



UNIVERSIDAD DE CÓRDOBA
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

SOLUCIÓN PARA LA GESTIÓN DE EMPRESAS CÁRNICAS
BASADA EN GNU/LINUX Y SOFTWARE LIBRE

MANUAL CÓDIGO

Autor: Rafael Ernesto Escobar Zurita

Director: Domingo Ortiz Boyer

Córdoba, Febrero de 2018

A mi familia, a la que he quitado tanto tiempo.

Sumario

Capítulo 1. INTRODUCCIÓN.....	5
Capítulo 2. CÓDIGO FUENTE.....	7
2.1 Base de Datos.....	7
2.2 Código de la aplicación.....	20
2.2.1 Menú Principal. Fmain.cass.....	20
2.2.2 Modulo Gestión.....	25
2.2.2.1 Frm_Articulos.class.....	25
2.2.2.2 Frm_Clientes.class.....	43
2.2.2.3 Frm_Proveedores.class.....	53
2.2.2.4 Frm_Vendedores.class.....	66
2.2.3 Modulo Facturación.....	72
2.2.3.1 Frm_Facturacion.class.....	72
2.2.3.2 Frm_Edicion_Facturas_Albaranes.class.....	85
2.2.3.3 Frm_Visualizar_Factura_Albaran.class.....	96
2.2.3.4 Frm_Mostrar_Factura.class.....	111
2.2.3.5 Frm_FacturacionAlbaranes.class.....	112
2.2.3.6 Frm_Pasocontabilidad.class.....	125
2.2.3.7 Frm_Averias.class.....	128
2.2.4 Modulo Almacén.....	136
2.2.4.1 Frm_Compras.class.....	136
2.2.4.2 Frm_Stock.class.....	154
2.2.5 Modulo Lotes.....	163
2.2.5.1 Frm_Lotes.class.....	163
2.2.5.2 Frm_ConsultaLotes.class.....	184
2.2.5.3 Frm_Visualiza_Lote.class.....	189
2.2.5.4 Frm_Trazabilidad.class.....	191
2.2.5.5 Frm_Visualiza_Lote_Trazabilidad.class.....	193
2.2.6 Modulo Utilidades.....	196
2.2.6.1 Frm_Conexionbasedatos.class.....	196
2.2.6.2 Frm_CopiasSeguridad.class.....	198
2.2.6.3 Frm_Informes.class.....	200
2.2.6.4 Frm_Listado.class.....	207
2.2.6.5 Frm_Utilidades.class.....	220
2.2.7 Utilidades.....	233

Capítulo 1. INTRODUCCIÓN

En este documento se recogerá el código de la aplicación generada así con el fichero con la creación de la base de datos en formato SQL.

La aplicación ha sido crea con el IDE Gambas, cuyo lenguaje de programación posee el mismo nombre, aunque es muy parecido a la programación en Visual Basic .Net, el nombre de Gambas es un acrónimo de Gambas Almost Means Basic.

Capitulo 2. CÓDIGO FUENTE

2.1 Base de Datos.

En la carpeta *carneaerp/documentacion/basededatos* se encuentra el fichero *carnea.sql* con el siguiente código que genera la base de datos.

```
-- phpMyAdmin SQL Dump
-- version 4.6.6deb5
-- https://www.phpmyadmin.net/
--
-- Servidor: localhost:3306
-- Tiempo de generación: 02-12-2017 a las 18:18:23
-- Versión del servidor: 5.7.20-0ubuntu0.17.10.1
-- Versión de PHP: 7.1.11-0ubuntu0.17.10.1

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Base de datos: `carnea`
--
CREATE DATABASE IF NOT EXISTS `carnea` DEFAULT CHARACTER SET utf8 COLLATE
utf8_general_ci;
USE `carnea`;

--
-- Estructura de tabla para la tabla `albaranes`
--
DROP TABLE IF EXISTS `albaranes`;
CREATE TABLE `albaranes` (
  `numeroalbaran` varchar(10) NOT NULL COMMENT 'Numero Albaran',
  `fecha` varchar(10) NOT NULL COMMENT 'Fecha creacion albaran',
  `hora` varchar(10) NOT NULL COMMENT 'Hora creacion albaran',
  `base` float NOT NULL COMMENT 'Base imponible',
  `iva` int(11) NOT NULL COMMENT 'Iva que lleva la albaran',
  `recargo` int(11) NOT NULL COMMENT 'Recargo del albaran',
  `importeiva` float NOT NULL COMMENT 'Importe en euros de aplicar el iva a la
base',
  `importerecargo` float NOT NULL COMMENT 'Importe en euros de aplicar el
recargo a la base',
  `fechacobro` varchar(10) NOT NULL COMMENT 'Fecha de cobro del albaran',
  `descuentopp` float DEFAULT NULL COMMENT 'Descuento pronto pago',
  `importedescuentopp` float DEFAULT NULL COMMENT 'Valor del descuento pronto
pago',
```

Capítulo 2. CÓDIGO FUENTE

```
`descuentoesp` float DEFAULT NULL COMMENT 'Descuento especial',
`importedescuentoesp` float DEFAULT NULL COMMENT 'Valor del descuento
especial',
`pagado` varchar(1) NOT NULL COMMENT 'Si ha sido cobrado o no',
`formapago` varchar(20) DEFAULT NULL COMMENT 'Forma del pago del cliente',
`vencimiento` varchar(10) DEFAULT NULL,
`facturado` varchar(1) NOT NULL COMMENT 'Si se ha facturado',
`total` float NOT NULL COMMENT 'Total del Albaran',
`cliente` varchar(10) NOT NULL COMMENT 'Cliente de la factura',
`vendedor` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
-- Estructura de tabla para la tabla `albaranesdetalle`
--
```

```
DROP TABLE IF EXISTS `albaranesdetalle`;
CREATE TABLE `albaranesdetalle` (
  `numeroalbaran` varchar(10) NOT NULL COMMENT 'Numero de factura a la que
pertenece la linea detalle',
  `linea` int(11) NOT NULL COMMENT 'Linea de la factura',
  `precio` float NOT NULL COMMENT 'Precio kilo del articulo',
  `peso` float NOT NULL COMMENT 'Peso del articulo',
  `importe` float NOT NULL COMMENT 'Importe total del articulo',
  `lote` varchar(10) NOT NULL COMMENT 'Lote del articulo',
  `articulo` varchar(10) NOT NULL COMMENT 'Articulo'
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
-- Estructura de tabla para la tabla `articulo`
--
```

```
DROP TABLE IF EXISTS `articulo`;
CREATE TABLE `articulo` (
  `codigoarticulo` varchar(10) NOT NULL,
  `nombre` varchar(30) NOT NULL,
  `familia` varchar(25) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos los articulos';
```

```
--
-- Estructura de tabla para la tabla `articulomateriaprima`
--
```

```
DROP TABLE IF EXISTS `articulomateriaprima`;
CREATE TABLE `articulomateriaprima` (
  `articulo` varchar(10) NOT NULL,
  `materiaprima` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Composicion de los articulos';
```

```
--
-- Estructura de tabla para la tabla `averia`
--
```


Capítulo 2. CÓDIGO FUENTE

```
DROP TABLE IF EXISTS `averia`;
CREATE TABLE `averia` (
  `codigoaveria` varchar(10) NOT NULL,
  `fecha` varchar(10) NOT NULL,
  `vendedor` varchar(10) NOT NULL,
  `cliente` varchar(10) NOT NULL,
  `articulo` varchar(10) NOT NULL,
  `lote` varchar(10) NOT NULL,
  `peso` float NOT NULL,
  `descripcion` varchar(200) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Averias';

--
-- Estructura de tabla para la tabla `clientefax`
--

DROP TABLE IF EXISTS `clientefax`;
CREATE TABLE `clientefax` (
  `fax` varchar(9) NOT NULL,
  `cliente` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-----

--
-- Estructura de tabla para la tabla `clientes`
--

DROP TABLE IF EXISTS `clientes`;
CREATE TABLE `clientes` (
  `codigocliente` varchar(10) NOT NULL,
  `razonsocial` varchar(50) NOT NULL,
  `direccion` varchar(50) NOT NULL,
  `localidad` varchar(20) NOT NULL,
  `provincia` varchar(20) NOT NULL,
  `codigopostal` varchar(6) NOT NULL,
  `nif` varchar(10) NOT NULL,
  `subcuentaiva` int(7) NOT NULL,
  `descuentopp` float NOT NULL,
  `descuentoesp` float NOT NULL,
  `recargo` varchar(1) DEFAULT NULL,
  `iva` varchar(1) DEFAULT NULL,
  `email` varchar(20) DEFAULT NULL,
  `enviofacturaemail` varchar(1) DEFAULT NULL,
  `formapago` varchar(20) NOT NULL,
  `tarifa` varchar(25) NOT NULL,
  `vendedor` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos los clientes';

--
-- Estructura de tabla para la tabla `clientetelefono`
--

DROP TABLE IF EXISTS `clientetelefono`;
CREATE TABLE `clientetelefono` (
  `telefono` varchar(9) NOT NULL,
  `cliente` varchar(10) NOT NULL
```

Capítulo 2. CÓDIGO FUENTE

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos los telefonos del
cliente';

--
-- Estructura de tabla para la tabla `compras`
--

DROP TABLE IF EXISTS `compras`;
CREATE TABLE `compras` (
  `codigocompra` int(11) NOT NULL COMMENT 'Codigo de la compra auto
incremental',
  `fecharecepcion` varchar(10) NOT NULL COMMENT 'Fecha de la recepcion de la
compra',
  `fechafacturacompra` varchar(10) NOT NULL,
  `cantidad` float NOT NULL COMMENT 'Cantidad en kilos',
  `lotecompra` varchar(20) NOT NULL COMMENT 'Lote de la compra',
  `nfacturacompra` varchar(20) NOT NULL COMMENT 'Numero de factura de la
compra',
  `higiene` varchar(12) DEFAULT NULL COMMENT 'Higiene solo acepta Aceptable y No
aceptable',
  `carolep` varchar(12) NOT NULL COMMENT 'Caro leptidos solo acepta Aceptable y
No aceptable',
  `precio` float NOT NULL COMMENT 'Coste de la compra',
  `firma` varchar(20) DEFAULT NULL COMMENT 'Datos de la compra',
  `observaciones` varchar(200) DEFAULT NULL COMMENT 'Observaciones de la
compra',
  `transporte` int(11) DEFAULT NULL COMMENT 'Valor',
  `stock` varchar(1) NOT NULL COMMENT 'Si se ha pasado a stock',
  `proveedor` varchar(10) NOT NULL,
  `materiaprima` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Compras a proveedores';

--
-- Estructura de tabla para la tabla `cuentascontabilidad`
--

DROP TABLE IF EXISTS `cuentascontabilidad`;
CREATE TABLE `cuentascontabilidad` (
  `id` varchar(1) NOT NULL,
  `venta` varchar(7) NOT NULL COMMENT 'Cuenta contabilidad de la venta',
  `iva` varchar(7) NOT NULL COMMENT 'Cuenta contablidad del iva',
  `ivare` varchar(7) NOT NULL COMMENT 'Cuenta contablidad iva repercutido',
  `re` varchar(7) NOT NULL COMMENT 'Cuenta contablidad del recargo',
  `cobro` varchar(7) NOT NULL COMMENT 'Cuenta contablidad del cobro'
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Estructura de tabla para la tabla `dispositivomovil`
--

DROP TABLE IF EXISTS `dispositivomovil`;
CREATE TABLE `dispositivomovil` (
  `codigodispositivomovil` varchar(10) NOT NULL,
  `nombre` varchar(50) NOT NULL,
  `especificaciones` varchar(500) NOT NULL
```

Capítulo 2. CÓDIGO FUENTE

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos los dispositivos
moviles adquiridos';
```

```
--
```

```
-- Estructura de tabla para la tabla `dispositivomovilvendedor`
```

```
--
```

```
DROP TABLE IF EXISTS `dispositivomovilvendedor`;
CREATE TABLE `dispositivomovilvendedor` (
  `dispositivomovil` varchar(10) NOT NULL COMMENT 'Codigo dispositivo',
  `vendedor` varchar(10) NOT NULL COMMENT 'Codigo vendedor'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Dispositivos y vendedores';
```

```
--
```

```
-- Estructura de tabla para la tabla `facturas`
```

```
--
```

```
DROP TABLE IF EXISTS `facturas`;
CREATE TABLE `facturas` (
  `numerofactura` varchar(10) NOT NULL COMMENT 'Numero factura',
  `fecha` varchar(10) NOT NULL COMMENT 'Fecha creacion factura',
  `hora` varchar(10) NOT NULL COMMENT 'Hora creacion factura',
  `base` float NOT NULL COMMENT 'Base imponible',
  `iva` int(11) NOT NULL COMMENT 'Iva que lleva la factura',
  `recargo` int(11) NOT NULL COMMENT 'Recargo de la factura',
  `importeiva` float NOT NULL COMMENT 'Importe en euros de aplicar el iva a la
base',
  `importerrecargo` float NOT NULL COMMENT 'Importe en euros de aplicar el
recargo a la base',
  `fechacobro` varchar(10) NOT NULL COMMENT 'Fecha de cobro de la factura',
  `descuentopp` float DEFAULT NULL COMMENT 'Descuento pronto pago',
  `importedescuentopp` float DEFAULT NULL COMMENT 'Valor del descuento pronto
pago',
  `descuentoesp` float DEFAULT NULL COMMENT 'Descuento especial',
  `importedescuentoesp` float DEFAULT NULL COMMENT 'Valor del descuento
especial',
  `cobrada` varchar(1) NOT NULL COMMENT 'Si ha sigo cobrada o no',
  `formapago` varchar(20) DEFAULT NULL COMMENT 'Forma del pago del cliente',
  `vencimiento` varchar(10) DEFAULT NULL,
  `contabilizada` varchar(1) NOT NULL COMMENT 'Si se ha pasado a contabilidad',
  `total` float NOT NULL COMMENT 'Total de la factura',
  `cliente` varchar(10) NOT NULL COMMENT 'Cliente de la factura',
  `vendedor` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Facturas';
```

```
--
```

```
-- Estructura de tabla para la tabla `facturasdetalles`
```

```
--
```

```
DROP TABLE IF EXISTS `facturasdetalles`;
CREATE TABLE `facturasdetalles` (
  `numerofactura` varchar(10) NOT NULL COMMENT 'Numero de factura a la que
pertenece la linea detalle',
  `linea` int(11) NOT NULL COMMENT 'Linea de la factura',
  `precio` float NOT NULL COMMENT 'Precio kilo del articulo',
  `peso` float NOT NULL COMMENT 'Peso del articulo',
```

Capítulo 2. CÓDIGO FUENTE

```
`importe` float NOT NULL COMMENT 'Importe total del articulo',
`lote` varchar(10) NOT NULL COMMENT 'Lote del articulo',
`articulo` varchar(10) NOT NULL COMMENT 'Articulo'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Lineas detalle de las facturas';

--
-- Estructura de tabla para la tabla `familiaarticulo`
--

DROP TABLE IF EXISTS `familiaarticulo`;
CREATE TABLE `familiaarticulo` (
  `nombrefamilia` varchar(25) NOT NULL COMMENT 'Almacena nombre de la familia',
  `recargo` float NOT NULL COMMENT 'Almacena el recargo que lleva la familia por
desechos durante el proceso de fabricacion'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacena las familias';

--
-- Estructura de tabla para la tabla `formapago`
--

DROP TABLE IF EXISTS `formapago`;
CREATE TABLE `formapago` (
  `nombreformapago` varchar(20) NOT NULL COMMENT 'Almacena el nombre de la forma
de pago',
  `dias` int(11) NOT NULL,
  `porcentaje` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos las formas de pago con
sus dias y el porcentaje';

--
-- Estructura de tabla para la tabla `lotedetalle`
--

DROP TABLE IF EXISTS `lotedetalle`;
CREATE TABLE `lotedetalle` (
  `codigolote` varchar(10) NOT NULL COMMENT 'Codigo del lote al que pertenece la
linea',
  `linea` int(11) NOT NULL COMMENT 'Numero de linea detalle',
  `cantidad` float NOT NULL COMMENT 'Cantidad en kilos de mateira prima',
  `coste` float NOT NULL COMMENT 'Coste medido en euros/kilo de la materia
prima',
  `numerolotecompra` varchar(12) NOT NULL COMMENT 'Lote de compra de la mateira
prima',
  `proveedor` varchar(10) NOT NULL COMMENT 'Proveedor de la materia prima',
  `nfacturacompra` varchar(10) NOT NULL COMMENT 'Numero de la facrua de compra
de la materia prima',
  `fechafacturacompra` varchar(10) NOT NULL COMMENT 'Fecha factura de compra',
  `materiaprima` varchar(10) NOT NULL COMMENT 'Materia prima de la linea
detalle'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Lineas detalle de los lotes';

--
-- Estructura de tabla para la tabla `lotes`
--
```

Capítulo 2. CÓDIGO FUENTE

```
DROP TABLE IF EXISTS `lotes`;
CREATE TABLE `lotes` (
  `codigolote` varchar(10) NOT NULL COMMENT 'Codigo del lote',
  `fecha` varchar(10) NOT NULL COMMENT 'Fecha creacion de lote',
  `cantidad` float NOT NULL COMMENT 'Kilos del lote',
  `coste` float NOT NULL COMMENT 'Coste en euros del lote',
  `cantidadbruta` float NOT NULL COMMENT 'Cantidad bruta en kilos del lote',
  `merma` float NOT NULL COMMENT 'Kilos perdidos en el proceso de creacion del
lote',
  `articulo` varchar(10) NOT NULL COMMENT 'Articulo creado en el lote'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Cabecera de lotes';
--
-- Estructura de tabla para la tabla `materiaprima`
--

DROP TABLE IF EXISTS `materiaprima`;
CREATE TABLE `materiaprima` (
  `codigomateriaprima` varchar(10) NOT NULL COMMENT 'Codigo de la materia
prima',
  `nombre` varchar(30) NOT NULL COMMENT 'Nombre de la materiaprima'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos las materias primas
con las que estan fabridados';

--
-- Estructura de tabla para la tabla `proveedor`
--

DROP TABLE IF EXISTS `proveedor`;
CREATE TABLE `proveedor` (
  `codigoproveedor` varchar(10) NOT NULL COMMENT 'Codigo del proveedor',
  `razonsocial` varchar(50) NOT NULL COMMENT 'Razon Social proveedor',
  `nif` varchar(10) NOT NULL COMMENT 'Nif del proveedor',
  `direccion` varchar(50) NOT NULL COMMENT 'Direccion del proveedor',
  `localidad` varchar(20) NOT NULL COMMENT 'Localidad del proveedor',
  `provincia` varchar(20) NOT NULL COMMENT 'Provincia del vendedor',
  `codigopostal` varchar(5) NOT NULL COMMENT 'Codigo postal de la localidad',
  `web` varchar(100) DEFAULT NULL COMMENT 'Pagina web del proveedor',
  `email` varchar(20) DEFAULT NULL COMMENT 'Email del proveedor'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Proveedores.';

--
-- Estructura de tabla para la tabla `proveedorcomercial`
--

DROP TABLE IF EXISTS `proveedorcomercial`;
CREATE TABLE `proveedorcomercial` (
  `codigoproveedor` varchar(10) NOT NULL,
  `nombre` varchar(50) NOT NULL,
  `telefono` varchar(9) NOT NULL,
  `email` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Estructura de tabla para la tabla `proveedorfax`
--
```

Capítulo 2. CÓDIGO FUENTE

```
DROP TABLE IF EXISTS `proveedorfax`;
CREATE TABLE `proveedorfax` (
  `fax` varchar(9) NOT NULL COMMENT 'Fax',
  `proveedor` varchar(10) NOT NULL COMMENT 'Codigo proveedor'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Fax de los proveedores';

--
-- Estructura de tabla para la tabla `proveedormateriaprima`
--

DROP TABLE IF EXISTS `proveedormateriaprima`;
CREATE TABLE `proveedormateriaprima` (
  `proveedor` varchar(10) NOT NULL COMMENT 'Proveedor',
  `materiaprima` varchar(10) NOT NULL COMMENT 'Materia prima que vende el
proveedor'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Materias primas venden
proveedores';

--
-- Estructura de tabla para la tabla `proveedortelefono`
--

DROP TABLE IF EXISTS `proveedortelefono`;
CREATE TABLE `proveedortelefono` (
  `telefono` varchar(9) NOT NULL COMMENT 'Telefono',
  `proveedor` varchar(10) NOT NULL COMMENT 'Codigo proveedor'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Telefonos del proveedor';

--
-- Estructura de tabla para la tabla `stock`
--

DROP TABLE IF EXISTS `stock`;
CREATE TABLE `stock` (
  `codigostock` int(11) NOT NULL,
  `materiaprima` varchar(10) NOT NULL,
  `cantidad` float NOT NULL,
  `lotecompra` varchar(20) NOT NULL,
  `proveedor` varchar(10) NOT NULL,
  `nfacturacompra` varchar(20) NOT NULL,
  `ffacturacompra` varchar(10) NOT NULL,
  `precio` float NOT NULL,
  `usado` varchar(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Estructura de tabla para la tabla `tarifa`
--

DROP TABLE IF EXISTS `tarifa`;
CREATE TABLE `tarifa` (
  `nombretarifa` varchar(25) NOT NULL COMMENT 'Nombre de la familia'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos las familias a los
cuales pertenece los articulo';
```

```
--
-- Estructura de tabla para la tabla `tarifaarticulo`
--

DROP TABLE IF EXISTS `tarifaarticulo`;
CREATE TABLE `tarifaarticulo` (
  `articulo` varchar(10) NOT NULL,
  `tarifa` varchar(25) NOT NULL,
  `precio` float NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Estructura de tabla para la tabla `tclote`
--

DROP TABLE IF EXISTS `tclote`;
CREATE TABLE `tclote` (
  `codigolote` varchar(10) NOT NULL COMMENT 'Codigo lote',
  `temperatural` int(11) NOT NULL,
  `tiempol` int(11) NOT NULL,
  `temperatura2` int(11) NOT NULL,
  `tiempo2` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Temperaturas y tiempos de algunos procesos';

--
-- Estructura de tabla para la tabla `tipoiva`
--

DROP TABLE IF EXISTS `tipoiva`;
CREATE TABLE `tipoiva` (
  `nombretipoiva` varchar(25) NOT NULL COMMENT 'Almacena nombre tipo de iva',
  `iva` int(11) NOT NULL COMMENT 'Almacena el iva que se aplicara a factura y albaranes',
  `recargo` int(11) NOT NULL COMMENT 'Almacena el recargo que se aplicara a factura y albaranes'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacena los tipos de iva que existiran';

--
-- Estructura de tabla para la tabla `vendedor`
--

DROP TABLE IF EXISTS `vendedor`;
CREATE TABLE `vendedor` (
  `codigovendedor` varchar(10) NOT NULL COMMENT 'Almacena el codigo del vendedor',
  `nombre` varchar(30) NOT NULL COMMENT 'Nombre del Vendedor',
  `nif` varchar(10) NOT NULL COMMENT 'Nif del vendedor',
  `direccion` varchar(50) NOT NULL COMMENT 'Direccion del vendedor',
  `localidad` varchar(20) NOT NULL COMMENT 'localidad del vendedor',
  `provincia` varchar(20) NOT NULL COMMENT 'Provincia del vendedor',
  `codigopostal` varchar(6) NOT NULL COMMENT 'Codigo postal de la localidad'
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos los vendedores de la empresa.';
```

```
--
-- Estructura de tabla para la tabla `vendedortelefono`
--

DROP TABLE IF EXISTS `vendedortelefono`;
CREATE TABLE `vendedortelefono` (
  `telefono` varchar(9) NOT NULL,
  `vendedor` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Almacenamos los telefonos de los
vendedores';

--
-- Índices para tablas volcadas
--

--
-- Indices de la tabla `albaranes`
--
ALTER TABLE `albaranes`
  ADD PRIMARY KEY (`numeroalbaran`);

--
-- Indices de la tabla `articulo`
--
ALTER TABLE `articulo`
  ADD PRIMARY KEY (`codigoarticulo`);

--
-- Indices de la tabla `articulomateriaprima`
--
ALTER TABLE `articulomateriaprima`
  ADD PRIMARY KEY (`articulo`,`materiaprima`);

--
-- Indices de la tabla `averia`
--
ALTER TABLE `averia`
  ADD PRIMARY KEY (`codigoaveria`);

--
-- Indices de la tabla `clientefax`
--
ALTER TABLE `clientefax`
  ADD PRIMARY KEY (`fax`);

--
-- Indices de la tabla `clientes`
--
ALTER TABLE `clientes`
  ADD PRIMARY KEY (`codigocliente`);

--
-- Indices de la tabla `clientetelefono`
--
ALTER TABLE `clientetelefono`
  ADD PRIMARY KEY (`telefono`);
```



```
--
-- Indices de la tabla `compras`
--
ALTER TABLE `compras`
  ADD PRIMARY KEY (`codigocompra`);

--
-- Indices de la tabla `cuentascontabilidad`
--
ALTER TABLE `cuentascontabilidad`
  ADD PRIMARY KEY (`id`);

--
-- Indices de la tabla `dispositivomovil`
--
ALTER TABLE `dispositivomovil`
  ADD PRIMARY KEY (`codigodispositivomovil`);

--
-- Indices de la tabla `facturas`
--
ALTER TABLE `facturas`
  ADD PRIMARY KEY (`numerofactura`);

--
-- Indices de la tabla `familiaarticulo`
--
ALTER TABLE `familiaarticulo`
  ADD PRIMARY KEY (`nombrefamilia`);

--
-- Indices de la tabla `formapago`
--
ALTER TABLE `formapago`
  ADD PRIMARY KEY (`nombreformapago`);

--
-- Indices de la tabla `lotes`
--
ALTER TABLE `lotes`
  ADD PRIMARY KEY (`codigolote`);

--
-- Indices de la tabla `materiaprima`
--
ALTER TABLE `materiaprima`
  ADD PRIMARY KEY (`codigomateriaprima`);

--
-- Indices de la tabla `proveedor`
--
ALTER TABLE `proveedor`
  ADD PRIMARY KEY (`codigoproveedor`);

--
-- Indices de la tabla `proveedorfax`
--
ALTER TABLE `proveedorfax`
```

Capítulo 2. CÓDIGO FUENTE

```
ADD PRIMARY KEY (`fax`);

--
-- Indices de la tabla `proveedormateriaprima`
--
ALTER TABLE `proveedormateriaprima`
  ADD PRIMARY KEY (`proveedor`,`materiaprima`);

--
-- Indices de la tabla `proveedortelefono`
--
ALTER TABLE `proveedortelefono`
  ADD PRIMARY KEY (`telefono`);

--
-- Indices de la tabla `stock`
--
ALTER TABLE `stock`
  ADD PRIMARY KEY (`codigostock`);

--
-- Indices de la tabla `tarifa`
--
ALTER TABLE `tarifa`
  ADD PRIMARY KEY (`nombretarifa`);

--
-- Indices de la tabla `tarifaarticulo`
--
ALTER TABLE `tarifaarticulo`
  ADD PRIMARY KEY (`articulo`,`tarifa`);

--
-- Indices de la tabla `tclote`
--
ALTER TABLE `tclote`
  ADD PRIMARY KEY (`codigolote`);

--
-- Indices de la tabla `tipoiva`
--
ALTER TABLE `tipoiva`
  ADD PRIMARY KEY (`nombretipoiva`);

--
-- Indices de la tabla `vendedor`
--
ALTER TABLE `vendedor`
  ADD PRIMARY KEY (`codigovendedor`);

--
-- Indices de la tabla `vendedortelefono`
--
ALTER TABLE `vendedortelefono`
  ADD PRIMARY KEY (`telefono`);

--
-- AUTO_INCREMENT de las tablas volcadas
--
```

```
--  
-- AUTO_INCREMENT de la tabla `compras`  
--  
ALTER TABLE `compras`  
    MODIFY `codigocompra` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Codigo de la  
compra auto incremental', AUTO_INCREMENT=28;  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

2.2 Código de la aplicación

2.2.1 Menú Principal. Fmain.cass

```
' Gambas class
' Ruta de los directorios de trabajo
Public CarpetaCarnea As String = User.home & "/carnea"
Public Carpetafacturas As String = CarpetaCarnea & "/facturas"
Public Carpetaalbaranes As String = CarpetaCarnea & "/albaranes"
Public CarpetaLogo As String = CarpetaCarnea & "/logo"
Public CarpetaTrazabilidad As String = CarpetaCarnea & "/tazabilidad"
Public CarpetaCopiasSeguridad As String = CarpetaCarnea & "/copiasseguridad"
Public CarpetaListados As String = CarpetaCarnea & "/listados"
Public DirConexion As String = User.home & "/.config/carnea/conexion"
Public FicheroConexionBD As New Settings(DirConexion & "/conexion.conf")

Public Conexion_Base_Datos As New Connection
    ' Variables para la conexion a la base de datos

'Public ConexionBD_Tipo As String = "mysql"
'Public ConexionBD_Puerto As String = "3306"

Public FacturaAlbaranAMostrar As String ' Almacenamos la ruta de la factura a
mostrar que antes hemos generado en Html
Public Sub Form_Open()
    'Me.Center
    Settings.Read(Me)
    Me.Title = "CARNEA ERP. Version : " & Application.Version
    CrearDirectorios()
    Comprobarficheros()

End

Private Sub CargarDatosConexion()

    Conexion_Base_Datos.Close
    Conexion_Base_Datos.Type = "mysql"
    Conexion_Base_Datos.Host = FicheroConexionBD["ConexionBD/Host"]
    Conexion_Base_Datos.Port = FicheroConexionBD["ConexionBD/Puerto"]
    Conexion_Base_Datos.User = FicheroConexionBD["ConexionBD/Usuario"]
    Conexion_Base_Datos.Password = Utilidades.cripto(2,
FicheroConexionBD["ConexionBD/Password"])
    Conexion_Base_Datos.Name = FicheroConexionBD["ConexionBD/BaseDatos"]

End

Private Sub CrearDirectorios()
    'Message.Info(CarpetaCarnea)
    ' Si no existe el directorio de trabajo es la primera ejecucion del programa.
    If Not Exist(CarpetaCarnea) Then Mkdir CarpetaCarnea
    If Not Exist(DirConexion) Then
        Mkdir User.Home & "/.config/carnea"
        Mkdir DirConexion
    Endif
```

Capítulo 2. CÓDIGO FUENTE

```
If Not Exist(CarpetaFacturas) Then Mkdir CarpetaFacturas
If Not Exist(Carpetaalbaranes) Then Mkdir Carpetaalbaranes
If Not Exist(CarpetaLogo) Then Mkdir CarpetaLogo
If Not Exist(CarpetaTrazabilidad) Then Mkdir CarpetaTrazabilidad
If Not Exist(CarpetaCopiasSeguridad) Then Mkdir CarpetaCopiasSeguridad
If Not Exist(CarpetaListados) Then Mkdir CarpetaListados

Wait 0.5
If Not Exist(CarpetaLogo & "/logo.png") Then Copy
"./imagenes/imagenes/logo.png" To CarpetaLogo & "/logo.png"
End
Private Sub Comprobarficheros()

    If Not Exist(DirConexion & "/conexion.conf") Then
        Frm_Conexionbasedatos.ShowModal()
    Else
        CargarDatosConexion()
    Endif

End

Public Sub MenuSalir_Click()

    Me.Close

End

Public Sub ButtonSalir_Click()
    Settings.write(Me)
    FMain.Conexion_Base_Datos.Close
    Me.close
End

Public Sub ButtonArticulos_Click()

    Frm_Articulos.ShowModal()

End

Public Sub ButtonuUtilidades_Click()

    Frm_Utilidades.ShowModal()

End

Public Sub MenuConexionBd_Click()

    Frm_Conexionbasedatos.ShowModal()

End

Public Sub MenuArticulos_Click()

    Frm_Articulos.ShowModal()

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub ButtonClientes_Click()
    Frm_Clientes.ShowModal()
End

Public Sub MenuClientes_Click()
    Frm_Clientes.ShowModal()
End

Public Sub Menu1_Click()
    Frm_Utilidades.ShowModal()
End

Public Sub MenuVendedores_Click()
    Frm_Vendedores.ShowModal()
End

Public Sub ButtonVendedores_Click()
    Frm_Vendedores.ShowModal()
End

Public Sub ButtonProveedores_Click()
    Frm_Proveedores.ShowModal()
End

Public Sub MenuProveedores_Click()
    Frm_Proveedores.ShowModal()
End

Public Sub ButtonCompras_Click()
    Frm_Compras.ShowModal()
End

Public Sub MenuCompras_Click()
    Frm_Compras.ShowModal()
End

Public Sub ButtonStock_Click()
    Frm_Stock.Show()
End
```

```
Public Sub MenuStock_Click()  
    Frm_Stock.Show()  
End  
  
Public Sub ButtonLotes_Click()  
    Frm_Lotes.ShowModal()  
End  
  
Public Sub MenuLotes_Click()  
    Frm_Lotes.ShowModal()  
End  
  
Public Sub ButtonFacturacion_Click()  
    Frm_Facturacion.ShowModal()  
End  
  
Public Sub MenuFacturacion2_Click()  
    Frm_Facturacion.ShowModal()  
End  
  
Public Sub ButtonTrazabilidad_Click()  
    Frm_Trazabilidad.ShowModal()  
End  
  
Public Sub MenuTrazabilidad_Click()  
    Frm_Trazabilidad.ShowModal()  
End  
  
Public Sub ButtonAverias_Click()  
    Frm_Averias.Show()  
End  
  
Public Sub ButtonCopias_Click()  
    Frm_CopiasSeguridad.Show()  
End  
  
Public Sub MenuCopiasSeguridad_Click()  
    Frm_CopiasSeguridad.Show()
```

Capítulo 2. CÓDIGO FUENTE

```
End

Public Sub ButtonInformes_Click()

    Frm_Informes.Show()

End

Public Sub MenuInformes_Click()

    Frm_Informes.Show()

End

Public Sub Form_Close()

    FMain.Conexion_Base_Datos.Close

End

Public Sub MenuFacturacion3_Click()

    Frm_Facturacion.ShowModal()

End

Public Sub MenuEdicionFactuasAlbaranes_Click()

    Frm_Visualizar_Factura_Albaran.ShowModal()

End

Public Sub MenuFacturarAlbaranes_Click()

    Frm_FacturacionAlbaranes.ShowModal()

End

Public Sub MenuPasoContabilidad_Click()

    Frm_PasoContabilidad.Show()

End
```


2.2.2 Modulo Gestión.

2.2.2.1 Frm_Articulos.class

```
Public Accion_Visualizar_Tarifa As Boolean

Public Sub Form_Open()

    Me.Title = "GESTION DE ARTICULOS"

    Configurar_Tab_Articulos("TODO")
    Configurar_Tab_Familias()
    Configurar_Tab_MateriaPrima()
    Configurar_tab_Tarifa()
    TxtNombreFamilia.SetFocus
    Accion_Visualizar_Tarifa = False

End

Public Sub BtnSalirT_Click()

    Me.close

End

Public Sub BtnSalirA_Click()

    Me.close

End

Public Sub BtnSalirMP_Click()

    Me.close

End

Public Sub BtnSalirF_Click()

    Me.close

End

'''*****
'''                               INICIO CODIGO CONFIGURACION FAMILIAS
'''*****

''' Configuracion Tab Familias y sus objetos

Private Sub Configurar_Tab_Familias()
    Dim Cabeceras As New String[]
```

Capítulo 2. CÓDIGO FUENTE

```
Cabeceras.Add("Nombre")
Cabeceras.Add("Recargo")
DtsFamilia.Connection = FMain.Conexion_Base_Datos
DtsFamilia.table = "familiaarticulo"
DtbFamilia.Labels = Cabeceras
DtbFamilia.View.Columns[0].width = 220
DtbFamilia.View.Columns[1].width = 20
LCDNumeroFamilias.Text = CStr(DtbFamilia.View.Rows.Count)

End

Public Sub BtnAgregarFamilia_Click()

    Dim Recargo As Float
    Dim Sql As String

    If Comprobar_Campos_Familia("AGREGAR") Then
        Recargo = CFloat(TxtRecargoFamilia.text)
        Sql = "insert into familiaarticulo (nombrefamilia,recargo) values ('" &
        UCase(TxtNombreFamilia.Text) & "'," & Recargo & ")"
        Utilidades.In_Act_Bor(Sql)
        DtsFamilia.Refresh
        DtbFamilia.Refresh
        TxtNombreFamilia.text = ""
        TxtRecargoFamilia.Text = ""
        Configurar_Tab_Articulos("COMBOBOX")
        DtsFamilia.table = "familiaarticulo"
        LCDNumeroFamilias.Text = CStr(DtbFamilia.View.Rows.Count)
    Endif

End

Public Sub BtnBorrarFamilia_Click()
    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbFamilia.View.Row
    If dtbFamilia.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar Familia : " &
        DtbFamilia.View[Numero_fila, 0].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from familiaarticulo where nombrefamilia='" &
            DtbFamilia.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_Tab_Articulos("COMBOBOX")
            DtsFamilia.table = "familiaarticulo"
            LCDNumeroFamilias.Text = CStr(DtbFamilia.View.Rows.Count)
        Endif
    Endif
End

Public Sub TxtNombreFamilia_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtRecargoFamilia.SetFocus()
    Endif

End

Public Sub BtnModificarF_Click()
```

Capítulo 2. CÓDIGO FUENTE

```
Dim Sql As String
Dim Numero_fila As Integer

Numero_fila = DtbFamilia.View.Row
Numero_fila = DtbFamilia.View.Row
If LblModificarF.Text = "EDITAR" Then
    LblModificarF.Text = "ACTUALIZAR"
    TxtNombreFamilia.Text = DtbFamilia.View[Numero_fila, 0].Text
    TxtRecargoFamilia.Text = DtbFamilia.View[Numero_fila, 1].Text
    TxtNombreFamilia.Enabled = False
    BtnAgregarFamilia.Enabled = False
    BtnBorrarFamilia.Enabled = False
    TxtRecargoFamilia.SetFocus
Else
    If Comprobar_Campos_Familia("MODIFICAR") Then
        sql = "update familiaarticulo set recargo=" &
CFloat(TxtRecargoFamilia.Text) & " where nombrefamilia='" &
UCase(TxtNombreFamilia.Text) & "'"
        Utilidades.In_Act_Bor(sql)
        lblModificarF.Text = "EDITAR"
        TxtNombreFamilia.Text = ""
        TxtRecargoFamilia.Text = ""
        BtnAgregarFamilia.Enabled = True
        BtnBorrarFamilia.Enabled = True
        TxtNombreFamilia.Enabled = True
        DtsFamilia.table = "familiaarticulo"
    Endif
Endif

Endif

End

''' Funcion para comprobar los campos del Tab familias<br>
''' Parametros:<br>
'''     Accion - - > La accion desde la que comprobamos.<br>
'''             AGREGAR --> Para agregar datos.<br>
'''             MODIFICAR --> Cuando estamos modificando un dato.<br>
''' Devuelve:<br>
'''     TRUE --> Si los valores son correctos.<br>
'''     FALSE --> valores erroneos.

Private Function Comprobar_Campos_Familia(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    Sql = "select * from familiaarticulo where nombrefamilia='" &
UCase(TxtNombreFamilia.Text) & "'"
    If TxtNombreFamilia.Text = "" Then
        Balloon.Error("Debe introducir un nombre de Familia que va a crear",
TxtNombreFamilia)
        Validar = False
        TxtNombreFamilia.SetFocus
    Else If TxtRecargoFamilia.text = ""
        Balloon.Error("Debe el recargo en € que lleva la Familia", TxtRecargoFamilia)
        Validar = False
        TxtRecargoFamilia.SetFocus
    Else If Not IsFloat(TxtRecargoFamilia.Text)
        Balloon.Error("El recargo debe ser un numero decimal", TxtRecargoFamilia)
        Validar = False
    End If
```

Capítulo 2. CÓDIGO FUENTE

```
    TxtRecargoFamilia.SetFocus
Endif
If Accion = "AGREGAR" Then
    If Utilidades.Comprobar_existe_dato(Sql) Then
        Balloon.Error("El nombre de la familia ya existe", TxtNombreFamilia)
        Validar = False
        TxtNombreFamilia.SetFocus
    Endif

Endif
Return Validar
End

'*****
''
''          FIN CODIGO CONFIGURACION FAMILIAS
''*****

'*****
''
''          INICIO CODIGO CONFIGURACION MATERIA PRIMA
''*****

'' Configuracion Tab Materia Prima y sus Objetos
Private Sub Configurar_Tab_MateriaPrima()
    Dim Cabeceras As New String[]

    Cabeceras.Add("Codigo")
    Cabeceras.Add("Nombre")
    DtsMateriaPrima.Connection = FMain.Conexion_Base_Datos
    DtsMateriaPrima.table = "materiaprima"
    DtbMateriaPrima.Labels = Cabeceras
    DtbMateriaPrima.View.Columns[0].width = 100
    DtbMateriaPrima.View.Columns[1].width = 150
    LCDNumeroMP.Text = CStr(DtbMateriaPrima.View.Rows.Count)

End

Public Sub BtnAgregarMP_Click()

    Dim Sql As String

    If Comprobar_Campos_MateriaPrima("AGREGAR") Then

        Sql = "insert into materiaprima (codigomateriaprima,nombre) values ('" &
        UCase(TxtCodigoMateriaPrima.Text) & "',''" & UCase(TxtNombreMateriaPrima.Text) &
        "',''"
        Utilidades.In_Act_Bor(Sql)
        TxtNombreMateriaPrima.text = ""
        TxtCodigoMateriaPrima.Text = ""
        Configurar_Tab_Articulos("LISTBOX")
        DtsMateriaPrima.table = "materiaprima"
        LCDNumeroMP.Text = CStr(DtbMateriaPrima.View.Rows.Count)
    Endif

End

'' Funcion para comprobar los campos del Tab Materia Prima<br>
'' Parametros:<br>
```

Capítulo 2. CÓDIGO FUENTE

```
' '      Accion - - > La accion desde la que comprobamos.<br>
' '      AGREGAR --> Para agregar datos.<br>
' '      MODIFICAR --> Cuando estamos modificando un dato.<br>
' '      Devuelve:<br>
' '      TRUE --> Si los valores son correctos.<br>
' '      FALSE --> valores erroneos.

