch (Exception ex) {
229         System.out.println("Error" + ex);
230     }
231     DefaultTableModel model = new DefaultTableModel();
232     table.setModel(model);
233     try {
234         String url = "jdbc:oracle:thin:@localhost:1521:orcl21";
235         con = (Connection) DriverManager.getConnection(url, "aldoh", "2193035908");
236         s = con.createStatement();
237         rs = s.executeQuery("select * from dulceria");
238         ResultSetMetaData rsMd = rs.getMetaData();
239         int cantidadColumnas = rsMd.getColumnCount();
240         model.addColumn("Id_clave");
241         model.addColumn("Nombre_dulce");
242         model.addColumn("Tipo_dulce");
243         model.addColumn("descripcion");
244         model.addColumn("precio");
245         while (rs.next()) {
246             Object[] filas = new Object[cantidadColumnas];
247             for (int i = 0; i < cantidadColumnas; i++) {
248                 filas[i] = rs.getObject(i + 1);
249             }
250             model.addRow(filas);
251         }
252     } catch (SQLException ex) {

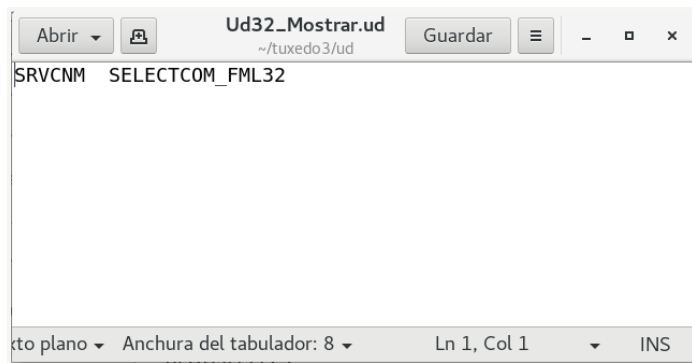
```

Datos de ver todo a través de la interfaz:

<div>Insertar</div> <div>Buscar</div> <div>Eliminar</div> <div>Actualizar</div>				
Id_clave	Nombre_dulce	Tipo_dulce	descripcion	precio
4321	Pica fresa	picante	picante	25
2331	panditas	Gomitas	Gomitas	2331
3172	CarlosV	Chocolate	Chocolate	3172
6233	Dubalin	Dubalin	Dubalin	6233

Mostrar

Archivo ud32:



Código de selec todo:

```
451 void SELECTCOM_FML32(TPSVCINFO *rqst)
452 {
453     FLDLEN32      flen;
454     paramConexion svL_paramCon;
455     char          msgbuf[256];
456
457     EXEC SQL BEGIN DECLARE SECTION;
458     int t_idClave;
459     char t_nombre[51];
460     char t_tipo[151];
461     char t_desc[20];
462     int t_precio[20]; //Nuevo
463     EXEC SQL END DECLARE SECTION;
464
465     FBFR32* fbfr = (FBFR32*)rqst->data ;
466
467     strcpy(svL_paramCon.usuario_DB, "aldoh");
468     strcpy(svL_paramCon.password_DB, "2193035908");
469     strcpy(svL_paramCon.name_DB, "orcl21");
470     t_idClave = 0;
471     memset(t_nombre, 0x00, sizeof(t_nombre));
472     memset(t_tipo, 0x00, sizeof(t_tipo));
473     memset(t_desc, 0x00, sizeof(t_desc));
474     memset(t_precio, 0x00, sizeof(t_precio));
475
476
477     userlog ("Servicio SELECT_FML32 Invocado") ;
478
479     /* Obtenemos los campos FML del buffer de entrada */
480     /*if(Fget32 (fbfr, CLAVE, 0, (char*)&t_idClave, 0) < 0) {
481         userlog ("Error Fget32(CLAVE): %s", Fstrerror32(Ferror32)) ;
482         tpreturn (TPFAIL, 0, rqst->data, 0L,0) ;
483     }
484
485     userlog (" CLAVE: [%d]", t_idClave) ;
486     */
487     Finit32 (fbfr, Fsizeof32 (fbfr));
488
489     userlog ("Conectando a la BD");
490     conectando_aBD(&svL_paramCon);
491
492     /* Declaramos y Abrimos Cursor */
```