Private Function Comprobar_Campos_MateriaPrima(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    Sql = "select * from materiaprima where codigomateriaprima='" &
    UCase(TxtCodigoMateriaPrima.Text) & "'"
    If TxtCodigoMateriaPrima.Text = "" Then
        Balloon.Error("Debe introducir unCodigo de Materia Prima que va a crear",
        TxtCodigoMateriaPrima)
        Validar = False
        TxtCodigoMateriaPrima.SetFocus
    Else If TxtNombreMateriaPrima.text = ""
        Balloon.Error("Debe introducir el nombre de la Materia Prima",
        TxtNombreMateriaPrima)
        Validar = False
        TxtNombreMateriaPrima.SetFocus
    Endif
    If Accion = "AGREGAR" Then
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El código Materia Prima ya existe", TxtCodigoMateriaPrima)
            Validar = False
            TxtCodigoMateriaPrima.SetFocus
        Endif
    Endif
    Return Validar
End

Public Sub BtnBorrarMP_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbMateriaPrima.View.Row
    If DtbMateriaPrima.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar la Materia Prima : " &
        DtbMateriaPrima.View[Numero_fila, 1].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from materiaprima where codigomateriaprima='" &
            DtbMateriaPrima.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_Tab_Articulos("LISTBOX")
            DtsMateriaPrima.table = "materiaprima"
            LCDNumeroMP.Text = CStr(DtbMateriaPrima.View.Rows.Count)
        Endif
    Endif
End

Public Sub BtnModificarMP_Click()
```

Capítulo 2. CÓDIGO FUENTE

```
Dim Sql As String
Dim Numero_fila As Integer

Numero_fila = DtbMateriaPrima.View.Row
If LblModificarMP.Text = "EDITAR" Then
    LblModificarMP.Text = "ACTUALIZAR"
    TxtCodigoMateriaPrima.Text = DtbMateriaPrima.View[Numero_fila, 0].Text
    TxtNombreMateriaPrima.Text = DtbMateriaPrima.View[Numero_fila, 1].Text
    TxtCodigoMateriaPrima.Enabled = False
    BtnAgregarMP.Enabled = False
    BtnBorrarMP.Enabled = False
    TxtNombreMateriaPrima.SetFocus
Else
    If Comprobar_Campos_MateriaPrima("MODIFICAR") Then
        sql = "update materiaprima set nombre='" &
        UCase(TxtNombreMateriaPrima.Text) & "' where codigomateriaprime='" &
        UCase(TxtCodigoMateriaPrima.Text) & "'"
        Utilidades.In_Act_Bor(sql)
        LblModificarMP.Text = "EDITAR"
        TxtCodigoMateriaPrima.Text = ""
        TxtNombreMateriaPrima.Text = ""
        TxtCodigoMateriaPrima.Enabled = True
        BtnAgregarMP.Enabled = True
        BtnBorrarMP.Enabled = True
        Configurar_Tab_Articulos("LISTBOX")
        DtsMateriaPrima.table = "materiaprime"
    Endif
Endif

End

Public Sub TxtCodigoMateriaPrima_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtNombreMateriaPrima.SetFocus()
    Endif

End

Public Sub TxtRecargoFamilia_KeyRelease()
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        BtnAgregarFamilia_Click()
        TxtNombreFamilia.SetFocus
    Endif
End

Public Sub TxtNombreMateriaPrima_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        BtnAgregarMP_Click()
        TxtCodigoMateriaPrima.SetFocus()
    Endif

End

Public Sub BtnAAgregarMP_Click()
    Dim Contador As Integer
    Dim i As Integer
```

```
Dim Lista_Seleccionados As New String[]
Dim Lista_NoSeleccionados As New String[]
Contador = LtBKMP.Count
For i = 0 To Contador - 1
    If LtBKMP[i].Selected Then
        Lista_Seleccionados.Add(LtBKMP[i].Text)
    Else
        Lista_NoSeleccionados.Add(LtBKMP[i].Text)
    Endif
Next
' Añadimos a LtBKMPComposicion los seleccionados
contador = Lista_Seleccionados.Count
For i = 0 To contador - 1
    LtBKMPComposicion.Add(Lista_Seleccionados[i])
Next
' Borramos ltBKMP y agregamos los que no hemos seleccionado.
contador = Lista_NoSeleccionados.Count
LtBKMP.Clear
For i = 0 To contador - 1
    LtBKMP.Add(Lista_NoSeleccionados[i])
Next
LtBKMP.UnselectAll
End

Public Sub BtnABorrarMP_Click()

    Dim Contador As Integer
    Dim i As Integer
    Dim Lista_Seleccionados As New String[]
    Dim Lista_NoSeleccionados As New String[]
    Contador = LtBKMPComposicion.Count
    For i = 0 To Contador - 1
        If LtBKMPComposicion[i].Selected Then
            Lista_Seleccionados.Add(LtBKMPComposicion[i].Text)
        Else
            Lista_NoSeleccionados.Add(LtBKMPComposicion[i].Text)
        Endif
    Next
    ' Añadimos a LtBKMP los seleccionados
    contador = Lista_Seleccionados.Count
    For i = 0 To contador - 1
        LtBKMP.Add(Lista_Seleccionados[i])
    Next
    ' Borramos ltBKMPComposicion y agregamos los que no hemos seleccionado.
    contador = Lista_NoSeleccionados.Count
    LtBKMPComposicion.Clear
    For i = 0 To contador - 1
        LtBKMPComposicion.Add(Lista_NoSeleccionados[i])
    Next
    ' Deseleccionamos todos ya que se quedan seleccionados
    LtBKMPComposicion.UnselectAll
End

''*****
''
''                               FIN CODIGO CONFIGURACION MATERIA PRIMA
```

Capítulo 2. CÓDIGO FUENTE

```
''*****
''*****
''          INICIO CODIGO CONFIGURACION ARTICULOS
''*****

'' Configuramos Tab Articulos y sus objetos <br>
'' Parametros:<br>
''     Accion:<br>
''         COMBOBOX --> Llenamos el Combobox de las Familias<br>
''         LISTBOX --> Llenamos el ListBox de la Materia Prima<br>
''         TODO --> Configuracion inicial, llenamos Combobox y Listobox<br>
Private Sub Configurar_Tab_Articulos(Accion As String)
    Dim Cabeceras As New String[]
    Dim Error_Tablas_Vacias As String
    Cabeceras.Add("Codigo")
    Cabeceras.Add("Nombre")
    Cabeceras.Add("Familia")
    Error_Tablas_Vacias = ""
    If Utilidades.Tabla_Vacia("select * from familiaarticulo") = 0 Then
        Error_Tablas_Vacias = " Debe crear Familias"
    Endif
    If Utilidades.Tabla_Vacia("select * from materiaprima") = 0 Then
        If Error_Tablas_Vacias = "" Then
            Error_Tablas_Vacias = " Debe crear materiaprima"
        Else
            Error_Tablas_Vacias = Error_Tablas_Vacias & " y agregar Materia Primas"
        Endif
    Endif
    If Error_Tablas_Vacias = "" Then
        TxtlerrorArticulos.Visible = False
        TxtlerrorArticulos.Text = ""
    Else
        TxtlerrorArticulos.Visible = True
        Error_Tablas_Vacias = Error_Tablas_Vacias & " antes de poder Crear
Articulos."
        TxtlerrorArticulos.Text = Error_Tablas_Vacias
    Endif

    If Accion = "COMBOBOX" Then
        Utilidades.Llenar_Combobox("select * from familiaarticulo",
CmbFamiliaArticulo, "nombrefamilia")
        CmbFamiliaArticulo.Index = -1
    Else If Accion = "LISTBOX"
        Utilidades.Llenar_Listbox("select * from materiaprima", LtBKMP, "nombre")
    Else If Accion = "TODO"
        Utilidades.Llenar_Combobox("select * from familiaarticulo",
CmbFamiliaArticulo, "nombrefamilia")
        Utilidades.Llenar_Listbox("select * from materiaprima", LtBKMP, "nombre")
        DtsArticulos.Connection = FMain.Conexion_Base_Datos
        DtsArticulos.table = "articulo"
        DtbArticulos.Labels = Cabeceras
        DtbArticulos.View.Columns[0].width = 100
        DtbArticulos.View.Columns[1].width = 200
        DtbArticulos.View.Columns[2].width = 200
    Endif
End

End
```



```
Public Sub TxtCodigoArticulo_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtNombreArticulo.SetFocus
    Endif

End

Public Sub TxtNombreArticulo_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        CmbFamiliaArticulo.SetFocus
    Endif

End

Public Sub BtnAgregarArticulo_Click()

    Dim Sql As String
    Dim Contador As Integer
    Dim i As Integer
    Dim C_MateriaPrima As String

    If Comprobar_Campos_Articulos("AGREGAR") Then
        ' Insertamos el articulo en la tabla Articulo
        Sql = "insert into articulo (codigoarticulo,nombre,familia) values('" &
        UCase(TxtCodigoArticulo.Text) & "',''" & UCase(TxtNombreArticulo.Text) & "',''" &
        UCase(CmbFamiliaArticulo.Text) & "')"
        Utilidades.In_Act_Bor(Sql)
        ' Insertamos en ArticuloMateriaPrima la composicion del articulo.
        Contador = LtBKMPComposicion.Count

        For i = 0 To Contador - 1
            sql = "select * from materiaprima where nombre='" &
            LtBKMPComposicion[i].Text & "'"
            C_Materiaprima = Utilidades.Obtener_Campo_Tabla(sql,
            "codigomateriaprima")
            sql = "insert into articulomateriaprima (articulo,materiaprima)
            values('" & UCase(TxtCodigoArticulo.Text) & "',''" & UCase(C_Materiaprima) & "')"
            Utilidades.In_Act_Bor(Sql)
        Next
        LtBKMPComposicion.Clear
        TxtCodigoArticulo.Text = ""
        TxtNombreArticulo.Text = ""
        Configurar_Tab_Articulos("TODO")
        TxtCodigoArticulo.SetFocus
        Configurar_tab_Tarifa()
    Endif

End

Public Sub BtnBorrarArticulo_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbArticulos.View.Row
    If DtbArticulos.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar el Articulo : " &
        DtbArticulos.View[Numero_fila, 1].Text, "Si", "No")
    End If

End
```

Capítulo 2. CÓDIGO FUENTE

```
        If respuesta = 1 Then
            sql = "delete from articulo where codigoarticulo='" &
DtbArticulos.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            sql = "delete from articulomateriaprima where articulo='" &
DtbArticulos.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            DtsArticulos.table = "articulo"
            Configurar_tab_Tarifa()
        Endif
    Endif

End

'' Funcion para comprobar los Campos del Tab Articulos<br>
'' Parametros:<br>
''     Accion - - > La accion desde la que comprobamos.<br>
''                     AGREGAR --> Para agregar datos.<br>
''                     MODIFICAR --> Cuando estamos modificando un dato.<br>
'' Devuelve:<br>
''     TRUE --> Si los valores son correctos.<br>
''     FALSE --> valores erroneos.
Private Function Comprobar_Campos_Articulos(Accion As String) As Boolean
    Dim Sql As String
    Dim Validar As Boolean
    Validar = True

    Sql = "select * from articulo where codigoarticulo='" &
UCase(TxtCodigoArticulo.Text) & "'"
    If TxtCodigoArticulo.Text = "" Then
        Balloon.Error("Debe introducir unCodigo de Articulo que va a crear",
TxtCodigoArticulo)
        Validar = False
        TxtCodigoArticulo.SetFocus
    Else If TxtNombreArticulo.text = ""
        Balloon.Error("Debe introducir el nombre del Articulo", TxtNombreArticulo)
        Validar = False
        TxtNombreArticulo.SetFocus
    Else If LtBKMPComposicion.Count = 0
        Balloon.Error("Debe añadir las materias primas que componen el articulo, si
el articulo no tiene composicion. <br> Cree una Materia Prima con el mismo
Nombre para despues crear los lotes", LtBKMPComposicion)
        Validar = False
    Endif
    If Accion = "AGREGAR" Then
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El codigo Articulo ya existe", TxtCodigoArticulo)
            Validar = False
            TxtCodigoArticulo.SetFocus
        Endif
    Endif

Endif
Return Validar

End

Public Sub BtnEditarArticulo_Click()
    Dim Sql As String
```

```

Dim Numero_fila As Integer
Dim Contador As Integer
Dim C_MateriaPrima As String
Dim i As Integer
Numero_fila = DtbArticulos.View.Row ' Posicion del DtbArticulos
If BtnEditarArticulo.Text = "EDITAR" Then
    BtnEditarArticulo.Text = "ACTUALIZAR"
    TxtCodigoArticulo.Text = DtbArticulos.View[Numero_fila, 0].Text
    TxtNombreArticulo.Text = DtbArticulos.View[Numero_fila, 1].Text
    CmbFamiliaArticulo.Text = DtbArticulos.View[Numero_fila, 2].Text
    BtnAgregarArticulo.Enabled = False
    BtnBorrarArticulo.Enabled = False
    TxtCodigoArticulo.Enabled = False
    TxtNombreArticulo.SetFocus
    ' Llenamos el Listbox con las materias primas.
    sql = "select * from articulomateriaprima where articulo='" &
TxtCodigoArticulo.text & "'"
    Llenar_Composicion_Articulo_MateriaPrima(sql)
    Eliminar_Materia_Prima_Listboxmp()
    DtsArticulos.table = "articulo"
    DtbArticulos.View.Columns[0].width = 100
    DtbArticulos.View.Columns[1].width = 200
    DtbArticulos.View.Columns[2].width = 200
Else
    ' Comprobamos que los datos son correctos
    If Comprobar_Campos_Articulos("MODIFICAR") Then
        ' Actualizamos los campos del Articulo
        sql = "update articulo set nombre='" & UCase(TxtNombreArticulo.text) &
        "','familia='" & UCase(CmbFamiliaArticulo.Text) & "'" where codigoarticulo='" &
        UCase(TxtCodigoArticulo.Text) & "'"
        Utilidades.In_Act_Bor(sql)
        ' Borramos Las materias primas que componen este articulo y añadimos las
        que tengamos en el ListBox de composicion
        sql = "delete from articulomateriaprima where articulo='" &
        TxtCodigoArticulo.text & "'"
        Utilidades.In_Act_Bor(Sql)
        Contador = LtBKMPComposicion.Count

        For i = 0 To Contador - 1
            sql = "select * from materiaprima where nombre='" &
            LtBKMPComposicion[i].Text & "'"
            C_Materiaprima = Utilidades.Obtener_Campo_Tabla(sql,
            "codigomateriaprima")
            sql = "insert into articulomateriaprima (articulo,materiaprima)
            values('" & UCase(TxtCodigoArticulo.Text) & "',''" & UCase(C_Materiaprima) & "'"")
            Utilidades.In_Act_Bor(Sql)
        Next
    Endif
    BtnAgregarArticulo.Enabled = True
    BtnBorrarArticulo.Enabled = True
    TxtCodigoArticulo.Enabled = True
    BtnEditarArticulo.Text = "EDITAR"
    TxtCodigoArticulo.Text = ""
    TxtNombreArticulo.Text = ""
    CmbFamiliaArticulo.text = ""
    DtbArticulos.Enabled = True
    LtBKMPComposicion.clear
    Configurar_Tab_Articulos("TODO")
Endif

```

End

'' Procedimiento que elimina los elementos que estan en el Listbox de la composicion de los elementos del listbos de materia prima.


```
Private Sub Eliminar_Materia_Prima_Listboxmp()  
    Dim i As Integer  
    Dim j As Integer  
    Dim Contador_Listobox_MP As Integer  
    Dim Contador_Listobox_Composicion As Integer  
    Dim Salida As Integer  
    salida = 0  
    j = 0  
    Contador_Listobox_Composicion = LtBKMPComposicion.Count  
    Contador_Listobox_MP = LtBKMP.Count  
    For i = 0 To Contador_Listobox_Composicion - 1  
        While salida = 0  
            If LtBKMP[j].Text = LtBKMPComposicion[i].Text Then  
                salida = 1  
                LtBKMP.Remove(j)  
            Endif  
            j = j + 1  
        Wend  
        salida = 0  
        j = 0  
    Next  
End
```

'' Procedimiento que nos llena el listbox de la composicion del articulo con el articulo seleccionado.

'' Recibe como parametro la sentencia SQL

```
Private Sub Llenar_Composicion_Articulo_MateriaPrima(Sql As String)  
    Dim Result_Trabajo As Result  
    Dim Materia_Prima As New String[]  
    Dim Materia_Prima_Nombre As String  
    Dim Sql_Trabajo As String  
    Dim i As Integer  
    Dim Contador As Integer  
    LtBKMPComposicion.Clear  
    FMain.Conexion_Base_Datos.Close  
    FMain.Conexion_Base_Datos.Open  
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)  
    Result_Trabajo.MoveFirst  
    ' Obtenemos los identifiacores de la materia prima  
    While Result_Trabajo.Available  
        Materia_Prima.Add(Result_Trabajo["materiaprima"])  
        Result_Trabajo.MoveNext  
    Wend  
    FMain.Conexion_Base_Datos.Close  
    contador = Materia_Prima.Count  
    LtBKMPComposicion.Clear  
    ' Llenamos con el nombre el Listbox.  
    For i = 0 To contador - 1  
        Sql_Trabajo = "select * from materiaprima where codigomateriaprima='" &  
Materia_Prima[i] & "'"  
        Materia_Prima_Nombre = Utilidades.Obtener_Campo_Tabla(Sql_Trabajo,  
"nombre")  
        LtBKMPComposicion.Add(Materia_Prima_Nombre)
```

Capítulo 2. CÓDIGO FUENTE

```
Next
LtBKMPComposicion.UnselectAll

End
Public Sub BtnVisualizarComposicion_Click()

    If BtnVisualizarComposicion.text = "COMPOSICION ARTICULOS" Then
        BtnAgregarArticulo.Enabled = False
        BtnBorrarArticulo.Enabled = False
        BtnEditarArticulo.Enabled = False
        BtnAAgregarMP.Enabled = False
        BtnABorrarMP.Enabled = False
        CmbVisualizarComposicion.Visible = True
        Utilidades.Llenar_Combobox("select nombre from articulo",
CmbVisualizarComposicion, "nombre")
        CmbVisualizarComposicion.text = ""
        BtnVisualizarComposicion.text = "S    A    L    I    R"
    Else
        BtnAgregarArticulo.Enabled = True
        BtnBorrarArticulo.Enabled = True
        BtnEditarArticulo.Enabled = True
        BtnAAgregarMP.Enabled = True
        BtnABorrarMP.Enabled = True
        CmbVisualizarComposicion.Visible = False
        CmbFamiliaArticulo.Index = -1
        LtBKMPComposicion.Clear
        BtnVisualizarComposicion.text = "COMPOSICION ARTICULOS"
    Endif
DtsArticulos.Table = "articulo"
DtbArticulos.View.Columns[0].width = 100
DtbArticulos.View.Columns[1].width = 200
DtbArticulos.View.Columns[2].width = 200
End
Public Sub CmbVisualizarComposicion_Click()
    Dim sql As String
    Dim Codigo_Articulo As String
    sql = "select * from articulo where nombre='" & CmbVisualizarComposicion.text
& "''"
    Codigo_Articulo = Utilidades.Obtener_Campo_Tabla(sql, "codigoarticulo")
    sql = "select familia from articulo where codigoarticulo='" & Codigo_Articulo
& "''"
    CmbFamiliaArticulo.text = Utilidades.Obtener_Campo_Tabla(sql, "familia")
    sql = "select * from articulomateriaprima where articulo='" & Codigo_Articulo
& "''"
    Llenar_Composicion_Articulo_MateriaPrima(sql)
    DtsArticulos.Table = "articulo"
    DtbArticulos.View.Columns[0].width = 100
    DtbArticulos.View.Columns[1].width = 200
    DtbArticulos.View.Columns[2].width = 200
End

'*****
'
'                               FIN CODIGO CONFIGURACION ARTICULOS
'*****

'*****
'
'                               INICIO CODIGO CONFIGURACION TARIFAS
'*****
```

Capítulo 2. CÓDIGO FUENTE

```
' ' Funcion para comprobar los Objetos del Tab Tarifas<br>
' ' Parametros:<br>
' '     Accion - - > La accion desde la que comprobamos.Si es a ñadir tenermos
que comprobar que el nombre de familia existe en caso contrario no.<br>
' '     AGREGAR --> Para agregar datos.<br>
' '     PRECIO --> Cuando estamos añadiendo precios a los
articulos.<br>
' '     EDITAR --> Para editar las tarifas.<br>
' '     GUARDAR --> Para guardar las tarifas.<br>
' '     BORRAR --> Para borrar las tarifas.<br>
' ' Devuelve:<br>
' '     TRUE --> Si los valores son correctos.<br>
' '     FALSE --> valores erroneos.
Private Function Comprobar_Campos_Tarifas(Accion As String) As Boolean
    Dim Sql As String
    Dim Validar As Boolean
    Validar = True

    If Accion = "GUARDAR" Then
        If DtsTarifa.Count = 0 Then
            Balloon.Error("Debe añadir al menos un articulo a la tarifa.",
CmbTArticulos)
            Validar = False
        Endif
    Endif
    If Accion = "PRECIO" Then
        If CmbTArticulos.text = ""
            Balloon.Error("Debe elegir un Articulo para asignarle precio.",
CmbTArticulos)
            Validar = False
            CmbTArticulos.SetFocus
        Else If Val(VlBPrecioArticulo.text) = 0
            Balloon.Error("El valor del articulo debe ser mayor que 0",
VlBPrecioArticulo)
            Validar = False
            VlBPrecioArticulo.SetFocus
        Endif
    Endif
    If Accion = "AGREGAR" Then
        Sql = "select * from tarifaarticulo where tarifa='" &
UCase(TxtNombreTarifa.Text) & "'"
        If TxtNombreTarifa.Text = "" Then
            Balloon.Error("Debe introducir un Nombre para la Tarifa ",
TxtNombreTarifa)
            Validar = False
            TxtNombreTarifa.SetFocus
        Else If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("Nombre de Tarifa duplicado.", TxtNombreTarifa)
            Validar = False
            TxtNombreTarifa.SetFocus
        Endif
    Endif
    If Accion = "EDITAR" Then
        If CmbTarifaEditar.text = "" Then
            Balloon.Error("Elija Tarifa a Editar.", CmbTarifaEditar)
            Validar = False
        Endif
    Endif
    If Accion = "BORRAR" Then
```

Capítulo 2. CÓDIGO FUENTE

```
    If CmbTarifaEditar.text = "" Then
        Balloon.Error("Elija Tarifa a Borrar.", CmbTarifaEditar)
        Validar = False
    Endif
Endif
Return Validar

End

Public Sub BtnAgregarTarifa_Click()
    Dim sql As String
    If Comprobar_Campos_Tarifas("AGREGAR") Then
        FrameTarifas.Enabled = True
        FrameTTarifas.Enabled = False
        TxtNombreTarifa.Enabled = False
        sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & TxtNombreTarifa.Text & "'"
        DtsTarifa.table = sql
        DtBTarifa.View.Columns[0].width = 200
        DtBTarifa.View.Columns[1].width = 100
        DtBTarifa.View.Columns[2].width = 100

    Endif

End

Public Sub BtnGuardarAT_Click()
    Dim Sql As String

    If Comprobar_Campos_Tarifas("GUARDAR") Then

        Sql = "select * from tarifa where nombretarifa='" &
UCase(TxtNombreTarifa.Text) & "'"

        If Utilidades.Comprobar_existe_dato(sql) Then
            Else
                Sql = "insert into tarifa(nombretarifa) values('" &
UCase(TxtNombreTarifa.Text) & "')"
                Utilidades.In_Act_Bor(Sql)
            Endif
            Utilidades.Llenar_Combobox("select * from tarifa", CmbTarifaEditar,
"nombretarifa")
            TxtNombreTarifa.text = ""
            VLBPrecioArticulo.Text = ""
            CmbTArticulos.Text = ""
            TxtNombreTarifa.Enabled = True
            FrameTarifas.Enabled = False
            FrameTTarifas.Enabled = True
            CmbTarifaEditar.enabled = True
            sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo order by T.tarifa"
            DtsTarifa.table = sql
            DtBTarifa.View.Columns[0].width = 200
            DtBTarifa.View.Columns[1].width = 100
            DtBTarifa.View.Columns[2].width = 100
        Endif
    End
End
Public Sub BtnAgregarAT_Click()
    Dim Sql As String
```

Capítulo 2. CÓDIGO FUENTE

```
Dim Codigo_Articulo As String
Dim Articulo_Eliminar As String
If Comprobar_Campos_Tarifas("PRECIO") Then
    Codigo_Articulo = Utilidades.Obtener_Campo_Tabla("select * from articulo
where nombre='" & CmbTArticulos.Text & "'", "codigoarticulo")
    sql = "insert into tarifaarticulo(articulo,tarifa,precio) values('" &
UCase(Codigo_Articulo) & "','" & UCase(TxtNombreTarifa.text) & "','" &
CFloat(VlBPrecioArticulo.text) & ")"
    Utilidades.In_Act_Bor(Sql)
    VlBPrecioArticulo.text = ""
    Articulo_Eliminar = CmbTArticulos.Text
    Utilidades.Eliminar_Elemento_ComboBox(CmbTArticulos, Articulo_Eliminar)
    sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & TxtNombreTarifa.Text & "'"
    DtsTarifa.table = sql
    DtBTarifa.View.Columns[0].width = 200
    DtBTarifa.View.Columns[1].width = 100
    DtBTarifa.View.Columns[2].width = 100
Endif

End

Public Sub BtnEditarTarifa_Click()
    Dim Sql As String
    Dim Numero_Elementos_Dts As Integer
    Dim i As Integer
    If Comprobar_Campos_Tarifas("EDITAR") Then
        FrameTarifas.Enabled = True
        FrameTTarifas.Enabled = False
        TxtNombreTarifa.text = CmbTarifaEditar.Text
        CmbTarifaEditar.Enabled = False
        TxtNombreTarifa.Enabled = False
        Utilidades.Llenar_Combobox("select * from articulo", CmbTArticulos,
"nombre")
        sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & CmbTarifaEditar.Text & "'"
        DtsTarifa.table = sql
        DtBTarifa.View.Columns[0].width = 200
        DtBTarifa.View.Columns[1].width = 100
        DtBTarifa.View.Columns[2].width = 100
        Numero_Elementos_Dts = DtsTarifa.Count
        For i = 0 To Numero_Elementos_Dts - 1
            Utilidades.Eliminar_Elemento_ComboBox(CmbTArticulos, DtBTarifa.View[i,
1].Text)
        Next
    Endif

End

Public Sub BtnBorrarTarifa_Click()
    Dim Respuesta As Integer
    Dim Sql As String
    If Comprobar_Campos_Tarifas("BORRAR") Then
        Respuesta = Message.Delete("Desea Borrar la Tarifa : " &
CmbTarifaEditar.Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from tarifa where nombretarifa='" & CmbTarifaEditar.Text &
"'"
```


Capítulo 2. CÓDIGO FUENTE

```
        Utilidades.In_Act_Bor(Sql)
        sql = "delete from tarifaarticulo where tarifa='" & CmbTarifaEditar.Text &
""
        Utilidades.In_Act_Bor(Sql)
        Utilidades.Eliminar_Elemento_ComboBox(CmbTarifaEditar,
CmbTarifaEditar.Text)
        CmbTarifaEditar.Text = ""
        sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo order by T.tarifa"
        DtsTarifa.table = sql
        DtBTarifa.View.Columns[0].width = 200
        DtBTarifa.View.Columns[1].width = 100
        DtBTarifa.View.Columns[2].width = 100
    Endif
Endif

End

Public Sub BtnBorrarAT_Click()
    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Dim Nombre_Articulo As String
    Dim Codigo_Articulo As String
    Numero_fila = DtBTarifa.View.Row
    If DtBTarifa.View.Rows.Count > 0 Then
        Codigo_Articulo = DtBTarifa.View[Numero_fila, 0].Text

        Nombre_articulo = DtBTarifa.View[Numero_fila, 1].Text
        sql = "select * from articulo where nombre='" & Nombre_articulo & "'"
        Codigo_Articulo = Utilidades.Obtener_Campo_Tabla(sql, "codigoarticulo")
        Respuesta = Message.Delete("Desea Borrar el Artículo : " & Nombre_Articulo,
"Si", "No")
        If respuesta = 1 Then
            sql = "delete from tarifaarticulo where articulo='" & codigo_Articulo & "'"
and tarifa='" & TxtNombreTarifa.Text & "'"
            CmbTArticulos.Add(Nombre_Articulo)
            Utilidades.In_Act_Bor(Sql)
            sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & TxtNombreTarifa.Text & "'"
            DtsTarifa.table = sql
            DtBTarifa.View.Columns[0].width = 200
            DtBTarifa.View.Columns[1].width = 100
            DtBTarifa.View.Columns[2].width = 100
        Else
            sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & TxtNombreTarifa.Text & "'"
            DtsTarifa.table = sql
            DtBTarifa.View.Columns[0].width = 200
            DtBTarifa.View.Columns[1].width = 100
            DtBTarifa.View.Columns[2].width = 100
        Endif
    Endif

End

Public Sub BtnVisualizarTarifa_Click()
    Dim Sql As String
```

Capítulo 2. CÓDIGO FUENTE

```
If CmbTarifaEditar.text = "" Then
    Balloon.Error("Debe elegir tarifa a visualizar.", CmbTarifaEditar)
Else
    CmbTarifaEditar.Enabled = True
    FrameTarifas.Enabled = True
    FrameTTarifas.Enabled = False
    BtnSalirVisualizarTarifas.visible = True
    LblSalirVisualizar.visible = True
    Accion_Visualizar_Tarifa = True
    BtnAgregarAT.Enabled = False
    BtnBorrarAT.Enabled = False
    BtnGuardarAT.Enabled = False
    CmbTArticulos.Enabled = False
    VLBPrecioArticulo.Enabled = False
    sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & CmbTarifaEditar.Text & "'"
    DtsTarifa.table = sql
    DtBTarifa.View.Columns[0].width = 200
    DtBTarifa.View.Columns[1].width = 100
    DtBTarifa.View.Columns[2].width = 100
Endif

End

Public Sub CmbTarifaEditar_Click()
    Dim Sql As String

    If Accion_Visualizar_Tarifa Then
        sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo and T.tarifa='" & CmbTarifaEditar.Text & "'"
        DtsTarifa.table = sql
        DtBTarifa.View.Columns[0].width = 200
        DtBTarifa.View.Columns[1].width = 100
        DtBTarifa.View.Columns[2].width = 100

    Endif

End

Public Sub BtnSalirVisualizarTarifas_Click()
    Dim Sql As String
    FrameTarifas.Enabled = False
    FrameTTarifas.Enabled = True
    BtnSalirVisualizarTarifas.visible = False
    LblSalirVisualizar.visible = False
    Accion_Visualizar_Tarifa = False
    CmbTarifaEditar.Text = ""
    BtnAgregarAT.Enabled = True
    BtnBorrarAT.Enabled = True
    BtnGuardarAT.Enabled = True
    CmbTArticulos.Enabled = True
    VLBPrecioArticulo.Enabled = True
    CmbTarifaEditar.Enabled = False
    sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T
where A.codigoarticulo=T.articulo order by T.tarifa"
    DtsTarifa.table = sql
    DtBTarifa.View.Columns[0].width = 200
```

Capítulo 2. CÓDIGO FUENTE

```
DtBTarifa.View.Columns[1].width = 100
DtBTarifa.View.Columns[2].width = 100
End

'' Configurarion Tab Tarifas y sus Objetos
Private Sub Configurar_tab_Tarifa()
    Dim Cabeceras As New String[]
    Dim Sql As String
    Cabeceras.Add("Tarifa")
    Cabeceras.Add("Articulo")
    Cabeceras.Add("Precio")
    If Utilidades.Tabla_Vacia("select * from articulo") = 0 Then
        TxtlerrorTarifas.Visible = True
        TxtlerrorTarifas.Text = "Debe ir a la Tab Articulos y crear los Articulos antes de poder Crear Tarifas."
    Else
        TxtlerrorTarifas.Visible = False
        TxtlerrorTarifas.Text = ""
    Endif
    Utilidades.Llenar_Combobox("select * from articulo", CmbTArticulos, "nombre")
    Utilidades.Llenar_Combobox("select * from tarifa", CmbTarifaEditar, "nombretarifa")
    DtsTarifa.Connection = FMain.Conexion_Base_Datos
    sql = "select T.tarifa,A.nombre,T.precio from articulo A, tarifaarticulo T where A.codigoarticulo=T.articulo order by T.tarifa"
    DtsTarifa.table = sql
    DtBTarifa.Labels = Cabeceras
    DtBTarifa.View.Columns[0].width = 200
    DtBTarifa.View.Columns[1].width = 100
    DtBTarifa.View.Columns[2].width = 100
End

''*****
''                               FIN CODIGO CONFIGURACION  TARIFAS
''*****
```

2.2.2.2 Frm_Clientes.class

```
Public Sub Form_Open()
    Me.Title = "GESTION DE CLIENTES"
    TxtRazonSocial.SetFocus
    Configurar_Objeto_Clientes()
    Configurar_DataSet_Clientes()
End

Public Sub Configurar_Objeto_Clientes()
    Dim Error_Tablas_Vacias As String
    Dim Error_V As Boolean
    Dim Valor_Subcuenta_Iva As Integer
    Error_V = False

    If Utilidades.Tabla_Vacia("select * from tarifa") = 0 Then
        Error_Tablas_Vacias = "Deber Cear Tarifas"
        Error_V = True
    End If
End Sub
```

```
Else
    Utilidades.Llenar_Combobox("select * from tarifa", CmbTarifa,
"nombretarifa")
Endif

If Utilidades.Tabla_Vacia("select * from vendedor") = 0 Then
    Error_V = True
    If Error_Tablas_Vacias = "" Then
        Error_Tablas_Vacias = "Deber Cear Vendedores"
    Else
        Error_Tablas_Vacias = Error_Tablas_Vacias & ", Vendedores"
    Endif
Else
    Utilidades.Llenar_Combobox("select * from vendedor", CmbVendedor,
"nombre")
Endif

If Utilidades.Tabla_Vacia("select * from formapago") = 0 Then

    Error_V = True
    If Error_Tablas_Vacias = "" Then
        Error_Tablas_Vacias = "Debe Cear Forma de Pago en Utilidades."
    Else
        Error_Tablas_Vacias = Error_Tablas_Vacias & " y Forma de pago en
Utilidades."
    Endif
Else
    Utilidades.Llenar_Combobox("select * from formapago", CmbFormaPago,
"nombreformapago")
Endif

If Error_V Then
    TxtlerrorClientes.Visible = True
    TxtlerrorClientes.text = Error_Tablas_Vacias
Endif

TxtlerrorClientes.text = Error_Tablas_Vacias
If Utilidades.Tabla_Vacia("select * from clientes") = 0 Then
    VlbSubcuentaIva.text = "43000001"
    TxtCodigo.text = "1"
Else
    Valor_Subcuenta_Iva = Val(utilidades.Valor_Maximo("select * from clientes
order by subcuentaiva", "subcuentaiva")) + 1
    VlbSubcuentaIva.text = Valor_Subcuenta_Iva
Endif

End

'' Procedimiento para configurar el Dataset.
Public Sub Configurar_DataSet_Clientes()
    Dim Cabeceras As New String[]
    Dim Sql As String

    Cabeceras.Add("Codigo")
    Cabeceras.Add("Razon Social")
    Cabeceras.Add("Direccion")
    Cabeceras.Add("Localidad")
    Cabeceras.Add("Provincia")
```

```
Cabeceras.Add("Codigo Postal")
Cabeceras.Add("Nif")
Cabeceras.Add("Subcuenta Iva")
Cabeceras.Add("Descuento PP")
Cabeceras.Add("Descuento ESP")
Cabeceras.Add("Recargo")
Cabeceras.Add("IVA")
Cabeceras.Add("Email")
Cabeceras.Add("Envio Email")
Cabeceras.Add("Forma Pago")
Cabeceras.Add("Tarifa")
Cabeceras.Add("Vendedor")
DtsClientes.Connection = FMain.Conexion_Base_Datos
sql = "select * from clientes"
DtsClientes.Table = sql
DtbClientes.Labels = Cabeceras
DtbClientes.view.Columns[0].Width = 100 'Codigo
DtbClientes.view.Columns[1].Width = 250
DtbClientes.view.Columns[2].Width = 250
DtbClientes.view.Columns[3].Width = 200
DtbClientes.view.Columns[4].Width = 200
DtbClientes.view.Columns[5].Width = 110 'Codigo Postal
DtbClientes.view.Columns[6].Width = 90 'Nif
DtbClientes.view.Columns[10].Width = 90 'Recargo
DtbClientes.view.Columns[11].Width = 90 'Iva
DtbClientes.view.Columns[12].Width = 100 'Email
DtbClientes.view.Columns[13].Width = 100 'Envio Email
DtbClientes.view.Columns[14].Width = 150 'Forma Pago
DtbClientes.view.Columns[15].Width = 150 'Tarifa
DtbClientes.view.Columns[16].Width = 150 'Vendedor
End

Public Sub BtnAgregarCliente_Click()

    Dim Sql As String
    Dim i As Integer
    Dim Tiene_Iva As String
    Dim Tiene_re As String
    Dim Email As String
    Dim Envio_Email As String
    If Comprobar_Datos_Clientes("AGREGAR") Then
        If ChkIva.value = True Then
            Tiene_Iva = "S"
        Else
            Tiene_Iva = "N"
        Endif
        If ChkRecargoEquivalente.Value = True Then
            Tiene_re = "S"
        Else
            Tiene_re = "N"
        Endif
        If ChkEnvioFacturasCorreo.Value = True Then
            Email = TxtEmail.Text
            Envio_Email = "S"
        Else
            Email = ""
            Envio_Email = "N"
        Endif
    Endif
```

```

        sql = "insert into clientes
(codigocliente,razonsocial,direccion,localidad,provincia,codigopostal,nif,subcuentaiva,descuentopp,descuentoesp,recargo,iva,email,enviofacturaemail,formapago,tarifa,vendedor)" &
        " values('" & UCase(TxtCodigo.Text) & "','" & UCase(TxtRazonSocial.Text) &
        "','" & UCase(TxtDireccion.Text) & "','" & UCase(TxtLocalidad.Text) & "','" &
        UCase(TxtProvincia.Text) &
        "','" & UCase(TxtCodigoPostal.Text) & "','" & UCase(TxtNif.Text) & "','" &
        Val(VlbSubcuentaIva.Text) & "','" & Val(VlbDescuentoPP.Text) & "','" &
        Val(VlbDescuentoESP.Text) & "','" &
        Tiene_re & "','" & Tiene_Iva & "','" & Email & "','" & Envio_Email & "','" &
        CmbFormaPago.Text & "','" & CmbTarifa.Text & "','" & CmbVendedor.Text & "')"
        utilidades.In_Act_Bor(sql)
        If LtbTelefono.Count > 0 Then
            For i = 0 To LtbTelefono.Count - 1
                sql = "insert into clientetelefono (telefono,cliente) values('" &
                LtbTelefono[i].Text & "','" & UCase(TxtCodigo.Text) & "')"
                Utilidades.In_Act_Bor(sql)
            Next
        Endif
        If LtbFax.Count > 0 Then
            For i = 0 To LtbFax.Count - 1
                sql = "insert into clientefax (fax,cliente) values('" & LtbFax[i].Text &
                "','" & UCase(TxtCodigo.Text) & "')"
                Utilidades.In_Act_Bor(sql)
            Next
        Endif
        Borrar_Campos_Clientes()
        Configurar_DataSet_Clientes()
    Endif

End

Public Sub BtnBorrarCliente_Click()
    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbClientes.View.Row
    If DtbClientes.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar el Cliente : " &
        DtbClientes.View[Numero_fila, 1].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from clientes where codigocliente='" &
            DtbClientes.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            sql = "delete from clientetelefono where cliente='" &
            DtbClientes.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            sql = "delete from clientefax where cliente='" &
            DtbClientes.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_DataSet_Clientes()
        Endif
    Endif

End

Public Sub BtnEditarCliente_Click()
    Dim Sql As String

```

```
Dim Numero_fila As Integer
Dim Tiene_Iva As String
Dim Tiene_re As String
Dim Email As String
Dim Envio_Email As String
Dim i As Integer
Dim Valor_Subcuenta_Iva As Integer
Valor_Subcuenta_Iva = VlbSubcuentaIva.Text
Numero_fila = DtbClientes.View.Row
If BtnEditarCliente.text = "EDITAR" Then
    TxtCodigo.Text = DtbClientes.View[Numero_fila, 0].Text
    TxtRazonSocial.Text = DtbClientes.View[Numero_fila, 1].Text
    TxtDireccion.Text = DtbClientes.View[Numero_fila, 2].Text
    TxtLocalidad.Text = DtbClientes.View[Numero_fila, 3].Text
    TxtProvincia.Text = DtbClientes.View[Numero_fila, 4].Text
    TxtCodigoPostal.Text = DtbClientes.View[Numero_fila, 5].Text
    TxtNif.Text = DtbClientes.View[Numero_fila, 6].Text
    VlbSubcuentaIva.Text = Val(DtbClientes.View[Numero_fila, 7].Text)
    VlbDescuentoPP.Text = Val(DtbClientes.View[Numero_fila, 8].Text)
    VlbDescuentoESP.Text = Val(DtbClientes.View[Numero_fila, 9].Text)
    If DtbClientes.View[Numero_fila, 10].Text = "S" Then
        ChkRecargoEquivalente.Value = True
    Else
        ChkRecargoEquivalente.Value = False
    Endif
    If DtbClientes.View[Numero_fila, 11].Text = "S" Then
        ChkIva.Value = True
    Else
        ChkIva.Value = False
    Endif
    If DtbClientes.View[Numero_fila, 13].Text = "S" Then
        ChkEnvioFacturasCorreo.Value = True
        TxtEmail.Text = DtbClientes.View[Numero_fila, 12].Text
    Else
        ChkRecargoEquivalente.Value = False
        TxtEmail.Text = ""
    Endif
    CmbFormaPago.Text = DtbClientes.View[Numero_fila, 14].Text
    CmbTarifa.Text = DtbClientes.View[Numero_fila, 15].Text
    CmbVendedor.Text = DtbClientes.View[Numero_fila, 16].Text
    sql = "select * from clientetelefono where cliente='" & TxtCodigo.Text & "'"
    utilidades.Llenar_Listbox(sql, LtbTelefono, "telefono")
    sql = "select * from clientefax where cliente='" & TxtCodigo.Text & "'"
    utilidades.Llenar_Listbox(sql, LtbFax, "fax")
    BtnAgregarCliente.Enabled = False
    BtnBorrarCliente.Enabled = False
    TxtCodigo.Enabled = False
    TxtRazonSocial.SetFocus
    BtnEditarCliente.text = "ACTUALIZAR"
Else
    If Comprobar_Datos_Clientes("MODIFICAR") Then
        If ChkIva.value = True Then
            Tiene_Iva = "S"
        Else
            Tiene_Iva = "N"
        Endif
        If ChkRecargoEquivalente.Value = True Then
            Tiene_re = "S"
        Else
```

```

        Tiene_re = "N"
    Endif
    If ChkEnvioFacturasCorreo.Value = True Then
        Email = TxtEmail.Text
        Envio_Email = "S"
    Else
        Email = ""
        Envio_Email = "N"
    Endif
    'Actualizamos los datos del Cliente.
    sql = "update clientes set razonsocial='" & UCase(TxtRazonSocial.text) &
    "','direccion='" & UCase(TxtDireccion.Text) & "','localidad='" &
    UCase(TxtLocalidad.Text) & "','provincia='" & UCase(TxtProvincia.Text) &
    "','codigopostal='" & UCase(TxtCodigoPostal.Text) &
    "','nif='" & UCase(TxtNif.Text) & "','subcuentaiva=" &
    Val(VlbSubcuentaIva.text) & ",descuentopp=" & Val(VlbDescuentoPP.Text) &
    ",descuentoesp=" & Val(VlbDescuentoESP.Text) &
    ",recargo='" & Tiene_re & "','iva='" & Tiene_Iva & "','email='" &
    TxtEmail.Text & "','enviofacturaemail='" & Envio_Email & "','formapago='" &
    CmbFormaPago.Text & "','tarifa='" & CmbTarifa.text &
    "','vendedor='" & CmbVendedor.Text & "' where codigocliente='" &
    UCase(TxtCodigo.Text) & "'"
    utilidades.In_Act_Bor(sql)
    ' Actualizamos Telefonos, borramos los telefonos, si el listbox de los
    telefonos esta vacio no hacemos nada.
    ' Si tenemos datos en el listbox de telefono los añadimos.
    sql = "delete from clientetelefono where cliente='" & TxtCodigo.Text & "'"
    utilidades.In_Act_Bor(sql)
    If LtbTelefono.Count > 0 Then
        For i = 0 To LtbTelefono.Count - 1
            sql = "insert into clientetelefono (telefono,cliente) values('" &
            LtbTelefono[i].Text & "',''" & UCase(TxtCodigo.Text) & "'"")"
            Utilidades.In_Act_Bor(sql)
        Next
    Endif

    ' Actualizamos Fax, borramos los telefonos, si el listbox de los telefonos
    esta vacio no hacemos nada.
    ' Si tenemos datos en el listbox de telefono los añadimos.
    sql = "delete from clientefax where cliente='" & TxtCodigo.Text & "'"
    Utilidades.In_Act_Bor(sql)
    If LtbFax.Count > 0 Then
        For i = 0 To LtbFax.Count - 1
            sql = "insert into clientefax (fax,cliente) values('" & LtbFax[i].Text &
            "',''" & UCase(TxtCodigo.Text) & "'"")"
            Utilidades.In_Act_Bor(sql)
        Next
    Endif
    BtnAgregarCliente.Enabled = True
    BtnBorrarCliente.Enabled = True
    TxtCodigo.Enabled = True
    BtnEditarCliente.text = "EDITAR"
    Borrar_Campos_Clientes()
    Valor_Subcuenta_Iva = Val(utilidades.Valor_Maximo("select * from clientes
    order by subcuentaiva", "subcuentaiva")) + 1
    VlbSubcuentaIva.Text = Valor_Subcuenta_Iva
Endif
Endif
Configurar_DataSet_Clientes()

```



```
End

'' Funcion para comprobar los Campos de los Vendedoreses <br>
'' Parametros:<br>
''     Accion - - > La accion desde la que comprobamos.<br>
''             AGREGAR --> Para agregar datos.<br>
''             MODIFICAR --> Cuando estamos modificando un dato.<br>
'' Devuelve:<br>
''     TRUE --> Si los valores son correctos.<br>
''     FALSE --> valores erroneos.
Private Function Comprobar_Datos_Clientes(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    If TxtCodigo.Text = "" Then
        Balloon.Error("Debe introducir unCodigo de Cliente que va a crear",
TxtCodigo)
        Validar = False
        TxtCodigo.SetFocus
    Else If TxtRazonSocial.Text = ""
        Balloon.Error("Debe introducir una Razon Social de Cliente que va a crear",
TxtRazonSocial)
        Validar = False
        TxtRazonSocial.SetFocus
    Else If ChkEnvioFacturasCorreo.Value = True And TxtEmail.Text = ""
        Balloon.Error("Debe introducir un Email Valido", TxtEmail)
        Validar = False
        TxtEmail.SetFocus
    Else If TxtCodigoPostal.Text <> "" And (Not IsNumber(TxtCodigoPostal.Text))
        Balloon.Error("El codigo postal debe ser numerico ", TxtCodigoPostal)
        Validar = False
        TxtCodigoPostal.SetFocus
    Else If TxtCodigoPostal.Length < 5 And TxtCodigoPostal.Length > 0
        Balloon.Error("El codigo postal debe tener 5 digitos ", TxtCodigoPostal)
        Validar = False
        TxtCodigoPostal.SetFocus
    Else If CmbTarifa.text = ""
        Balloon.Error("Debe elegir una Tarifa para el cliente.", CmbTarifa)
        Validar = False
    Else If CmbFormaPago.text = ""
        Balloon.Error("Debe elegir una forma de pago para el cliente.",
CmbFormaPago)
        Validar = False
    Endif
    If Accion = "AGREGAR" Then
        Sql = "select * from clientes where codigocliente='" & UCase(TxtCodigo.Text)
& ""
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El codigo Cliente ya existe", TxtCodigo)
            Validar = False
            TxtCodigo.SetFocus
        Endif
    Endif
    Return Validar
End

Public Sub BtnAgregarTelefono_Click()

    If TxtTelefono.text = "" Then
```

```
Balloon.Error("Introduzca telefono antes de añadir", TxtTelefono)
TxtTelefono.SetFocus
Else If Not IsNumber(TxtTelefono.Text)
    Balloon.Error("El Telefono es un dato Numerico", TxtTelefono)
    TxtTelefono.SetFocus
Else If TxtTelefono.Length < 9
    Balloon.Error("Al Telefono le faltan digitos ", TxtTelefono)
    TxtTelefono.SetFocus
Else
    LtbTelefono.Add(TxtTelefono.Text)
    TxtTelefono.Text = ""
Endif

End

Public Sub BtnBorrarTelefono_Click()

    Dim Contar_Seleccionados As Integer
    Dim Numero_Telefonos As Integer
    Dim i As Integer
    Numero_Telefonos = LtbTelefono.Count
    Contar_Seleccionados = 0
    For i = 0 To Numero_Telefonos - 1
        If LtbTelefono[i].Selected Then
            Contar_Seleccionados = Contar_Seleccionados + 1
        Endif
    Next
    If Numero_Telefonos > 0 And Contar_Seleccionados > 0 Then
        If LtbTelefono.Current.Selected Then
            LtbTelefono.Remove(LtbTelefono.Index)
        Endif
    Endif

End

Public Sub BtnAgregarFax_Click()

    If TxtFax.text = "" Then
        Balloon.Error("Introduzca Fax antes de añadir", TxtFax)
        TxtFax.SetFocus
    Else If Not IsNumber(TxtFax.Text)
        Balloon.Error("El Fax es un dato Numerico", TxtFax)
        TxtFax.SetFocus
    Else If TxtFax.Length < 9
        Balloon.Error("Al Fax le faltan digitos ", TxtFax)
        TxtFax.SetFocus
    Else
        LtbFax.Add(TxtFax.Text)
        TxtFax.Text = ""
    Endif

End

Public Sub BtnBorrarFax_Click()

    Dim Contar_Seleccionados As Integer
    Dim Numero_Fax As Integer
    Dim i As Integer
    Numero_Fax = LtbFax.Count
```

Capítulo 2. CÓDIGO FUENTE

```
    Contar_Seleccionados = 0
    For i = 0 To Numero_Fax - 1
        If LtbFax[i].Selected Then
            Contar_Seleccionados = Contar_Seleccionados + 1
        Endif
    Next
    If Numero_Fax > 0 And Contar_Seleccionados > 0 Then
        If LtbFax.Current.Selected Then
            LtbFax.Remove(LtbFax.Index)
        Endif
    Endif
End

Private Sub Borrar_Campos_Clientes()

    LtbFax.Clear
    LtbTelefono.Clear
    ChkIva.Value = True
    ChkEnvioFacturasCorreo.value = False
    ChkRecargoEquivalente.value = False
    TxtEmail.text = ""
    TxtEmail.Enabled = False
    TxtCodigo.text = ""
    TxtCodigoPostal.Text = ""
    TxtDireccion.Text = ""
    TxtLocalidad.Text = ""
    TxtNif.Text = ""
    TxtProvincia.Text = ""
    TxtRazonSocial.Text = ""
    TxtFax.Text = ""
    TxtTelefono.Text = ""
    VlbDescuentoESP.Text = ""
    VlbDescuentoPP.Text = ""
    CmbFormaPago.Text = ""
    CmbTarifa.Text = ""
    CmbVendedor.Text = ""
    VlbSubcuentaIva.text = Val(VlbSubcuentaIva.text) + 1
End

Public Sub BtnSalirClientes_Click()

    Me.Close

End

Public Sub ChkEnvioFacturasCorreo_Click()

    If ChkEnvioFacturasCorreo.Value = True Then
        TxtEmail.Enabled = True
        TxtEmail.SetFocus
    Else
        TxtEmail.Enabled = False
    Endif
End

Public Sub TxtRazonSocial_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
```

```
        TxtDireccion.SetFocus
    Endif

End

Public Sub TxtDireccion_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtLocalidad.SetFocus
    Endif

End

Public Sub TxtLocalidad_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtProvincia.SetFocus
    Endif

End

Public Sub TxtProvincia_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtNif.SetFocus
    Endif

End

Public Sub TxtNif_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtCodigoPostal.SetFocus
    Endif

End

Public Sub TxtCodigoPostal_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtCodigo.SetFocus
    Endif

End

Public Sub VlbSubcuentaIva_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlbDescuentoPP.SetFocus
    Endif

End

Public Sub TxtCodigo_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlbSubcuentaIva.SetFocus
    Endif
```

```
End

Public Sub VlbDescuentoPP_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlbDescuentoESP.SetFocus
    Endif

End
```

2.2.2.3 Frm_Proveedores.class

```
Public Sub Form_Open()

    Me.Title = "GESTION DE PROVEEDORES"
    TxtCodigo.SetFocus
    Utilidades.Llenar_Listbox("select * from materiaprima", LtBMP, "nombre")
    Configurar_DataSet_Proveedores()
    Configurar_Datagrid_Comerciales()
End

Public Sub BtnSalirProveedores_Click()

    Me.Close

End

'' Botones de Añadir y Borrar Telefonos
Public Sub BtnAgregarTelefono_Click()

    If TxtTelefono.text = "" Then
        Balloon.Error("Introduzca telefono antes de añadir", TxtTelefono)
        TxtTelefono.SetFocus
    Else If Not IsNumber(TxtTelefono.Text)
        Balloon.Error("El Telefono es un dato Numerico", TxtTelefono)
        TxtTelefono.SetFocus
    Else If TxtTelefono.Length < 9
        Balloon.Error("Al Telefono le faltan digitos ", TxtTelefono)
        TxtTelefono.SetFocus
    Else
        LtbTelefono.Add(TxtTelefono.Text)
        TxtTelefono.Text = ""
    Endif

End

Public Sub BtnBorrarTelefono_Click()

    Dim Contar_Seleccionados As Integer
    Dim Numero_Telefonos As Integer
    Dim i As Integer
    Numero_Telefonos = LtbTelefono.Count
    Contar_Seleccionados = 0
    For i = 0 To Numero_Telefonos - 1
        If LtbTelefono[i].Selected Then
```

```
        Contar_Seleccionados = Contar_Seleccionados + 1
    Endif
Next
If Numero_Telefonos > 0 And Contar_Seleccionados > 0 Then
    If LtbTelefono.Current.Selected Then
        LtbTelefono.Remove(LtbTelefono.Index)
    Endif
Endif

End

''Botones de Añadir y Borrar Fax

Public Sub BtnAgregarFax_Click()

    If TxtFax.text = "" Then
        Balloon.Error("Introduzca Fax antes de añadir", TxtFax)
        TxtFax.SetFocus
    Else If Not IsNumber(TxtFax.Text)
        Balloon.Error("El Fax es un dato Numerico", TxtFax)
        TxtFax.SetFocus
    Else If TxtFax.Length < 9
        Balloon.Error("Al Fax le faltan digitos ", TxtFax)
        TxtFax.SetFocus
    Else
        LtbFax.Add(TxtFax.Text)
        TxtFax.Text = ""
    Endif

End

Public Sub BtnBorrarFax_Click()

    Dim Contar_Seleccionados As Integer
    Dim Numero_Fax As Integer
    Dim i As Integer
    Numero_Fax = LtbFax.Count
    Contar_Seleccionados = 0
    For i = 0 To Numero_Fax - 1
        If LtbFax[i].Selected Then
            Contar_Seleccionados = Contar_Seleccionados + 1
        Endif
    Next
    If Numero_Fax > 0 And Contar_Seleccionados > 0 Then
        If LtbFax.Current.Selected Then
            LtbFax.Remove(LtbFax.Index)
        Endif
    Endif

End

'' Configuración de los botones de pasar Las materias primas
Public Sub BtnAgregarMP_Click()

    Dim Contador As Integer
    Dim i As Integer
    Dim Lista_Seleccionados As New String[]
    Dim Lista_NoSeleccionados As New String[]
    Contador = LtBMP.Count
    For i = 0 To Contador - 1
        If LtBMP[i].Selected Then
```

```
        Lista_Seleccionados.Add(LtBMP[i].Text)
    Else
        Lista_NoSeleccionados.Add(LtBMP[i].Text)

    Endif
Next
' Añadimos a LtBKMPComposicion los seleccionados
contador = Lista_Seleccionados.Count
For i = 0 To contador - 1
    LtBMPProveedor.Add(Lista_Seleccionados[i])
Next
' Borramos ltBKMP y agregamos los que no hemos seleccionado.
contador = Lista_NoSeleccionados.Count
LtBMP.Clear
For i = 0 To contador - 1
    LtBMP.Add(Lista_NoSeleccionados[i])
Next
LtBMP.UnselectAll

End

Public Sub BtnBorrarMP_Click()

    Dim Contador As Integer
    Dim i As Integer
    Dim Lista_Seleccionados As New String[]
    Dim Lista_NoSeleccionados As New String[]
    Contador = LtBMPProveedor.Count
    For i = 0 To Contador - 1
        If LtBMPProveedor[i].Selected Then
            Lista_Seleccionados.Add(LtBMPProveedor[i].Text)
        Else
            Lista_NoSeleccionados.Add(LtBMPProveedor[i].Text)

        Endif
    Next
    ' Añadimos a LtBKMP los seleccionados
    contador = Lista_Seleccionados.Count
    For i = 0 To contador - 1
        LtBMP.Add(Lista_Seleccionados[i])
    Next
    ' Borramos ltBKMPComposicion y agregamos los que no hemos seleccionado.
    contador = Lista_NoSeleccionados.Count
    LtBMPProveedor.Clear
    For i = 0 To contador - 1
        LtBMPProveedor.Add(Lista_NoSeleccionados[i])
    Next
    ' Deseleccionamos todos ya que se quedan seleccionados
    LtBMPProveedor.UnselectAll

End

'' Botones de Agregar, Borrar y Editar proveedores
Public Sub BtnAgregarProveedor_Click()

    Dim Sql As String
    Dim Contador As Integer
    Dim i As Integer
    Dim C_MateriaPrima As String
    Dim Numero_fila As Integer
```

```

If Comprobar_Campos_Proveedor("AGREGAR") Then
    'Insertamos en la tabla Vendedor
    sql = "insert into proveedor
(codigoproveedor,razonsocial,direccion,localidad,provincia,codigopostal,nif,web,
email)" &
    " values('" & UCase(TxtCodigo.Text) & "','" & UCase(TxtRazonSocial.Text) &
    "','" & UCase(TxtDireccion.Text) & "','" & UCase(TxtLocalidad.Text) & "','" &
    UCase(TxtProvincia.Text) &
    "','" & UCase(TxtCodigoPostal.Text) & "','" & UCase(TxtNif.Text) & "','" &
    TxtWeb.Text & "','" & TxtEmail.Text & "')"
    utilidades.In_Act_Bor(sql)
    'Insertamos datos en la tabla Proveedor Telefono
    If LtbTelefono.Count > 0 Then
        For i = 0 To LtbTelefono.Count - 1
            sql = "insert into proveedortelefono (telefono,proveedor) values('" &
            LtbTelefono[i].Text & "','" & UCase(TxtCodigo.Text) & "')"
            Utilidades.In_Act_Bor(sql)
        Next
    Endif
    'Insertamos datos en la tabla Proveedor Fax
    If LtbFax.Count > 0 Then
        For i = 0 To LtbFax.Count - 1
            sql = "insert into proveedorfax (fax,proveedor) values('" &
            LtbFax[i].Text & "','" & UCase(TxtCodigo.Text) & "')"
            Utilidades.In_Act_Bor(sql)
        Next
    Endif
    ' Insertamos los datos en al tabla Proveedor MateriaPrima
    Contador = LtBMPProveedor.Count
    For i = 0 To Contador - 1
        sql = "select * from materiaprima where nombre='" & LtBMPProveedor[i].Text
        & "'"
        C_Materiaprima = Utilidades.Obtener_Campo_Tabla(sql, "codigomateriaprima")
        sql = "insert into proveedormateriaprima (proveedor,materiaprima)
        values('" & UCase(TxtCodigo.Text) & "','" & UCase(C_Materiaprima) & "')"
        Utilidades.In_Act_Bor(Sql)
    Next
    ' Añadimos los comerciales a la tabla, si el data grid no esta vacio
    If GridComerciales.Rows.Count > 0 Then
        Numero_fila = GridComerciales.Rows.Count
        For i = 0 To Numero_fila - 1
            sql = "insert into proveedorcomercial
(codigoproveedor,nombre,telefono,email) values ('" & UCase(TxtCodigo.Text) &
            "','" & UCase(GridComerciales[i, 0].Text) & "','" & UCase(GridComerciales[i,
            1].Text) & "','" & UCase(GridComerciales[i, 2].Text) & "')"
            utilidades.In_Act_Bor(sql)
        Next
    Endif
    Borrar_Campos_Proveedor()
Endif
End

Public Sub BtnBorrarProveedor_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbProveedor.View.Row
    If DtbProveedor.View.Rows.Count > 0 Then

```



```

    Respuesta = Message.Delete("Desea Borrar el Proveedor : " &
DtbProveedor.View[Numero_fila, 1].Text, "Si", "No")
    If respuesta = 1 Then
        ' Tabla Proveedor
        sql = "delete from proveedor where codigoproveedor='" &
DtbProveedor.View[Numero_fila, 0].Text & "'"
        Utilidades.In_Act_Bor(Sql)
        ' Tabla Proveedor Telefono
        sql = "delete from provedortelefono where proveedor='" &
DtbProveedor.View[Numero_fila, 0].Text & "'"
        Utilidades.In_Act_Bor(Sql)
        ' Tabla Proveedor Fax
        sql = "delete from provedorfax where proveedor='" &
DtbProveedor.View[Numero_fila, 0].Text & "'"
        Utilidades.In_Act_Bor(Sql)
        ' Tabla Proveedor Comercial
        sql = "delete from provedorcomercial where codigoproveedor='" &
DtbProveedor.View[Numero_fila, 0].Text & "'"
        Utilidades.In_Act_Bor(Sql)
        ' Tabla Proveedor Materia Prima
        sql = "delete from provedormateriaprima where proveedor='" &
DtbProveedor.View[Numero_fila, 0].Text & "'"
        Utilidades.In_Act_Bor(Sql)
        Configurar_DataSet_Proveedores()
    Endif
Endif

End

Public Sub BtnEditarProveedor_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim i As Integer
    Dim C_MateriaPrima As String

    Numero_fila = DtbProveedor.View.Row
    If BtnEditarProveedor.text = "EDITAR" Then
        TxtCodigo.Text = DtbProveedor.View[Numero_fila, 0].Text
        TxtRazonSocial.text = DtbProveedor.View[Numero_fila, 1].Text
        TxtNif.Text = DtbProveedor.View[Numero_fila, 2].Text
        TxtDireccion.Text = DtbProveedor.View[Numero_fila, 3].Text
        TxtLocalidad.Text = DtbProveedor.View[Numero_fila, 4].Text
        TxtProvincia.Text = DtbProveedor.View[Numero_fila, 5].Text
        TxtCodigoPostal.Text = DtbProveedor.View[Numero_fila, 6].Text
        TxtWeb.Text = DtbProveedor.View[Numero_fila, 7].Text
        TxtEmail.Text = DtbProveedor.View[Numero_fila, 8].Text
        ' Llenamos los Listbox de Telefono y Fax
        sql = "select * from provedortelefono where proveedor='" & TxtCodigo.Text &
""""
        utilidades.Llenar_Listbox(sql, LtbTelefono, "telefono")
        sql = "select * from provedorfax where proveedor='" & TxtCodigo.Text & """"
        utilidades.Llenar_Listbox(sql, LtbFax, "fax")
        ' Llenamos el Listbox con las materias primas.
        sql = "select * from provedormateriaprima where proveedor='" &
TxtCodigo.text & """"
        ' Carga los listbox de las materias primas
        Llenar_Composicion_Proveedor_MateriaPrima(sql)
    End If
End Sub

```

```

Eliminar_Materia_Prima_Listboxmp()
' Cargamos los datos de los comerciales en el Grid
Llenar_DataGrid_Comerciales()

BtnAgregarProveedor.Enabled = False
BtnBorrarProveedor.Enabled = False
TxtCodigo.Enabled = False
BtnEditarProveedor.text = "ACTUALIZAR"

Else
    sql = "update proveedor set razonsocial='" & UCase(TxtRazonSocial.Text) &
    "','nif='" & UCase(TxtNif.Text) & "','direccion='" & UCase(TxtDireccion.Text) &
    "','localidad='" &
    UCase(TxtLocalidad.Text) & "','provincia='" & UCase(TxtProvincia.Text) &
    "','codigopostal='" & UCase(TxtCodigoPostal.Text) & "','web='" &
    UCase(TxtWeb.Text) &
    "','email='" & UCase(TxtEmail.Text) & "' where codigoproveedor='" &
    UCase(TxtCodigo.Text) & "'"
    utilidades.In_Act_Bor(sql)
    ' Actualizamos Telefonos, borramos los telefonos, si el listbox de los
    telefonos esta vacio no hacemos nada.
    ' Si tenemos datos en el listbox de telefono los añadimos.
    sql = "delete from provedortelefono where proveedor='" & TxtCodigo.Text &
    "'"
    utilidades.In_Act_Bor(sql)
    If LtbTelefono.Count > 0 Then
        For i = 0 To LtbTelefono.Count - 1
            sql = "insert into provedortelefono (telefono,proveedor) values('" &
            LtbTelefono[i].Text & "',''" & UCase(TxtCodigo.Text) & "'"")"
            Utilidades.In_Act_Bor(sql)
        Next
    Endif

    ' Actualizamos Fax, borramos los telefonos, si el listbox de los telefonos
    esta vacio no hacemos nada.
    ' Si tenemos datos en el listbox de telefono los añadimos.
    sql = "delete from proveedorfax where proveedor='" & TxtCodigo.Text & "'"
    Utilidades.In_Act_Bor(sql)
    If LtbFax.Count > 0 Then
        For i = 0 To LtbFax.Count - 1
            sql = "insert into proveedorfax (fax,proveedor) values('" &
            LtbFax[i].Text & "',''" & UCase(TxtCodigo.Text) & "'"")"
            Utilidades.In_Act_Bor(sql)
        Next
    Endif

    ' Actualizamos La materia prima que distribuye, borramos los datos de la
    tabla Proveedor Materia Prima y
    ' añadimos los elementos del list box
    sql = "delete from provedormateriaprima where proveedor='" & TxtCodigo.Text
    & "'"
    Utilidades.In_Act_Bor(sql)
    For i = 0 To LtBMPProveedor.Count - 1
        sql = "select * from materiaprima where nombre='" & LtBMPProveedor[i].Text
        & "'"
        C_Materiaprima = Utilidades.Obtener_Campo_Tabla(sql, "codigomateriaprima")
        sql = "insert into provedormateriaprima (proveedor,materiaprima)
        values('" & UCase(TxtCodigo.Text) & "',''" & UCase(C_Materiaprima) & "'"")"
        Utilidades.In_Act_Bor(Sql)
    Next

```

```
' Añadimos los comerciales a la tabla, si el data grid no esta vacio
sql = "delete from proveedorcomercial where codigoproveedor=" &
TxtCodigo.Text & ""
Utilidades.In_Act_Bor(sql)
If GridComerciales.Rows.Count > 0 Then
    Numero_fila = GridComerciales.Rows.Count
    For i = 0 To Numero_fila - 1
        sql = "insert into proveedorcomercial
(codigoproveedor,nombre,telefono,email) values ('" & UCase(TxtCodigo.Text) &
"', '" & UCase(GridComerciales[i, 0].Text) & "', '" & UCase(GridComerciales[i,
1].Text) & "', '" & UCase(GridComerciales[i, 2].Text) & "'"
        utilidades.In_Act_Bor(sql)
    Next
Endif
BtnAgregarProveedor.Enabled = True
BtnBorrarProveedor.Enabled = True
TxtCodigo.Enabled = True
BtnEditarProveedor.text = "EDITAR"
Borrar_Campos_Proveedor()
Endif
Configurar_DataSet_Proveedores()
End
'' Configuracion de los botones de Agregar, Borra y Modificar Comerciales
Public Sub BtnAgregarComercial_Click()
    Dim Numero_fila As Integer
    If Comprobar_Campos_Comercial("AGREGAR") Then

        Numero_fila = GridComerciales.Rows.Count
        GridComerciales.Rows.Insert(Numero_fila, 1)
        GridComerciales[Numero_fila, 0].Text = UCase(TxtNombreComercial.Text)
        GridComerciales[Numero_fila, 1].Text = UCase(TxtTelefonoComercial.Text)
        GridComerciales[Numero_fila, 2].Text = UCase(TxtEmailComercial.Text)
        TxtNombreComercial.Text = ""
        TxtTelefonoComercial.text = ""
        TxtEmailComercial.text = ""

    Endif

End

Public Sub BtnBorrarComercial_Click()

    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    If GridComerciales.Rows.Selection.Count > 0 Then
        Numero_fila = GridComerciales.Rows.Selection[0]
        Respuesta = Message.Delete("Desea Borrar el Comercial : " &
GridComerciales[Numero_fila, 0].Text, "Si", "No")
        If respuesta = 1 Then
            GridComerciales.Rows.Remove(Numero_fila, 1)
        End If
    End If
End

Public Sub BtnModificarComercial_Click()

    Dim Numero_fila As Integer

    If GridComerciales.Rows.Selection.Count > 0 Then
```

```
Numero_fila = GridComerciales.Rows.Selection[0]

If LblModificarComercial.Text = "EDITAR" Then
    LblModificarComercial.Text = "ACTUALIZAR"
    TxtNombreComercial.Text = GridComerciales[Numero_fila, 0].Text
    TxtTelefonoComercial.Text = GridComerciales[Numero_fila, 1].Text
    TxtEmailComercial.Text = GridComerciales[Numero_fila, 2].Text
    TxtNombreComercial.Enabled = False
    BtnAgregarComerdial.Enabled = False
    BtnBorrarComercial.Enabled = False
    TxtNombreComercial.SetFocus
Else
    If Comprobar_Campos_Comercial("MODIFICAR") Then
        GridComerciales.Rows.Remove(Numero_fila, 1)
        Numero_fila = GridComerciales.Rows.Count
        GridComerciales.Rows.Insert(Numero_fila, 1)
        GridComerciales[Numero_fila, 0].Text = UCase(TxtNombreComercial.Text)
        GridComerciales[Numero_fila, 1].Text = UCase(TxtTelefonoComercial.Text)
        GridComerciales[Numero_fila, 2].Text = UCase(TxtEmailComercial.Text)
        LblModificarComercial.Text = "EDITAR"
        TxtNombreComercial.Text = ""
        TxtTelefonoComercial.Text = ""
        TxtEmailComercial.Text = ""
        TxtNombreComercial.Enabled = True
        BtnAgregarComerdial.Enabled = True
        BtnBorrarComercial.Enabled = True
    Endif
Endif
End If
End

''***** Funciones Privadas*****

'' Configuracion del Dataset de Proveedores
Private Sub Configurar_DataSet_Proveedores()

    Dim Cabeceras As New String[]
    Dim Sql As String

    Cabeceras.Add("Codigo")
    Cabeceras.Add("Razon Social")
    Cabeceras.Add("Nif")
    Cabeceras.Add("Direccion")
    Cabeceras.Add("Localidad")
    Cabeceras.Add("Provincia")
    Cabeceras.Add("Codigo Postal")
    Cabeceras.Add("Web")
    Cabeceras.Add("Email")
    DtsProveedor.Connection = FMain.Conexion_Base_Datos
    sql = "select * from proveedor"
    DtsProveedor.Table = sql
    DtbProveedor.Labels = Cabeceras
    DtbProveedor.view.Columns[0].Width = 100
    DtbProveedor.view.Columns[1].Width = 250
    DtbProveedor.view.Columns[2].Width = 100
    DtbProveedor.view.Columns[3].Width = 250
    DtbProveedor.view.Columns[4].Width = 200
```

Capítulo 2. CÓDIGO FUENTE

```
DtbProveedor.view.Columns[5].Width = 200
DtbProveedor.view.Columns[6].Width = 150
DtbProveedor.view.Columns[7].Width = 250
DtbProveedor.view.Columns[8].Width = 100
End

'' Configuracion del Datas de Comerciales<br>
'' Parametros:<br>
''     Sql --> Sentencia para cargar datos en el Dataset
Private Sub Configurar_Datagrid_Comerciales()

    GridComerciales.Columns.Count = 3
    GridComerciales.Header = 3
    GridComerciales.Columns[0].Title = "Nombre"
    GridComerciales.Columns[1].Title = "Telefono"
    GridComerciales.Columns[2].Title = "Email"
    GridComerciales.Columns[0].Width = 200
    GridComerciales.Columns[1].Width = 100
    GridComerciales.Columns[2].Width = 150

End

'' Funcion para comprobar los Campos Proveedores <br>
'' Parametros:<br>
''     Accion - - > La accion desde la que comprobamos.<br>
''                     AGREGAR --> Para agregar datos.<br>
''                     MODIFICAR --> Cuando estamos modificando un dato.<br>
'' Devuelve:<br>
''     TRUE --> Si los valores son correctos.<br>
''     FALSE --> valores erroneos.
Private Function Comprobar_Campos_Proveedor(Accion As String) As Boolean
    Dim Sql As String
    Dim Validar As Boolean
    Validar = True

    If TxtCodigo.Text = "" Then
        Balloon.Error("Debe introducir unCodigo de Proveedor que va a crear",
TxtCodigo)
        Validar = False
        TxtCodigo.SetFocus
    Else If TxtRazonSocial.text = ""
        Balloon.Error("Debe introducir el Nombre del Proveedor", TxtRazonSocial)
        Validar = False
        TxtRazonSocial.SetFocus
    Else If LtBMPPProveedor.Count = 0
        Balloon.Error("Debe añadir las materias primas que distribuye el proveedor",
LtBMPPProveedor)
        Validar = False
    Endif
    If Accion = "AGREGAR" Then
        Sql = "select * from proveedor where codigoproveedor='" &
UCase(TxtCodigo.Text) & "'"
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El codigo Proveedor ya existe", TxtCodigo)
            Validar = False
            TxtCodigo.SetFocus
        Endif
    Endif
End Function
```

```
Endif
Return Validar
End

'' Funcion para comprobar los Campos Comercial <br>
'' Parametros:<br>
''     Accion --> Accion a realizar<br>
''     AGREGAR --> Para agregar Comerciales<br>
''     MODIFICAR --> Para modificar Comerciales<br>
'' Devuelve:<br>
''     TRUE --> Si los valores son correctos.<br>
''     FALSE --> valores erroneos.
Private Function Comprobar_Campos_Comercial(Accion As String) As Boolean
    Dim Sql As String
    Dim Validar As Boolean
    Validar = True
    If TxtNombreComercial.Text = ""
        Balloon.Error("Introduzca nombre del comercial", TxtNombreComercial)
        Validar = False
        TxtNombreComercial.SetFocus
    Else If TxtTelefonoComercial.Text = ""
        Balloon.Error("Introduzca telefono del comercial", TxtTelefonoComercial)
        Validar = False
        TxtTelefonoComercial.SetFocus
    Else If Not IsNumber(TxtTelefonoComercial.Text)
        Balloon.Error("El Telefono es un dato Numerico", TxtTelefonoComercial)
        Validar = False
        TxtTelefonoComercial.SetFocus
    Else If TxtTelefonoComercial.Length < 9 And TxtTelefonoComercial.Length > 1
        Balloon.Error("Al Telefono le faltan digitos ", TxtTelefonoComercial)
        Validar = False
        TxtTelefonoComercial.SetFocus
    Endif
    If Accion = "AGREGAR" Then
        Sql = "select * from proveedorcomercial where nombre='" &
        UCase(TxtNombreComercial.Text) & "'"
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El Nombre del Comercial ya existe", TxtNombreComercial)
            Validar = False
            TxtNombreComercial.SetFocus
        Endif
    Endif
    Return Validar
End

'' Limpia los campos despues de uan Insercion o Modificacion de Proveedor
Private Sub Borrar_Campos_Proveedor()
    Dim Numero_fila As Integer
    TxtCodigo.Text = ""
    TxtCodigoPostal.Text = ""
    TxtDireccion.Text = ""
    TxtTelefonoComercial.Text = ""
    TxtEmail.Text = ""
    TxtEmailComercial.Text = ""
    TxtFax.Text = ""
    TxtLocalidad.Text = ""
    TxtNif.Text = ""
    TxtNombreComercial.Text = ""
```

```
TxtProvincia.Text = ""
TxtRazonSocial.Text = ""
TxtTelefono.Text = ""
TxtWeb.Text = ""
LtBMPProveedor.Clear
LtbFax.Clear
LtBMP.Clear
LtbTelefono.Clear
Utilidades.Llenar_Listbox("select * from materiaprima", LtBMP, "nombre")
Numero_fila = GridComerciales.Rows.Count
If GridComerciales.Rows.Count > 0 Then
    GridComerciales.Rows.Remove(0, Numero_fila)
Endif

Configurar_DataSet_Proveedores()

End

'' Procedimiento que elimina los elementos que estan en el Listbox de la
composicion de los elementos del listbos de materia prima.<br>

Private Sub Eliminar_Materia_Prima_Listboxmp()
    Dim i As Integer
    Dim j As Integer
    Dim Contador_Listobox_MP As Integer
    Dim Contador_Listobox_Composicion As Integer
    Dim Salida As Integer
    salida = 0
    j = 0
    Contador_Listobox_Composicion = LtBMPProveedor.Count
    Contador_Listobox_MP = LtBMP.Count
    For i = 0 To Contador_Listobox_Composicion - 1
        While salida = 0
            If LtBMP[j].Text = LtBMPProveedor[i].Text Then
                salida = 1
                LtBMP.Remove(j)
            Endif
            j = j + 1
        Wend
        salida = 0
        j = 0
    Next
End

'' Procedimiento que nos llena el listbox de la composicion del articulo con el
articulo seleccionado.<br>
'' Recibe como parametro la sentencia SQL
Private Sub Llenar_Composicion_Proveedor_MateriaPrima(Sql As String)
    Dim Result_Trabajo As Result
    Dim Materia_Prima As New String[]
    Dim Materia_Prima_Nombre As String
    Dim Sql_Trabajo As String
    Dim i As Integer
    Dim Contador As Integer
    LtBMPProveedor.Clear
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
```

Capítulo 2. CÓDIGO FUENTE

```
Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
Result_Trabajo.MoveFirst
' Obtenemos los identificadores de la materia prima
While Result_Trabajo.Available
    Materia_Prima.Add(Result_Trabajo["materiaprima"])
    Result_Trabajo.MoveNext
Wend
FMain.Conexion_Base_Datos.Close
contador = Materia_Prima.Count
LtBMPPProveedor.Clear
' Llenamos con el nombre el Listbox.
For i = 0 To contador - 1
    Sql_Trabajo = "select * from materiaprima where codigomateriaprima='" &
Materia_Prima[i] & "'"
    Materia_Prima_Nombre = Utilidades.Obtener_Campo_Tabla(Sql_Trabajo,
"nombre")
    LtBMPPProveedor.Add(Materia_Prima_Nombre)
Next
LtBMPPProveedor.UnselectAll

End

'' Procedimiento para llenar el Datagrid con los datos de los Comerciales del
Proveedor
Private Sub Llenar_DataGrid_Comerciales()
    Dim Result_Trabajo As Result
    Dim Sql As String
    Dim Numero_fila As Integer
    sql = "Select nombre,telefono,email from proveedorcomercial where
codigoproveedor='" & UCase(TxtCodigo.Text) & "'"
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst

    While Result_Trabajo.Available
        Numero_fila = GridComerciales.Rows.Count
        GridComerciales.Rows.Insert(Numero_fila, 1)
        GridComerciales[Numero_fila, 0].Text = Result_Trabajo["nombre"]
        GridComerciales[Numero_fila, 1].Text = Result_Trabajo["telefono"]
        GridComerciales[Numero_fila, 2].Text = Result_Trabajo["email"]
        Result_Trabajo.MoveNext
    Wend

    FMain.Conexion_Base_Datos.Close

End

'*****
Public Sub TxtCodigo_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtRazonSocial.SetFocus
    Endif

End
```


Capítulo 2. CÓDIGO FUENTE

```
Public Sub TxtRazonSocial_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtDireccion.SetFocus  
    Endif  
  
End  
  
Public Sub TxtDireccion_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtLocalidad.SetFocus  
    Endif  
  
End  
  
Public Sub TxtLocalidad_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtProvincia.SetFocus  
    Endif  
  
End  
  
Public Sub TxtProvincia_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtNif.SetFocus  
    Endif  
  
End  
  
Public Sub TxtNif_KeyPress()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtCodigoPostal.SetFocus  
    Endif  
  
End  
  
Public Sub TxtCodigoPostal_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtWeb.SetFocus  
    Endif  
  
End  
  
Public Sub TxtWeb_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtEmail.SetFocus  
    Endif  
  
End  
!*****  
  
Public Sub TxtNombreComercial_KeyRelease()
```

```
If Key.Code = Key.Return Or Key.Code = Key.Enter Then
    TxtTelefonoComercial.SetFocus
Endif
```

```
End
```

```
Public Sub TxtTelefonoComercial_KeyRelease()
```

```
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtEmailComercial.SetFocus
    Endif
```

```
End
```

2.2.2.4 Frm_Vendedores.class

```
Public Sub Form_Open()
```

```
    Me.Title = "GESTION DE VENDEDORES"
```

```
    ' Llenamos los Dispositivos moviles que no estan en uso.
    Utilidades.Llenar_Combobox("select nombre from dispositivomovil where
codigodispositivomovil not in (select dispositivomovil from
dispositivomovilmvendedor)", CmbDispositivoMovil, "nombre")
    TxtCodigo.SetFocus
    Configurar_Dataset_Vendedores()
End
```

```
'' Configuramos el dataset Vendedores
Private Sub Configurar_Dataset_Vendedores()
    Dim Cabeceras As New String[]
    Dim Sql As String
```

```
    Cabeceras.Add("Codigo")
    Cabeceras.Add("Nombre")
    Cabeceras.Add("Nif")
    Cabeceras.Add("Direccion")
    Cabeceras.Add("Localidad")
    Cabeceras.Add("Provincia")
    Cabeceras.Add("Codigo Postal")
```

```
    DtsVendedor.Connection = FMain.Conexion_Base_Datos
    sql = "select * from vendedor"
    DtsVendedor.Table = sql
    DtbVendedor.Labels = Cabeceras
    DtbVendedor.view.Columns[0].Width = 100
    DtbVendedor.view.Columns[1].Width = 250
    DtbVendedor.view.Columns[2].Width = 100
    DtbVendedor.view.Columns[3].Width = 200
    DtbVendedor.view.Columns[4].Width = 100
    DtbVendedor.view.Columns[5].Width = 100
    DtbVendedor.view.Columns[6].Width = 110
```

```
End
```

```
'' Funcion para comprobar los Campos de los Vendedoreses <br>
'' Parametros:<br>
```

Capítulo 2. CÓDIGO FUENTE

```
' '      Accion - - > La accion desde la que comprobamos.<br>
' '      AGREGAR --> Para agregar datos.<br>
' '      MODIFICAR --> Cuando estamos modificando un dato.<br>
' '      Devuelve:<br>
' '      TRUE --> Si los valores son correctos.<br>
' '      FALSE --> valores erroneos.
Private Function Comprobar_Datos_Vendedores(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True