```

493 EXEC SQL AT ORACLE2 DECLARE cu_bibAutor CURSOR FOR
494     SELECT id_clave, nombre_dulce, tipo_dulce, descripcion, precio
495     FROM Dulceria;
496     //WHERE id_clave = :t_idClave;
497
498     if(sqlca.sqlcode != 0) {
499         userlog ("Error Declare Cursor: %d-%s",
500             sqlca.sqlcode, sqlca.sqlerrm.sqlerrmc) ;
501         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
502     }
503
504 EXEC SQL AT ORACLE2 OPEN cu_bibAutor;
505 if(sqlca.sqlcode != 0) {
506     userlog ("Error Open Cursor: %d-%s",
507         sqlca.sqlcode, sqlca.sqlerrm.sqlerrmc) ;
508     tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
509 }
510
511 EXEC SQL AT ORACLE2 FETCH cu_bibAutor
512 INTO t_idClave, :t_nombre, :t_tipo, :t_desc, :t_precio;
513
514 while (sqlca.sqlcode == 0) {
515     Fadd32 (fbfr, CLAVE, (char *) &t_idClave, 0) ;
516     userlog ("Codigo: [%d]", t_idClave) ;
517     Fadd32 (fbfr, NOMBRE, ajustaCampos(t_nombre), 0) ;
518     userlog (" Autor : [%s]", ajustaCampos(t_nombre));
519     Fadd32 (fbfr, TIPO, ajustaCampos( t_tipo), 0) ;
520     userlog (" Titulo: [%s]", ajustaCampos(t_tipo)) ;
521     Fadd32 (fbfr, DESC, ajustaCampos(t_desc), 0) ;
522     userlog (" Desc : [%s]", ajustaCampos(t_desc)) ;
523     Fadd32 (fbfr, PRECIO, (char *) &t_precio, 0) ;
524     userlog (" Precio: [%d]", t_precio) ;
525
526     EXEC SQL AT ORACLE2 FETCH cu_bibAutor
527     INTO t_idClave, :t_nombre, :t_tipo, :t_desc, :t_precio;
528 }
529 EXEC SQL CLOSE cu_bibAutor;
530 userlog ("Select exitoso");
531 //Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
532 cierraConexion();
533 tpreturn(TPSUCCESS, 0,(char *)fbfr, sizeof(fbfr), 0);
534 }

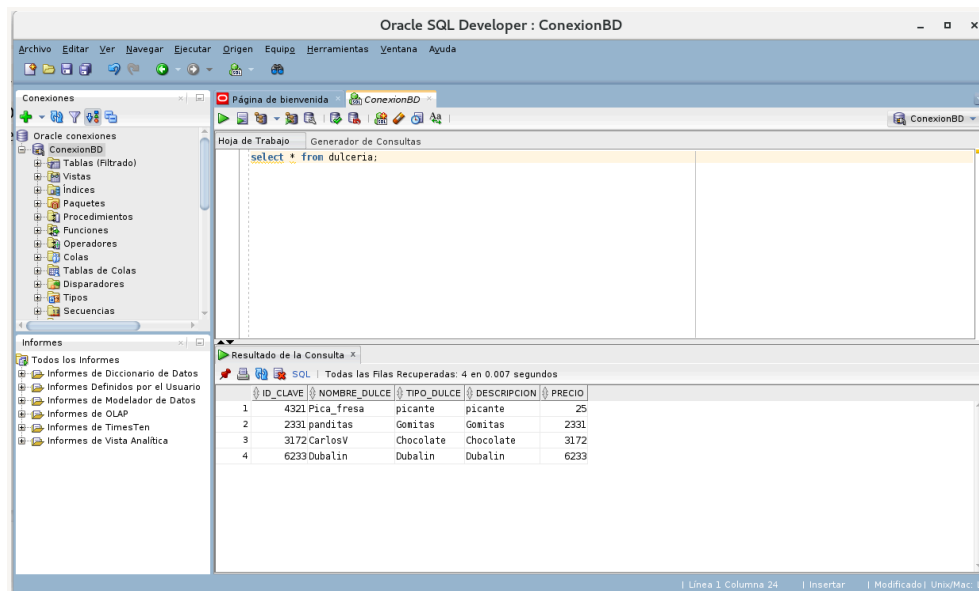
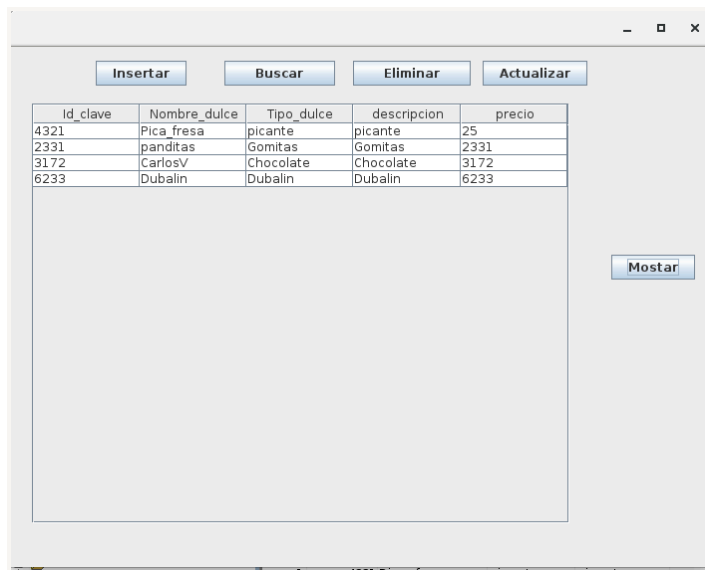
```

Descripción de la tabla ver todo:

```

CREATE TABLE Dulceria( id_clave NUMBER(20),nombre_dulce VARCHAR2(20),tipo_dulce
VARCHAR2(20),descripcion VARCHAR2(20),precio NUMBER(20));

```



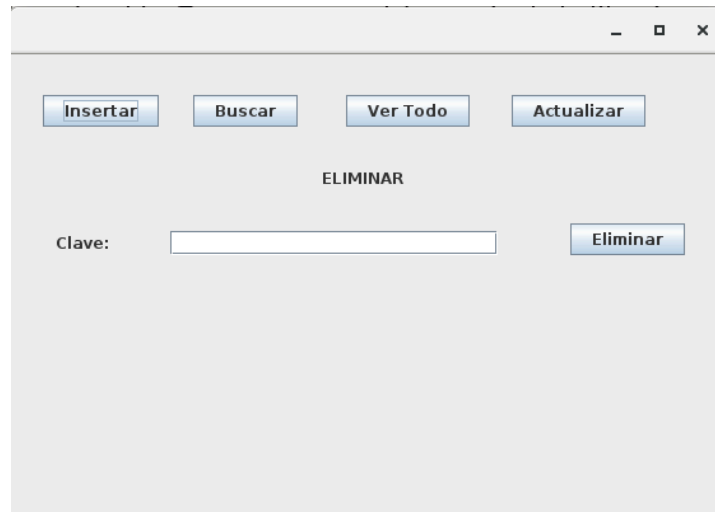
En la imagen de la interfaz tenemos un botón el cual al dar clic este nos permitirá visualizar todo el contenido de la base de datos.

Operación Eliminar:

Para la operación eliminar, el usuario dará de baja un registro en la tabla contenida en la BD por medio de un dato del registro que sea único.

Interfaz Eliminar:

La interfaz gráfica se puede representar como la figura que se muestra a continuación.



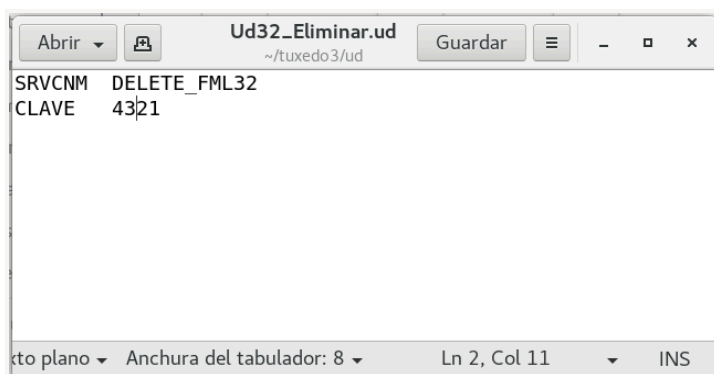
Código de la Interfaz delete:

```
177 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
178     // TODO add your handling code here:
179     String nombreArchivo = "Ud32_Eliminar";
180     String carpeta = System.getProperty("user.dir");
181     String direccionCompleta = "/home/oracle/tuxedo3/ud/"+nombreArchivo+".ud";
182     FileWriter Ubicacion = null;
183     try{
184         Ubicacion = new FileWriter (direccionCompleta);
185     }catch(IOException ex){
186         System.out.println("Error Carpeta:"+ex);
187     }
188     try{
189         BufferedWriter escritor = new BufferedWriter(Ubicacion);
190         String Cadena = "SRVCNM\t"+"DELETE_FML32\n"+
191             "CLAVE\t"+txt_clave.getText()+"\n\n";
192
193         escritor.write(Cadena);
194         JOptionPane.showMessageDialog(null, "Eliminado Correctamente");
195         escritor.close();
196     }catch(Exception ex){
197         System.out.println("Error Contenido:"+ex);
198     }
199
200     Process proc = null;
201     try {
202         proc = Runtime.getRuntime().exec("/home/oracle/tuxedo3/ud/"+"scrip4.sh");
203     } catch (Exception ex) {
204         System.out.println("Error" + ex);
205     }
206
207     BufferedReader buf = new BufferedReader(new InputStreamReader(proc.getInputStream()));
208     String s;
209     try{
210         while ((s = buf.readLine()) != null){
211             System.out.println(s);
212         }
213     } catch (IOException ex){
214         System.out.println(ex.toString());
215     }
216 }
217
218 }
```

Eliminar datos a través de la interfaz:



Archivo ud32:



Código de delete:

```

306 void DELETE_FML32(TPSVCINFO *rqst)
307 {
308     FLDLEN32      flen;
309     paramConexion svL_paramCon;
310     char          msgbuf[256];
311
312     EXEC SQL BEGIN DECLARE SECTION;
313     int t_idClave;
314     char t_nombre[51];
315     char t_tipo[151];
316     char t_desc[20];
317     int t_precio[20]; //Nuevo
318     EXEC SQL END DECLARE SECTION;
319
320     FBFR32* fbfr = (FBFR32*)rqst->data ;
321
322     strcpy(svL_paramCon.usuario_DB, "aldoh");
323     strcpy(svL_paramCon.password_DB, "2193035908");
324     strcpy(svL_paramCon.name_DB, "orcl21");
325
326     memset(&t_idClave, 0x00, sizeof(t_idClave));
327     memset(t_nombre, 0x00, sizeof(t_nombre));
328     memset(t_tipo, 0x00, sizeof(t_tipo));
329     memset(t_desc, 0x00, sizeof(t_desc));
330     memset(&t_precio, 0x00, sizeof(t_precio)); //Nuevo
331
332     userlog ("Servicio INSERT_FML32 Invocado") ;
333
334     if(Fget32 (fbfr, CLAVE, 0, (char*)&t_idClave, 0) < 0) {
335         userlog ("Error Fget32(CLAVE): %s", Fstrerror32(Ferror32)) ;
336         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
337     }
338     userlog (" CLAVE: [%d]", t_idClave) ;
339
340     userlog ("Conectando a la BD");
341     conectando_aBD(&svL_paramCon);
342
343     EXEC SQL AT ORACLE2
344     DELETE FROM Dulceria
345     WHERE id_clave=:t_idClave;
346
347     if
348     (sqlca.sqlcode != 0){

```

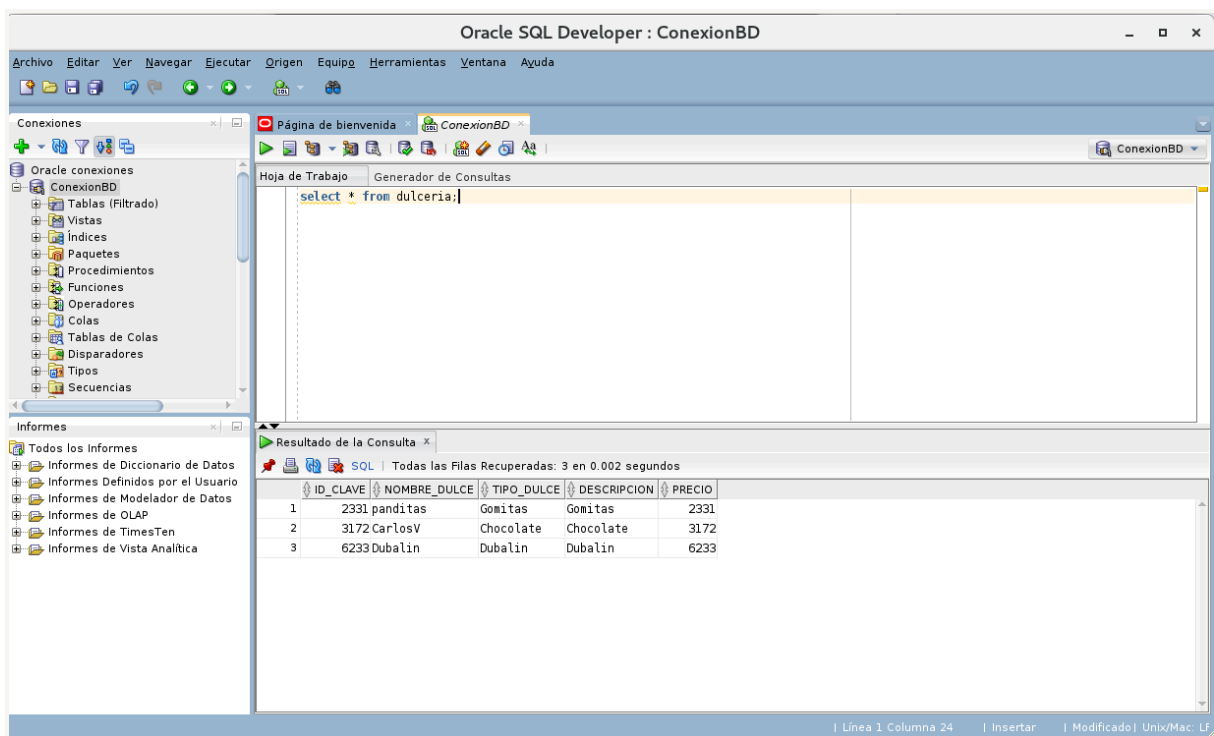
```

349         sprintf(msgbuf, "Fallo insert: sqlcode = %d(%s)\n",
350             sqlca.sqlcode, sqlca.sqlerrm.sqlerrmc);
351         sqlError("Error al realizar Insert");
352         Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
353         cierraConexion();
354         tpreturn(TPFAIL, 0, rqst->data, 0L, 0);
355     }
356
357     strcpy(msgbuf, "Insert exitoso!");
358     userlog ("Insert exitoso");
359     Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
360     cierraConexion();
361     tpreturn(TPSUCCESS, 0, rqst->data, strlen(rqst->data), 0);
362 }
363

```

Descripción del producto Eliminado de la tabla:

```
CREATE TABLE Dulceria( id_clave NUMBER(20),nombre_dulce VARCHAR2(20),tipo_dulce  
VARCHAR2(20),descripcion VARCHAR2(20),precio NUMBER(20));
```



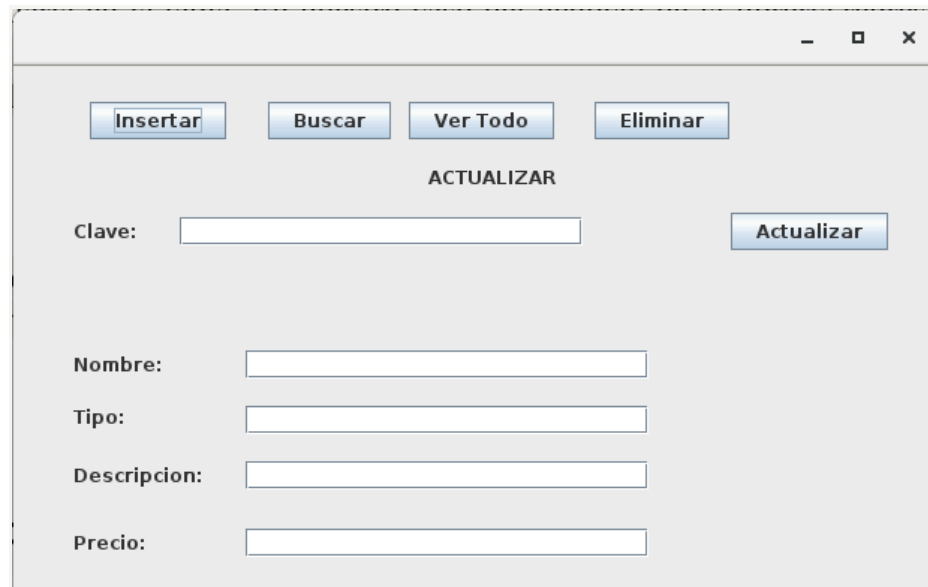
En la base de datos dulcería podemos observar que un producto se ha eliminado, ya que a través de la interfaz podemos visualizar que introducimos el id clave del dulce que queremos eliminar.

Operación Actualizar:

Para esta operación, el usuario actualiza los datos de un registro en la tabla por medio del valor único que tiene registrado. El dato será enviado por medio de la interfaz gráfica proporcionada por el usuario.

Interfaz Actualizar:

La interfaz es como la que se muestra en la siguiente figura.



Código de la Interfaz update:

```
213 private void ActualizarActionPerformed(java.awt.event.ActionEvent evt) {
214     // TODO add your handling code here:
215     String nombreArchivo = "Ud32_Actualizar";
216     String carpeta = System.getProperty("user.dir");
217     String direccionCompleta = "/home/oracle/tuxedo3/ud/" + nombreArchivo + ".ud";
218     FileWriter Ubicacion = null;
219     try {
220         Ubicacion = new FileWriter(direccionCompleta);
221     } catch (IOException ex) {
222         System.out.println("Error Carpeta: " + ex);
223     }
224     try {
225         BufferedWriter escritor = new BufferedWriter(Ubicacion);
226         String Cadena = "SRVCNM\t" + "UPDATE_FML32\n" +
227             "CLAVE\t" + txt_clave.getText() + "\n" +
228             "NOMBRE\t" + txt_nombre.getText() + "\n" +
229             "TIPO\t" + txt_tipo.getText() + "\n" +
230             "DESC\t" + txt_descripcion.getText() + "\n" +
231             "PRECIO\t" + txt_precio.getText() + "\n\n";
232         escritor.write(Cadena);
233         JOptionPane.showMessageDialog(null, "Actualizado Correctamente");
234         escritor.close();
235     } catch (Exception ex) {
236         System.out.println("Error Contenido: " + ex);
237     }
238     Process proc = null;
239     try {
240         proc = Runtime.getRuntime().exec("/home/oracle/tuxedo3/ud/" + "scrip5.sh");
241     } catch (Exception ex) {
242         System.out.println("Error" + ex);
243     }
244     BufferedReader buf = new BufferedReader(new InputStreamReader(proc.getInputStream()));
245     String s;
246     try {
247         while ((s = buf.readLine()) != null) {
248             System.out.println(s);
249         }
250     } catch (IOException ex) {
251         System.out.println(ex.toString());
252     }
253 }
254
255
256 }
```

Actualizar datos a través de la interfaz:

Insertar Buscar Ver Todo Eliminar

ACTUALIZAR

Clave:

Nombre:

Tipo:

Descripción:

Precio:

Message

Actualizado Correctamente

Archivo ud32:

Abrir Guardar

~/tuxedo3/ud

```
SRVCNM UPDATE_FML32
CLAVE  6233
NOMBRE Rockaleta
TIPO   Paleta
DESC   Paleta
PRECIO 15
```

cto plano ▾ Anchura del tabulador: 8 ▾ Ln 1, Col 1 ▾ INS

Código de update:

```

364 //UPDETE
365 void UPDATE_FML32(TPSVCINFO *rqst)
366 {
367     FLDLEN32      flen;
368     paramConexion svL_paramCon;
369     char          msgbuf[256];
370
371     EXEC SQL BEGIN DECLARE SECTION;
372     int t_idClave;
373     char t_nombre[51];
374     char t_tipo[151];
375     char t_desc[20];
376     int t_precio[20]; //Nuevo
377     EXEC SQL END DECLARE SECTION;
378
379     FBFR32* fbfr = (FBFR32*)rqst->data ;
380
381     strcpy(svL_paramCon.usuario_DB, "aldoh");
382     strcpy(svL_paramCon.password_DB, "2193035908");
383     strcpy(svL_paramCon.name_DB, "orcl21");
384
385     memset(&t_idClave, 0x00, sizeof(t_idClave));
386     memset(t_nombre, 0x00, sizeof(t_nombre));
387     memset(t_tipo, 0x00, sizeof(t_tipo));
388     memset(t_desc, 0x00, sizeof(t_desc));
389     memset(&t_precio, 0x00, sizeof(t_precio)); //Nuevo
390
391     userlog ("Servicio INSERT_FML32 Invocado") ;
392
393     if(Fget32 (fbfr, CLAVE, 0, (char*)&t_idClave, 0) < 0) {
394         userlog ("Error Fget32(CLAVE): %s", Fstrerror32(Ferror32)) ;
395         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
396     }
397     userlog (" CLAVE: [%d]", t_idClave) ;
398
399     if(Fget32 (fbfr, NOMBRE, 0, t_nombre, 0) < 0) {
400         userlog ("Error Fget32(NOMBRE): %s", Fstrerror32(Ferror32)) ;
401         tpreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
402     }
403     userlog (" NOMBRE : [%s]", t_nombre) ;
404
405     if(Fget32 (fbfr, TIPO, 0, t_tipo, 0) < 0) {

```