    If TxtCodigo.Text = "" Then
        Balloon.Error("Debe introducir unCodigo de Vendedor que va a crear",
TxtCodigo)
        Validar = False
        TxtCodigo.SetFocus
    Else If TxtNombre.text = ""
        Balloon.Error("Debe introducir el nombre de Vendedor", TxtNombre)
        Validar = False
        TxtNombre.SetFocus
    Else If TxtCodigoPostal.Text <> "" And (Not IsNumber(TxtCodigoPostal.Text))
        Balloon.Error("El codigo postal debe ser numerico ", TxtCodigoPostal)
        Validar = False
        TxtCodigoPostal.SetFocus
    Else If TxtCodigoPostal.Length < 5 And TxtCodigoPostal.Length > 0
        Balloon.Error("El codigo postal debe tener 5 digitos ", TxtCodigoPostal)
        Validar = False
        TxtCodigoPostal.SetFocus
    Endif
    'Si no seleccionamos ninguno borramos el asociado, si tenemmos asignado lo
    eliminamos y si no actualizamos.
    If Accion = "AGREGAR" Then
        Sql = "select * from vendedor where codigovendedor='" &
UCase(TxtCodigo.Text) & "'"
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El codigo Vendedor ya existe", TxtCodigo)
            Validar = False
            TxtCodigo.SetFocus
        Endif
    Endif
    Return Validar
End

Public Sub BtnSalirVendedores_Click()

    Me.Close

End

Public Sub BtnAgregarVendedor_Click()
    Dim Sql As String
    Dim i As Integer
    Dim Codigo_DM As String
    If Comprobar_Datos_Vendedores("AGREGAR") Then
        ' Insertamos datos en la tabla Vendedor
        sql = "insert into  vendedor
(codigovendedor,nombre,nif,direccion,localidad,provincia,codigopostal) values('"
& UCase(TxtCodigo.Text) & "','" & UCase(TxtNombre.text) & "','" &
```

Capítulo 2. CÓDIGO FUENTE

```
        UCase(TxtNif.Text) & "',' & UCase(TxtDireccion.Text) & "',' &
UCase(TxtLocalidad.Text) & "',' & UCase(TxtProvincia.Text) & "',' &
UCase(TxtCodigoPostal.Text) & "')"
        Utilidades.In_Act_Bor(sql)
        'Insertamos si hemos seleccionado Dispositivo Movil insertamos en la tabla
DispositivomovilVendedor
        If CmbDispositivoMovil.Text <> "" Then
            sql = "select * from dispositivomovil where nombre='" &
CmbDispositivoMovil.Text & "'"
            Codigo_DM = Utilidades.Obtener_Campo_Tabla(sql, "codigodispositivomovil")
            sql = "insert into dispositivomovilvendedor(dispositivomovil,vendedor)
values('" & UCase(Codigo_DM) & "',' & UCase(TxtCodigo.Text) & "'"
            Utilidades.In_Act_Bor(sql)
        Endif
        ' Insertamos los telefonos si hemos añadido en la tabla
        If LtbTelefono.Count > 0 Then
            For i = 0 To LtbTelefono.Count - 1
                sql = "insert into vendordtelefono (telefono,vendedor) values('" &
LtbTelefono[i].Text & "',' & UCase(TxtCodigo.Text) & "'"
                Utilidades.In_Act_Bor(sql)
            Next
        Endif
        Borrar_Campos_Vendedor()
        Configurar_Dataset_Vendedores()

    Endif

End

Public Sub BtnAgregarTelefono_Click()

    If TxtTelefono.Text = "" Then
        Balloon.Error("Introduzca telefono antes de añadir", TxtTelefono)
        TxtTelefono.SetFocus
    Else If Not IsNumber(TxtTelefono.Text)
        Balloon.Error("El Telefono es un dato Numerico", TxtTelefono)
        TxtTelefono.SetFocus
    Else If TxtTelefono.Length < 9
        Balloon.Error("Al Telefono le faltan digitos ", TxtTelefono)
        TxtTelefono.SetFocus
    Else
        LtbTelefono.Add(TxtTelefono.Text)
        TxtTelefono.Text = ""
    Endif

End

Public Sub BtnBorrarTelefono_Click()
    Dim Contar_Seleccionados As Integer
    Dim Numero_Telefonos As Integer
    Dim i As Integer
    Numero_Telefonos = LtbTelefono.Count
    Contar_Seleccionados = 0
    For i = 0 To Numero_Telefonos - 1
        If LtbTelefono[i].Selected Then
            Contar_Seleccionados = Contar_Seleccionados + 1
        Endif
    Next
    If Numero_Telefonos > 0 And Contar_Seleccionados > 0 Then
```

Capítulo 2. CÓDIGO FUENTE

```
        If LtbTelefono.Current.Selected Then
            LtbTelefono.Remove(LtbTelefono.Index)
        Endif
    Endif

End

'' Limpiamos los campos del Vendedor.
Private Sub Borrar_Campos_Vendedor()

    TxtCodigo.text = ""
    TxtCodigoPostal.text = ""
    TxtDireccion.text = ""
    TxtLocalidad.text = ""
    TxtNif.text = ""
    TxtNombre.text = ""
    TxtProvincia.text = ""
    LtbTelefono.Clear
    CmbDispositivoMovil.Text = ""
    TxtTelefono.Text = ""

End

Public Sub BtnBorrarVendedor_Click()
    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbVendedor.View.Row
    If DtbVendedor.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar el Vendedor y toda la informacion asociada : " & DtbVendedor.View[Numero_fila, 1].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from vendedor where codigovendedor='" &
DtbVendedor.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            sql = "delete from dispositivomovilvendedor where vendedor='" &
DtbVendedor.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            sql = "delete from vendedortelefono where vendedor='" &
DtbVendedor.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_Dataset_Vendedores()
        Endif
    Endif

End

Public Sub BtnEditarVendedor_Click()
    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Codigo_DM As String
    Dim i As Integer
    Numero_fila = DtbVendedor.View.Row
    If BtnEditarVendedor.Text = "EDITAR" Then
        TxtCodigo.text = DtbVendedor.View[Numero_fila, 0].Text
        TxtNombre.text = DtbVendedor.View[Numero_fila, 1].Text
        TxtNif.text = DtbVendedor.View[Numero_fila, 2].Text
        TxtDireccion.text = DtbVendedor.View[Numero_fila, 3].Text
    End If
End Sub
```

```

    TxtLocalidad.text = DtbVendedor.View[Numero_fila, 4].Text
    TxtProvincia.text = DtbVendedor.View[Numero_fila, 5].Text
    TxtCodigoPostal.text = DtbVendedor.View[Numero_fila, 6].Text
    'Llenamos el Los telefonos del vendedor.
    sql = "select * from vendedortelefono where vendedor='" & TxtCodigo.Text &
""
    Utilidades.Llenar_Listbox(sql, LtbTelefono, "telefono")
    ' Cargamos los datos de los dispositivos moviles y seleccionamos el asociado
al cliente
    Utilidades.Llenar_Combobox("select nombre from dispositivomovil where
codigodispositivomovil not in (select dispositivomovil from
dispositivomoviltendedor)", CmbDispositivoMovil, "nombre")
    sql = "select * from dispositivomoviltendedor where vendedor='" &
TxtCodigo.Text & ""
    If Utilidades.Tabla_Vacia(sql) > 0 Then
        Codigo_DM = Utilidades.Obtener_Campo_Tabla(sql, "dispositivomovil")
        sql = "select * from dispositivomovil where codigodispositivomovil='" &
Codigo_DM & ""
        CmbDispositivoMovil.Add(Utilidades.Obtener_Campo_Tabla(sql, "nombre"))
        CmbDispositivoMovil.text = Utilidades.Obtener_Campo_Tabla(sql, "nombre")
    Endif
    BtnEditarVendedor.text = "ACTUALIZAR"
    BtnAgregarVendedor.Enabled = False
    BtnBorrarVendedor.Enabled = False
    TxtCodigo.Enabled = False
Else
    If Comprobar_Datos_Vendedores("MODIFICAR") Then
        'Actualizamos la tabla Vendedor
        sql = "update vendedor set nombre='" & UCase(TxtNombre.text) & "',nif='" &
UCase(TxtNif.Text) & "',direccion='" & UCase(TxtDireccion.Text) &
        "',localidad='" & UCase(TxtLocalidad.Text) & "',provincia='" &
UCase(TxtProvincia.Text) & "',codigopostal='" & UCase(TxtCodigoPostal.Text) & "'
        where codigovendedor='" & UCase(TxtCodigo.Text) & ""
        Utilidades.In_Act_Bor(sql)
        ' Actualizamos Telefonos, borramos los telefonos, si el listbox de los
telefonos esta vacio no hacemos nada.
        ' Si tenemos datos en el listbox de telefono los añadimos.
        sql = "delete from vendedortelefono where vendedor='" & TxtCodigo.Text &
""
        Utilidades.In_Act_Bor(sql)
        If LtbTelefono.Count > 0 Then
            For i = 0 To LtbTelefono.Count - 1
                sql = "insert into vendedortelefono (telefono,vendedor) values('" &
LtbTelefono[i].Text & "','" & UCase(TxtCodigo.Text) & "')"
                Utilidades.In_Act_Bor(sql)
            Next
        Endif
        ' Actualizamos lo referente al Dispositivo Movil Asignado. Borramos los
dispositivos moviles, y si tenemos asignado lo añadimos
        sql = "delete from dispositivomoviltendedor where vendedor='" &
TxtCodigo.Text & ""
        Utilidades.In_Act_Bor(sql)
        If CmbDispositivoMovil.text <> "" Then ' Si no hemos seleccionado DM, si
teniamos asignado borramos.
            sql = "select * from dispositivomovil where nombre='" &
CmbDispositivoMovil.Text & ""
            Codigo_DM = Utilidades.Obtener_Campo_Tabla(sql,
"codigodispositivomovil")

```

```
        sql = "insert into dispositivomovilvendedor(dispositivomovil,vendedor)
values('" & UCase(Codigo_DM) & "',''" & UCase(TxtCodigo.Text) & "','")"
        Utilidades.In_Act_Bor(sql)
    Endif
    BtnEditarVendedor.text = "EDITAR"
    BtnAgregarVendedor.Enabled = True
    BtnBorrarVendedor.Enabled = True
    TxtCodigo.Enabled = True
    Borrar_Campos_Vendedor()

Endif

Endif
Configurar_Dataset_Vendedores()
End

Public Sub TxtCodigo_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtNombre.SetFocus
    Endif

End

Public Sub TxtNombre_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtDireccion.SetFocus
    Endif

End

Public Sub TxtDireccion_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtLocalidad.SetFocus
    Endif

End

Public Sub TxtLocalidad_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtProvincia.SetFocus
    Endif

End

Public Sub TxtNif_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtCodigoPostal.SetFocus
    Endif

End

Public Sub TxtProvincia_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
```

```
    TxtNif.SetFocus
Endif
```

```
End
```

2.2.3 Modulo Facturación.

2.2.3.1 *Frm_Facturacion.class*

```
Public Tarifa_Cliente As String ' Almacenamos la tarifa del Cliente al cual se
esta reazliando Factura/Albaran.
Public Linea_Detalle As Integer ' Controlamos en la linea de detalle que
estamos posicionados.
Public Linea As Integer ' Contador de Lineas detalle de la Factura
```

```
Public Sub Form_Open()
```

```
    Me.title = "Facturacion"
    Configurar_Objetos_Facturacion()
```

```
End
```

```
Public Sub MenuEdicionFacturas_Click()
```

```
    Frm_Visualizar_Factura_Albaran.ShowModal()
```

```
End
```

```
Public Sub BtnSalirFacturacion_Click()
```

```
    Me.Close
```

```
End
```

```
Public Sub MenuSalir_Click()
```

```
    Me.close
```

```
End
```

```
Public Sub MenuPasoContabilidad_Click()
```

```
    Frm_PasoContabilidad.Show()
```

```
End
```

```
Public Sub MenuFacturarAlbaranes_Click()
```

```
    Frm_FacturacionAlbaranes.ShowModal()
```

```
End
```

```
Public Sub RdbFacturas_Click()
```

```
    lblNumero.text = "N° Factura :"  
    ChkCobradaPagado.Text = "Factura Cobrada"  
    FrameDetalles.Text = "Factura Detalle"  
    lblTotal.Text = "Total Factura :"  
    TxtNumero.Text = Utilidades.Calculo_Numero_FA("F")
```

```
End
```


Capítulo 2. CÓDIGO FUENTE

```
Public Sub RdbAlbaranes_Click()

    LblNumero.text = "N° Albaran :"
    ChkCobradaPagado.Text = "Albaran Pagado"
    FrameDetalles.Text = "Albaran Detalle"
    LblTotal.Text = "Total Albaran :"
    TxtNumero.Text = Utilidades.Calculo_Numero_FA("A")
End

Public Sub CmbFormaPago_Click()

    Dim dias As Integer
    dias = CInt(Utilidades.Obtener_Campo_Tabla("select dias from formapago where
nombreformapago='" & CmbFormaPago.Text & "'", "dias"))
    If dias > 0 Then
        DteFechavencimiento.Value = DateAdd(Now, gb.Day, dias)
    Else
        DteFechavencimiento.Value = Date(Year(Now), Month(Now), Day(Now))
    Endif

End

Public Sub CmbCliente_Click()

    Dim Codigo_Cliente As String
    Dim Iva As Integer
    Dim RE As Integer
    Dim Dias As Integer
    If CmbCliente.Text = "" Then
    Else

        iva = Utilidades.Obtener_Campo_Tabla("select iva from tipoiva", "iva")
        re = Utilidades.Obtener_Campo_Tabla("select recargo from tipoiva",
"recargo")
        Codigo_Cliente = Utilidades.Obtener_Campo_Tabla("select codigocliente from
clientes where razonsocial='" & CmbCliente.Text & "'", "codigocliente")
        CmbVendedor.Text = Utilidades.Obtener_Campo_Tabla("select vendedor from
clientes where razonsocial='" & CmbCliente.Text & "'", "vendedor")
        CmbFormaPago.Text = Utilidades.Obtener_Campo_Tabla("select formapago from
clientes where razonsocial='" & CmbCliente.Text & "'", "formapago")
        Tarifa_Cliente = Utilidades.Obtener_Campo_Tabla("select tarifa from clientes
where razonsocial='" & CmbCliente.Text & "'", "tarifa")

        dias = CInt(Utilidades.Obtener_Campo_Tabla("select dias from formapago where
nombreformapago='" & CmbFormaPago.Text & "'", "dias"))
        If dias > 0 Then
            DteFechavencimiento.Value = DateAdd(Now, gb.Day, dias)
        Else
            DteFechavencimiento.Value = Date(Year(Now), Month(Now), Day(Now))
        Endif
        VlbDescuentoESP.Value = Utilidades.Obtener_Campo_Tabla("select descuentoesp
from clientes where razonsocial='" & CmbCliente.Text & "'", "descuentoesp")
        VlbDescuentoPP.Value = Utilidades.Obtener_Campo_Tabla("select descuentopp
from clientes where razonsocial='" & CmbCliente.Text & "'", "descuentopp")
        If Utilidades.Obtener_Campo_Tabla("select iva from clientes where
razonsocial='" & CmbCliente.Text & "'", "iva") = "S" Then
            VlbIva.Value = iva
        Else

    End If

End If

End
```

```
VlbIva.Value = 0
Endif
If Utilidades.Obtener_Campo_Tabla("select recargo from clientes where
razonsocial='" & CmbCliente.Text & "'", "recargo") = "S" Then
    VlbRecargo.Value = re
Else
    VlbRecargo.Value = 0
Endif
If Utilidades.Tabla_Vacia("select * from articulo") = 0 Then

Else
    CmbArticulo.clear
    Utilidades.Llenar_Combobox("select * from articulo", CmbArticulo,
"nombre")
    CmbArticulo.Index = -1
Endif
Endif
End
```

```
Public Sub BtnAgregarFactura_Click()

If Comprobar_Datos_Factura("NUEVO") Then
    If BtnAgregarFactura.text = "NUEVO" Then
        BtnAgregarFactura.text = "GRABAR"
        FrameDetalles.Enabled = True
        PanelFacturaAlbaran.Enabled = False
        TxtNumero.Enabled = False
        CmbCliente.Enabled = False
        ChkGenerarFactura.Enabled = True
    Else
        BtnAgregarFactura.text = "NUEVO"
        FrameDetalles.Enabled = False
        PanelFacturaAlbaran.Enabled = True
        TxtNumero.Enabled = True
        CmbCliente.Enabled = True
        ChkGenerarFactura.value = False
        ChkGenerarFactura.Enabled = False
        Grabar_Factura()
        If ChkGenerarFactura.Value = True Then
            Utilidades.Generar_Factura_Albaran(TxtNumero.text)
        Endif

        Limpiar_Datos_Factura()
        linea = 0
        CmbCliente.Index = -1
        CmbVendedor.Index = -1
        CmbFormaPago.Index = -1
    Endif

Endif

End
Public Sub ChkCobradaPagado_Click()

    If ChkCobradaPagado.Value = True Then
```

Capítulo 2. CÓDIGO FUENTE

```
DteFechaCobro.Enabled = True

Else
    DteFechaCobro.Enabled = False
Endif

End

Public Sub BtnEdicionFacturas_Click()
    If Utilidades.Tabla_Vacia("select * from facturas") > 0 Or
    Utilidades.Tabla_Vacia("select * from albaranes") > 0 Then
        Frm_Visualizar_Factura_Albaran.ShowModal()
    Else
        Message.Error("No hay Facturas y Albaranes para editar")
    Endif
End

''***** Lineas Detalle *****
''
''*****
Public Sub BtnAgregarLineaDetalle_Click()
    Dim Aux_Importe As Float
    If Comprobar_Datos_Factura("DETALLES") Then
        Aux_Importe = vlbpeso.Value * VlbPrecio.Value
        VlbImporte.text = Format(Aux_Importe, "##0.00")

        Agrega_Linea_Detalle_Factura()
        Limpiar_Detalles_Facturas()

    Endif
End

Public Sub BtnBorrarLineaDetalle_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Dim Texto As String
    Dim Importe As Float
    If GridLineasDetalle.Rows.Selection.Count > 0 Then
        Numero_fila = GridLineasDetalle.Rows.Selection[0]
        Importe = CFloat(GridLineasDetalle[Numero_fila, 4].Text)
        Texto = "Desea borrar la Linea Detalle : <b>" &
GridLineasDetalle[Numero_fila, 1].Text & "</b> y Artículo: <b>" &
GridLineasDetalle[Numero_fila, 6].Text & "</b>"
        Respuesta = Message.Delete(Texto, "Si", "No")
        If respuesta = 1 Then
            If RdbFacturas.value = True Then
                sql = "delete from facturasdetalles where linea='" &
GridLineasDetalle[Numero_fila, 1].Text & "'"
                Utilidades.In_Act_Bor(sql)
                sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & UCase(TxtNumero.Text) & "'"
                Llenar_DataGrid_LineasDetalle(sql, "F")
            End If
        End If
    End If
End Sub
```

```
Endif
If RdbAlbaranes.Value = True Then
    sql = "delete from albaranesdetalle where linea='" &
GridLineasDetalle[Numero_fila, 1].Text & "'"
    Utilidades.In_Act_Bor(sql)
    sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & UCase(TxtNumero.Text) & "'"
    Llenar_DataGrid_LineasDetalle(sql, "A")
Endif
VlbImporteBase.Value = Format(VlbImporteBase.Value - Importe, "##0.00")
Calculo_Descuentos_Ivas()
Endif

Endif
End

Public Sub VlbPeso_KeyRelease()

If Key.Code = Key.Return Or Key.Code = Key.Enter Then
    If vlbpeso.Value = "" Or vlbpeso.value = 0 Then
        vlbpeso.SetFocus
    Else
        VlbPrecio.SetFocus
    Endif
Endif

End

Public Sub VlbPrecio_KeyRelease()

If Key.Code = Key.Return Or Key.Code = Key.Enter Then
    If VlbPrecio.Value = "" Or VlbPrecio.value = 0 Then
        VlbPrecio.SetFocus
    Else
        If vlbpeso.value > 0 And VlbPrecio.Value > 0 Then
            VlbImporte.Value = Format(vlbpeso.Value * VlbPrecio.Value, "##0.00")

        Endif
    Endif
Endif

Endif

End

Public Sub CmbArticulo_Click()

Dim Codigo As String
codigo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from articulo
where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
If Utilidades.Tabla_Vacia("select * from lotes where articulo='" & codigo &
"'") > 0 Then
    Utilidades.Llenar_Combobox("select * from lotes where articulo='" & codigo &
"'", CmbLote, "codigolote")
    CmbLote.Text = ""
    CmbLote.Index = -1
Else
    CmbLote.Text = ""
    CmbLote.Clear
Endif

Endif
```

Capítulo 2. CÓDIGO FUENTE

```
    vlbpeso.Value = 0
    VlbPrecio.Value = 0
    VlbImporte.Value = 0
    CmbLote.SetFocus
End

Public Sub CmbLote_Click()
    Dim Codigo As String
    Dim Sql As String
    codigo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from articulo
where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    If Utilidades.Tabla_Vacia("select cantidad from lotes where codigolote='" &
CmbLote.Text & "'") > 0 Then
        vlbpeso.Value = Utilidades.Obtener_Campo_Tabla("select cantidad from lotes
where codigolote='" & CmbLote.Text & "'", "cantidad")
        sql = "select precio from tarifaarticulo where articulo='" & codigo & "' and
tarifa='" & Tarifa_Cliente & "'"
        If Utilidades.Tabla_Vacia(sql) > 0 Then
            VlbPrecio.value = Utilidades.Obtener_Campo_Tabla(sql, "precio")
        Endif
    Endif
    vlbpeso.SetFocus
End

Public Sub CmbLote_KeyRelease()
    Dim Codigo As String
    Dim Sql As String
    codigo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from articulo
where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        If CmbLote.Text = "" Then
            CmbLote.SetFocus
        Else
            sql = "select precio from tarifaarticulo where articulo='" & codigo & "'
and tarifa='" & Tarifa_Cliente & "'"
            If Utilidades.Tabla_Vacia(sql) > 0 Then
                VlbPrecio.value = Utilidades.Obtener_Campo_Tabla(sql, "precio")
            Endif
            vlbpeso.SetFocus
        Endif
    Endif
End

Public Sub BtnInicio_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        GridLineasDetalle.Rows.UnselectAll
        Linea_Detalle = 0
        GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
    Endif

End

Public Sub BtnAnterior_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        If Linea_Detalle = 0 Then
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Else
```

```
        GridLineasDetalle.Rows.UnselectAll
        Linea_Detalle = Linea_Detalle - 1
        GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
    Endif
Endif

End

Public Sub BtnSiguiente_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        If Linea_Detalle = GridLineasDetalle.Rows.Count - 1 Then
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Else
            GridLineasDetalle.Rows.UnselectAll
            Linea_Detalle = Linea_Detalle + 1
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Endif
    Endif

End

Public Sub BtnFin_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        GridLineasDetalle.Rows.UnselectAll
        Linea_Detalle = GridLineasDetalle.Rows.Count - 1
        GridLineasDetalle.Rows.Select(Linea_Detalle, 1)

    Endif

End

''***** Funciones privadas*****
''
''*****

'' Configuramos los Objetos de los Lotes
Private Sub Configurar_Objetos_Facturacion()
    Dim Error_Tablas_Vacias As String
    Dim Error_V As Boolean

    error_V = False

    If Utilidades.Tabla_Vacia("select * from clientes") = 0 Then
        Error_Tablas_Vacias = "Deber Cear Clientes"
        Error_V = True
    Else
        CmbCliente.clear
        Utilidades.Llenar_Combobox("select * from clientes", CmbCliente,
"razonsocial")
        CmbCliente.Index = -1
    Endif
    If Utilidades.Tabla_Vacia("select * from vendedor") = 0 Then
        If Error_Tablas_Vacias = "" Then
            Error_Tablas_Vacias = "Deber Cear Vendedores"
        Else
            Error_Tablas_Vacias &= " y Crear Vendedores."
        Endif
    Endif
End Sub
```

```
Error_V = True
Else
    CmbVendedor.clear
    Utilidades.Llenar_Combobox("select * from vendedor", CmbVendedor,
"nombre")
Endif
If Utilidades.Tabla_Vacia("select * from formapago") = 0 Then
    If Error_Tablas_Vacias = "" Then
        Error_Tablas_Vacias = "Deber Cear Formas de Pago en Utilidades"
    Else
        Error_Tablas_Vacias &= " y Crear Formas de Pago en Utilidades."
    Endif
    Error_V = True
Else
    CmbFormaPago.clear
    Utilidades.Llenar_Combobox("select * from formapago", CmbFormaPago,
"nombreformapago")
    CmbFormaPago.Index = -1
Endif
If Utilidades.Tabla_Vacia("select * from articulo") = 0 Then
    If Error_Tablas_Vacias = "" Then
        Error_Tablas_Vacias = "Deber Cear Articulos para crear Facturas"
    Else
        Error_Tablas_Vacias &= " y Crear Articulos."
    Endif
    Error_V = True
Else
    'CmbArticulo.clear
    'Utilidades.Llenar_Combobox("select * from articulo", CmbArticulo,
"nombre")
    'CmbArticulo.Index = -1
Endif
If Error_V Then
    TxtErrorFacturacion.Visible = True
    TxtErrorFacturacion.text = Error_Tablas_Vacias
Endif
DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaCobro.Value = Date(Year(Now), Month(Now), Day(Now))
DteHora.Value = Format$(Now, "h:n:s")
LblTotalFactura.text = Format(0, "##0.00")
VlbImporteBase.Value = Format(0, "##0.00")
TxtNumero.Text = Utilidades.Calculo_Numero_FA("F")
Configurar_DataGrid_LineasDetalle()
Linea_Detalle = 0
linea = 0
RdbFacturas.Value = True
End

'' Procedimiento para configurar Grid LineasDetalle
Private Sub Configurar_DataGrid_LineasDetalle()

    GridLineasDetalle.Columns.Count = 7
    GridLineasDetalle.Header = GridLineasDetalle.Both
    GridLineasDetalle.Columns[0].Title = "Numero"
    GridLineasDetalle.Columns[1].Title = "Linea"
    GridLineasDetalle.Columns[2].Title = "Precio"
    GridLineasDetalle.Columns[3].Title = "Peso"
    GridLineasDetalle.Columns[4].Title = "Importe"
```

```
GridLineasDetalle.Columns[5].Title = "Lote"
GridLineasDetalle.Columns[6].Title = "Articulo"
GridLineasDetalle.Columns[0].Width = 100
GridLineasDetalle.Columns[1].Width = 100
GridLineasDetalle.Columns[2].Width = 100
GridLineasDetalle.Columns[3].Width = 100
GridLineasDetalle.Columns[4].Width = 100
GridLineasDetalle.Columns[5].Width = 150
GridLineasDetalle.Columns[6].Width = 200

End

'' Procedimiento para Llenar de datos el DataGrid LineasDetalle<br>
'' Parametros:<br>
''     Tipo --> Tomara los siguientes valores<br>
''     A -- Albaran.<br>
''     F -- Factura.<br>
''     Sql --> Setncia Sql para llenar el Data Grid<br>
Private Sub Llenar_DataGrid_LineasDetalle(Sql As String, Tipo As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridLineasDetalle.Rows.Count > 0 Then
        GridLineasDetalle.Rows.Remove(0, GridLineasDetalle.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst

    If Tipo = "F" Then
        While Result_Trabajo.Available
            Numero_fila = GridLineasDetalle.Rows.Count
            GridLineasDetalle.Rows.Insert(Numero_fila, 1)
            GridLineasDetalle[Numero_fila, 0].Text = Result_Trabajo["L.numerofactura"]
            GridLineasDetalle[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
            GridLineasDetalle[Numero_fila, 2].Text = Result_Trabajo["L.precio"]
            GridLineasDetalle[Numero_fila, 3].Text = Result_Trabajo["L.peso"]
            GridLineasDetalle[Numero_fila, 4].Text = Result_Trabajo["L.importe"]
            GridLineasDetalle[Numero_fila, 5].Text = Result_Trabajo["L.lote"]
            GridLineasDetalle[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
            Result_Trabajo.MoveNext
        Wend
    Else
        While Result_Trabajo.Available
            Numero_fila = GridLineasDetalle.Rows.Count
            GridLineasDetalle.Rows.Insert(Numero_fila, 1)
            GridLineasDetalle[Numero_fila, 0].Text = Result_Trabajo["L.numeroalbaran"]
            GridLineasDetalle[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
            GridLineasDetalle[Numero_fila, 2].Text = Result_Trabajo["L.precio"]
            GridLineasDetalle[Numero_fila, 3].Text = Result_Trabajo["L.peso"]
            GridLineasDetalle[Numero_fila, 4].Text = Result_Trabajo["L.importe"]
            GridLineasDetalle[Numero_fila, 5].Text = Result_Trabajo["L.lote"]
            GridLineasDetalle[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
            Result_Trabajo.MoveNext
        Wend
    Endif
    FMain.Conexion_Base_Datos.Close
    GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
```


End

```
' ' Funcion para comprobar los datos de los lotes y las lineas de lotes.<br>
' ' Parametros:<br>
' '     Accion Tomara los siguientes valores<br>
' '         NUEVO -- Comprobar los datos para crear nueva Factura.<br>
' '         DETALLES -- Comprobar los datos antes de añadir línea detallle.<br>
' ' Devuelve:<br>
' '     TRUE -- Datos corecctos<br>
' '     FALSE -- Datos erroneos
Private Function Comprobar_Datos_Factura(Accion As String) As Boolean

    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    If Accion = "NUEVO" Then
        sql = "select *from facturas where numerofactura='" & UCase(TxtNumero.Text)
    & ""
        If TxtNumero.Text = "" Then
            Balloon.Error("Debe introducir Numero Facatura/Albaran", TxtNumero)
            Validar = False
            TxtNumero.SetFocus
        Else If Utilidades.Comprobar_existe_dato(Sql)
            Balloon.Error("Debe introducir Numero Facatura/Albaran ya exite",
TxtNumero)
            Validar = False
            TxtNumero.SetFocus
        Else If CmbCliente.Text = ""
            Balloon.Error("Debe elegir Cliente para crear Factura/Albaran",
CmbCliente)
            Validar = False
            CmbCliente.SetFocus
        Endif
    Else If accion = "DETALLES"
        If CmbArticulo.Text = "" Then
            Balloon.Error("Debe Elegir Articulo", CmbArticulo)
            Validar = False
        Else If CmbLote.Text = ""
            Balloon.Error("Debe Elegir o Introducir Numero de Lote", CmbLote)
            Validar = False
            CmbLote.SetFocus
        Else If vlbpeso.Value = 0
            Balloon.Error("Debe Introducir Peso", vlbpeso)
            Validar = False
            vlbpeso.SetFocus
        Else If VlbPrecio.Value = 0
            Balloon.Error("Debe Introducir Precio", VlbPrecio)
            Validar = False
            VlbPrecio.SetFocus
        Endif
    Endif

    Return Validar
End

' Procedimiento para limpiar los datos de Los Detalles Facturas
Private Sub Limpiar_Detalles_Facturas()
```

```

CmbArticulo.index = -1
CmbLote.text = ""
CmbLote.Clear
vlbpeso.Value = 0
VlbPrecio.Value = 0
VlbImporte.Value = 0

End

' Añadimos la linea detalle a la factura y calculamos descuentos e ivas.
Private Sub Agrega_Lineas_Detalle_Factura()

    Dim Sql As String
    Dim Codigo_articulo As String
    linea = linea + 1
    Codigo_articulo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from
articulo where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    If RdbFacturas.Value = True Then
        sql = "insert into
facturasdetalles( numerofactura, linea, precio, peso, importe, lote, articulo)
values(' " &
        UCase(TxtNumero.Text) & "'," & linea & "," & CFloat(VlbPrecio.Value) & "," &
CFloat(vlbpeso.Value) & "," & CFloat(Vlbimporte.Value) &
        ",'" & UCase(CmbLote.text) & "','" & Codigo_articulo & "')"
        Utilidades.In_Act_Bor(sql)
        sql = "select
L.numerofactura, L.linea, L.precio, L.peso, L.importe, L.lote, A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numerofactura='" & UCase(TxtNumero.Text) & "'"
        Llenar_DataGrid_LineasDetalle(sql, "F")
    Endif
    If RdbAlbaranes.Value = True Then
        sql = "insert into
albaranesdetalle( numeroalbaran, linea, precio, peso, importe, lote, articulo)
values(' " &
        UCase(TxtNumero.Text) & "'," & linea & "," & CFloat(VlbPrecio.Value) & "," &
CFloat(vlbpeso.Value) & "," & CFloat(Vlbimporte.Value) &
        ",'" & UCase(CmbLote.text) & "','" & Codigo_articulo & "')"
        Utilidades.In_Act_Bor(sql)
        sql = "select
L.numeroalbaran, L.linea, L.precio, L.peso, L.importe, L.lote, A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numeroalbaran='" & UCase(TxtNumero.Text) & "'"
        Llenar_DataGrid_LineasDetalle(sql, "A")
    Endif
    VlbImporteBase.Value = Format(VlbImporteBase.Value + VlbImporte.Value,
"##0.00")
    Calculo_Descuentos_Ivas()
End

'' Procedimiento para calcular Descuentos e Iva cuando añadimos o borramos una
linea detalle.

Private Sub Calculo_Descuentos_Ivas()

    Dim Descuento_PP As Float
    Dim Descuento_ESP As Float
    Dim Iva As Float

```

```
Dim RE As Float
Dim AuxBase As Float
If VlbDescuentoESP.Value > 0 Then
    Descuento_ESP = (VlbImporteBase.Value * VlbDescuentoESP.Value) / 100
Else
    Descuento_ESP = 0
Endif
If VlbDescuentoPP.Value > 0 Then
    Descuento_PP = (VlbImporteBase.Value * VlbDescuentoPP.Value) / 100
Else
    Descuento_PP = 0
Endif
VlbImporteDescuentoESP.Value = Format(Descuento_ESP, "##0.00")
VlbImporteDescuentoPP.Value = Format(Descuento_PP, "##0.00")
auxbase = VlbImporteBase.Value - Descuento_ESP - Descuento_PP
If VlbIva.Value > 0 Then
    iva = (auxbase * VlbIva.value) / 100
Else
    iva = 0
Endif
If VlbRecargo.Value > 0 Then
    re = (auxbase * VlbRecargo.value) / 100
Else
    re = 0
Endif
VlbImporteIva.Value = Format(iva, "##0.00")
VlbImporteRecargo.Value = Format(re, "##0.00")
LblTotalFactura.Text = Format(auxbase + iva + re, "##0.00")
End
'' Graba la factura.
Private Sub Grabar_Factura()

    Dim Codigo_Vendedor As String
    Dim Codigo_Cliente As String
    Dim Sql As String
    Dim Cobrado_Pagada As String
    If ChkCobradaPagado.Value = True Then
        Cobrado_Pagada = "S"
    Else
        Cobrado_Pagada = "N"
    Endif
    Codigo_Cliente = Utilidades.Obtener_Campo_Tabla("select codigocliente from
clientes where razonsocial='" & CmbCliente.Text & "'", "codigocliente")
    Codigo_Vendedor = Utilidades.Obtener_Campo_Tabla("select codigovendedor from
vendedor where nombre='" & CmbVendedor.Text & "'", "codigovendedor")
    If RdbFacturas.Value = True Then
        sql = "insert into
facturas(numerofactura,fecha,hora,base,iva,recargo,importeiva,importerecargo,fec
hacobro,descuentopp,importedescuentopp," &

"descuentoesp,importedescuentoesp,coorada,formapago,vencimiento,contabilizada,to
tal,cliente,vendedor)values( '" &
        UCase(TxtNumero.Text) & "','" &
Utilidades.Fecha_Correcta(DteFechaCreacion.Value) & "','" & DteHora.value & "','"
& VlbImporteBase.Value & "," &
        VlbIva.Value & "," & VlbRecargo.Value & "," & VlbImporteIva.Value & "," &
VlbImporteRecargo.Value & "','" & Utilidades.Fecha_Correcta(DteFechaCobro.Value)
& "','" &
```

Capítulo 2. CÓDIGO FUENTE

```
VlbDescuentoPP.Value & "," & VlbImporteDescuentoPP.Value & "," &
VlbDescuentoESP.Value & "," & VlbImporteDescuentoESP.Value & "," &
Cobrado_Pagada & "','" &
CmbFormaPago.Text & "','" &
Utilidades.Fecha_Correcta(DteFechavencimiento.Value) & "','"N'," &
LblTotalFactura.Text & "','" & Codigo_Cliente & "','" & Codigo_Vendedor & "')"
Utilidades.In_Act_Bor(sql)
TxtNumero.Text = Utilidades.Calculo_Numero_FA("F")
Endif

If RdbAlbaranes.Value = True Then
    sql = "insert into
albaranes(numeroalbaran,fecha,hora,base,iva,recargo,importeiva,importerecargo,fe
chacobro,descuentopp,importedescuentopp," &

"descuentoesp,importedescuentoesp,pagado,formapago,vencimiento,facturado,total,c
liente,vendedor)values( '" &
    UCase(TxtNumero.Text) & "','" &
Utilidades.Fecha_Correcta(DteFechaCreacion.Value) & "','" & DteHora.value & "','"
& VlbImporteBase.Value & "," &
    VlbIva.Value & "," & VlbRecargo.Value & "," & VlbImporteIva.Value & "," &
VlbImporteRecargo.Value & "','" & Utilidades.Fecha_Correcta(DteFechaCobro.Value)
& "','" &
    VlbDescuentoPP.Value & "," & VlbImporteDescuentoPP.Value & "," &
VlbDescuentoESP.Value & "," & VlbImporteDescuentoESP.Value & "','" &
Cobrado_Pagada & "','" &
    CmbFormaPago.Text & "','" &
Utilidades.Fecha_Correcta(DteFechavencimiento.Value) & "','"N'," &
LblTotalFactura.Text & "','" & Codigo_Cliente & "','" & Codigo_Vendedor & "')"
    Utilidades.In_Act_Bor(sql)
    TxtNumero.Text = Utilidades.Calculo_Numero_FA("A")
Endif
End

'' Limpia los datos de la cabecera de la factura.
Private Sub Limpiar_Datos_Factura()

    linea = 0
    Linea_Detalle = 0
    'RdbFacturas.Value = True
    'RdbAlbaranes.Value = False
    ChkCobradaPagado.Value = False
    DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFechaCobro.Value = Date(Year(Now), Month(Now), Day(Now))
    DteHora.Value = Format$(Now, "h:n:s")
    LblTotalFactura.text = Format(0, "##0.00")
    VlbDescuentoESP.Value = 0
    VlbDescuentoPP.Value = 0
    VlbImporteDescuentoESP.Value = 0
    VlbImporteDescuentoPP.Value = 0
    VlbIva.Value = 0
    VlbImporteIva.Value = 0
    VlbRecargo.Value = 0
    VlbImporteRecargo.Value = 0
    VlbImporteBase.value = 0
    CmbCliente.Index = -1
    CmbVendedor.Index = -1
    CmbFormaPago.Index = -1
    If GridLineasDetalle.Rows.Count > 0 Then
```

```
GridLineasDetalle.Rows.Remove(0, GridLineasDetalle.Rows.Count)
End If
End
```

2.2.3.2 *Frm_Edicion_Facturas_Albaranes.class*

```
Public Tarifa_Cliente As String ' Almacenamos la tarifa del Cliente al cual se
esta reazliando Factura/Albaran.
Public Tipo As String ' Almacena F si es una factura y A si es un albaran para
editar.
Public Linea_Detalle As Integer ' Controlamos en la linea de detalle que
estamos posicionados.
Public Linea As Integer ' Contador de Lineas detalle de la Factura

Public Sub Form_Open()
    Dim Sql_Cabecera As String
    Dim Sql_Detalles As String
    Me.Title = "EDICION DE FACTURAS/ALBARANES"
    TxtNumero.Text = Frm_Visualizar_Factura_Albaran.Factura_Albaran_Editar
    Configurar_DataGrid_LineasDetalle()
    Linea_Detalle = 0
    Tipo = Mid(TxtNumero.Text, 1, 1)
    Utilidades.Llenar_Combobox("select * from clientes", CmbCliente,
"razonsocial")
    Utilidades.Llenar_Combobox("select * from vendedor", CmbVendedor, "nombre")
    Utilidades.Llenar_Combobox("select * from formapago", CmbFormaPago,
"nombreformapago")
    Utilidades.Llenar_Combobox("select * from articulo", CmbArticulo, "nombre")
    If tipo = "F" Then
        LblNumero.text = "N° Factura :"
        ChkCobradaPagado.Text = "Factura Cobrada"
        FrameDetalles.Text = "Factura Detalle"
        FrameFacturacion.Text = "Factura"
        ChkContabilizada.Text = "Contabilizada"
        Sql_Cabecera = "select
F.numerofactura,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.cobrada,F.formapago,F.vencimiento,F.con
tabilizada,F.total,C.razonsocial,V.nombre from facturas F left join clientes C
on" &
" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigoendedor where F.numerofactura='" & TxtNumero.Text & "'"
        Sql_Detalles = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & UCase(TxtNumero.Text) & "'"
    Else
        LblNumero.text = "N° Albaran :"
        ChkCobradaPagado.Text = "Albaran Pagado"
        FrameDetalles.Text = "Albaran Detalle"
        FrameFacturacion.Text = "Albaran"
        ChkContabilizada.Text = "Facturado"
        Sql_Cabecera = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &
```

```
",F.descuentoesp,F.importedescountoesp,F.pagado,F.formapago,F.vencimiento,F.facturado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor where F.numeroalbaran='" & TxtNumero.Text & "'"
    Sql_Detalles = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & UCase(TxtNumero.Text) & "'"

Endif
Visualizar_Datos_Factura_Albaran(Sql_Cabecera)
Llenar_DataGrid_LineasDetalle(Sql_Detalles)
Calculo_Descuentos_Ivas()

End

Public Sub BtnSalirEdicion_Click()

    Me.Close

End

Public Sub BtnAgregarFactura_Click()

    Grabar_Factura()
    If Chkgenerarfactura.Value = True Then
        Utilidades.Generar_Factura_Albaran(TxtNumero.Text)
    Endif
End

'***** Lineas Detalle *****
'
'*****
Public Sub BtnAgregarLineaDetalle_Click()
    Dim Aux_Importe As Float
    If Comprobar_Datos_Factura("DETALLES") Then
        Aux_Importe = vlbpeso.Value * VlbPrecio.Value
        VlbImporte.text = Format(Aux_Importe, "##0.00")
        Agrega_Linea_Detalle_Factura()
        Limpiar_Detalles_Facturas()

    Endif
End

Public Sub BtnBorrarLineaDetalle_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Dim Texto As String
    Dim Importe As Float
    If GridLineasDetalle.Rows.Selection.Count > 0 Then
        Numero_fila = GridLineasDetalle.Rows.Selection[0]
        Importe = CFloat(GridLineasDetalle[Numero_fila, 4].Text)
        Texto = "Desea borrar la Linea Detalle : <b>" &
GridLineasDetalle[Numero_fila, 1].Text & "</b> y Articulo: <b>" &
GridLineasDetalle[Numero_fila, 6].Text & "</b>"
```

Capítulo 2. CÓDIGO FUENTE

```
Respuesta = Message.Delete(Texto, "Si", "No")
If respuesta = 1 Then
    If tipo = "F" Then
        sql = "delete from facturasdetalles where linea='" &
GridLineasDetalle[Numero_fila, 1].Text & "'"
        Utilidades.In_Act_Bor(sql)
        sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & UCase(TxtNumero.Text) & "'"
        Llenar_DataGrid_LineasDetalle(sql)
    Endif
    If tipo = "A" Then
        sql = "delete from albaranesdetalle where linea='" &
GridLineasDetalle[Numero_fila, 1].Text & "'"
        Utilidades.In_Act_Bor(sql)
        sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & UCase(TxtNumero.Text) & "'"
        Llenar_DataGrid_LineasDetalle(sql)
    Endif
    VlbImporteBase.Value = Format(VlbImporteBase.Value - Importe, "##0.00")
    Calculo_Descuentos_Ivas()
Endif

Endif
End

Public Sub VlbPeso_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        If vlbpeso.Value = "" Or vlbpeso.value = 0 Then
            vlbpeso.SetFocus
        Else
            VlbPrecio.SetFocus
        Endif
    Endif

End

Public Sub VlbPrecio_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        If VlbPrecio.Value = "" Or VlbPrecio.value = 0 Then
            VlbPrecio.SetFocus
        Else
            If vlbpeso.value > 0 And VlbPrecio.Value > 0 Then
                VlbImporte.Value = Format(vlbpeso.Value * VlbPrecio.Value, "##0.00")

            Endif
        Endif
    Endif

End

Public Sub CmbArticulo_Click()

    Dim Codigo As String
```

```
codigo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from articulo
where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
If Utilidades.Tabla_Vacia("select * from lotes where articulo='" & codigo &
"'" ) > 0 Then
    Utilidades.Llenar_Combobox("select * from lotes where articulo='" & codigo &
"'" , CmbLote, "codigolote")
    CmbLote.Text = ""
    CmbLote.Index = -1
Else
    CmbLote.Text = ""
    CmbLote.Clear
Endif
vlbpeso.Value = 0
VlbPrecio.Value = 0
VlbImporte.Value = 0
CmbLote.SetFocus
End

Public Sub CmbLote_Click()
    Dim Codigo As String
    Dim Sql As String
    codigo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from articulo
where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    If Utilidades.Tabla_Vacia("select cantidad from lotes where codigolote='" &
CmbLote.Text & "'" ) > 0 Then
        vlbpeso.Value = Utilidades.Obtener_Campo_Tabla("select cantidad from lotes
where codigolote='" & CmbLote.Text & "'", "cantidad")
        sql = "select precio from tarifaarticulo where articulo='" & codigo & "'" and
tarifa='" & Tarifa_Cliente & "'"
        If Utilidades.Tabla_Vacia(sql) > 0 Then
            VlbPrecio.value = Utilidades.Obtener_Campo_Tabla(sql, "precio")
        Endif
    Endif
    vlbpeso.SetFocus
End

Public Sub CmbLote_KeyRelease()
Dim Codigo As String
Dim Sql As String
codigo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from articulo
where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
If Key.Code = Key.Return Or Key.Code = Key.Enter Then
    If CmbLote.Text = "" Then
        CmbLote.SetFocus
    Else
        sql = "select precio from tarifaarticulo where articulo='" & codigo & "'"
and tarifa='" & Tarifa_Cliente & "'"
        If Utilidades.Tabla_Vacia(sql) > 0 Then
            VlbPrecio.value = Utilidades.Obtener_Campo_Tabla(sql, "precio")
        Endif
        vlbpeso.SetFocus
    Endif
Endif
End

Public Sub BtnInicio_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        GridLineasDetalle.Rows.UnselectAll
    End If
End Sub
```


Capítulo 2. CÓDIGO FUENTE

```
        Linea_Detalle = 0
        GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
    Endif

End

Public Sub BtnAnterior_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        If Linea_Detalle = 0 Then
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Else
            GridLineasDetalle.Rows.UnselectAll
            Linea_Detalle = Linea_Detalle - 1
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Endif
    Endif
End

Public Sub BtnSiguiente_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        If Linea_Detalle = GridLineasDetalle.Rows.Count - 1 Then
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Else
            GridLineasDetalle.Rows.UnselectAll
            Linea_Detalle = Linea_Detalle + 1
            GridLineasDetalle.Rows.Select(Linea_Detalle, 1)
        Endif
    Endif
End

Public Sub BtnFin_Click()

    If GridLineasDetalle.Rows.Count > 0 Then
        GridLineasDetalle.Rows.UnselectAll
        Linea_Detalle = GridLineasDetalle.Rows.Count - 1
        GridLineasDetalle.Rows.Select(Linea_Detalle, 1)

    Endif

End

'***** Funciones privadas*****
'
'*****
'
' Procedimiento para Llenar de datos el DataGrid LineasDetalle<br>
' Parametros:<br>
'     Tipo --> Tomara los siguientes valores<br>
'         A -- Albaran.<br>
'         F -- Factura.<br>
'     Sql --> Setncia Sql para llenar el Data Grid<br>
Private Sub Llenar_DataGrid_LineasDetalle(Sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridLineasDetalle.Rows.Count > 0 Then
        GridLineasDetalle.Rows.Remove(0, GridLineasDetalle.Rows.Count)
    Endif
End
```

Capítulo 2. CÓDIGO FUENTE

```
FMain.Conexion_Base_Datos.Close
FMain.Conexion_Base_Datos.Open
Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
Result_Trabajo.MoveFirst

If Tipo = "F" Then
    While Result_Trabajo.Available
        Numero_fila = GridLineasDetalle.Rows.Count
        GridLineasDetalle.Rows.Insert(Numero_fila, 1)
        GridLineasDetalle[Numero_fila, 0].Text = Result_Trabajo["L.numerofactura"]
        GridLineasDetalle[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
        GridLineasDetalle[Numero_fila, 2].Text = Result_Trabajo["L.precio"]
        GridLineasDetalle[Numero_fila, 3].Text = Result_Trabajo["L.peso"]
        GridLineasDetalle[Numero_fila, 4].Text = Result_Trabajo["L.importe"]
        GridLineasDetalle[Numero_fila, 5].Text = Result_Trabajo["L.lote"]
        GridLineasDetalle[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
        Result_Trabajo.MoveNext
    Wend
Else
    While Result_Trabajo.Available
        Numero_fila = GridLineasDetalle.Rows.Count
        GridLineasDetalle.Rows.Insert(Numero_fila, 1)
        GridLineasDetalle[Numero_fila, 0].Text = Result_Trabajo["L.numeroalbaran"]
        GridLineasDetalle[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
        GridLineasDetalle[Numero_fila, 2].Text = Result_Trabajo["L.precio"]
        GridLineasDetalle[Numero_fila, 3].Text = Result_Trabajo["L.peso"]
        GridLineasDetalle[Numero_fila, 4].Text = Result_Trabajo["L.importe"]
        GridLineasDetalle[Numero_fila, 5].Text = Result_Trabajo["L.lote"]
        GridLineasDetalle[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
        Result_Trabajo.MoveNext
    Wend
Endif
FMain.Conexion_Base_Datos.Close
GridLineasDetalle.Rows.Select(Linea_Detalle, 1)

End

'' Procedimiento para configurar Grid LineasDetalle
Private Sub Configurar_DataGrid_LineasDetalle()

    GridLineasDetalle.Columns.Count = 7
    GridLineasDetalle.Header = GridLineasDetalle.Both
    GridLineasDetalle.Columns[0].Title = "Numero"
    GridLineasDetalle.Columns[1].Title = "Linea"
    GridLineasDetalle.Columns[2].Title = "Precio"
    GridLineasDetalle.Columns[3].Title = "Peso"
    GridLineasDetalle.Columns[4].Title = "Importe"
    GridLineasDetalle.Columns[5].Title = "Lote"
    GridLineasDetalle.Columns[6].Title = "Articulo"
    GridLineasDetalle.Columns[0].Width = 100
    GridLineasDetalle.Columns[1].Width = 100
    GridLineasDetalle.Columns[2].Width = 100
    GridLineasDetalle.Columns[3].Width = 100
    GridLineasDetalle.Columns[4].Width = 100
    GridLineasDetalle.Columns[5].Width = 150
    GridLineasDetalle.Columns[6].Width = 200

End

'' Funcion para comprobar los datos de los lotes y las lineas de lotes.<br>
```

Capítulo 2. CÓDIGO FUENTE

```
' ' Parametros:<br>
' '     Accion Tomara los siguientes valores<br>
' '         NUEVO -- Comprobar los datos para crear nueva Factura.<br>
' '         DETALLES -- Comprobar los datos antes de añadir linea detallle.<br>
' ' Devuelve:<br>
' '     TRUE -- Datos corecctos<br>
' '     FALSE -- Datos erroneos
Private Function Comprobar_Datos_Factura(Accion As String) As Boolean

    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    If Accion = "NUEVO" Then
        sql = "select *from facturas where numerofactura='" & UCase(TxtNumero.Text)
    & " '"
        If TxtNumero.Text = "" Then
            Balloon.Error("Debe introducir Numero Facatura/Albaran", TxtNumero)
            Validar = False
            TxtNumero.SetFocus
        Else If Utilidades.Comprobar_existe_dato(Sql)
            Balloon.Error("Debe introducir Numero Facatura/Albaran ya exite",
TxtNumero)
            Validar = False
            TxtNumero.SetFocus
        Else If CmbCliente.Text = ""
            Balloon.Error("Debe elegir Cliente para crear Factura/Albaran",
CmbCliente)
            Validar = False
            CmbCliente.SetFocus
        Endif
    Else If accion = "DETALLES"
        If CmbArticulo.Text = "" Then
            Balloon.Error("Debe Elegir Articulo", CmbArticulo)
            Validar = False
        Else If CmbLote.Text = ""
            Balloon.Error("Debe Elegir o Introducir Numero de Lote", CmbLote)
            Validar = False
            CmbLote.SetFocus
        Else If vlbpeso.Value = 0
            Balloon.Error("Debe Introducir Peso", vlbpeso)
            Validar = False
            vlbpeso.SetFocus
        Else If VlbPrecio.Value = 0
            Balloon.Error("Debe Introducir Precio", VlbPrecio)
            Validar = False
            VlbPrecio.SetFocus
        Endif
    Endif

    Return Validar
End

' Procedimiento para limpiar los datos de Los Detalles Facturas
Private Sub Limpiar_Detalles_Facturas()

    CmbArticulo.index = -1
    CmbLote.text = ""
    CmbLote.Clear
    vlbpeso.Value = 0
```

Capítulo 2. CÓDIGO FUENTE

```
VlbPrecio.Value = 0
VlbImporte.Value = 0

End

' Añadimos la linea detalle a la factura y calculamos descuentos e ivas.
Private Sub Agrega_Linea_Detalle_Factura()

    Dim Sql As String
    Dim Codigo_articulo As String
    linea = linea + 1
    Codigo_articulo = Utilidades.Obtener_Campo_Tabla("select codigoarticulo from
articulo where nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    If tipo = "F" Then
        sql = "insert into
facturasdetalles(numerofactura,linea,precio,peso,importe,lote,articulo)
values('" &
        UCase(TxtNumero.Text) & "','" & linea & "','" & CFloat(VlbPrecio.Value) & "','" &
CFloat(vlbpeso.Value) & "','" & CFloat(Vlbimporte.Value) &
        "','" & UCase(CmbLote.text) & "','" & Codigo_articulo & "')"
        Utilidades.In_Act_Bor(sql)
        sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numerofactura='" & UCase(TxtNumero.Text) & "'"
        Llenar_DataGrid_LineasDetalle(sql)
    Endif
    If Tipo = "A" Then
        sql = "insert into
albaranesdetalle(numeroalbaran,linea,precio,peso,importe,lote,articulo)
values('" &
        UCase(TxtNumero.Text) & "','" & linea & "','" & CFloat(VlbPrecio.Value) & "','" &
CFloat(vlbpeso.Value) & "','" & CFloat(Vlbimporte.Value) &
        "','" & UCase(CmbLote.text) & "','" & Codigo_articulo & "')"
        Utilidades.In_Act_Bor(sql)
        sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numeroalbaran='" & UCase(TxtNumero.Text) & "'"
        Llenar_DataGrid_LineasDetalle(sql)
    Endif
    VlbImporteBase.Value = Format(VlbImporteBase.Value + VlbImporte.Value,
"##0.00")
    Calculo_Descuentos_Ivas()
End

'' Procedimiento para calcular Descuentos e Iva cuando añadimos o borramos una
linea detalle.

Private Sub Calculo_Descuentos_Ivas()

    Dim Descuento_PP As Float
    Dim Descuento_ESP As Float
    Dim Iva As Float
    Dim RE As Float
    Dim AuxBase As Float
    If VlbDescuentoESP.Value > 0 Then
        Descuento_ESP = (VlbImporteBase.Value * VlbDescuentoESP.Value) / 100
    Else
```

```

    Descuento_ESP = 0
Endif
If VlbDescuentoPP.Value > 0 Then
    Descuento_PP = (VlbImporteBase.Value * VlbDescuentoPP.Value) / 100
Else
    Descuento_PP = 0
Endif
VlbImporteDescuentoESP.Value = Format(Descuento_ESP, "##0.00")
VlbImporteDescuentoPP.Value = Format(Descuento_PP, "##0.00")
auxbase = VlbImporteBase.Value - Descuento_ESP - Descuento_PP
If VlbIva.Value > 0 Then
    iva = (auxbase * VlbIva.value) / 100
Else
    iva = 0
Endif
If VlbRecargo.Value > 0 Then
    re = (auxbase * VlbRecargo.value) / 100
Else
    re = 0
Endif
VlbImporteIva.Value = Format(iva, "##0.00")
VlbImporteRecargo.Value = Format(re, "##0.00")
LblTotalFactura.Text = Format(auxbase + iva + re, "##0.00")
End
'' Graba la factura.
Private Sub Grabar_Factura()

    Dim Codigo_Vendedor As String
    Dim Codigo_Cliente As String
    Dim Sql As String
    Dim Cobrado_Pagada As String
    Dim Contabilizada_Facturado As String
    If ChkCobradaPagado.Value = True Then
        Cobrado_Pagada = "S"
    Else
        Cobrado_Pagada = "N"
    Endif
    If ChkContabilizada.value = True Then
        Contabilizada_Facturado = "S"
    Else
        Contabilizada_Facturado = "N"
    Endif
    Codigo_Cliente = Utilidades.Obtener_Campo_Tabla("select codigocliente from
clientes where razonsocial='" & CmbCliente.Text & "'", "codigocliente")
    Codigo_Vendedor = Utilidades.Obtener_Campo_Tabla("select codigovendedor from
vendedor where nombre='" & CmbVendedor.Text & "'", "codigovendedor")
    If tipo = "F" Then

        sql = "update facturas set fecha='" &
Utilidades.Fecha_Correcta(DteFechaCreacion.value) & "',hora='" & DteHora.value &
"',base=" & VlbImporteBase.Value &
        ",iva=" & VlbIva.Value & ",recargo=" & VlbRecargo.Value & ",importeiva=" &
VlbImporteIva.Value & ",importerecargo=" & VlbImporteRecargo.Value &
        ",fechacobro='" & Utilidades.Fecha_Correcta(DteFechaCobro.Value) &
        "',descuentopp=" & VlbDescuentoPP.Value & ",importedescuentopp=" &
VlbImporteDescuentoPP.Value &
        ",descuentoesp=" & VlbDescuentoESP.Value & ",importedescuentoesp=" &
VlbImporteDescuentoESP.Value & ",cobrada='" & Cobrado_Pagada &

```

```
        "','formapago='" & CmbFormaPago.Text & "','vencimiento='" &
Utilidades.Fecha_Correcta(DteFechavencimiento.Value) & "','contabilizada='" &
Contabilizada_Facturado &
        "','total='" & LblTotalFactura.Text & "','cliente='" & Codigo_Cliente &
        "','vendedor='" & Codigo_Vendedor & "' where numerofactura='" & TxtNumero.Text &
        ""

        Utilidades.In_Act_Bor(sql)
        Me.Close
    Endif

    If Tipo = "A" Then
        sql = "update albaranes set fecha='" &
Utilidades.Fecha_Correcta(DteFechaCreacion.value) & "','hora='" & DteHora.value &
        "','base='" & VlbImporteBase.Value &
        "','iva='" & VlbIva.Value & "','recargo='" & VlbRecargo.Value & "','importeiva='" &
VlbImporteIva.Value & "','importerecargo='" & VlbImporteRecargo.Value &
        "','fechacobro='" & Utilidades.Fecha_Correcta(DteFechaCobro.Value) &
        "','descuentopp='" & VlbDescuentoPP.Value & "','importedescentopp='" &
VlbImporteDescuentoPP.Value &
        "','descuentoesp='" & VlbDescuentoESP.Value & "','importedescentoesp='" &
VlbImporteDescuentoESP.Value & "','pagado='" & Cobrado_Pagada &
        "','formapago='" & CmbFormaPago.Text & "','vencimiento='" &
Utilidades.Fecha_Correcta(DteFechavencimiento.Value) & "','facturado='" &
Contabilizada_Facturado &
        "','total='" & LblTotalFactura.Text & "','cliente='" & Codigo_Cliente &
        "','vendedor='" & Codigo_Vendedor & "' where numeroalbaran='" & TxtNumero.Text &
        ""

        Utilidades.In_Act_Bor(sql)
        Me.Close
    Endif

End

'' Visualizamos los datos de la cabecera de la Factura o Albaran.
Private Sub Visualizar_Datos_Factura_Albaran(sql As String)

    Dim Result_Trabajo As Result
    Dim Cobrada As String
    Dim Contabilizada As String
    Dim Dia As String
    Dim Mes As String
    Dim Ano As String
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst

    If Tipo = "F" Then
        While Result_Trabajo.Available

            Dia = Mid(Result_Trabajo["F.fecha"], 1, 2)
            Mes = Mid(Result_Trabajo["F.fecha"], 4, 2)
            Ano = Mid(Result_Trabajo["F.fecha"], 7, 4)
            DteFechaCreacion.Value = Date(ano, mes, dia)

            DteHora.Value = Result_Trabajo["F.hora"]
            VlbImporteBase.Value = Result_Trabajo["F.base"]
            VlbIva.Value = Result_Trabajo["F.iva"]
            VlbRecargo.Value = Result_Trabajo["F.recargo"]
```

```
VlbImporteIva.Value = Result_Trabajo["F.importeiva"]
VlbImporteRecargo.Value = Result_Trabajo["F.importerecargo"]

Dia = Mid(Result_Trabajo["F.fechacobro"], 1, 2)
Mes = Mid(Result_Trabajo["F.fechacobro"], 4, 2)
Año = Mid(Result_Trabajo["F.fechacobro"], 7, 4)
DteFechaCobro.Value = Date(año, mes, día)

VlbDescuentoPP.Value = Result_Trabajo["F.descuentopp"]
VlbImporteDescuentoPP.Value = Result_Trabajo["F.importedescuentopp"]
VlbDescuentoESP.Value = Result_Trabajo["F.descuentoesp"]
VlbImporteDescuentoESP.Value = Result_Trabajo["F.importedescuentoesp"]
Cobrada = Result_Trabajo["F.cobrada"]
If Cobrada = "S" Then
    ChkCobradaPagado.Value = True
Else
    ChkCobradaPagado.Value = False
Endif
CmbFormaPago.Text = Result_Trabajo["F.formapago"]

Dia = Mid(Result_Trabajo["F.vencimiento"], 1, 2)
Mes = Mid(Result_Trabajo["F.vencimiento"], 4, 2)
Año = Mid(Result_Trabajo["F.vencimiento"], 7, 4)
DteFechavencimiento.Value = Date(año, mes, día)

Contabilizada = Result_Trabajo["F.contabilizada"]
If Contabilizada = "S" Then
    ChkContabilizada.Value = True
Else
    ChkContabilizada.Value = False
Endif
CmbCliente.Text = Result_Trabajo["C.razonsocial"]
CmbVendedor.Text = Result_Trabajo["V.nombre"]
Result_Trabajo.MoveNext
Wend
Else
    While Result_Trabajo.Available
        Dia = Mid(Result_Trabajo["F.fecha"], 1, 2)
        Mes = Mid(Result_Trabajo["F.fecha"], 4, 2)
        Año = Mid(Result_Trabajo["F.fecha"], 7, 4)
        DteFechaCreacion.Value = Date(año, mes, día)
        DteHora.Value = Result_Trabajo["F.hora"]
        VlbImporteBase.Value = Result_Trabajo["F.base"]
        VlbIva.Value = Result_Trabajo["F.iva"]
        VlbRecargo.Value = Result_Trabajo["F.recargo"]
        VlbImporteIva.Value = Result_Trabajo["F.importeiva"]
        VlbImporteRecargo.Value = Result_Trabajo["F.importerecargo"]
        Dia = Mid(Result_Trabajo["F.fechacobro"], 1, 2)
        Mes = Mid(Result_Trabajo["F.fechacobro"], 4, 2)
        Año = Mid(Result_Trabajo["F.fechacobro"], 7, 4)
        DteFechaCobro.Value = Date(año, mes, día)
        VlbDescuentoPP.Value = Result_Trabajo["F.descuentopp"]
        VlbImporteDescuentoPP.Value = Result_Trabajo["F.importedescuentopp"]
        VlbDescuentoESP.Value = Result_Trabajo["F.descuentoesp"]
        VlbImporteDescuentoESP.Value = Result_Trabajo["F.importedescuentoesp"]
        Cobrada = Result_Trabajo["F.pagado"]
        If Cobrada = "S" Then
            ChkCobradaPagado.Value = True
        Else
```

```
        ChkCobradaPagado.Value = False
    Endif
    CmbFormaPago.Text = Result_Trabajo["F.formapago"]
    Dia = Mid(Result_Trabajo["F.vencimiento"], 1, 2)
    Mes = Mid(Result_Trabajo["F.vencimiento"], 4, 2)
    Ano = Mid(Result_Trabajo["F.vencimiento"], 7, 4)
    DteFechavencimiento.Value = Date(ano, mes, dia)
    Contabilizada = Result_Trabajo["F.facturado"]
    If Contabilizada = "S" Then
        ChkContabilizada.Value = True
    Else
        ChkContabilizada.Value = False
    Endif

    Result_Trabajo.MoveNext
Wend
Endif
FMain.Conexion_Base_Datos.Close
End
```

2.2.3.3 *Frm_Visualizar_Factura_Albaran.class*

```
Public Factura_Albaran_Editar As String ' Variable para pasar al formulario de
edicion la factura y albaran seleccionada.
Public Linea_Cabeceras As Integer ' Variable para controlar la linea de
posicion del GridCabeceras y poder movernos con flechas.
Public Linea_Detalles As Integer ' Variable para controlar la linea de posicion
del GridDetalles y poder movernos con flechas.
Public Uso_Filtro As Boolean 'Controlamos que estamos filtrando datos para pdoer
controlar ordenar por columna.
Public Actualiza_datos As Boolean

Public Sub BtnSalirVisualizar_Click()

    Me.Close

End

Public Sub BtnBorrar_Click()

    Dim sql As String
    Dim respuesta As Integer

    If Comprobar_Campos() Then

        If RdbFacturas.Value = True Then
            Respuesta = Message.Delete("Desea Borrar Factura :<b> " &
UCase(CmbFacturaAlbaran.Text) & "</b>", "Si", "No")
        Else
            Respuesta = Message.Delete("Desea Borrar Albaran :<b> " &
UCase(CmbFacturaAlbaran.Text) & "</b>", "Si", "No")
        Endif
        If respuesta = 1 Then
            ' Cabecera de Factura / Albaran
            If RdbFacturas.Value = True Then
                sql = "delete from facturas where numerofactura='" &
UCase(CmbFacturaAlbaran.Text) & "'"
            Endif
        Endif
    Endif
End Sub
```



```

        Utilidades.In_Act_Bor(Sql)
    ' linea Detalles
        sql = "delete from facturasdetalles where numerofactura='" &
UCase(CmbFacturaAlbaran.Text) & "'"
        Utilidades.In_Act_Bor(Sql)
        RdbFacturas_Click()
    Else
        sql = "delete from albaranes where numeroalbaran='" &
UCase(CmbFacturaAlbaran.Text) & "'"
        Utilidades.In_Act_Bor(Sql)
    ' linea Detalles
        sql = "delete from albaranesdetalle where numeroalbaran='" &
UCase(CmbFacturaAlbaran.Text) & "'"
        Utilidades.In_Act_Bor(Sql)
        RdbAlbaranes_Click()
    Endif
Endif

Endif

End

Public Sub BtnEditar_Click()

    Dim Sql As String
    If Comprobar_Campos() Then
        Factura_Albaran_Editar = CmbFacturaAlbaran.Text
        Frm_Edicion_Factura_Albaranes.ShowModal()
        If RdbFacturas.Value = True Then
            sql = "select
F.numerofactura,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.cobrada,F.formapago,F.vencimiento,F.con
tabilizada,F.total,C.razonsocial,V.nombre from facturas F left join clientes C
on" &

" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor"
            Llenar_DataGrid_Cabeceras(sql, "F")
            sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & GridCabeceras[0, 0].Text & "'"
            Llenar_DataGrid_Detalles(sql, "F")
        Else
            sql = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.pagado,F.formapago,F.vencimiento,F.fact
urado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor"
            Llenar_DataGrid_Cabeceras(sql, "A")
            sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & GridCabeceras[0, 0].Text & "'"
            Llenar_DataGrid_Detalles(sql, "A")
        End If
    End If
End Sub

```

Capítulo 2. CÓDIGO FUENTE

```
        Endif
        CmbFacturaAlbaran.Index = -1
    Endif

End

Public Sub BtnGenerar_Click()

    If Comprobar_Campos() Then
        Utilidades.Generar_Factura_Albaran(UCCase(CmbFacturaAlbaran.Text))
    Endif
End

Public Sub RdbFacturas_Click()
    Dim sql As String
    If Utilidades.Tabla_Vacia("select * from facturas") > 0 Then
        LblNumero.text = "Nº Factura :"
        FrameDetalles.Text = "Factura Detalle"
        FrameFacturas.Text = "Facturas"
        FrameFiltrar.Text = "Filtrar Facturas"
        Utilidades.Llenar_Combobox("select * from facturas", CmbFacturaAlbaran,
"numerofactura")
        CmbFacturaAlbaran.Text = ""
        CmbFacturaAlbaran.Index = -1
        Configurar_DataGrid_Cabeceras_Facturas()
        sql = "select
F.numerofactura,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.cobrada,F.formapago,F.vencimiento,F.con
tabilizada,F.total,C.razonsocial,V.nombre from facturas F left join clientes C
on" &
        " F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigoendedor order by F.numerofactura"
        Llenar_DataGrid_Cabeceras(sql, "F")
        sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numerofactura='" & GridCabeceras[0, 0].Text & "'"
        Llenar_DataGrid_Detalles(sql, "F")
    Else
        CmbFacturaAlbaran.Clear
        If GridCabeceras.Rows.Count > 0 Then
            GridCabeceras.Rows.Remove(0, GridCabeceras.Rows.Count)
        End If
        If GridDetalles.Rows.Count > 0 Then
            GridDetalles.Rows.Remove(0, GridDetalles.Rows.Count)
        End If
    End If
End

Public Sub RdbAlbaranes_Click()
    Dim sql As String
    If Utilidades.Tabla_Vacia("select * from albaranes") > 0 Then
        LblNumero.text = "Nº Albaran :"
        FrameDetalles.Text = "Albaran Detalle"
        FrameFacturas.Text = "Albaranes"
        FrameFiltrar.Text = "Filtrar Albaranes"
```

Capítulo 2. CÓDIGO FUENTE

```
    Utilidades.Llenar_Combobox("select * from albaranes", CmbFacturaAlbaran,
"numeroalbaran")
    CmbFacturaAlbaran.Text = ""
    CmbFacturaAlbaran.Index = -1
    Configurar_DataGrid_Cabeceras_Albaranes()
    sql = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.pagado,F.formapago,F.vencimiento,F.fact
urado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigo vendedor order by F.numeroalbaran"
    Llenar_DataGrid_Cabeceras(sql, "A")
    sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & GridCabeceras[0, 0].Text & "'"
    Llenar_DataGrid_Detalles(sql, "A")
Else
    CmbFacturaAlbaran.Clear
    If GridCabeceras.Rows.Count > 0 Then
        GridCabeceras.Rows.Remove(0, GridCabeceras.Rows.Count)
    End If
    If GridDetalles.Rows.Count > 0 Then
        GridDetalles.Rows.Remove(0, GridDetalles.Rows.Count)
    End If
End If
End If
End

Public Sub Form_Open()
    Me.Title = "VISUALIZACION DE FACTURAS/ALBARANES"
    Configurar_Objeto()
End

'*****INICIO CABECERAS*****
'
'*****
Public Sub BtnInicioCabecera_Click()
    Dim Sql As String
    If GridCabeceras.Rows.Count > 0 Then
        GridCabeceras.Rows.UnselectAll
        Linea_Cabeceras = 0
        GridCabeceras.Rows.Select(Linea_Cabeceras, 1)
        If RdbFacturas.Value = True Then
            sql = "select
L.numero de factura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numero de factura='" & GridCabeceras[Linea_Cabeceras, 0].Text &
""
            Llenar_DataGrid_Detalles(sql, "F")
        Else
            sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & GridCabeceras[Linea_Cabeceras, 0].Text &
""
            Llenar_DataGrid_Detalles(sql, "A")
        End If
    End If
End Sub
```

```
        Endif
    Endif
    CmbFacturaAlbaran.Text = GridCabeceras[Linea_Cabeceras, 0].Text
End

Public Sub BtnAnteriorCabecera_Click()
    Dim Sql As String
    If GridCabeceras.Rows.Count > 0 Then
        If Linea_Cabeceras = 0 Then
            GridCabeceras.Rows.Select(Linea_Cabeceras, 1)
        Else
            GridCabeceras.Rows.UnselectAll
            Linea_Cabeceras = Linea_Cabeceras - 1
            GridCabeceras.Rows.Select(Linea_Cabeceras, 1)
        Endif
        If RdbFacturas.Value = True Then
            sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & GridCabeceras[Linea_Cabeceras, 0].Text &
""
            Llenar_DataGrid_Detalles(sql, "F")
        Else
            sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & GridCabeceras[Linea_Cabeceras, 0].Text &
""
            Llenar_DataGrid_Detalles(sql, "A")
        Endif
    Endif
    CmbFacturaAlbaran.Text = GridCabeceras[Linea_Cabeceras, 0].Text
End

Public Sub BtnSiguienteCabecera_Click()
    Dim Sql As String
    If GridCabeceras.Rows.Count > 0 Then
        If Linea_Cabeceras = GridCabeceras.Rows.Count - 1 Then
            GridCabeceras.Rows.Select(Linea_Cabeceras, 1)
        Else
            GridCabeceras.Rows.UnselectAll
            Linea_Cabeceras = Linea_Cabeceras + 1
            GridCabeceras.Rows.Select(Linea_Cabeceras, 1)
        Endif
        If RdbFacturas.Value = True Then
            sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & GridCabeceras[Linea_Cabeceras, 0].Text &
""
            Llenar_DataGrid_Detalles(sql, "F")
        Else
            sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numeroalbaran='" & GridCabeceras[Linea_Cabeceras, 0].Text &
""
            Llenar_DataGrid_Detalles(sql, "A")
        Endif
    Endif
End
```

```

        Endif
    Endif
    CmbFacturaAlbaran.Text = GridCabeceras[Linea_Cabeceras, 0].Text
End

Public Sub BtnFinCabecera_Click()
    Dim Sql As String
    If GridCabeceras.Rows.Count > 0 Then
        GridCabeceras.Rows.UnselectAll
        Linea_Cabeceras = GridCabeceras.Rows.Count - 1
        GridCabeceras.Rows.Select(Linea_Cabeceras, 1)
        If RdbFacturas.Value = True Then
            sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
            " where L.numerofactura='" & GridCabeceras[Linea_Cabeceras, 0].Text &
            "'"
            Llenar_DataGrid_Detalles(sql, "F")
        Else
            sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
            " where L.numeroalbaran='" & GridCabeceras[Linea_Cabeceras, 0].Text &
            "'"
            Llenar_DataGrid_Detalles(sql, "A")
        Endif
    Endif
    CmbFacturaAlbaran.Text = GridCabeceras[Linea_Cabeceras, 0].Text
End

Public Sub GridCabeceras_Click()
    Dim Numero_fila As Integer
    Dim sql As String
    If GridCabeceras.Rows.Selection.Count > 0 Then
        Numero_fila = GridCabeceras.Rows.Selection[0]
        CmbFacturaAlbaran.Text = GridCabeceras[Numero_fila, 0].Text
        Linea_Cabeceras = Numero_fila
        If RdbFacturas.Value = True Then
            sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
            " where L.numerofactura='" & GridCabeceras[Linea_Cabeceras, 0].Text &
            "'"
            Llenar_DataGrid_Detalles(sql, "F")
        Else
            sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
            " where L.numeroalbaran='" & GridCabeceras[Linea_Cabeceras, 0].Text &
            "'"
            Llenar_DataGrid_Detalles(sql, "A")
        Endif
    End If
End

'''*****FIN CABECERAS *****
''
'''*****

```

Capítulo 2. CÓDIGO FUENTE

```
'*****INICIO DETALLES*****'  
'  
'*****  
  
Public Sub BtnInicioDetalles_Click()  
  
    If GridDetalles.Rows.Count > 0 Then  
        GridDetalles.Rows.UnselectAll  
        Linea_Detalles = 0  
        GridDetalles.Rows.Select(Linea_Detalles, 1)  
    Endif  
  
End  
  
Public Sub BtnAnteriorDetalles_Click()  
  
    If GridDetalles.Rows.Count > 0 Then  
        If Linea_Detalles = 0 Then  
            GridDetalles.Rows.Select(Linea_Detalles, 1)  
        Else  
            GridDetalles.Rows.UnselectAll  
            Linea_Detalles = Linea_Detalles - 1  
            GridDetalles.Rows.Select(Linea_Detalles, 1)  
        Endif  
    Endif  
  
End  
  
Public Sub BtnSiguienteDetalles_Click()  
  
    If GridDetalles.Rows.Count > 0 Then  
        If Linea_Detalles = GridDetalles.Rows.Count - 1 Then  
            GridDetalles.Rows.Select(Linea_Detalles, 1)  
        Else  
            GridDetalles.Rows.UnselectAll  
            Linea_Detalles = Linea_Detalles + 1  
            GridDetalles.Rows.Select(Linea_Detalles, 1)  
        Endif  
    Endif  
  
End  
  
Public Sub BtnFinDetalles_Click()  
  
    If GridDetalles.Rows.Count > 0 Then  
        GridDetalles.Rows.UnselectAll  
        Linea_Detalles = GridDetalles.Rows.Count - 1  
        GridDetalles.Rows.Select(Linea_Detalles, 1)  
    Endif  
  
End  
  
'*****FIN DETALLES*****'  
'  
'*****
```

Capítulo 2. CÓDIGO FUENTE

```
''*****INICIO FILTROS *****''
''*****
Public Sub ChkFechaCreacion_Click()

    If ChkFechaCreacion.Value = True Then
        PanelFCreacion.Enabled = True
        DteFICreacion.Value = Date(Year(Now), Month(Now), Day(Now))
        DteFFCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    Else
        PanelFCreacion.Enabled = False
    Endif

End

Public Sub ChkCliente_Click()

    If ChkCliente.Value = True Then
        PanelCliente.Enabled = True
    Else
        PanelCliente.Enabled = False
    Endif

End

Public Sub ChkVendedor_Click()

    If ChkVendedor.Value = True Then
        PanelVendedor.Enabled = True
    Else
        PanelVendedor.Enabled = False
    Endif

End

Public Sub BtnFiltrar_Click()

    Dim Validar As Boolean
    Validar = True
    Uso_filtro = True
    If ChkFechaCreacion.value = False And Chkvendedor.Value = False And
Chkcliente.value = False Then
        validar = False
        Balloon.Error("Debe elegir un filtro.", PanelFiltros)
    Else If Chkcliente.value = True And Cmbfcliente.Text = ""
        validar = False
        Balloon.Error("Si activa el filtro de los Clientes debe seleccionar el
Cliente para filtrar.", Cmbfcliente)
    Else If Chkvendedor.value = True And Cmbfvendedor.Text = ""
        validar = False
        Balloon.Error("Si activa el filtro de los Vendedores debe seleccionar el
Vendedor para filtrar.", Cmbfvendedor)
    Endif
    If validar Then
        Ejecutar_Filtros()
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
End

Public Sub BtnQuitarFiltro_Click()

    Quitar_Filtros()

End

'''*****FIN FILTROS *****'''

'''***** Funciones privadas*****'''

''' Procedimiento para configurar los objetos del Formulario.
Private Sub Configurar_Objeto()

    CmbFCliente.Clear
    Utilidades.Llenar_Combobox("select * from clientes", CmbFCliente,
"razonsocial")
    CmbFCliente.Index = -1
    CmbFVendedor.Clear
    Utilidades.Llenar_Combobox("select * from vendedor", CmbFVendedor, "nombre")
    CmbFVendedor.Index = -1
    Utilidades.Llenar_Combobox("select * from facturas", CmbFacturaAlbaran,
"numerofactura")
    CmbFacturaAlbaran.Index = -1
    Configurar_DataGrid_Cabeceras_Facturas()
    Configurar_DataGrid_Detalles()
    Uso_Filtro = False
    Actualiza_datos = False
    Linea_Cabeceras = 0
    Linea_Detalles = 0

    If Utilidades.Tabla_Vacia("select * from facturas") > 0 Then
        RdbFacturas.Value = True
    Else If Utilidades.Tabla_Vacia("select * from albaranes") > 0
        RdbAlbaranes.Value = True
    Endif
End

''' Procedimiento para configurar Grid LineasDetalle
Private Sub Configurar_DataGrid_Detalles()

    GridDetalles.Columns.Count = 7
    GridDetalles.Header = GridDetalles.Both
    GridDetalles.Columns[0].Title = "Numero"
    GridDetalles.Columns[1].Title = "Linea"
    GridDetalles.Columns[2].Title = "Precio"
    GridDetalles.Columns[3].Title = "Peso"
    GridDetalles.Columns[4].Title = "Importe"
    GridDetalles.Columns[5].Title = "Lote"
    GridDetalles.Columns[6].Title = "Articulo"
    GridDetalles.Columns[0].Width = 100
    GridDetalles.Columns[1].Width = 100
    GridDetalles.Columns[2].Width = 100
```


Capítulo 2. CÓDIGO FUENTE

```
GridDetalles.Columns[3].Width = 100
GridDetalles.Columns[4].Width = 100
GridDetalles.Columns[5].Width = 150
GridDetalles.Columns[6].Width = 200

End

'' Configuracion de las cabeceras para las facturas
Private Sub Configurar_DataGrid_Cabeceras_Facturas()

    GridCabeceras.Columns.Count = 20
    GridCabeceras.Header = GridCabeceras.Both
    GridCabeceras.Columns[0].Title = "Numero"
    GridCabeceras.Columns[1].Title = "Fecha"
    GridCabeceras.Columns[2].Title = "Hora"
    GridCabeceras.Columns[3].Title = "Base"
    GridCabeceras.Columns[4].Title = "Iva"
    GridCabeceras.Columns[5].Title = "Recargo"
    GridCabeceras.Columns[6].Title = "Importe Iva"
    GridCabeceras.Columns[7].Title = "Importe RE"
    GridCabeceras.Columns[8].Title = "Fecha Cobro"
    GridCabeceras.Columns[9].Title = "Descuento PP"
    GridCabeceras.Columns[10].Title = "Importe DPP"
    GridCabeceras.Columns[11].Title = "Descuento ESP"
    GridCabeceras.Columns[12].Title = "Importe DESP"
    GridCabeceras.Columns[13].Title = "Cobrada"
    GridCabeceras.Columns[14].Title = "Forma Pago"
    GridCabeceras.Columns[15].Title = "Vencimiento"
    GridCabeceras.Columns[16].Title = "Contabilizada"
    GridCabeceras.Columns[17].Title = "Total"
    GridCabeceras.Columns[18].Title = "Cliente"
    GridCabeceras.Columns[19].Title = "Vendedor"
    GridCabeceras.Columns[0].Width = 100
    GridCabeceras.Columns[1].Width = 100
    GridCabeceras.Columns[2].Width = 100
    GridCabeceras.Columns[3].Width = 100
    GridCabeceras.Columns[4].Width = 100
    GridCabeceras.Columns[5].Width = 150
    GridCabeceras.Columns[6].Width = 200
    GridCabeceras.Columns[7].Width = 200
    GridCabeceras.Columns[8].Width = 200
    GridCabeceras.Columns[9].Width = 200
    GridCabeceras.Columns[10].Width = 200
    GridCabeceras.Columns[11].Width = 200
    GridCabeceras.Columns[12].Width = 200
    GridCabeceras.Columns[13].Width = 200
    GridCabeceras.Columns[14].Width = 200
    GridCabeceras.Columns[15].Width = 200
    GridCabeceras.Columns[16].Width = 200
    GridCabeceras.Columns[17].Width = 150
    GridCabeceras.Columns[18].Width = 200
    GridCabeceras.Columns[19].Width = 200