```

406     userlog ("Error Fget32(TIPO): %s", |
407         Fstrerror32(Ferror32)) ;
408     tpretreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
409 }
410 userlog (" TIPO: [%s]", t_tipo) ;
411
412 if(Fget32 (fbfr, DESC, 0, t_desc, 0) < 0) {
413     userlog ("Error Fget32(DESC): %s", Fstrerror32(Ferror32)) ;
414     tpretreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
415 }
416 userlog (" DESC : [%s]", t_desc) ;
417
418 //Nuevo
419 if(Fget32 (fbfr, PRECIO, 0, (char*)&t_precio, 0) < 0) {
420     userlog ("Error Fget32(PRECIO): %s", Fstrerror32(Ferror32)) ;
421     tpretreturn (TPFAIL, 0, rqst->data, 0L, 0) ;
422 }
423 userlog (" PRECIO: [%d]", t_precio) ;
424
425
426 userlog ("Conectando a la BD");
427 conectando_aBD(&svL_paramCon);
428
429 EXEC SQL AT ORACLE2
430 UPDATE Dulceria SET nombre_dulce=:t_nombre, tipo_dulce=:t_tipo, descripcion=:t_desc, precio=:t_precio
431 WHERE id_clave=:t_idClave;
432
433
434 if
435 (sqlca.sqlcode != 0){
436     sprintf(msgbuf, "Fallo update: sqlcode = %d(%s)\n",
437         sqlca.sqlcode, sqlca.sqlerrm.sqlerrmc);
438     sqlError("Error al realizar update");
439     Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
440     cierraConexion();
441     tpretreturn(TPFAIL, 0, rqst->data, 0L, 0);
442 }
443
444 strcpy(msgbuf, "Actualizacion exitoso!");
445 userlog ("Insert exitoso");
446 Fchg32(fbfr, OUTPUT, 0, msgbuf, 0);
447 cierraConexion();
448 tpretreturn(TPSUCCESS, 0, rqst->data, strlen(rqst->data), 0);