End

'' Configuracion de las cabeceras para las albaranes
Private Sub Configurar_DataGrid_Cabeceras_Albaranes()

    GridCabeceras.Columns.Count = 20
    GridCabeceras.Header = GridCabeceras.Both
```

```

GridCabeceras.Columns[0].Title = "Numero"
GridCabeceras.Columns[1].Title = "Fecha"
GridCabeceras.Columns[2].Title = "Hora"
GridCabeceras.Columns[3].Title = "Base"
GridCabeceras.Columns[4].Title = "Iva"
GridCabeceras.Columns[5].Title = "Recargo"
GridCabeceras.Columns[6].Title = "Importe Iva"
GridCabeceras.Columns[7].Title = "Importe RE"
GridCabeceras.Columns[8].Title = "Fecha Cobro"
GridCabeceras.Columns[9].Title = "Descuento PP"
GridCabeceras.Columns[10].Title = "Importe DPP"
GridCabeceras.Columns[11].Title = "Descuento ESP"
GridCabeceras.Columns[12].Title = "Importe DESP"
GridCabeceras.Columns[13].Title = "Pagado"
GridCabeceras.Columns[14].Title = "Forma Pago"
GridCabeceras.Columns[15].Title = "Vencimiento"
GridCabeceras.Columns[16].Title = "Facturado"
GridCabeceras.Columns[17].Title = "Total"
GridCabeceras.Columns[18].Title = "Cliente"
GridCabeceras.Columns[19].Title = "Vendedor"
GridCabeceras.Columns[0].Width = 100
GridCabeceras.Columns[1].Width = 100
GridCabeceras.Columns[2].Width = 100
GridCabeceras.Columns[3].Width = 100
GridCabeceras.Columns[4].Width = 100
GridCabeceras.Columns[5].Width = 150
GridCabeceras.Columns[6].Width = 200
GridCabeceras.Columns[7].Width = 200
GridCabeceras.Columns[8].Width = 200
GridCabeceras.Columns[9].Width = 200
GridCabeceras.Columns[10].Width = 200
GridCabeceras.Columns[11].Width = 200
GridCabeceras.Columns[12].Width = 200
GridCabeceras.Columns[13].Width = 200
GridCabeceras.Columns[14].Width = 200
GridCabeceras.Columns[15].Width = 200
GridCabeceras.Columns[16].Width = 200
GridCabeceras.Columns[17].Width = 150
GridCabeceras.Columns[18].Width = 200
GridCabeceras.Columns[19].Width = 200
End

'' Procedimiento para Llenar de datos el DataGrid Detalles<br>
'' Parametros:<br>
''     Tipo --> Tomara los siguientes valores<br>
''         A -- Albaran.<br>
''         F -- Factura.<br>
''     Sql --> Setncia Sql para llenar el Data Grid<br>
Private Sub Llenar_DataGrid_Detalles(Sql As String, Tipo As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridDetalles.Rows.Count > 0 Then
        GridDetalles.Rows.Remove(0, GridDetalles.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst

```

```

If Tipo = "F" Then
    While Result_Trabajo.Available
        Numero_fila = GridDetalles.Rows.Count
        GridDetalles.Rows.Insert(Numero_fila, 1)
        GridDetalles[Numero_fila, 0].Text = Result_Trabajo["L.numeroofactura"]
        GridDetalles[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
        GridDetalles[Numero_fila, 2].Text = Result_Trabajo["L.precio"]
        GridDetalles[Numero_fila, 3].Text = Result_Trabajo["L.peso"]
        GridDetalles[Numero_fila, 4].Text = Result_Trabajo["L.importe"]
        GridDetalles[Numero_fila, 5].Text = Result_Trabajo["L.lote"]
        GridDetalles[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
        Result_Trabajo.MoveNext
    Wend
Else
    While Result_Trabajo.Available
        Numero_fila = GridDetalles.Rows.Count
        GridDetalles.Rows.Insert(Numero_fila, 1)
        GridDetalles[Numero_fila, 0].Text = Result_Trabajo["L.numeroalbaran"]
        GridDetalles[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
        GridDetalles[Numero_fila, 2].Text = Result_Trabajo["L.precio"]
        GridDetalles[Numero_fila, 3].Text = Result_Trabajo["L.peso"]
        GridDetalles[Numero_fila, 4].Text = Result_Trabajo["L.importe"]
        GridDetalles[Numero_fila, 5].Text = Result_Trabajo["L.lote"]
        GridDetalles[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
        Result_Trabajo.MoveNext
    Wend
Endif
FMain.Conexion_Base_Datos.Close
GridDetalles.Rows.Select(0, 1)

End

''' Procedimiento para Llenar de datos el DataGridView Cabeceras<br>
''' Parametros:<br>
''' Tipo --> Tomara los siguientes valores<br>
''' A -- Albaran.<br>
''' F -- Factura.<br>
''' Sql --> Setncia Sql para llenar el Data Grid<br>
Private Sub Llenar_DataGrid_Cabeceras(Sql As String, Tipo As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridCabeceras.Rows.Count > 0 Then
        GridCabeceras.Rows.Remove(0, GridCabeceras.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst

    If Tipo = "F" Then
        While Result_Trabajo.Available
            Numero_fila = GridCabeceras.Rows.Count
            GridCabeceras.Rows.Insert(Numero_fila, 1)
            GridCabeceras[Numero_fila, 0].Text = Result_Trabajo["F.numeroofactura"]
            GridCabeceras[Numero_fila, 1].Text = Result_Trabajo["F.fecha"]
            GridCabeceras[Numero_fila, 2].Text = Result_Trabajo["F.hora"]
            GridCabeceras[Numero_fila, 3].Text = Result_Trabajo["F.base"]

```

```

        GridCabeceras[Numero_fila, 4].Text = Result_Trabajo["F.iva"]
        GridCabeceras[Numero_fila, 5].Text = Result_Trabajo["F.recargo"]
        GridCabeceras[Numero_fila, 6].Text = Result_Trabajo["F.importeiva"]
        GridCabeceras[Numero_fila, 7].Text = Result_Trabajo["F.importerecargo"]
        GridCabeceras[Numero_fila, 8].Text = Result_Trabajo["F.fechacobro"]
        GridCabeceras[Numero_fila, 9].Text = Result_Trabajo["F.descuentopp"]
        GridCabeceras[Numero_fila, 10].Text =
Result_Trabajo["F.importedescuentopp"]
        GridCabeceras[Numero_fila, 11].Text = Result_Trabajo["F.descuentoesp"]
        GridCabeceras[Numero_fila, 12].Text =
Result_Trabajo["F.importedescuentoesp"]
        GridCabeceras[Numero_fila, 13].Text = Result_Trabajo["F.cobrada"]
        GridCabeceras[Numero_fila, 14].Text = Result_Trabajo["F.formapago"]
        GridCabeceras[Numero_fila, 15].Text = Result_Trabajo["F.vencimiento"]
        GridCabeceras[Numero_fila, 16].Text = Result_Trabajo["F.contabilizada"]
        GridCabeceras[Numero_fila, 17].Text = Result_Trabajo["F.total"]
        GridCabeceras[Numero_fila, 18].Text = Result_Trabajo["C.razonsocial"]
        GridCabeceras[Numero_fila, 19].Text = Result_Trabajo["V.nombre"]
        Result_Trabajo.MoveNext
    Wend
Else
    While Result_Trabajo.Available
        Numero_fila = GridCabeceras.Rows.Count
        GridCabeceras.Rows.Insert(Numero_fila, 1)
        GridCabeceras[Numero_fila, 0].Text = Result_Trabajo["F.numeroalbaran"]
        GridCabeceras[Numero_fila, 1].Text = Result_Trabajo["F.fecha"]
        GridCabeceras[Numero_fila, 2].Text = Result_Trabajo["F.hora"]
        GridCabeceras[Numero_fila, 3].Text = Result_Trabajo["F.base"]
        GridCabeceras[Numero_fila, 4].Text = Result_Trabajo["F.iva"]
        GridCabeceras[Numero_fila, 5].Text = Result_Trabajo["F.recargo"]
        GridCabeceras[Numero_fila, 6].Text = Result_Trabajo["F.importeiva"]
        GridCabeceras[Numero_fila, 7].Text = Result_Trabajo["F.importerecargo"]
        GridCabeceras[Numero_fila, 8].Text = Result_Trabajo["F.fechacobro"]
        GridCabeceras[Numero_fila, 9].Text = Result_Trabajo["F.descuentopp"]
        GridCabeceras[Numero_fila, 10].Text =
Result_Trabajo["F.importedescuentopp"]
        GridCabeceras[Numero_fila, 11].Text = Result_Trabajo["F.descuentoesp"]
        GridCabeceras[Numero_fila, 12].Text =
Result_Trabajo["F.importedescuentoesp"]
        GridCabeceras[Numero_fila, 13].Text = Result_Trabajo["F.pagado"]
        GridCabeceras[Numero_fila, 14].Text = Result_Trabajo["F.formapago"]
        GridCabeceras[Numero_fila, 15].Text = Result_Trabajo["F.vencimiento"]
        GridCabeceras[Numero_fila, 16].Text = Result_Trabajo["F.facturado"]
        GridCabeceras[Numero_fila, 17].Text = Result_Trabajo["F.total"]
        GridCabeceras[Numero_fila, 18].Text = Result_Trabajo["C.razonsocial"]
        GridCabeceras[Numero_fila, 19].Text = Result_Trabajo["V.nombre"]
        Result_Trabajo.MoveNext
    Wend
Endif
FMain.Conexion_Base_Datos.Close
GridCabeceras.Rows.Select(0, 1)

End

'' Funcion para comprobar los Campos de Compras <br>
'' Parametros:<br>
'' Devuelve:<br>
'' TRUE --> Si los valores son correctos.<br>

```

Capítulo 2. CÓDIGO FUENTE

```
' ' FALSE --> valores erroneos.
Private Function Comprobar_Campos() As Boolean

    Dim Validar As Boolean
    Dim sql As String
    Validar = True
    If CmbFacturaAlbaran.text = "" Then
        Balloon.Error("Debe elegir Factura/Albaran para editar", CmbFacturaAlbaran)
        validar = False
    Else If RdbFacturas.Value = True Then
        Sql = "select * from facturas where numerofactura='" &
    UCase(CmbFacturaAlbaran.Text) & "'"
        If Not Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El Numro de Factura no existe", CmbFacturaAlbaran)
            Validar = False
            CmbFacturaAlbaran.SetFocus
        Endif
    Else
        Sql = "select * from albaranes where numeroalbaran='" &
    UCase(CmbFacturaAlbaran.Text) & "'"
        If Not Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("El Numro de Albaran no existe", CmbFacturaAlbaran)
            Validar = False
            CmbFacturaAlbaran.SetFocus
        Endif
    Endif
Return validar
End

' ' Para quitar los filtros
Private Sub Quitar_Filtros()
    Dim Sql As String
    If RdbFacturas.Value = True Then
        sql = "select
F.numerofactura,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.cobrada,F.formapago,F.vencimiento,F.con
tabilizada,F.total,C.razonsocial,V.nombre from facturas F left join clientes C
on" &
" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor order by F.numerofactura"
        Llenar_DataGrid_Cabeceras(sql, "F")
        sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
" where L.numerofactura='" & GridCabeceras[0, 0].Text & "'"
        Llenar_DataGrid_Detalles(sql, "F")
    Else
        sql = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.pagado,F.formapago,F.vencimiento,F.fact
urado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
" F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor order by F.numeroalbaran"
```

```

        Llenar_DataGrid_Cabeceras(sql, "A")
        sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numeroalbaran='" & GridCabeceras[0, 0].Text & "'"
        Llenar_DataGrid_Detalles(sql, "A")
    Endif
    Linea_Cabeceras = 0
    Linea_Detalles = 0
    Cmbfcliente.Index = -1
    Cmbfvendedor.Index = -1
    ChkCliente.Value = False
    ChkVendedor.Value = False
    ChkFechaCreacion.Value = False
End

'' Ejecutamos los filtros cuando se han configurado Correctamente.
Private Sub Ejecutar_Filtros()
    Dim Filtrado As Integer
    Dim Sql As String
    filtrado = 0
    If RdbFacturas.Value = True Then
        sql = "select
F.numerofactura,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.cobrada,F.formapago,F.vencimiento,F.con
tabilizada,F.total,C.razonsocial,V.nombre from facturas F left join clientes C
on" &
        " F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor where "
    Else
        sql = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.pagado,F.formapago,F.vencimiento,F.fact
urado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
        " F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor where "
    Endif

    ' Seleccion del filtro Fecha Creacion
    If ChkFechaCreacion.value = True Then
        If DteFICreacion.Value = DteFFCreacion.value Then
            sql &= " F.fecha='" & utilidades.Fecha_Correcta(DteFICreacion.value) &
            """"
        Else
            sql &= " F.fecha between '" &
            utilidades.Fecha_Correcta(DteFICreacion.value) & "' and '" &
            utilidades.Fecha_Correcta(DteFFCreacion.value) & "'"
        Endif
        filtrado = 1
    Endif
    ' Seleccion del filtro Cliente
    If Chkcliente.value = True Then
        If filtrado = 0 Then
            sql &= " C.razonsocial='" & Cmbfcliente.text & "'"
        Else

```

```
        sql &= " and C.razonsocial='" & Cmbfcliente.text & "'"
    Endif
    filtrado = 1
Endif
' Seleccion del filtro Vendedor
If Chkvendedor.value = True Then
    If filtrado = 0 Then
        sql &= " V.nombre='" & Cmbfvendedor.text & "'"
    Else
        sql &= " and V.nombre='" & Cmbfvendedor.text & "'"
    Endif
Endif

If RdbFacturas.Value = True Then
    sql &= " order by F. numerofactura"
    Llenar_DataGrid_Cabeceras(sql, "F")
    If GridCabeceras.Rows.Count > 0 Then
        sql = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numerofactura='" & GridCabeceras[0, 0].Text & "'"
        Llenar_DataGrid_Detalles(sql, "F")
        Linea_Cabeceras = 0
        Linea_Detalles = 0
    Else
        If GridDetalles.Rows.Count > 0 Then
            GridDetalles.Rows.Remove(0, GridDetalles.Rows.Count)
        Endif
    Endif
Else
    sql &= " order by F.numeroalbaran"
    Llenar_DataGrid_Cabeceras(sql, "A")
    Print sql
    If GridCabeceras.Rows.Count > 0 Then
        sql = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
        " where L.numeroalbaran='" & GridCabeceras[0, 0].Text & "'"
        Llenar_DataGrid_Detalles(sql, "A")
        Linea_Cabeceras = 0
        Linea_Detalles = 0
        If GridDetalles.Rows.Count > 0 Then
            GridDetalles.Rows.Remove(0, GridDetalles.Rows.Count)
        Endif
    Endif
Endif
Endif

End
```

2.2.3.4 Frm_Mostrar_Factura.class

```
Public Sub Form_Open()

    WebView.Url = FMain.FacturaAlbaranAMostrar & ".html"
    Me.Title = "VISUALIZACION FACTURA/ALBARAN"
End

Public Sub BtnSalirVisualizar_Click()
```

```
Me.Close

End

Public Sub BtnGenerar_Click()

    Exec ["wkhtmltopdf", FMain.FacturaAlbaranAMostrar & ".html",
FMain.FacturaAlbaranAMostrar & ".pdf"] Wait
    Exec ["evince", FMain.FacturaAlbaranAMostrar & ".pdf"]

End
```

2.2.3.5 Frm_FacturacionAlbaranes.class

```
Public Linea_Albaran As Integer
Public Linea_FacturasDetalle As Integer
Public Linea As Integer

Public Sub Form_Open()
    Me.Title = "FACTURAR ALBARANES"
    Configurar_Objeto_FacturacionAlbaranes()
End

Public Sub BtnSalirFacturarAlbaranes_Click()
    Me.Close()
End

Public Sub BtnAgregarAlbaran_Click()
    Dim Numero_fila As Integer
    Dim Albaran As String

    If GridAlbaranes.Rows.Selection.Count > 0 Then
        Numero_fila = GridAlbaranes.Rows.Selection[0]
        Albaran = GridAlbaranes[Numero_fila, 0].Text
        If Comprobar_Albaranes(Albaran) Then
            ltbAlbaranes.Add(Albaran)
            Agregar_LineaAlbaran_LineaFactura(Albaran)
            TxtNumero.Enabled = False
        Else
            Balloon.Info("Albaran " & Albaran & " ya añadido", GridAlbaranes)
        Endif
    Endif

End

Public Sub BtnAgregarTodosAlbaranes_Click()

    Dim i As Integer
    Dim Albaran As String
    If GridAlbaranes.Rows.Selection.Count > 0 Then
        If GridAlbaranes.Rows.Count = ltbAlbaranes.Count Then
        Else
            For i = 0 To GridAlbaranes.Rows.Count - 1
                albaran = GridAlbaranes[i, 0].Text
                If Comprobar_Albaranes(albaran) Then
                    ltbAlbaranes.Add(albaran)
                    Agregar_LineaAlbaran_LineaFactura(albaran)
                End If
            Next i
        End If
    End If
```



```
        Endif
    Next
Endif
Endif
End

Public Sub BtnBorrarAlbaran_Click()

    Dim Contador As Integer
    Dim i As Integer
    Dim j As Integer
    Dim Albaran As String
    Dim Contador2 As Integer
    Contador = ltbAlbaranes.Count
    For i = 0 To Contador - 1
        If ltbAlbaranes[i].Selected Then
            albaran = ltbAlbaranes[i].Text
            ltbAlbaranes.Remove(i)
            contador2 = utilidades.Tabla_Vacia("select * from albaranesdetalle where
numeroalbaran='" & albaran & "'")
            For j = 1 To contador2
                EliminarAlbaran(Albaran)
            Next
        Endif
    Next
    If ltbAlbaranes.Count = 0 Then
        TxtNumero.Enabled = True
    Endif
End

Public Sub CmbCliente_Click()

    Dim Codigo_Cliente As String
    Dim Iva As Integer
    Dim RE As Integer
    If CmbCliente.Text = "" Then
    Else
        iva = Utilidades.Obtener_Campo_Tabla("select iva from tipoiva", "iva")
        re = Utilidades.Obtener_Campo_Tabla("select recargo from tipoiva",
"recargo")
        Codigo_Cliente = Utilidades.Obtener_Campo_Tabla("select codigocliente from
clientes where razonsocial='" & CmbCliente.Text & "'", "codigocliente")
        CmbVendedor.Text = Utilidades.Obtener_Campo_Tabla("select vendedor from
clientes where razonsocial='" & CmbCliente.Text & "'", "vendedor")
        CmbFormaPago.Text = Utilidades.Obtener_Campo_Tabla("select formapago from
clientes where razonsocial='" & CmbCliente.Text & "'", "formapago")
        VlbDescuentoESP.Value = Utilidades.Obtener_Campo_Tabla("select descuentoesp
from clientes where razonsocial='" & CmbCliente.Text & "'", "descuentoesp")
        VlbDescuentoPP.Value = Utilidades.Obtener_Campo_Tabla("select descuentopp
from clientes where razonsocial='" & CmbCliente.Text & "'", "descuentopp")
        If Utilidades.Obtener_Campo_Tabla("select iva from clientes where
razonsocial='" & CmbCliente.Text & "'", "iva") = "S" Then
            VlbIva.Value = iva
        Else
            VlbIva.Value = 0
        Endif
        If Utilidades.Obtener_Campo_Tabla("select recargo from clientes where
razonsocial='" & CmbCliente.Text & "'", "recargo") = "S" Then
            VlbRecargo.Value = re
        Endif
    Endif
End
```

```
Else
    VlbRecargo.Value = 0
Endif
BtnFiltrar_Click()
ChkCobradaPagado.Value = False
'ChkEntreFechas.Value = False
'FrameFiltroEntreFechas.Enabled = False
DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaInicio.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaFin.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaCobro.Value = Date(Year(Now), Month(Now), Day(Now))
lbtAlbaranes.Clear
Endif
End
Public Sub BtnFiltrar_Click()

    Dim Sql As String
    If CmbCliente.Text = "" Then
    Else
        sql = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.pagado,F.formapago,F.vencimiento,F.fact
urado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
        " F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigo vendedor where C.razonsocial='" & CmbCliente.Text & "' and
F.facturado='N'"
        If ChkEntreFechas.value = True Then
            If DteFechaInicio.Value = DteFechaFin.value Then
                sql &= " and F.fecha='" &
utilidades.Fecha_Correcta(DteFechaInicio.value) & "'"
            Else
                sql &= " and F.fecha between '" &
utilidades.Fecha_Correcta(DteFechaInicio.value) & "' and '" &
utilidades.Fecha_Correcta(DteFechaFin.value) & "'"
            Endif
        Endif
        Llenar_DataGrid_Albaran(sql)
    Endif

End
Public Sub ChkEntreFechas_Click()

    If ChkEntreFechas.Value = True Then
        FrameFiltroEntreFechas.Enabled = True
    Else
        FrameFiltroEntreFechas.Enabled = False
    Endif

End
Public Sub ChkCobradaPagado_Click()

    If ChkCobradaPagado.Value = True Then
        DteFechaCobro.Enabled = True

    Else
        DteFechaCobro.Enabled = False
    Endif
```

```
End

Public Sub CmbFormaPago_Click()

    Dim dias As Integer
    dias = CInt(Utilidades.Obtener_Campo_Tabla("select dias from formapago where
nombreformapago='" & CmbFormaPago.Text & "'", "dias"))
    If dias > 0 Then
        DteFechavencimiento.Value = DateAdd(Now, gb.Day, dias)
    Else
        DteFechavencimiento.Value = Date(Year(Now), Month(Now), Day(Now))
    Endif

End

Public Sub BtnGenerar_Click()

    If GridFacturasDetalle.Rows.Count = 0 Then
        Balloon.Info("No ha añadido ningun albaran debe añadir al menos un
albaran.", ltbAlbaranes)
    Else
        Actualiza_Facturado_Albaran()
        Grabar_DetalleFactura()
        Grabar_Factura()
        Limpiar_Datos_Factura()
        TxtNumero.Text = Utilidades.Calculo_Numero_FA("F")
        If ChkVisualizarFacturta.Value = True Then
            Utilidades.Generar_Factura_Albaran(TxtNumero.text)
        Endif
    Endif

End

'''*****INICIO ALBARANES GRID*****'''

Public Sub BtnInicio_Click()

    If GridAlbaranes.Rows.Count > 0 Then
        GridAlbaranes.Rows.UnselectAll
        Linea_Albaran = 0
        GridAlbaranes.Rows.Select(Linea_Albaran, 1)
    Endif

End

Public Sub BtnAnterior_Click()

    If GridAlbaranes.Rows.Count > 0 Then
        If Linea_Albaran = 0 Then
            GridAlbaranes.Rows.Select(Linea_Albaran, 1)
        Else
            GridAlbaranes.Rows.UnselectAll
            Linea_Albaran = Linea_Albaran - 1
            GridAlbaranes.Rows.Select(Linea_Albaran, 1)
        Endif
    Endif

End
```

```
Endif

End

Public Sub BtnSiguiente_Click()

    If GridAlbaranes.Rows.Count > 0 Then
        If Linea_Albaran = GridAlbaranes.Rows.Count - 1 Then
            GridAlbaranes.Rows.Select(Linea_Albaran, 1)
        Else
            GridAlbaranes.Rows.UnselectAll
            Linea_Albaran = Linea_Albaran + 1
            GridAlbaranes.Rows.Select(Linea_Albaran, 1)
        Endif
    Endif

End

Public Sub BtnFin_Click()

    If GridAlbaranes.Rows.Count > 0 Then
        GridAlbaranes.Rows.UnselectAll
        Linea_Albaran = GridAlbaranes.Rows.Count - 1
        GridAlbaranes.Rows.Select(Linea_Albaran, 1)
    Endif

End

'''*****FIN ALBARANES GRID*****'''
''
'''*****

'''*****INICIO FACTURAS DETALLE GRID*****'''
''
'''*****

Public Sub BtnInicioFacturasDetalle_Click()

    If GridFacturasDetalle.Rows.Count > 0 Then
        GridFacturasDetalle.Rows.UnselectAll
        Linea_FacturasDetalle = 0
        GridFacturasDetalle.Rows.Select(Linea_FacturasDetalle, 1)
    Endif

End

Public Sub BtnAnteriorFacturasDetalle_Click()

    If GridFacturasDetalle.Rows.Count > 0 Then
        If Linea_FacturasDetalle = 0 Then
            GridFacturasDetalle.Rows.Select(Linea_FacturasDetalle, 1)
        Else
            GridFacturasDetalle.Rows.UnselectAll
            Linea_FacturasDetalle = Linea_FacturasDetalle - 1
            GridFacturasDetalle.Rows.Select(Linea_FacturasDetalle, 1)
        Endif
    Endif

End
```

```
Endif

End

Public Sub BtnSiguienteFacturasDetalle_Click()

    If GridFacturasDetalle.Rows.Count > 0 Then
        If Linea_FacturasDetalle = GridFacturasDetalle.Rows.Count - 1 Then
            GridFacturasDetalle.Rows.Select(Linea_FacturasDetalle, 1)
        Else
            GridFacturasDetalle.Rows.UnselectAll
            Linea_FacturasDetalle = Linea_FacturasDetalle + 1
            GridFacturasDetalle.Rows.Select(Linea_FacturasDetalle, 1)
        Endif
    Endif

End

Public Sub BtnFinFacturasDetalle_Click()

    If GridFacturasDetalle.Rows.Count > 0 Then
        GridFacturasDetalle.Rows.UnselectAll
        Linea_FacturasDetalle = GridFacturasDetalle.Rows.Count - 1
        GridFacturasDetalle.Rows.Select(Linea_FacturasDetalle, 1)
    Endif

End

'''*****FIN FACTURAS DETALLE GRID*****'''
'''*****'''

'''***** Funciones privadas*****'''
'''*****'''

''' Procedimiento para añadir las lineas de detalle al la factura<br>
''' solo introduce los datos en el GridFacturasDetalle, si ya existe un <br>
''' articulo le suma la cantidad y recalcula el total.<br>
''' Parametros:<br>
'''     Albaran --> Numero Albaran que hay que añadir los elementos.<br>
'''

Private Sub Agregar_LineaAlbaran_Lineafactura(Albaran As String)

    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    Dim Sql As String
    Dim NombreArticulo As String
    'Dim Linea As Integer
    ' Datos del Albaran
    Dim Albaran_Peso As Float
    Dim Albaran_Precio As Float
    Dim Albaran_Lote As Float
    Dim Albaran_Articulo As String
    Dim Albaran_Importe As Float
    linea = 1
    If Not FMain.Conexion_Base_Datos.Opened Then
```

Capítulo 2. CÓDIGO FUENTE

```
FMain.Conexion_Base_Datos.Open
Endif
sql = "select * from albaranesdetalle where numeroalbaran='" & Albaran & "'"
Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
Result_Trabajo.MoveFirst
While Result_Trabajo.Available
    Albaran_Precio = Result_Trabajo["precio"]
    Albaran_Peso = Result_Trabajo["peso"]
    Albaran_Importe = Result_Trabajo["importe"]
    Albaran_Lote = Result_Trabajo["lote"]
    Albaran_Articulo = Result_Trabajo["articulo"]
    NombreArticulo = utilidades.Obtener_Campo_Tabla("select nombre from articulo
where codigoarticulo='" & Albaran_Articulo & "'", "nombre")
    linea = CompruebaArticulo(NombreArticulo, Albaran_Lote)
    If linea = 0 Then
        Numero_fila = GridFacturasDetalle.Rows.Count
        GridFacturasDetalle.Rows.Insert(Numero_fila, 1)
        GridFacturasDetalle[Numero_fila, 0].Text = TxtNumero.Text
        GridFacturasDetalle[Numero_fila, 1].Text = Albaran
        GridFacturasDetalle[Numero_fila, 2].Text = Albaran_Precio
        GridFacturasDetalle[Numero_fila, 3].Text = Albaran_Peso
        GridFacturasDetalle[Numero_fila, 4].Text = Albaran_Importe
        GridFacturasDetalle[Numero_fila, 5].Text = Albaran_Lote
        GridFacturasDetalle[Numero_fila, 6].Text = NombreArticulo
    Else

    Endif
    VlbImporteBase.Value = VlbImporteBase.Value + CFloat(Albaran_Importe)
    Result_Trabajo.MoveNext
Wend
Calculo_Descuentos_Ivas()
End

'' Eliminamos el albaran de la factura y recalculamos.<br>
'' Parametros:<br>
''     Albaran - - > Numero de albaran a eliminar<br>

Private Sub EliminarAlbaran(Albaran As String)

    Dim i As Integer
    Dim Total As Float
    Dim Salir As Integer
    salir = 0
    i = 0
    If GridFacturasDetalle.Rows.Count > 0 Then
        While salir = 0
            If GridFacturasDetalle[i, 1].Text = Albaran Then
                total = CFloat(GridFacturasDetalle[i, 5].Text)
                VlbImporteBase.Value = VlbImporteBase.Value - total
                GridFacturasDetalle.Rows.Remove(i)
                salir = 1
            Endif
            i = i + 1
        Wend
        Calculo_Descuentos_Ivas()
    Endif
End
```

Capítulo 2. CÓDIGO FUENTE

```
' ' Funcion que comprueba si un articulo ya existe en el GridFacturasDetalle,
debe coincidir nombre y lote<br>
' ' Parametros:<br>
' '     Articulo -- > Nombre del articulo a buscar<br>
' '     Lote -- > Lote del articulo a buscar<br>
' ' Devuelve:<br>
' '     Numero_Linea -- Si existe<br>
' '     0 -- Si no existe<br>
Private Function CompruebaArticulo(Articulo As String, Lote As String) As
Integer

    Dim Validar As Integer
    Dim i As Integer
    validar = 0
    If GridFacturasDetalle.Rows.Count > 0 Then
        For i = 0 To GridFacturasDetalle.Rows.Count - 1
            If GridFacturasDetalle[i, 6].Text = Articulo And GridFacturasDetalle[i,
5].Text = Lote Then
                validar = i
            Endif
        Next
    Endif
    Return validar
End

' ' Configuracion de los objetos iniciales del formularios
Private Sub Configurar_Objetos_FacturacionAlbaranes()

    Dim Error_Tablas_Vacias As String
    Dim Error_V As Boolean

    error_V = False

    If Utilidades.Tabla_Vacia("select * from clientes") = 0 Then
        Error_Tablas_Vacias = "Deber Cear Clientes"
        Error_V = True
    Else
        CmbCliente.clear
        Utilidades.Llenar_Combobox("select * from clientes", CmbCliente,
"razonsocial")
        CmbCliente.Index = -1
    Endif
    If Utilidades.Tabla_Vacia("select * from vendedor") = 0 Then
        If Error_Tablas_Vacias = "" Then
            Error_Tablas_Vacias = "Deber Cear Vendedores"
        Else
            Error_Tablas_Vacias &= " y Crear Vendedores."
        Endif
        Error_V = True
    Else
        CmbVendedor.clear
        Utilidades.Llenar_Combobox("select * from vendedor", CmbVendedor,
"nombre")
    Endif
    If Utilidades.Tabla_Vacia("select * from formapago") = 0 Then
        If Error_Tablas_Vacias = "" Then
            Error_Tablas_Vacias = "Deber Cear Formas de Pago en Utilidades"
        Else
```

Capítulo 2. CÓDIGO FUENTE

```
Error_Tablas_Vacias &= " y Crear Formas de Pago en Utilidades."
Endif
Error_V = True
Else
    CmbFormaPago.clear
    Utilidades.Llenar_Combobox("select * from formapago", CmbFormaPago,
"nombreformapago")
    CmbFormaPago.Index = -1
Endif
If Error_V Then
    'TxtErrorFacturacion.Visible = True
    'TxtErrorFacturacion.text = Error_Tablas_Vacias
Endif
DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaInicio.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaFin.Value = Date(Year(Now), Month(Now), Day(Now))
DteFechaCobro.Value = Date(Year(Now), Month(Now), Day(Now))
DteHora.Value = Format$(Now, "h:n:s")
Linea_Albaran = 0
Linea = 1
'LblTotalFactura.text = Format(0, "##0.00")
'VlbImporteBase.Value = Format(0, "##0.00")
TxtNumero.Text = Utilidades.Calculo_Numero_FA("F")
Configurar_DataGrid_Albaranes()
Configurar_DataGrid_FacturasDetalle()

End

'' Procedimiento para configurar Grid LineasDetalle
Private Sub Configurar_DataGrid_Albaranes()

    GridAlbaranes.Columns.Count = 20
    GridAlbaranes.Header = GridAlbaranes.Both
    GridAlbaranes.Columns[0].Title = "Numero"
    GridAlbaranes.Columns[1].Title = "Fecha"
    GridAlbaranes.Columns[2].Title = "Hora"
    GridAlbaranes.Columns[3].Title = "Base"
    GridAlbaranes.Columns[4].Title = "Iva"
    GridAlbaranes.Columns[5].Title = "Recargo"
    GridAlbaranes.Columns[6].Title = "Importe Iva"
    GridAlbaranes.Columns[7].Title = "Importe RE"
    GridAlbaranes.Columns[8].Title = "Fecha Cobro"
    GridAlbaranes.Columns[9].Title = "Descuento PP"
    GridAlbaranes.Columns[10].Title = "Importe DPP"
    GridAlbaranes.Columns[11].Title = "Descuento ESP"
    GridAlbaranes.Columns[12].Title = "Importe DESP"
    GridAlbaranes.Columns[13].Title = "Pagado"
    GridAlbaranes.Columns[14].Title = "Forma Pago"
    GridAlbaranes.Columns[15].Title = "Vencimiento"
    GridAlbaranes.Columns[16].Title = "Facturado"
    GridAlbaranes.Columns[17].Title = "Total"
    GridAlbaranes.Columns[18].Title = "Cliente"
    GridAlbaranes.Columns[19].Title = "Vendedor"
    GridAlbaranes.Columns[0].Width = 100
    GridAlbaranes.Columns[1].Width = 100
    GridAlbaranes.Columns[2].Width = 100
    GridAlbaranes.Columns[3].Width = 100
    GridAlbaranes.Columns[4].Width = 100
    GridAlbaranes.Columns[5].Width = 150
```



```
GridAlbaranes.Columns[6].Width = 200
GridAlbaranes.Columns[7].Width = 200
GridAlbaranes.Columns[8].Width = 200
GridAlbaranes.Columns[9].Width = 200
GridAlbaranes.Columns[10].Width = 200
GridAlbaranes.Columns[11].Width = 200
GridAlbaranes.Columns[12].Width = 200
GridAlbaranes.Columns[13].Width = 200
GridAlbaranes.Columns[14].Width = 200
GridAlbaranes.Columns[15].Width = 200
GridAlbaranes.Columns[16].Width = 200
GridAlbaranes.Columns[17].Width = 150
GridAlbaranes.Columns[18].Width = 200
GridAlbaranes.Columns[19].Width = 200

End

'' Procedimiento para configurar Grid FacturasDetalle
Private Sub Configurar_DataGrid_FacturasDetalle()

    GridFacturasDetalle.Columns.Count = 7
    GridFacturasDetalle.Header = GridFacturasDetalle.Both
    GridFacturasDetalle.Columns[0].Title = "Factura"
    GridFacturasDetalle.Columns[1].Title = "Albaran"
    GridFacturasDetalle.Columns[2].Title = "Precio"
    GridFacturasDetalle.Columns[3].Title = "Peso"
    GridFacturasDetalle.Columns[4].Title = "Importe"
    GridFacturasDetalle.Columns[5].Title = "Lote"
    GridFacturasDetalle.Columns[6].Title = "Articulo"
    GridFacturasDetalle.Columns[0].Width = 100
    GridFacturasDetalle.Columns[1].Width = 100
    GridFacturasDetalle.Columns[2].Width = 100
    GridFacturasDetalle.Columns[3].Width = 100
    GridFacturasDetalle.Columns[4].Width = 150
    GridFacturasDetalle.Columns[5].Width = 150
    GridFacturasDetalle.Columns[6].Width = 200

End

'' Procedimiento para LLenar de datos el DataGrid Cabeceras<br>
'' Parametros:<br>
''     Sql --> Setncia Sql para llenar el Data Grid<br>
Private Sub Llenar_DataGrid_Albaran(Sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridAlbaranes.Rows.Count > 0 Then
        GridAlbaranes.Rows.Remove(0, GridAlbaranes.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst

    While Result_Trabajo.Available
        Numero_fila = GridAlbaranes.Rows.Count
        GridAlbaranes.Rows.Insert(Numero_fila, 1)
        GridAlbaranes[Numero_fila, 0].Text = Result_Trabajo["F.numeroalbaran"]
        GridAlbaranes[Numero_fila, 1].Text = Result_Trabajo["F.fecha"]
        GridAlbaranes[Numero_fila, 2].Text = Result_Trabajo["F.hora"]
        GridAlbaranes[Numero_fila, 3].Text = Result_Trabajo["F.base"]
    End While
End Sub
```

```

GridAlbaranes[Numero_fila, 4].Text = Result_Trabajo["F.iva"]
GridAlbaranes[Numero_fila, 5].Text = Result_Trabajo["F.recargo"]
GridAlbaranes[Numero_fila, 6].Text = Result_Trabajo["F.importeiva"]
GridAlbaranes[Numero_fila, 7].Text = Result_Trabajo["F.importerrecargo"]
GridAlbaranes[Numero_fila, 8].Text = Result_Trabajo["F.fechacobro"]
GridAlbaranes[Numero_fila, 9].Text = Result_Trabajo["F.descuentopp"]
GridAlbaranes[Numero_fila, 10].Text = Result_Trabajo["F.importedescuentopp"]
GridAlbaranes[Numero_fila, 11].Text = Result_Trabajo["F.descuentoesp"]
GridAlbaranes[Numero_fila, 12].Text =
Result_Trabajo["F.importedescuentoesp"]
GridAlbaranes[Numero_fila, 13].Text = Result_Trabajo["F.pagado"]
GridAlbaranes[Numero_fila, 14].Text = Result_Trabajo["F.formapago"]
GridAlbaranes[Numero_fila, 15].Text = Result_Trabajo["F.vencimiento"]
GridAlbaranes[Numero_fila, 16].Text = Result_Trabajo["F.facturado"]
GridAlbaranes[Numero_fila, 17].Text = Result_Trabajo["F.total"]
GridAlbaranes[Numero_fila, 18].Text = Result_Trabajo["C.razonsocial"]
GridAlbaranes[Numero_fila, 19].Text = Result_Trabajo["V.nombre"]
Result_Trabajo.MoveNext
Wend

FMain.Conexion_Base_Datos.Close
GridAlbaranes.Rows.Select(0, 1)

End
'' Comprueba que no ha sido añadido ya el albaran.<br>
'' Parametros :<br>
'' Albaran --> Albaran a comprobar.<br>
'' Devuelve :<br>
'' True --> Albaran no ha sido añadido.<br>
'' False --> Albaran ha sido añadido.<br>
Private Function Comprobar_Albaranes(Albaran As String) As Boolean

Dim Validar As Boolean
Dim i As Integer
Dim contador As Integer
Validar = True
Contador = ltbAlbaranes.Count
For i = 0 To Contador - 1
If ltbAlbaranes[i].Text = Albaran Then
validar = False
Endif
Next
Return Validar
End

'' Procedimiento para calcular Descuentos e Iva cuando añadimos o borramos una
linea detalle.

Private Sub Calculo_Descuentos_Ivas()

Dim Descuento_PP As Float
Dim Descuento_ESP As Float
Dim Iva As Float
Dim RE As Float
Dim AuxBase As Float
If VlbDescuentoESP.Value > 0 Then
Descuento_ESP = (VlbImporteBase.Value * VlbDescuentoESP.Value) / 100
Else
Descuento_ESP = 0

```

```
Endif
If VlbDescuentoPP.Value > 0 Then
    Descuento_PP = (VlbImporteBase.Value * VlbDescuentoPP.Value) / 100
Else
    Descuento_PP = 0
Endif
VlbImporteDescuentoESP.Value = Format(Descuento_ESP, "##0.00")
VlbImporteDescuentoPP.Value = Format(Descuento_PP, "##0.00")
auxbase = VlbImporteBase.Value - Descuento_ESP - Descuento_PP
If VlbIva.Value > 0 Then
    iva = (auxbase * VlbIva.value) / 100
Else
    iva = 0
Endif
If VlbRecargo.Value > 0 Then
    re = (auxbase * VlbRecargo.value) / 100
Else
    re = 0
Endif
VlbImporteIva.Value = Format(iva, "##0.00")
VlbImporteRecargo.Value = Format(re, "##0.00")
LblTotalFactura.Text = Format(auxbase + iva + re, "##0.00")
End

'' Limpia los datos de la cabecera de la factura.
Private Sub Limpiar_Datos_Factura()
    TxtNumero.Enabled = True
    linea = 1
    ChkCobradaPagado.Value = False
    DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFechaCobro.Value = Date(Year(Now), Month(Now), Day(Now))
    DteHora.Value = Format$(Now, "h:n:s")
    LblTotalFactura.text = Format(0, "##0.00")
    VlbDescuentoESP.Value = 0
    VlbDescuentoPP.Value = 0
    VlbImporteDescuentoESP.Value = 0
    VlbImporteDescuentoPP.Value = 0
    VlbIva.Value = 0
    VlbImporteIva.Value = 0
    VlbRecargo.Value = 0
    VlbImporteRecargo.Value = 0
    VlbImporteBase.value = 0
    CmbCliente.Index = -1
    CmbVendedor.Index = -1
    CmbFormaPago.Index = -1
    ltbAlbaranes.Clear
    GridFacturasDetalle.Rows.Remove(0, GridFacturasDetalle.Rows.Count)
    GridAlbaranes.Rows.Remove(0, GridAlbaranes.Rows.Count)
End

'' Graba la factura.
Private Sub Grabar_Factura()

    Dim Codigo_Vendedor As String
    Dim Codigo_Cliente As String
    Dim Sql As String
    Dim Cobrado_Pagada As String
    If ChkCobradaPagado.Value = True Then
        Cobrado_Pagada = "S"
```

Capítulo 2. CÓDIGO FUENTE

```
Else
    Cobrado_Pagada = "N"
Endif
Codigo_Cliente = Utilidades.Obtener_Campo_Tabla("select codigocliente from
clientes where razonsocial='" & CmbCliente.Text & "'", "codigocliente")
Codigo_Vendedor = Utilidades.Obtener_Campo_Tabla("select codigovendedor from
vendedor where nombre='" & CmbVendedor.Text & "'", "codigovendedor")
sql = "insert into
facturas(numerofactura, fecha, hora, base, iva, recargo, importeiva, importerecargo, fec
hacobro, descuentopp, importedes descuentopp, " &

"descentoesp, importedescentoesp, cobrada, formapago, vencimiento, contabilizada, to
tal, cliente, vendedor) values ( '" &
    UCase(TxtNumero.Text) & "', '" &
Utilidades.Fecha_Correcta(DteFechaCreacion.Value) & "', '" & DteHora.value & "', "
& VlbImporteBase.Value & ", " &
    VlbIva.Value & ", " & VlbRecargo.Value & ", " & VlbImporteIva.Value & ", " &
VlbImporteRecargo.Value & ", '" & Utilidades.Fecha_Correcta(DteFechaCobro.Value)
& "', " &
    VlbDescuentoPP.Value & ", " & VlbImporteDescuentoPP.Value & ", " &
VlbDescuentoESP.Value & ", " & VlbImporteDescuentoESP.Value & ", '" &
Cobrado_Pagada & "', '" &
    CmbFormaPago.Text & "', '" &
Utilidades.Fecha_Correcta(DteFechavencimiento.Value) & "', 'N', " &
LblTotalFactura.Text & ", '" & Codigo_Cliente & "', '" & Codigo_Vendedor & "')"
    Utilidades.In_Act_Bor(sql)
End

'' Graba las lineas de detalle de la factura.
Private Sub Grabar_DetalleFactura()

    Dim i As Integer
    Dim Sql As String
    Dim Codigo_Articulo As String
    For i = 0 To GridFacturasDetalle.Rows.Count - 1
        Codigo_Articulo = utilidades.Obtener_Campo_Tabla("select codigoarticulo from
articulo where nombre='" & GridFacturasDetalle[i, 6].Text & "'",
"codigoarticulo")
        sql = "insert into
facturasdetalles(numerofactura, linea, precio, peso, importe, lote, articulo)
values('" &
            UCase(TxtNumero.Text) & "', " & linea & ", " & CFloat(GridFacturasDetalle[i,
2].Text) & ", " & CFloat(GridFacturasDetalle[i, 3].Text) & ", " &
CFloat(GridFacturasDetalle[i, 4].Text) &
            ", '" & UCase(GridFacturasDetalle[i, 5].Text) & "', '" & Codigo_articulo &
            "')"
        Utilidades.In_Act_Bor(sql)
        linea = linea + 1
    Next
End

'' Actualiza los Albaranes a Facturados.
Private Sub Actualiza_Facturado_Albaran()

    Dim i As Integer
    Dim Sql As String
    Dim Contador As Integer
    Contador = ltbAlbaranes.Count
```

```
For i = 0 To Contador - 1
    sql = "update albaranes set facturado='S' where numeroalbaran='" &
ltbAlbaranes[i].Text & "'"
    Utilidades.In_Act_Bor(sql)
Next
End
```

2.2.3.6 *Frm_Pasocontabilidad.class*

```
Public Sub Form_Open()
    Me.Title = "PASO A CONTABILIDAD CONTAPLUS"
    Configurar_Cuentas_Contabilidad()
    DirPasoContabilidad.Value = FMain.CarpetaCarnea
End

Public Sub BtnSalirPasoContabilidad_Click()
    Me.Close
End

Public Sub BtnIniciar_Click()

    Dim xDiario As File
    Dim XSubCta As File
    Dim RutaPasoContabilidad As String
    Dim Facturas_NoContabilizadas As Result
    Dim Asiento As Float
    ' Lineas para montar los asientos
    Dim Aux_Linea_Subcuenta As String
    Dim Linea_Subcuenta As String
    Dim Linea_Cliente As String
    Dim Linea_Venta As String
    Dim Linea_Iva As String
    Dim Linea_Recargo As String
    Dim Linea_Cobro As String
    Dim Sql As String
    ' Datos para pasar a contabilidad
    Dim Cliente_Codigo As String
    Dim Cliente_Recargo As String
    Dim Cliente_Subcuenta As String
    ' Datos Facturas
    Dim Factura_Numero As String
    Dim Factura_Fecha As String
    Dim Factura_Base As String
    Dim Factura_Iva As String
    Dim Factura_Recargo As String
    Dim Factura_ImporteIva As String
    Dim Factura_ImporteRecargo As String
    Dim Factura_Total As String

    Aux_Linea_Subcuenta = "\";;;;;;;;;;;;;"
    Asiento = 1.0
    If DirPasoContabilidad.value = "" Then
        Balloon.Error("Debe elegir directorio con permisos para poder realizar el
paso a contabilidad.", DirPasoContabilidad)
    Else
        RutaPasoContabilidad = DirPasoContabilidad.Value
        If Exist(RutaPasoContabilidad & "/xDiario.txt") Or
Exist(RutaPasoContabilidad & "/xSubcta.txt") Then
```

```

Kill RutaPasoContabilidad & "/xSubcta.txt"
Kill RutaPasoContabilidad & "/xDiario.txt"
xDiario = Open RutaPasoContabilidad & "/xDiario.txt" For Write Append
xSubCta = Open RutaPasoContabilidad & "/xSubcta.txt" For Write Append
Else
    xDiario = Open RutaPasoContabilidad & "/xDiario.txt" For Write Append
    xSubCta = Open RutaPasoContabilidad & "/xSubcta.txt" For Write Append
Endif
FMain.Conexion_Base_Datos.Close
FMain.Conexion_Base_Datos.Open
sql = "select
numerosfactura, fecha, base, iva, recargo, importeiva, importerecargo, total, cliente
from facturas where contabilizada='N'"
Facturas_NoContabilizadas = FMain.Conexion_Base_Datos.Exec(Sql)
Facturas_NoContabilizadas.MoveFirst
While Facturas_NoContabilizadas.Available

    Cliente_Codigo = Facturas_NoContabilizadas["cliente"]
    Factura_Numero = Facturas_NoContabilizadas["numerosfactura"]
    Factura_Fecha = Facturas_NoContabilizadas["fecha"]
    Factura_Base = Facturas_NoContabilizadas["base"]
    Factura_Iva = Facturas_NoContabilizadas["iva"]
    Factura_Recargo = Facturas_NoContabilizadas["recargo"]
    Factura_ImporteIva = Facturas_NoContabilizadas["importeiva"]
    Factura_ImporteRecargo = Facturas_NoContabilizadas["importerecargo"]
    Factura_Total = Facturas_NoContabilizadas["total"]
    Cliente_Recargo = Utilidades.Obtener_Campo_Tabla("select recargo from
clientes where codigocliente='" & Cliente_Codigo & "'", "recargo")
    Cliente_Subcuenta = Utilidades.Obtener_Campo_Tabla("select subcuentaiva
from clientes where codigocliente='" & Cliente_Codigo & "'", "subcuentaiva")
    Linea_Cliente = CStr(Asiento) & ";" & Factura_Fecha & ";" & "
Cliente_Subcuenta & "\" & "
"\" & "
Factura_Total & ";" & "
Linea_Subcuenta = "\" & Cliente_Subcuenta & Aux_Linea_Subcuenta
Print #xDiario, Linea_Cliente
Print #xSubCta, Linea_Subcuenta
Linea_Venta = CStr(Asiento) & ";" & Factura_Fecha & ";" & TtxtVenta.Text
& "\" & "
Factura_Base & ";" & "
Linea_Subcuenta = "\" & TtxtVenta.Text & Aux_Linea_Subcuenta
Print #xDiario, Linea_Venta
Print #xSubCta, Linea_Subcuenta
Linea_Iva = CStr(Asiento) & ";" & Factura_Fecha & ";" & TtxtIva.Text &
"\" & "
Factura_Numero & ";" & Factura_Iva & ";" & CStr(CFloat(Factura_Recargo))
& "
" & Factura_ImporteIva & ";" & Factura_Base &
" & Factura_Fecha & ";"
Print #xDiario, Linea_Iva
If Cliente_Recargo = "S" Then
    Linea_Subcuenta = "\" & TtxtIvaRe.Text & Aux_Linea_Subcuenta
    Print #xSubCta, Linea_Subcuenta
    Linea_Recargo = CStr(Asiento) & ";" & Factura_Fecha & ";" & "
TtxtRecargoE.Text & "\" & Cliente_Subcuenta & "\" & "
" & "
" & Factura_ImporteRecargo &
" & "
Print #xDiario, Linea_Recargo
Linea_Subcuenta = "\" & TtxtRecargoE.Text & Aux_Linea_Subcuenta

```

```

        Print #xSubCta, Linea_Subcuenta
    Else
        Linea_Subcuenta = "\" & TxtIva.Text & Aux_Linea_Subcuenta
        Print #xSubCta, Linea_Subcuenta
    Endif
    Asiento = Asiento + 1
    Linea_Cobro = CStr(Asiento) & ";" & Factura_Fecha & ";" & "\" & TxtCobro.Text
& "\";;;"Fra-Cobro-" & Factura_Numero & "\";;0.00;;0.00;0.00;;;"
&
        Factura_Total & ";0.00;0.00;;;"
        Linea_Subcuenta = "\" & TxtCobro.Text & Aux_Linea_Subcuenta
        Print #xDiario, Linea_Cobro
        Print #xSubCta, Linea_Subcuenta
        Linea_Cliente = CStr(Asiento) & ";" & Factura_Fecha & ";" & "\" &
Cliente_Subcuenta & "\";;;"Fra-Cobro-" & Factura_Numero &
"\";;0.00;;0.00;0.00;;;"0.00;" &
        Factura_Total & ";0.00;;;"
        Linea_Subcuenta = "\" & TxtVenta.Text & Aux_Linea_Subcuenta
        Print #xDiario, Linea_Cliente
        Print #xSubCta, Linea_Subcuenta
        Asiento = Asiento + 1
        sql = "update facturas set contabilizada='S' where numerofactura='" &
Factura_Numero & "'"
        Utilidades.In_Act_Bor(sql)
        Facturas_NoContabilizadas.MoveNext
    Wend

    xSubCta.Close
    xDiario.Close
    FMain.Conexion_Base_Datos.Close
    LcdAsientosCreados.Text = Asiento - 1
    lblproceso.Visible = True
Endif
FMain.Conexion_Base_Datos.Close
End

'***** Funciones privadas*****
'
'*****
Private Sub Configurar_Cuentas_Contabilidad()

    TxtVenta.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "venta")
    TxtIva.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "iva")
    TxtIvaRe.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "ivare")
    TxtRecargoE.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "re")
    TxtCobro.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "cobro")
End

```

2.2.3.7 *Frm_Averias.class*

```
Public Linea_Averia As Integer

Public Sub Form_Open()
    Me.Title = "GESTION DE AVERIAS"
    Configurar_Objetos_Averias()
End

Public Sub BtnSalirAverias_Click()
    Me.Close
End

Public Sub BtnAgregarAveria_Click()

    Dim Sql As String
    Dim Codigo_Articulo As String
    Dim Codigo_Vendedor As String
    Dim Codigo_Cliente As String
    Dim Fecha_Averia As String
    If Comprobar_Datos_Averia("NUEVO") Then
        sql = "select codigoarticulo from articulo where nombre='" &
CmbArticulo.Text & "'"
        Codigo_Articulo = Utilidades.Obtener_Campo_Tabla(sql, "codigoarticulo")
        sql = "select codigocliente from clientes where razonsocial='" &
CmbCliente.Text & "'"
        Codigo_Cliente = Utilidades.Obtener_Campo_Tabla(sql, "codigocliente")
        sql = "select codigovendedor from vendedor where nombre='" &
CmbVendedor.Text & "'"
        Codigo_Vendedor = Utilidades.Obtener_Campo_Tabla(sql, "codigovendedor")
        Fecha_Averia = Utilidades.Fecha_Correcta(DteFecha.Value)
        sql = "insert into
averia(codigoaveria,fecha,vendedor,cliente,articulo,peso,lote, descripcion)
values('" &
        UCase(TxtCodigoAveria.Text) & "','" & Fecha_Averia & "','" & Codigo_Vendedor
& "','" & Codigo_Cliente &
        "','" & Codigo_Articulo & "','" & vlbpeso.Value & "','" & UCase(txtlote.Text)
& "','" & UCase(txtDescripcion.Text) & "')"
        Utilidades.In_Act_Bor(sql)
        Calculo_Numero_Averia()
        Borrar_Datos_Averias()
        sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripci
on from averia A" &
        " left join vendedor V on A.vendedor=V.codigovendedor left join clientes C
on A.cliente=C.codigocliente left join " &
        " articulo T on A.articulo=T.codigoarticulo"
        Llenar_DataGrid_Averias(sql)
    Endif

End

Public Sub BtnBorrarAveria_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Dim Texto As String
    Dim Codigo_Averia As String
```



```

If GridAverias.Rows.Selection.Count > 0 Then
    Numero_fila = GridAverias.Rows.Selection[0]
    Codigo_Averia = GridAverias[Numero_fila, 0].Text
    Texto = "Desea borrar la Averia : <b>" & Codigo_Averia & "</b> y Artículo:
<b>" & GridAverias[Numero_fila, 4].Text & "</b>"
    Respuesta = Message.Delete(Texto, "Si", "No")
    'Respuesta = Message.Delete("Desea Borrar Averia : " &
GridAverias.View[Numero_fila, 0].Text, "Si", "No")
    If respuesta = 1 Then
        sql = "delete from averia where codigoaveria='" & Codigo_Averia & "'"
        Utilidades.In_Act_Bor(Sql)
        sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripci
on from averia A" &
        " left join vendedor V on A.vendedor=V.codigovendedor left join clientes C
on A.cliente=C.codigocliente left join " &
        " articulo T on A.articulo=T.codigoarticulo"
        Llenar_DataGrid_Averias(sql)
    Endif
Endif

End

Public Sub BtnEditarAveria_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Dia As String
    Dim Mes As String
    Dim Ano As String
    Dim Fecha_Averia As String
    Dim Codigo_Articulo As String
    Dim Codigo_Vendedor As String
    Dim Codigo_Cliente As String
    If BtnEditarAveria.text = "EDITAR" And GridAverias.Rows.Count > 0 Then '
Cargamos los datos del lote seleccionado.
        Numero_fila = GridAverias.Rows.Selection[0]
        TxtCodigoAveria.text = GridAverias[Numero_fila, 0].Text
        Dia = Mid(GridAverias[Numero_fila, 1].Text, 1, 2)
        Mes = Mid(GridAverias[Numero_fila, 1].Text, 4, 2)
        Ano = Mid(GridAverias[Numero_fila, 1].Text, 7, 4)
        DteFecha.Value = Date(ano, mes, dia)
        CmbVendedor.Text = GridAverias[Numero_fila, 2].Text
        CmbCliente.Text = GridAverias[Numero_fila, 3].Text
        CmbArticulo.Text = GridAverias[Numero_fila, 4].Text
        vlbpeso.Value = GridAverias[Numero_fila, 5].Text
        txtlote.Text = GridAverias[Numero_fila, 6].Text
        txtDescripcion.Text = GridAverias[Numero_fila, 7].Text
        BtnEditarAveria.text = "ACTUALIZAR"
        TxtCodigoAveria.Enabled = False
        BtnAgregarAveria.Enabled = False
        BtnBorrarAveria.Enabled = False
    Else
        If Comprobar_Datos_Averia("EDITAR") Then
            Fecha_Averia = Utilidades.Fecha_Correcta(DteFecha.Value)
            sql = "select codigoarticulo from articulo where nombre='" &
CmbArticulo.Text & "'"
            Codigo_Articulo = Utilidades.Obtener_Campo_Tabla(sql, "codigoarticulo")

```

Capítulo 2. CÓDIGO FUENTE

```
        sql = "select codigocliente from clientes where razonsocial='" &
CmbCliente.Text & "'"
        Codigo_Cliente = Utilidades.Obtener_Campo_Tabla(sql, "codigocliente")
        sql = "select codigovendedor from vendedor where nombre='" &
CmbVendedor.Text & "'"
        Codigo_Vendedor = Utilidades.Obtener_Campo_Tabla(sql, "codigovendedor")
        sql = "update averia set fecha='" & Fecha_Averia & "',vendedor='" &
Codigo_Vendedor & "',cliente='" & Codigo_Cliente &
        "',articulo='" & Codigo_Articulo & "',peso=" & vlbpeso.Value & ",lote='" &
UCase(txtlote.Text) & "',descripcion='" & UCase(txtDescripcion.Text) &
        "'" where codigoaveria='" & TxtCodigoAveria.Text & "'"
        Utilidades.In_Act_Bor(Sql)
        sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripcion
on from averia A" &
        " left join vendedor V on A.vendedor=V.codigovendedor left join clientes C
on A.cliente=C.codigocliente left join " &
        " articulo T on A.articulo=T.codigoarticulo"
        Llenar_DataGrid_Averias(sql)
        BtnEditarAveria.text = "EDITAR"
        TxtCodigoAveria.Enabled = True
        BtnAgregarAveria.Enabled = True
        BtnBorrarAveria.Enabled = True
        Borrar_Datos_Averias()
        Calculo_Numero_Averia()
    Endif
Endif

End

''***** Botones GridAverias*****
''
''*****

Public Sub BtnInicio_Click()

    If GridAverias.Rows.Count > 0 Then
        GridAverias.Rows.UnselectAll
        Linea_Averia = 0
        GridAverias.Rows.Select(Linea_Averia, 1)
    Endif

End

Public Sub BtnAnterior_Click()

    If GridAverias.Rows.Count > 0 Then
        If Linea_Averia = 0 Then
            GridAverias.Rows.Select(Linea_Averia, 1)
        Else
            GridAverias.Rows.UnselectAll
            Linea_Averia = Linea_Averia - 1
            GridAverias.Rows.Select(Linea_Averia, 1)
        Endif
    Endif

End

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnSiguiente_Click()

    If GridAverias.Rows.Count > 0 Then
        If Linea_Averia = GridAverias.Rows.Count - 1 Then
            GridAverias.Rows.Select(Linea_Averia, 1)
        Else
            GridAverias.Rows.UnselectAll
            Linea_Averia = Linea_Averia + 1
            GridAverias.Rows.Select(Linea_Averia, 1)
        Endif
    Endif

End

Public Sub BtnFin_Click()

    If GridAverias.Rows.Count > 0 Then
        GridAverias.Rows.UnselectAll
        Linea_Averia = GridAverias.Rows.Count - 1
        GridAverias.Rows.Select(Linea_Averia, 1)

    Endif
End

''***** Funciones privadas*****''
''*****''

Public Sub Configurar_Objetos_Averias()

    Dim Error_Tablas_Vacias As String
    Dim Error_V As Boolean
    Dim Sql As String

    Error_V = False
    If Utilidades.Tabla_Vacia("select * from articulo") = 0 Then
        Error_Tablas_Vacias = "Deber Cear Articulos"
        Error_V = True
    Else
        CmbArticulo.clear
        Utilidades.Llenar_Combobox("select * from articulo", CmbArticulo,
"nombre")
        CmbArticulo.Index = -1
    Endif
    If Utilidades.Tabla_Vacia("select * from vendedor") = 0 Then
        If Error_Tablas_Vacias = "" Then
            Error_Tablas_Vacias = "Deber Cear Vendedores"
        Else
            Error_Tablas_Vacias &= " y Crear Vendedores."
        Endif
        Error_V = True
    Else
        CmbVendedor.clear
        Utilidades.Llenar_Combobox("select * from vendedor", CmbVendedor,
"nombre")
        CmbVendedor.Index = -1
    Endif
    If Utilidades.Tabla_Vacia("select * from clientes") = 0 Then
```

Capítulo 2. CÓDIGO FUENTE

```

    If Error_Tablas_Vacias = "" Then
        Error_Tablas_Vacias = "Deber Cear Clientes"
    Else
        Error_Tablas_Vacias &= " y Crear Clientes."
    Endif
    Error_V = True
Else
    CmbCliente.clear
    Utilidades.Llenar_Combobox("select * from clientes", CmbCliente,
"razonsocial")
    CmbCliente.Index = -1
Endif
If Error_V Then
    TxtErrorAveria.Visible = True
    TxtErrorAveria.text = Error_Tablas_Vacias
Endif
Configurar_GridAveria()
Linea_Averia = 0
DteFecha.Value = Date(Year(Now), Month(Now), Day(Now))
Calculo_Numero_Averia()
sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripci
on from averia A" &
    " left join vendedor V on A.vendedor=V.codigovendedor left join clientes C
on A.cliente=C.codigocliente left join " &
    " articulo T on A.articulo=T.codigoarticulo"
    Llenar_DataGrid_Averias(sql)

End

'' Configuracion de GridAveria
Public Sub Configurar_GridAveria()

    Gridaverias.Columns.Count = 8
    Gridaverias.Header = Gridaverias.Both
    GridAverias.Columns[0].Title = "Codigo"
    GridAverias.Columns[1].Title = "Fecha"
    GridAverias.Columns[2].Title = "Vendedor"
    GridAverias.Columns[3].Title = "Cliente"
    GridAverias.Columns[4].Title = "Articulo"
    GridAverias.Columns[5].Title = "Lote"
    GridAverias.Columns[6].Title = "Peso"
    GridAverias.Columns[7].Title = "Descripcion"
    GridAverias.Columns[0].Width = 100
    GridAverias.Columns[1].Width = 100
    GridAverias.Columns[2].Width = 150
    GridAverias.Columns[3].Width = 150
    GridAverias.Columns[4].Width = 150
    GridAverias.Columns[5].Width = 90
    GridAverias.Columns[6].Width = 90
    GridAverias.Columns[7].Width = 250

End
```

Capítulo 2. CÓDIGO FUENTE

```
' ' Funcion para comprobar los datos de las Averias.<br>
' ' Parametros:<br>
' '     Accion Tomara los siguientes valores<br>
' '         NUEVO -- Comprobar los datos para crear nueva Averia.<br>
' '         EDITAR -- Comprobar los datos antes Modificar una Averia.<br>
' ' Devuelve:<br>
' '     TRUE -- Datos corecctos<br>
' '     FALSE -- Datos erroneos
Private Function Comprobar_Datos_Averia(Accion As String) As Boolean

    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    If Accion = "NUEVO" Then
        sql = "select * from averia where codigoaveria='" &
        UCase(TxtCodigoAveria.Text) & "'"
        If TxtCodigoAveria.Text = "" Then
            Balloon.Error("Debe introducir Numero Averia", TxtCodigoAveria)
            Validar = False
            TxtCodigoAveria.SetFocus
        Else If Utilidades.Comprobar_existe_dato(Sql)
            Balloon.Error("Codigo Averia ya existe", TxtCodigoAveria)
            Validar = False
            TxtCodigoAveria.SetFocus
        Else If CmbCliente.Text = ""
            Balloon.Error("Debe elegir Cliente para crear Averia", CmbCliente)
            Validar = False
            CmbCliente.SetFocus
        Else If CmbVendedor.Text = ""
            Balloon.Error("Debe elegir Vendedor para crear Averia", CmbVendedor)
            Validar = False
            CmbVendedor.SetFocus
        Else If CmbArticulo.Text = ""
            Balloon.Error("Debe elegir Articulo para crear Averia", CmbArticulo)
            Validar = False
            CmbArticulo.SetFocus
        Else If vlbpeso.Value = "0"
            Balloon.Error("Debe introducir peso para crear Averia", vlbpeso)
            Validar = False
            vlbpeso.SetFocus
        Else If txtlote.Text = ""
            Balloon.Error("Debe introducir lote articulo para crear Averia", txtlote)
            Validar = False
            txtlote.SetFocus
        Endif
    Else If accion = "EDITAR"
        If CmbCliente.Text = ""
            Balloon.Error("Debe elegir Cliente para crear Averia", CmbCliente)
            Validar = False
            CmbCliente.SetFocus
        Else If CmbVendedor.Text = ""
            Balloon.Error("Debe elegir Vendedor para crear Averia", CmbVendedor)
            Validar = False
            CmbVendedor.SetFocus
        Else If CmbArticulo.Text = ""
            Balloon.Error("Debe elegir Articulo para crear Averia", CmbArticulo)
            Validar = False
            CmbArticulo.SetFocus
        Else If vlbpeso.Value = "0"
```

```
        Balloon.Error("Debe introducir peso  para crear Averia", vlbpeso)
        Validar = False
        vlbpeso.SetFocus
    Else If txtlote.Text = ""
        Balloon.Error("Debe introducir lote articulo para crear Averia", txtlote)
        Validar = False
        txtlote.SetFocus
    Endif
Endif
Return Validar
End
```

```
' ' Llenar DataGridView Averias <br>
' ' Parametros: <br>
' '  Sql --> Sentencia Sql con los datos a rellenar<br>
```

```
Private Sub Llenar_DataGrid_Averias(sql As String)

    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridAverias.Rows.Count > 0 Then
        GridAverias.Rows.Remove(0, GridAverias.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    While Result_Trabajo.Available
        Numero_fila = GridAverias.Rows.Count
        GridAverias.Rows.Insert(Numero_fila, 1)
        GridAverias[Numero_fila, 0].Text = Result_Trabajo["A.codigoaveria"]
        GridAverias[Numero_fila, 1].Text = Result_Trabajo["A.fecha"]
        GridAverias[Numero_fila, 2].Text = Result_Trabajo["V.nombre"]
        GridAverias[Numero_fila, 3].Text = Result_Trabajo["C.razonsocial"]
        GridAverias[Numero_fila, 4].Text = Result_Trabajo["T.nombre"]
        GridAverias[Numero_fila, 5].Text = Result_Trabajo["A.peso"]
        GridAverias[Numero_fila, 6].Text = Result_Trabajo["A.lote"]
        GridAverias[Numero_fila, 7].Text = Result_Trabajo["A.descripcion"]
        Result_Trabajo.MoveNext
    Wend
    FMain.Conexion_Base_Datos.Close
    GridAverias.Rows.Select(Linea_Averia, 1)

End
```

```
' ' Calculo automatico del numnero de Averia.
Private Sub Calculo_Numero_Averia()

    Dim Ano As String
    Dim Numero As String
    Dim Aux As Integer
    ano = Format(Now, "yy")

    If Utilidades.Tabla_Vacia("select * from averia") > 0 Then
        numero = Utilidades.Valor_Maximo("select * from averia order by 1",
"codigoaveria")
        aux = CInt(Mid(numero, 2, 9))
        TxtCodigoAveria.Text = "A" & CInt(aux + 1)
    End If
End Sub
```

```
Else
    TxtCodigoAveria.Text = "A" & ano & "0000001"
Endif

End
'' Borramos los datos despues de dar de alta una averia.
Private Sub Borrar_Datos_Averias()

    CmbArticulo.Index = -1
    CmbCliente.Index = -1
    CmbVendedor.Index = -1
    txtlote.Text = ""
    txtDescripcion.Text = ""
    DteFecha.Value = Date(Year(Now), Month(Now), Day(Now))
    vlbpeso.Value = 0

End
```

2.2.4 Modulo Almacén.

2.2.4.1 *Frm_Compras.class*

```
Public Codigo_Stock_Para_Edicion As String
Public Kilos_Stock_Para_Edicion As String
Public Materia_Prima_Para_Edicion As String
Public Proveedor_Stock_Para_Edicion As String
Public Linea_Compras As Integer
Public Uso_Filtro As Boolean

Public Sub Form_Open()
    Me.Title = "GESTION DE COMPRAS"
    Configurar_Objetos_Compras()
End

'''***** Comnfiguracion Botones *****'''

Public Sub CmbProveedor_Click()
    Dim Sql As String
    Dim Codigo_Proveedor As String
    If CmbProveedor.Text = "" Then
        CmbMateriaPrima.Clear
    Else
        Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla("select * from proveedor
where razonsocial = '" & CmbProveedor.text & "'", "codigoproveedor")
        sql = "select nombre from materiaprima where codigomateriaprima in (select
materiaprima from proveedormateriaprima where proveedor='" & Codigo_Proveedor &
"') order by 1"
        Utilidades.Llenar_Combobox(sql, CmbMateriaPrima, "nombre")
    Endif
End

Public Sub BtnAgregarCompra_Click()

    Dim Sql As String
    Dim Higiene As String
    Dim Carolep As String
    Dim Stock As String
    Dim Fecha_Recepcion As String
    Dim Fecha_Factura_Compra As String
    Dim Aux As String
    Dim Codigo_Proveedor As String
    Dim Codigo_Materia_Prima As String
    Dim Codigo_Maximo_Compra As String

    If Comprobar_Campos_Compras() Then
        Aux = DteFechaRecepcion.Value
        Fecha_Recepcion = utilidades.Fecha_Correcta(Aux)
        Aux = DteFFacturaCompra.Value
        Fecha_Factura_Compra = utilidades.Fecha_Correcta(Aux)
        If RdbHigieneAceptable.Value = True Then
            Higiene = "ACEPTABLE"
        Else
            higiene = "NO ACEPTABLE"
```



```

Endif
If RdbCarolepAceptable.Value = True Then
    Carolep = "ACEPTABLE"
Else
    Carolep = "NO ACEPTABLE"
Endif
If ChkAgregarStock.Value = True Then
    Stock = "S"
Else
    Stock = "N"
Endif
sql = "select * from proveedor where razonsocial='" & CmbProveedor.Text &
""
Codigo_Proveedor = utilidades.Obtener_Campo_Tabla(sql, "codigoproveedor")
sql = "select * from materiaprima where nombre='" & CmbMateriaPrima.Text &
""
Codigo_Materia_Prima = utilidades.Obtener_Campo_Tabla(sql,
"codigomateriaprima")
'Insertamos en la tabla Vendedor
sql = "insert into compras
(fecharecepcion,fechafacturacompra,cantidad,lotecompra,nfacturacompra,higiene,ca
rolep,precio,firma,observaciones,transporte,stock,proveedor,materiaprima)" &
" values('" & Fecha_Recepcion & "','" & Fecha_Factura_Compra & "','" &
CFloat(VlbCantidad.Text) & "','" & UCase(TxtNLoteCompra.Text) & "','" &
UCase(TxtNFacturaCompra.text) & "','" & Higiene &
"','" & Carolep & "','" & CFloat(VlbPrecio.Text) & "','" & TxtFirma.Text &
"','" & TxtObservaciones.text & "','" & CInt(VlbTransporte.Text) & "','" & Stock &
"','" & Codigo_Proveedor &
"','" & Codigo_Materia_Prima & "')"
utilidades.In_Act_Bor(sql)
' Añadimos a Stock
If ChkAgregarStock.Value = True Then

    Codigo_Maximo_Compra = utilidades.Valor_Maximo("select codigocompra from
compras order by codigocompra", "codigocompra")
    'Codigo_Maximo_Compra = CStr(Val(Codigo_Maximo_Compra) + 1)
    sql = "insert into stock
(codigostock,materiaprima,cantidad,lotecompra,proveedor,nfacturacompra,ffacturac
ompra,precio,usado)" &
" values('" & Codigo_Maximo_Compra & "','" & Codigo_Materia_Prima & "','" &
CFloat(VlbCantidad.Text) & "','" & UCase(TxtNLoteCompra.Text) & "','" &
Codigo_Proveedor & "','" & UCase(TxtNFacturaCompra.text) &
"','" & Fecha_Factura_Compra & "','" & CFloat(VlbPrecio.Text) & "','" & 'N')"
    utilidades.In_Act_Bor(sql)
Endif

Borrar_Campos_Compras()
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
Llenar_Datagrid_Compras(sql)
Endif

End

```

```
Public Sub BtnBorrarCompra_Click()  
    Dim Sql As String  
    Dim Numero_fila As Integer  
    Dim Respuesta As Integer  
    Dim Texto As String  
    Dim Producto As String  
    Dim Proveedor As String  
    Dim Lote_Compra As String  
    Dim Numero_Factura_Compra As String  
    Dim Stock_Usado As String  
    Dim Stock_Kilos As String  
    Dim Codigo_Proveedor As String  
    Dim Codigo_Materia_Prima As String  
    Dim Pasado_Stock As Boolean  
    Dim Id_Stock As String  
    Dim Codigo_Compra As String  
    Pasado_Stock = False  
  
    If GridCompras.Rows.Selection.Count > 0 Then  
        Numero_fila = GridCompras.Rows.Selection[0]  
        Codigo_Compra = GridCompras[Numero_fila, 0].Text  
        Producto = GridCompras[Numero_fila, 4].Text  
        Proveedor = GridCompras[Numero_fila, 3].Text  
        Lote_Compra = GridCompras[Numero_fila, 6].Text  
        Numero_Factura_Compra = GridCompras[Numero_fila, 7].Text  
        Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla("select * from proveedor  
where razonsocial = '" & Proveedor & "'", "codigoproveedor")  
        Codigo_Materia_Prima = Utilidades.Obtener_Campo_Tabla("select * from  
materiaprima where nombre='" & Producto & "'", "codigomateriaprima")  
        sql = "select * from stock where materiaprima='" & Codigo_Materia_Prima & "'  
and proveedor='" & Codigo_Proveedor & "' and lotecompra='" &  
        Lote_Compra & "' and nfacturacompra='" & Numero_Factura_Compra & "'" &  
        Texto = "Desea Borrar la compra con Producto:<b> " & Producto & "  
</b>, Proveedor:<b> " & Proveedor &  
        "</b> , Lote Compra:<b> " & Lote_Compra & " </b>, N° Factura Compra:<b> " &  
        Numero_Factura_Compra & " </b><br>"  
        If Utilidades.Tabla_Vacia(sql) > 0 Then  
            Pasado_Stock = True  
            Stock_Usado = Utilidades.Obtener_Campo_Tabla(sql, "usado")  
            Stock_Kilos = Utilidades.Obtener_Campo_Tabla(sql, "cantidad")  
            Texto = Texto & "La compra esta en Stock y "  
            If Stock_Usado = "N" Then  
                Texto = Texto & "<b>NO</b> ha sido usada quedando <b>" & Stock_Kilos &  
                "</b> Kilos en Stock"  
            Else  
                Texto = Texto & "<b>SI</b> ha sido usada quedando <b>" & Stock_Kilos &  
                "</b> Kilos en Stock"  
            Endif  
            Respuesta = Message.Delete(Texto, "Compras", "Compras y Stock", "Salir")  
        Else  
            Texto = Texto & "<b>La compra aun no se ha pasado a stock.</b>"  
            Respuesta = Message.Delete(Texto, "Si", "No")  
        End If  
  
        If respuesta = 1 Then  
            ' Tabla Compras  
            sql = "delete from compras where codigocompra=" & Codigo_Compra & "  
            Utilidades.In_Act_Bor(Sql)
```

```

        sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
        "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
        Llenar_Datagrid_Compras(sql)
        Else If respuesta = 2 And Pasado_Stock
        ' Tabla Stock
        sql = "delete from compras where codigocompra=" & Codigo_Compra & ""
        Utilidades.In_Act_Bor(Sql)
        sql = "select * from stock where materiaprima='" & Codigo_Materia_Prima &
"" and proveedor='" & Codigo_Proveedor & "' and lotecompra='" &
Lote_Compra & "' and nfacturacompra='" & Numero_Factura_Compra & ""
Id_Stock = utilidades.Obtener_Campo_Tabla(sql, "codigostock")
sql = "delete from stock where codigostock=" & Id_Stock & ""
        Utilidades.In_Act_Bor(Sql)
        sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
        "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
        Llenar_Datagrid_Compras(sql)
        Endif

    Endif
End

Public Sub BtnEditarCompra_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Dim Texto As String
    Dim Dia As String
    Dim Mes As String
    Dim Ano As String
    Dim Codigo_Compra As String
    Dim Higiene As String
    Dim Carolep As String
    Dim Stock As String
    Dim Fecha_Recepcion As String
    Dim Fecha_Factura_Compra As String
    Dim Aux As String
    Dim Codigo_Proveedor As String
    Dim Codigo_Materia_Prima As String
    Dim Usado_stock As String
    Dim Pasado_Stock As String
    Dim Codigo_Maximo_Compra As String

    If BtnEditarCompra.text = "EDITAR" And GridCompras.Rows.Selection.Count > 0
Then
        Numero_fila = GridCompras.Rows.Selection[0]
        Codigo_Compra = GridCompras[Numero_fila, 0].Text
        Dia = Mid(GridCompras[Numero_fila, 1].Text, 1, 2)
        Mes = Mid(GridCompras[Numero_fila, 1].Text, 4, 2)
        Ano = Mid(GridCompras[Numero_fila, 1].Text, 7, 4)
        DteFechaRecepcion.Value = Date(ano, mes, dia)

```

```

Dia = Mid(GridCompras[Numero_fila, 2].Text, 1, 2)
Mes = Mid(GridCompras[Numero_fila, 2].Text, 4, 2)
Ano = Mid(GridCompras[Numero_fila, 2].Text, 7, 4)
DteFFacturaCompra.Value = Date(ano, mes, dia)
CmbProveedor.Text = GridCompras[Numero_fila, 3].Text
CmbProveedor_Click()
CmbMateriaPrima.Text = GridCompras[Numero_fila, 4].Text
VlbCantidad.Value = GridCompras[Numero_fila, 5].Text
TxtNLoteCompra.text = GridCompras[Numero_fila, 6].Text
TxtNFacturaCompra.Text = GridCompras[Numero_fila, 7].Text
If GridCompras[Numero_fila, 8].Text = "ACEPTABLE" Then
    RdbHigieneAceptable.Value = True
Else
    RdbHigieneNoAceptable.value = True
Endif
If GridCompras[Numero_fila, 9].Text = "ACEPTABLE" Then
    RdbCarolepAceptable.Value = True
Else
    RdbCarolepNoAceptable.value = True
Endif
VlbPrecio.Value = GridCompras[Numero_fila, 10].Text
TxtFirma.text = GridCompras[Numero_fila, 11].Text
TxtObservaciones.Text = GridCompras[Numero_fila, 12].Text
VlbTransporte.Value = GridCompras[Numero_fila, 13].Text
If GridCompras[Numero_fila, 14].Text = "S" Then
    ChkAgregarStock.Value = True
    Pasado_Stock = "S"
Else
    ChkAgregarStock.Value = False
    Pasado_Stock = "N"
Endif

sql = "select * from proveedor where razonsocial='" & CmbProveedor.Text &
""
Codigo_Proveedor = utilidades.Obtener_Campo_Tabla(sql, "codigoproveedor")
sql = "select * from materiaprima where nombre='" & CmbMateriaPrima.Text &
""
Codigo_Materia_Prima = utilidades.Obtener_Campo_Tabla(sql,
"codigomateriaprima")
sql = "select * from stock where materiaprima='" & Codigo_Materia_Prima & "'
and proveedor='" & Codigo_Proveedor & "' and lote compra='" &
UCase(TxtNLoteCompra.Text) & "' and nfactura compra='" &
TxtNFacturaCompra.text & ""
If Pasado_Stock = "S" Then
    Usado_stock = utilidades.Obtener_Campo_Tabla(sql, "usado")
    Codigo_Stock_Para_Edicion = utilidades.Obtener_Campo_Tabla(sql,
"codigostock")
    Kilos_Stock_Para_Edicion = utilidades.Obtener_Campo_Tabla(sql, "cantidad")
    Materia_Prima_Para_Edicion = utilidades.Obtener_Campo_Tabla(sql,
"materiaprima")
    Proveedor_Stock_Para_Edicion = utilidades.Obtener_Campo_Tabla(sql,
"proveedor")
    If Usado_stock = "S" Then
        TxtlerrorCompras.Visible = True
        TxtlerrorCompras.Text = "La compra ha sido usada en Stock."
    Endif
End If

BtnBorrarCompra.Enabled = False

```

```

    BtnAgregarCompra.Enabled = False
    FrameFiltrar.Enabled = False
    GridCompras.Enabled = False
    BtnEditarCompra.text = "ACTUALIZAR"
Else
    If Comprobar_Campos_Compras() Then
        Aux = DteFechaRecepcion.Value
        Fecha_Recepcion = utilidades.Fecha_Correcta(Aux)
        Aux = DteFFacturaCompra.Value
        Fecha_Factura_Compra = utilidades.Fecha_Correcta(Aux)
        Codigo_Compra = GridCompras[Numero_fila, 0].Text
        If RdbHigieneAceptable.Value = True Then
            Higiene = "ACEPTABLE"
        Else
            higiene = "NO ACEPTABLE"
        Endif
        If RdbCarolepAceptable.Value = True Then
            Carolep = "ACEPTABLE"
        Else
            Carolep = "NO ACEPTABLE"
        Endif
        If ChkAgregarStock.Value = True Then
            Stock = "S"
        Else
            Stock = "N"
        Endif
        sql = "select * from proveedor where razonsocial='" & CmbProveedor.Text &
    ""
        Codigo_Proveedor = utilidades.Obtener_Campo_Tabla(sql, "codigoproveedor")
        sql = "select * from materiaprima where nombre='" & CmbMateriaPrima.Text &
    ""
        Codigo_Materia_Prima = utilidades.Obtener_Campo_Tabla(sql,
"codigomateriaprima")
        'Actualizamos en la tabla Compras
        sql = "update compras set fecharecepcion='" & Fecha_Recepcion &
        "'", fechafacturacompra='" & Fecha_Factura_Compra & "'", cantidad=" &
CFloat(VlbCantidad.Text) &
        "', lotecompra='" & UCase(TxtNLoteCompra.Text) & "', nfacturacompra='" &
UCase(TxtNFacturaCompra.Text) & "', higiene='" & higiene & "', carolep='" &
carolep & "', precio=" & CFloat(VlbPrecio.Text) &
        "', firma='" & TxtFirma.Text & "', observaciones='" & TxtObservaciones.Text &
        "', transporte=" & CInt(VlbTransporte.Text) & "', stock='" & Stock &
        "', proveedor='" &
        Codigo_Proveedor & "', materiaprima='" & Codigo_Materia_Prima & "'" where
codigocompra=" & Codigo_Compra
        Print sql
        utilidades.In_Act_Bor(sql)
        ' Añadimos a Stock si hemos marcado Y no esta aun en stock
        sql = "select * from stock where materiaprima='" & Codigo_Materia_Prima & "'"
and proveedor='" & Codigo_Proveedor & "'" and lotecompra='" &
        TxtNLoteCompra.Text & "'" and nfacturacompra='" & TxtNFacturaCompra.Text &
    ""
        If Not utilidades.Comprobar_existe_dato(sql) Then
            ' Si no existe en Stock pero hemos pulsado para añadir.
            If ChkAgregarStock.Value = True Then
                Codigo_Maximo_Compra = utilidades.Valor_Maximo("select codigocompra
from compras order by codigocompra", "codigocompra")
                ' Codigo_Maximo_Compra = CStr(Val(Codigo_Maximo_Compra) + 1)

```