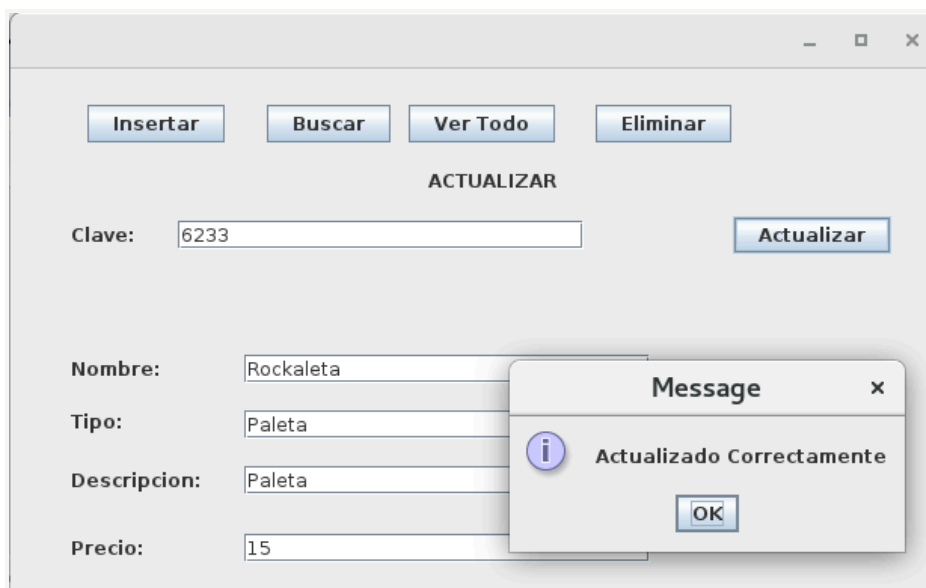
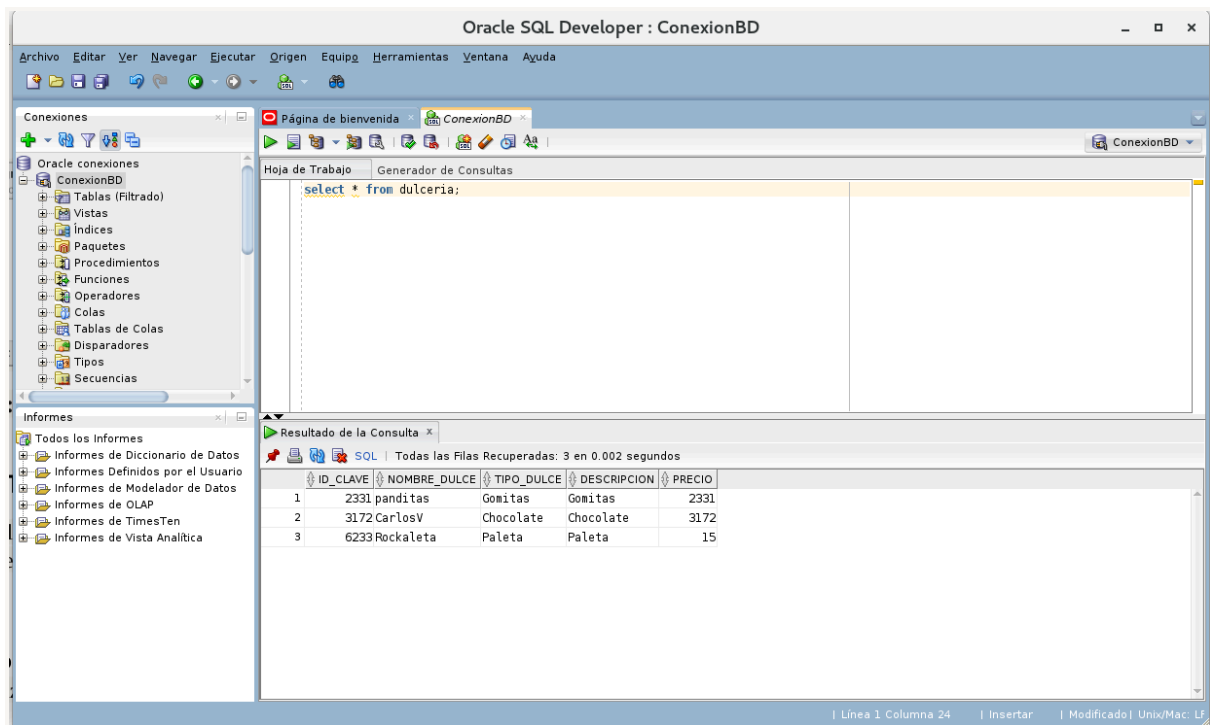
```

Descripción de la actualización de un dato de la tabla:

```


CREATE TABLE Dulceria( id_clave NUMBER(20),nombre_dulce VARCHAR2(20),tipo_dulce
VARCHAR2(20),descripcion VARCHAR2(20),precio NUMBER(20));

```

Como podemos notar en la interfaz de actualizar introducir la clave del producto de nuestra base de datos y en este caso agregamos los nuevos datos del producto a actualizar dicho producto.

Archivo fml

Abrir ▾				biblioBD01.fml ~/tuxedo3/fml		Guardar		☰		- □ ×	
#name	number	type	flag	comment							
OUTPUT	5	string	-	-							
CLAVE	10	long	-	- codigo del libro							
NOMBRE	20	string	-	- nombre del dulce							
TIPO	30	string	-	- tipo de dulce							
DESC	30	string	-	- descripcion del dulce							
PRECIO	30	long	-	- precio del dulce							
to plano ▾ Anchura del tabulador: 8 ▾ Ln 1, Col 1 ▾ INS											

Conclusiones.

Como podemos observar en la documentación de este proyecto, podemos notar que podemos conectar a una base de datos como lo es oracle a través de interfaces, la interfaz podemos ver que introducimos datos que esto al ejecutar un boton tiene código que realiza la creación de archivos ud32 y que estos lo ejecutamos al mismo tiempo y que esto hace que la conexión con tuxedo y oracle sea más fácil.

Todo esto es más sencillo ya que podemos visualizar de mejor manera, no se nos complica a la hora de visualizar desde ejecutable sino desde una interfaz con la que un usuario interactúa a diario y se le hace más sencillo.

Bibliografía.

<https://www.oracle.com/middleware/technologies/tuxedo.html>

<https://www.oracle.com/database/sqldeveloper/>

<https://www.tutorialesprogramacionya.com/javaya/detalleconcepto.php?punto=92&codigo=172&inicio=80>

<https://decodigo.com/java-crear-archivos-de-texto>