```

        sql = "insert into stock
(codigostock,materiaprima,cantidad,lotecompra,proveedor,nfacturacompra,ffacturac
ompra,precio,usado)" &
        " values('" & Codigo_Maximo_Compra & "','" & Codigo_Materia_Prima &
        "','" & CFloat(VlbCantidad.Text) & "','" & UCase(TxtNLoteCompra.Text) & "','" &
Codigo_Proveedor & "','" & UCase(TxtNFacturaCompra.text) &
        "','" & Fecha_Factura_Compra & "','" & CFloat(VlbPrecio.Text) & "','" & 'N')"

        utilidades.In_Act_Bor(sql)
    Endif
Else
    ' Si ya estaba en Stock.
    If ChkAgregarStock.Value = False Then
        ' Lo queremos quitar del Stock
        Texto = "La compra esta en Stock Materia Prima: <b>" &
Materia_Prima_Para_Edicion & " </b>con la siguiente cantidad de Kilos: <b> " &
Kilos_Stock_Para_Edicion & " </b> Proveedor: <b>" &
        Proveedor_Stock_Para_Edicion & "</b> <br> ¿Desea Borrarla de Stock y
Agregar la nueva?"
        Respuesta = Message.Delete(Texto, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from stock where codigostock=" &
Codigo_Stock_Para_Edicion & ""
            utilidades.In_Act_Bor(sql)
            Codigo_Maximo_Compra = utilidades.Valor_Maximo("select codigocompra
from compras order by codigocompra", "codigocompra")
            'Codigo_Maximo_Compra = CStr(Val(Codigo_Maximo_Compra) + 1)
            sql = "insert into stock
(codigostock,materiaprima,cantidad,lotecompra,proveedor,nfacturacompra,ffacturac
ompra,precio,usado)" &
            " values('" & Codigo_Maximo_Compra & "','" & Codigo_Materia_Prima &
            "','" & CFloat(VlbCantidad.Text) & "','" & UCase(TxtNLoteCompra.Text) & "','" &
Codigo_Proveedor & "','" & UCase(TxtNFacturaCompra.text) &
            "','" & Fecha_Factura_Compra & "','" & CFloat(VlbPrecio.Text) &
            "','" & 'N')"

            utilidades.In_Act_Bor(sql)
        Endif
    Endif
Endif
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
Llenar_Datagrid_Compras(sql)
BtnBorrarCompra.Enabled = True
BtnAgregarCompra.Enabled = True
FrameFiltrar.Enabled = True
GridCompras.Enabled = True
Borrar_Campos_Compras()
BtnEditarCompra.text = "EDITAR"
TxtlerrorCompras.Visible = False
TxtlerrorCompras.text = ""
Borrar_Campos_Compras()

Endif

```

```
Endif

End

Public Sub BtnSalirCompras_Click()

    Me.Close

End

Public Sub ChkVariasCompras_Click()

    If ChkVariasCompras.value = True Then
        If Comprobar_Campos_Varias_Compras() Then
            DteFechaRecepcion.Enabled = False
            DteFFacturaCompra.Enabled = False
            TxtNFacturaCompra.Enabled = False
            TxtNLoteCompra.Enabled = False
            CmbProveedor.Enabled = False
        Else
            ChkVariasCompras.Value = False
        Endif
    Else
        CmbProveedor.Enabled = True
        TxtNFacturaCompra.Enabled = True
        TxtNLoteCompra.Enabled = True
        DteFechaRecepcion.Enabled = True
        DteFFacturaCompra.Enabled = True
    Endif
End

Public Sub BtnComprasInicio_Click()

    If GridCompras.Rows.Count > 0 Then
        GridCompras.Rows.UnselectAll
        Linea_Compras = 0
        GridCompras.Rows.Select(Linea_Compras, 1)
    Endif

End

Public Sub BtnComprasAnterior_Click()

    If GridCompras.Rows.Count > 0 Then
        If Linea_Compras = 0 Then
            GridCompras.Rows.Select(Linea_Compras, 1)
        Else
            GridCompras.Rows.UnselectAll
            Linea_Compras = Linea_Compras - 1
            GridCompras.Rows.Select(Linea_Compras, 1)
        Endif
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnComprasSiguiente_Click()

    If GridCompras.Rows.Count > 0 Then
        If Linea_Compras = GridCompras.Rows.Count - 1 Then
            GridCompras.Rows.Select(Linea_Compras, 1)
        Else
            GridCompras.Rows.UnselectAll
            Linea_Compras = Linea_Compras + 1
            GridCompras.Rows.Select(Linea_Compras, 1)
        Endif
    Endif

End

Public Sub BtnComprasFin_Click()

    If GridCompras.Rows.Count > 0 Then
        GridCompras.Rows.UnselectAll
        Linea_Compras = GridCompras.Rows.Count - 1
        GridCompras.Rows.Select(Linea_Compras, 1)

    Endif

End

Public Sub GridCompras_DblClick()

    Dim sql As String
    Dim Columna_Seleccionada As Integer
    If Not Uso_Filtro Then
        Columna_Seleccionada = GridCompras.Column
        Select Columna_Seleccionada
        Case 0
            sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
            "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
1"
        Case 1
            sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
            "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
2"
        Case 2
            sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
            "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
3"
        Case 3
            sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
```


Capítulo 2. CÓDIGO FUENTE

```
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
4"

Case 4
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
5"

Case 5
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
6"

Case 6
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
7"

Case 7
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
8"

Case 8
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
9"

Case 9
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
10"

Case 10
sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
```

Capítulo 2. CÓDIGO FUENTE

```
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
11"
    Case 11
        sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
        "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
12"
    Case 12
        sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
        "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
13"
    Case 13
        sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
        "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
14"
    Case 14
        sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
        "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima order by
15"
    End Select
    Llenar_Datagrid_Compras(sql)
End If

End

Public Sub TxtNFacturaCompra_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtNLoteCompra.SetFocus
    Endif

End

Public Sub TxtNLoteCompra_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlbCantidad.SetFocus
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub VlbCantidad_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlbPrecio.SetFocus
    Endif

End

Public Sub VlbPrecio_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlbTransporte.SetFocus
    Endif

End

Public Sub VlbTransporte_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtFirma.SetFocus
    Endif

End

Public Sub TxtFirma_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtObservaciones.SetFocus
    Endif

End

'''***** Fin Conmfiguracion Botones *****'''
'''*****'''

'''***** FILTROS *****'''
'''*****'''

Public Sub BtnFiltrar_Click()
    Dim Validar As Boolean
    Validar = True
    Uso_Filtro = True
    If ChkFechaCompra.value = False And ChkFechaRecepcion.value = False And
    ChkMateriaPrima.Value = False And ChkProveedor.value = False And
    ChkPasadoAStock.value = False Then
        validar = False
        Balloon.Error("Debe elegir un filtro.", PanelFiltrar)
    Else If ChkProveedor.value = True And CmbFProveedor.Text = ""
        validar = False
        Balloon.Error("Si activa el filtro de los Proveedores debe seleccionar el
Proveedor para filtrar.", CmbFProveedor)
    Else If ChkMateriaPrima.value = True And CmbfMateriaPrima.Text = ""
        validar = False
        Balloon.Error("Si activa el filtro de Materia Prima debe seleccionar la
Materia Prima para filtrar.", CmbfMateriaPrima)
    Endif
End
```

```
If validar Then
    Ejecutar_Filtros()
Endif

End

Public Sub BtnQuitarFiltro_Click()
    Uso_Filtro = False
    Quitar_Filtros()

End

Public Sub ChkFechaRecepcion_Click()

    If ChkFechaRecepcion.value = True Then
        PanelFRecepcion.Enabled = True
    Else
        PanelFRecepcion.Enabled = False
    Endif

End

Public Sub ChkFechaCompra_Click()

    If ChkFechaCompra.value = True Then
        PanelFCompra.Enabled = True

    Else
        PanelFCompra.Enabled = False
    Endif

End

Public Sub ChkProveedor_Click()

    If ChkProveedor.value = True Then
        PanelProveedor.Enabled = True
    Else
        PanelProveedor.Enabled = True
    Endif

End

Public Sub ChkMateriaPrima_Click()

    If ChkMateriaPrima.Value = True Then
        PanelMatreriaprima.Enabled = True
    Else
        PanelMatreriaprima.Enabled = False
    Endif

End

Public Sub ChkPasadoAStock_Click()

    If ChkPasadoAStock.Value = True Then
        PanelStock.Enabled = True
    Else
        PanelStock.Enabled = False
    Endif
```

End

```

''***** FIN FILTROS *****+
''
''*****

''***** Funciones privadas*****
''
''*****

'' Para quitar los filtros
Private Sub Quitar_Filtros()
    Dim Sql As String
    ChkFechaCompra.value = False
    ChkFechaRecepcion.value = False
    ChkMateriaPrima.value = False
    ChkProveedor.value = False
    ChkPasadoAStock.value = False
    CmbFMateriaPrima.Text = ""
    CmbFProveedor.text = ""
    PanelFRecepcion.Enabled = False
    PanelFCompra.Enabled = False
    PanelProveedor.Enabled = False
    PanelMatreriaprima.Enabled = False
    PanelStock.Enabled = False
    sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
    Llenar_Datagrid_Compras(sql)
End

'' Ejecutamos los filtros cuando se han configurado Correctamente.
Private Sub Ejecutar_Filtros()

    Dim Sql As String

    sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
"C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
    ' Seleccion del filtro Fecha Recepcion
    If ChkFechaRecepcion.value = True Then
        If DteFRecepcion.Value = DteFFRecepcion.value Then
            sql &= " and C.fecharecepcion='" &
Utilidades.Fecha_Correcta(DteFRecepcion.value) & "'"
        Else
            sql &= " and C.fecharecepcion between '" &
utilidades.Fecha_Correcta(DteFRecepcion.value) & "' and '" &
utilidades.Fecha_Correcta(DteFFRecepcion.value) & "'"
        Endif
    Endif

```

```

Endif
' Seleccion del filtro Fecha Compra
If ChkFechaCompra.value = True Then
    If DteFICompra.Value = DteFFCompra.value Then
        sql &= " and C.fechafacturacompra='" &
utilidades.Fecha_Correcta(DteFICompra.value) & "'"
    Else
        sql &= " and C.fechafacturacompra between '" &
utilidades.Fecha_Correcta(DteFICompra.value) & "' and '" &
utilidades.Fecha_Correcta(DteFFCompra.value) & "'"
    Endif
Endif
' Seleccion del filtro Proveedor
If ChkProveedor.value = True Then
    sql &= " and P.razonsocial='" & CmbFProveedor.text & "'"
Endif
' Seleccion del filtro Materia Prima
If ChkMateriaPrima.value = True Then
    sql &= " and M.nombre='" & CmbfMateriaPrima.text & "'"
Endif
' Seleccion del filtro Stock
If ChkPasadoAStock.value = True Then
    If RdbEnStock.value = True
        sql &= " and C.stock='S'"
    Else
        sql &= " and C.stock='N'"
    Endif
Endif

Llenar_Datagrid_Compras(sql)

End

'' Configuracion del Grid Compras<br>

Private Sub Configurar_Grid_Compras()
    Dim Sql As String

    GridCompras.Columns.Count = 15
    GridCompras.Header = GridCompras.Both
    GridCompras.Columns[0].Title = "Codigo"
    GridCompras.Columns[1].Title = "Fecha Recepcion"
    GridCompras.Columns[2].Title = "Fecha F. Compra"
    GridCompras.Columns[3].Title = "Proveedor"
    GridCompras.Columns[4].Title = "Materia Prima"
    GridCompras.Columns[5].Title = "Cantidad"
    GridCompras.Columns[6].Title = "Lote Compra"
    GridCompras.Columns[7].Title = "N. F. Compra"
    GridCompras.Columns[8].Title = "Higiene"
    GridCompras.Columns[9].Title = "C. O. Leptidos"
    GridCompras.Columns[10].Title = "Precio"
    GridCompras.Columns[11].Title = "Firma"
    GridCompras.Columns[12].Title = "Observaciones"
    GridCompras.Columns[13].Title = "Transporte"
    GridCompras.Columns[14].Title = "Stock"

    GridCompras.Columns[0].Width = 90
    GridCompras.Columns[1].Width = 150

```

```

GridCompras.Columns[2].Width = 150
GridCompras.Columns[3].Width = 200
GridCompras.Columns[4].Width = 150
GridCompras.Columns[5].Width = 100
GridCompras.Columns[6].Width = 150
GridCompras.Columns[7].Width = 150
GridCompras.Columns[8].Width = 90
GridCompras.Columns[9].Width = 120
GridCompras.Columns[10].Width = 100
GridCompras.Columns[11].Width = 150
GridCompras.Columns[12].Width = 150
GridCompras.Columns[13].Width = 150
GridCompras.Columns[14].Width = 90
    sql = "select
C.codigocompra,C.fecharecepcion,C.fechafacturacompra,P.razonsocial,
M.nombre,C.cantidad,C.Lotecompra,C.nfacturacompra,C.higiene,C.carolep,C.precio,C
.firma,C.observaciones,C.transporte," &
    "C.stock From compras C, proveedor P,materiaprima M where
C.proveedor=P.codigoproveedor and C.materiaprima=M.codigomateriaprima"
    Llenar_Datagrid_Compras(sql)

End
'' Llenamos DataGridView Lote <br>
'' Parametros:<br>
''     Sql --> Sentencia para cargar datos en el Dataset
Private Sub Llenar_Datagrid_Compras(sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    Numero_fila = GridCompras.Rows.Count
    If GridCompras.Rows.Count > 0 Then
        GridCompras.Rows.Remove(0, Numero_fila)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    While Result_Trabajo.Available
        Numero_fila = GridCompras.Rows.Count
        GridCompras.Rows.Insert(Numero_fila, 1)
        GridCompras[Numero_fila, 0].Text = Result_Trabajo["C.codigocompra"]
        GridCompras[Numero_fila, 1].Text = Result_Trabajo["C.fecharecepcion"]
        GridCompras[Numero_fila, 2].Text = Result_Trabajo["C.fechafacturacompra"]
        GridCompras[Numero_fila, 3].Text = Result_Trabajo["P.razonsocial"]
        GridCompras[Numero_fila, 4].Text = Result_Trabajo["M.nombre"]
        GridCompras[Numero_fila, 5].Text = Result_Trabajo["C.cantidad"]
        GridCompras[Numero_fila, 6].Text = Result_Trabajo["C.Lotecompra"]
        GridCompras[Numero_fila, 7].Text = Result_Trabajo["C.nfacturacompra"]
        GridCompras[Numero_fila, 8].Text = Result_Trabajo["C.higiene"]
        GridCompras[Numero_fila, 9].Text = Result_Trabajo["C.carolep"]
        GridCompras[Numero_fila, 10].Text = Result_Trabajo["C.precio"]
        GridCompras[Numero_fila, 11].Text = Result_Trabajo["C.firma"]
        GridCompras[Numero_fila, 12].Text = Result_Trabajo["C.observaciones"]
        GridCompras[Numero_fila, 13].Text = Result_Trabajo["C.transporte"]
        GridCompras[Numero_fila, 14].Text = Result_Trabajo["C.stock"]
        Result_Trabajo.MoveNext
    Wend
    FMain.Conexion_Base_Datos.Close
    GridCompras.Rows.Select(Linea_Compras, 1)

```

```
End

'' Configurarnos cuando abrimos el formulario.
'' Llenamos los combobox de Proveedores y Materias Primas.
'' Si las tablas estan vacias muestra un error.

Private Sub Configurar_Objeto_Compras()
    Dim Error_Tablas_Vacias As String
    Dim Error_V As Boolean
    Error_V = False
    Linea_Compras = 0
    Uso_Filtro = False
    If Utilidades.Tabla_Vacia("select * from proveedor") = 0 Then
        Error_Tablas_Vacias = "Deber Cear Proveedores"
        Error_V = True
    Else
        CmbProveedor.Clear
        CmbFProveedor.clear
        Utilidades.Llenar_Combobox("select * from proveedor", CmbProveedor,
"razonsocial")
        CmbProveedor.text = ""
        Utilidades.Llenar_Combobox("select * from proveedor", CmbFProveedor,
"razonsocial")
        CmbFProveedor.text = ""
    Endif

    If Utilidades.Tabla_Vacia("select * from materiaprima") = 0 Then

        Error_V = True
        If Error_Tablas_Vacias = "" Then
            Error_Tablas_Vacias = "Debe Cear Materias Primas en Articulos."
        Else
            Error_Tablas_Vacias = Error_Tablas_Vacias & " y Materias Primas en
Articulos."
        Endif
    Else
        CmbFMateriaPrima.Clear
        CmbMateriaPrima.Clear
        'Utilidades.Llenar_Combobox("select * from materiaprima", CmbMateriaPrima,
"nombre")
        Utilidades.Llenar_Combobox("select * from materiaprima", CmbFMateriaPrima,
"nombre")
        CmbfMateriaPrima.text = ""
    Endif

    If Error_V Then
        TxtlerrorCompras.Visible = True
        TxtlerrorCompras.text = Error_Tablas_Vacias
    Endif
    DteFechaRecepcion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFFacturaCompra.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFFRecepcion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFIRecepcion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFFCompra.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFICompra.Value = Date(Year(Now), Month(Now), Day(Now))
    TxtlerrorCompras.text = Error_Tablas_Vacias
    Configurar_Grid_Compras()
End
```


Capítulo 2. CÓDIGO FUENTE

```
' ' Procedimiento para borrar los campos del formulario
Private Sub Borrar_Campos_Compras()

    If ChkVariasCompras.Value = True Then
        TxtObservaciones.Text = ""
        TxtFirma.Text = ""
        RdbCarolepAceptable.Value = True
        RdbHigieneAceptable.Value = True
        CmbMateriaPrima.Index = -1
        VlbTransporte.text = ""
        VlbCantidad.text = ""
        VlbPrecio.text = ""
        ChkAgregarStock.Value = False
    Else
        DteFechaRecepcion.Value = Date(Year(Now), Month(Now), Day(Now))
        DteFacturaCompra.Value = Date(Year(Now), Month(Now), Day(Now))
        TxtObservaciones.Text = ""
        TxtFirma.Text = ""
        TxtNFacturaCompra.Text = ""
        TxtNLoteCompra.Text = ""
        RdbCarolepAceptable.Value = True
        RdbHigieneAceptable.Value = True
        CmbMateriaPrima.index = -1
        CmbProveedor.index = -1
        VlbTransporte.text = ""
        VlbCantidad.text = ""
        VlbPrecio.text = ""
        ChkAgregarStock.Value = False
    Endif

End

' ' Funcion para comprobar los Campos de Compras <br>
' ' Parametros:<br>
' ' Devuelve:<br>
' '     TRUE --> Si los valores son correctos.<br>
' '     FALSE --> valores erroneos.
Private Function Comprobar_Campos_Compras() As Boolean

    Dim Validar As Boolean
    Validar = True
    If TxtNFacturaCompra.Text = "" Then
        Balloon.Error("Debe introducir N° Facatura de la Compra", TxtNFacturaCompra)
        Validar = False
        TxtNFacturaCompra.SetFocus
    Else If TxtNLoteCompra.text = ""
        Balloon.Error("Debe introducir el N° Lote de Compra", TxtNLoteCompra)
        Validar = False
        TxtNLoteCompra.SetFocus
    Else If CmbMateriaPrima.Text = ""
        Balloon.Error("Debe elegir la Materia Prima de la compra.", CmbMateriaPrima)
        Validar = False
    Else If CmbProveedor.Text = ""
        Balloon.Error("Debe elegir un Proveedor de la compra.", CmbProveedor)
        Validar = False
    Else If VlbCantidad.Text = "0"
        Balloon.Error("Debe añadir Cantidad en Kilos de la Materia Prima de la compra.", VlbCantidad)
        Validar = False
    End If
```

```
VlbCantidad.SetFocus
Else If VlbPrecio.Text = "0"
    Balloon.Error("Debe elegir Precio en Euros de la Materia Prima de la compra.", VlbPrecio)
    Validar = False
    VlbPrecio.SetFocus
Endif

Return Validar
End

'' Funcion para comprobar los Campos Varias Compras <br>
'' Parametros:<br>
'' Devuelve:<br>
'' TRUE --> Si los valores son correctos.<br>
'' FALSE --> valores erroneos.
Private Function Comprobar_Campos_Varias_Compras() As Boolean

    Dim Validar As Boolean
    Validar = True
    If TxtNFacturaCompra.Text = "" Then
        Balloon.Error("Debe introducir N° Facatura de la Compra", TxtNFacturaCompra)
        Validar = False
        TxtNFacturaCompra.SetFocus
    Else If TxtNLoteCompra.text = ""
        Balloon.Error("Debe introducir el N° Lote de Compra", TxtNLoteCompra)
        Validar = False
        TxtNLoteCompra.SetFocus
    Else If CmbProveedor.Text = ""
        Balloon.Error("Debe elegir un Proveedor de la compra.", CmbProveedor)
        Validar = False
    Endif

    Return Validar
End
```

2.2.4.2 Frm_Stock.class

```
Public Linea_Stock As Integer
Public Uso_filtro As Boolean

Public Sub Form_Open()
    Me.Title = "GESTION DE STOCK"
    Configurar_Objeto_Stock()
End

Public Sub BtnSalirStock_Click()
    Me.Close
End

Public Sub BtnBorrarStocCero_Click()

    Dim Respuesta As Integer
    Dim Sql As String
    Respuesta = Message.Delete("Desea Borrar los Stock con Cantidad <b>0</b>",
    "Si", "No")
    If Respuesta = 1 Then
        utilidades.In_Act_Bor("delete from stock where cantidad=0")
    End If
End
```

Capítulo 2. CÓDIGO FUENTE

```
        sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
        " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima"
        Llenar_Datagrid_Stock(sql)
    Endif
End
Public Sub BtnBorrarLineaStock_Click()

    Dim Respuesta As Integer
    Dim Sql As String
    Dim Numero_fila As Integer

    If GridStock.Rows.Selection.Count > 0 Then
        Numero_fila = GridStock.Rows.Selection[0]
        Respuesta = Message.Delete("Desea Borrar linea Stock Materia Prima: <b>" &
GridStock[Numero_fila, 1].Text & " </b>", "Si", "No")
        If Respuesta = 1 Then
            utilidades.In_Act_Bor("delete from stock where codigostock=" &
GridStock[Numero_fila, 0].Text)
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima"
            Llenar_Datagrid_Stock(sql)
        Endif

    Endif

End

Public Sub BtnLInicio_Click()

    If GridStock.Rows.Count > 0 Then
        GridStock.Rows.UnselectAll
        Linea_Stock = 0
        GridStock.Rows.Select(Linea_Stock, 1)
    Endif

End

Public Sub BtnLPrevio_Click()

    If GridStock.Rows.Count > 0 Then
        If Linea_Stock = 0 Then
            GridStock.Rows.Select(Linea_Stock, 1)
        Else
            GridStock.Rows.UnselectAll
            Linea_Stock = Linea_Stock - 1
            GridStock.Rows.Select(Linea_Stock, 1)
        Endif
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnLSiguiente_Click()

    If GridStock.Rows.Count > 0 Then
        If Linea_Stock = GridStock.Rows.Count - 1 Then
            GridStock.Rows.Select(Linea_Stock, 1)
        Else
            GridStock.Rows.UnselectAll
            Linea_Stock = Linea_Stock + 1
            GridStock.Rows.Select(Linea_Stock, 1)
        Endif
    Endif

End

Public Sub BtnLFin_Click()

    If GridStock.Rows.Count > 0 Then
        GridStock.Rows.UnselectAll
        Linea_Stock = GridStock.Rows.Count - 1
        GridStock.Rows.Select(Linea_Stock, 1)
    Endif

End

Public Sub GridStock_DblClick()

    Dim sql As String
    Dim Columna_Seleccionada As Integer

    If Not Uso_filtro Then
        Columna_Seleccionada = GridStock.Column
        Select Columna_Seleccionada
        Case 0
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
1"
        Case 1
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
2"
        Case 2
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
3"
        Case 3
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
```

```

        " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
4"
        Case 4
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
5"
        Case 5
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
6"
        Case 6
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
7"
        Case 7
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
8"
        Case 8
            sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
            " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima order by
9"
        End Select
        Llenar_Datagrid_Stock(sql)
    Endif

End

'***** FILTROS *****
'
'*****
Public Sub BtnFiltrar_Click()
    Dim Validar As Boolean
    Validar = True
    Uso_filtro = True
    If ChkFechaCompra.value = False And ChkMateriaPrima.Value = False And
ChkProveedor.value = False And ChkUsado.value = False And ChkCantidad0.value =
False Then
        validar = False
        Balloon.Error("Debe elegir un filtro.", PanelFiltrar)
    Else If ChkProveedor.value = True And CmbProveedor.Text = ""
        validar = False

```

```
Balloon.Error("Si activa el filtro de los Proveedores debe seleccionar el
Proveedor para filtrar.", CmbProveedor)
Else If ChkMateriaPrima.value = True And CmbMateriaPrima.Text = ""
    validar = False
    Balloon.Error("Si activa el filtro de Materia Prima debe seleccionar la
Materia Prima para filtrar.", CmbMateriaPrima)
Endif
If validar Then
    Ejecutar_Filtros()
Endif

End
Public Sub BtnQuitarFiltro_Click()
    Uso_filtro = False
    Quitar_Filtros()

End

Public Sub ChkFechaCompra_Click()

    If ChkFechaCompra.value = True Then
        PanelFCompra.Enabled = True
    Else
        PanelFCompra.Enabled = False
    Endif

End

Public Sub ChkProveedor_Click()

    If ChkProveedor.value = True Then
        PanelProveedor.Enabled = True
    Else
        PanelProveedor.Enabled = False
    Endif

End

Public Sub ChkMateriaPrima_Click()

    If ChkMateriaPrima.Value = True Then
        PanelMatreriaprima.Enabled = True
    Else
        PanelMatreriaprima.Enabled = False
    Endif

End

Public Sub ChkUsado_Click()

    If ChkUsado.Value = True Then
        PanelStock.Enabled = True
    Else
        PanelStock.Enabled = False
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
'***** FIN FILTROS *****'+
''
''*****
''***** Funciones privadas*****
''
''*****

'' Para quitar los filtros
Private Sub Quitar_Filtros()
    Dim Sql As String
    ChkFechaCompra.value = False

    ChkMateriaPrima.value = False
    ChkProveedor.value = False
    ChkUsado.value = False
    CmbMateriaPrima.Text = ""
    CmbProveedor.text = ""
    PanelFCompra.Enabled = False
    PanelProveedor.Enabled = False
    PanelMatreriaprima.Enabled = False
    PanelStock.Enabled = False
    ChkCantidad0.value = False
    lblStockFiltrado.Visible = False
    sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
    " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima"
    Llenar_Datagrid_Stock(sql)

End

'' Ejecutamos los filtros cuando se han configurado Correctamente.
Private Sub Ejecutar_Filtros()

    Dim Sql As String

    sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
    " From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima"

    ' Seleccion del filtro Fecha Compra
    If ChkFechaCompra.value = True Then
        If DteFICompra.Value = DteFFCompra.value Then
            sql &= " and S.ffacturacompra='" &
utilidades.Fecha_Correcta(DteFICompra.value) & "'"
        Else
            sql &= " and S.ffacturacompra between '" &
utilidades.Fecha_Correcta(DteFICompra.value) & "' and '" &
utilidades.Fecha_Correcta(DteFFCompra.value) & "'"
        Endif
    Endif
    ' Seleccion del filtro Proveedor
    If ChkProveedor.value = True Then
        sql &= " and P.razonsocial='" & CmbProveedor.text & "'"
    Endif
```

```
' Seleccion del filtro Materia Prima
If ChkMateriaPrima.value = True Then
    sql &= " and M.nombre='" & CmbMateriaPrima.text & "'"
Endif
' Seleccion del filtro Stock
If ChkUsado.value = True Then
    If RdbUsado.value = True
        sql &= " and S.usado='S'"
    Else
        sql &= " and S.usado='N'"
    Endif
Endif
If ChkCantidad0.value = True Then
    sql &= " and S.cantidad = 0 "
Endif
LblStockFiltrado.Visible = True
Llenar_Datagrid_Stock(sql)

End

Private Sub Configurar_Objeto_Stock()
    Dim Sql As String
    CmbProveedor.Clear
    Linea_Stock = 0
    Uso_filtro = False
    Utilidades.Llenar_Combobox("select * from proveedor", CmbProveedor,
"razonsocial")
    CmbMateriaPrima.Clear
    Utilidades.Llenar_Combobox("select * from materiaprima", CmbMateriaPrima,
"nombre")
    sql = "select
S.codigostock,M.nombre,S.cantidad,S.lotecompra,P.razonsocial,S.nfacturacompra,S.
ffacturacompra,S.precio,S.usado" &
" From stock S, proveedor P,materiaprima M where
S.proveedor=P.codigoproveedor and S.materiaprima=M.codigomateriaprima"
    Configurar_Grid_Stock()
    Llenar_Datagrid_Stock(sql)
    DteFICompra.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFFCompra.Value = Date(Year(Now), Month(Now), Day(Now))

End

Private Sub Configurar_Grid_Stock()
    GridStock.Columns.Count = 9
    GridStock.Header = GridStock.Both
    GridStock.Columns[0].Title = "Codigo"
    GridStock.Columns[1].Title = "Materia Prima"
    GridStock.Columns[2].Title = "Cantidad"
    GridStock.Columns[3].Title = "Lote Compra"
    GridStock.Columns[4].Title = "Proveedor"
    GridStock.Columns[5].Title = "N Factura Compra"
    GridStock.Columns[6].Title = "F Factura Compra"
    GridStock.Columns[7].Title = "Precio"
    GridStock.Columns[8].Title = "Usado"
    GridStock.Columns[0].Width = 100
    GridStock.Columns[1].Width = 200
    GridStock.Columns[2].Width = 90
    GridStock.Columns[3].Width = 150
    GridStock.Columns[4].Width = 150
```


Capítulo 2. CÓDIGO FUENTE

```
GridStock.Columns[5].Width = 200
GridStock.Columns[6].Width = 200
GridStock.Columns[7].Width = 100
GridStock.Columns[8].Width = 90

End

'' Llenamos DataGridView Stock <br>
'' Parametros:<br>
''     Sql --> Sentencia para cargar datos en el Dataset
Private Sub Llenar_Datagrid_Stock(Sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer

    Numero_fila = GridStock.Rows.Count
    If GridStock.Rows.Count > 0 Then
        GridStock.Rows.Remove(0, Numero_fila)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    If Result_Trabajo.Count > 0 Then
        While Result_Trabajo.Available
            Numero_fila = GridStock.Rows.Count
            GridStock.Rows.Insert(Numero_fila, 1)
            GridStock[Numero_fila, 0].Text = Result_Trabajo["S.codigostock"]
            GridStock[Numero_fila, 1].Text = Result_Trabajo["M.nombre"]
            GridStock[Numero_fila, 2].Text = Result_Trabajo["S.cantidad"]
            GridStock[Numero_fila, 3].Text = Result_Trabajo["S.lotecompra"]
            GridStock[Numero_fila, 4].Text = Result_Trabajo["P.razonsocial"]
            GridStock[Numero_fila, 5].Text = Result_Trabajo["S.nfacturacompra"]
            GridStock[Numero_fila, 6].Text = Result_Trabajo["S.ffacturacompra"]
            GridStock[Numero_fila, 7].Text = Result_Trabajo["S.precio"]
            GridStock[Numero_fila, 8].Text = Result_Trabajo["S.usado"]
            Result_Trabajo.MoveNext
        Wend
        FMain.Conexion_Base_Datos.Close
        GridStock.Rows.Select(Linea_Stock, 1)
        Colorear_GridView_Stock()
    End If
End

'' Coloreamos las filas segun cantidad quede en stock<br>
'' cantidad < 20% Color Naranja<br>
'' 20% < cantidad < 60% Color Amarillo<br>
'' cantidad < 60% Color Verde<br>
'' cantidad 0% Color Blanco<br>

Private Sub Colorear_GridView_Stock()
    Dim Numero_fila As Integer
    Dim Sql As String
    Dim Cantidad_compra As String
    Dim Codigo_Materia_Prima_Compra As String
    Dim Valor As Float
    Dim Porciento As Float
    Dim Lote_stock As String
    Dim NFactura_stock As String
    Dim Cantidad_Stock As String
    Dim Nombre_MP As String
```

```

For Numero_fila = 0 To GridStock.Rows.Count - 1

    Lote_stock = GridStock[Numero_fila, 3].Text
    NFactura_stock = GridStock[Numero_fila, 5].Text
    Cantidad_Stock = GridStock[Numero_fila, 2].Text
    Nombre_MP = GridStock[Numero_fila, 1].Text
    ' Comprobamos cuanto queda de stock para colorear
    Sql = "select * from materiaprima where nombre='" & Nombre_MP & "'"
    Codigo_Materia_Prima_Compra = utilidades.Obtener_Campo_Tabla(Sql,
"codigomateriaprima")
    Sql = "select * from compras where lotecompra='" & Lote_stock & "'" and
nfacturacompra='" & NFactura_stock & "'" and materiaprima='" &
Codigo_Materia_Prima_Compra & "'"
    Cantidad_compra = utilidades.Obtener_Campo_Tabla(Sql, "cantidad")
    If Cantidad_compra = "" Or Cantidad_compra = "0" Then
        valor = 1.0
    Else
        valor = CFloat(Cantidad_compra)
    Endif
    Porciento = (CFloat(Cantidad_Stock) * 100) / valor

    If Porciento = 0 Then
    Else If Porciento < 20
        GridStock[Numero_fila, 0].Background = Color.Orange
        GridStock[Numero_fila, 1].Background = Color.Orange
        GridStock[Numero_fila, 2].Background = Color.Orange
        GridStock[Numero_fila, 3].Background = Color.Orange
        GridStock[Numero_fila, 4].Background = Color.Orange
        GridStock[Numero_fila, 5].Background = Color.Orange
        GridStock[Numero_fila, 6].Background = Color.Orange
        GridStock[Numero_fila, 7].Background = Color.Orange
        GridStock[Numero_fila, 8].Background = Color.Orange
    Else If porciento > 20 And Porciento < 60
        GridStock[Numero_fila, 0].Background = Color.Yellow
        GridStock[Numero_fila, 1].Background = Color.Yellow
        GridStock[Numero_fila, 2].Background = Color.Yellow
        GridStock[Numero_fila, 3].Background = Color.Yellow
        GridStock[Numero_fila, 4].Background = Color.Yellow
        GridStock[Numero_fila, 5].Background = Color.Yellow
        GridStock[Numero_fila, 6].Background = Color.Yellow
        GridStock[Numero_fila, 7].Background = Color.Yellow
        GridStock[Numero_fila, 8].Background = Color.Yellow
    Else If porciento > 60
        GridStock[Numero_fila, 0].Background = Color.Green
        GridStock[Numero_fila, 1].Background = Color.Green
        GridStock[Numero_fila, 2].Background = Color.Green
        GridStock[Numero_fila, 3].Background = Color.Green
        GridStock[Numero_fila, 4].Background = Color.Green
        GridStock[Numero_fila, 5].Background = Color.Green
        GridStock[Numero_fila, 6].Background = Color.Green
        GridStock[Numero_fila, 7].Background = Color.Green
        GridStock[Numero_fila, 8].Background = Color.Green
    Endif
Next

End

```

2.2.5 Modulo Lotes.

2.2.5.1 Frm_Lotes.class

```

Public Linea_Lote As Integer ' Variable para controlar la linea de posicion del
DatagridLote y poder movernos con flechas.
Public Linea_lote_Detalle As Integer ' Variable para controlar la linea de
posicion del DatagridLoteDetalle y poder movernos con flechas.
Public Uso_Filtro As Boolean 'Controlamos que estamos filtrando datos para pdoer
controlar ordenar por columna.
Public Contador_Linea_Detalle_lote As Integer ' Llevamos el control de las
lineas de detalle.
Public Articulo_Sin_stock As String ' Almacenamos el nombre del articulo que se
ha elegido en el combo sin stock para no dejar crear lote

Public Sub Form_Open()
    Me.Title = "GESTION DE LOTES"
    Configurar_Objetos_Lotes()
End

Public Sub Form_Close()
    Dim Numero_Lineas As Integer
    Dim i As Integer
    Dim Sql As String
    Dim Detalle_Cantidad As String
    Dim Detalle_Coste As String
    Dim Detalle_Codigo_Materia_Prima As String
    Dim Detalle_Codigo_Proveedor As String
    Dim Detalle_NLote As String
    Dim Detalle_Fecha As String
    Dim Detalle_NFactura As String
    Dim Detalle_Codigo As String
    Dim Stock_Cantidad As String
    Dim Compra_Id As String
    Dim Compra_Precio As String
    Dim Compra_Cantidad As String
    If GridLoteDetalle.Rows.Count > 0 And BtnAgregarLote.Text = "GRABAR" Then
        Numero_Lineas = GridLoteDetalle.Rows.Count
        For i = 0 To Numero_Lineas - 1
            Detalle_Codigo = GridLoteDetalle[i, 0].Text
            Detalle_Cantidad = GridLoteDetalle[i, 2].Text
            Detalle_Coste = GridLoteDetalle[i, 3].Text
            Detalle_NLote = GridLoteDetalle[i, 4].Text
            Detalle_Fecha = GridLoteDetalle[i, 7].Text
            Detalle_NFactura = GridLoteDetalle[i, 6].Text
            sql = "select codigomateriaprima from materiaprima where nombre='" &
GridLoteDetalle[i, 8].Text & "'"
            Detalle_Codigo_Materia_Prima = Utilidades.Obtener_Campo_Tabla(sql,
"codigomateriaprima")
            sql = "select codigoproveedor from proveedor where razonsocial='" &
GridLoteDetalle[i, 5].Text & "'"
            Detalle_Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla(sql,
"codigoproveedor")
            sql = "select * from stock where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and ffacturacompra='" & Detalle_Fecha &
"' and lotecompra='" & Detalle_NLote & "' and proveedor='" &
Detalle_Codigo_Proveedor & "' and nfacturacompra='" & Detalle_NFactura & "'"
            ' Comprobamos que tenemos stock de esta materia prima.
        
```

```

        If Utilidades.Tabla_Vacia(sql) = 1 Then
            ' Si tenemos Stock actualizamos la cantidad en Stock
            Stock_Cantidad = Format(CFloat(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad")) + CFloat(Detalle_Cantidad), "##0.00")
            sql = "update stock set cantidad=" & Stock_Cantidad & " where
materiaprima='" & Detalle_Codigo_Materia_Prima & "'and ffacturacompra='" &
Detalle_Fecha &
            "'" and lotecompra='" & Detalle_NLote & "'" and proveedor='" &
Detalle_Codigo_Proveedor & "'" and nfacturacompra='" & Detalle_NFactura & "'"
            Utilidades.In_Act_Bor(Sql)

            ' Obtenemos la cantidad de compras y si es igual no se ha usado el Stock
y actualizamos Usado a N
            sql = "select * from compras where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and fechafacturacompra='" & Detalle_Fecha &
            "'" and lotecompra='" & Detalle_NLote & "'" and proveedor='" &
Detalle_Codigo_Proveedor & "'" and nfacturacompra='" & Detalle_NFactura & "'"
            Compra_Cantidad = Format(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad"), "##0.00")
            If CFloat(Stock_Cantidad) = CFloat(Compra_Cantidad) Then
                sql = "update stock set usado='N' where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and ffacturacompra='" & Detalle_Fecha &
                "'" and lotecompra='" & Detalle_NLote & "'" and proveedor='" &
Detalle_Codigo_Proveedor & "'" and nfacturacompra='" & Detalle_NFactura & "'"
                Utilidades.In_Act_Bor(Sql)
            Endif
        Else
            ' Si no tenemos stock lo añadimos a Stock
            sql = "select * from compras where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and fechafacturacompra='" & Detalle_Fecha &
            "'" and lotecompra='" & Detalle_NLote & "'" and proveedor='" &
Detalle_Codigo_Proveedor & "'" and nfacturacompra='" & Detalle_NFactura & "'"
            Compra_Id = Utilidades.Obtener_Campo_Tabla(sql, "codigocompra")
            Compra_Precio = Utilidades.Obtener_Campo_Tabla(sql, "precio")
            sql = "insert into stock
(codigostock,materiaprima,cantidad,lotecompra,proveedor,nfacturacompra,ffacturac
ompra,precio,usado)" &
            " values('" & Compra_Id & "','" & Detalle_Codigo_Materia_Prima & "','" &
Detalle_Cantidad & "','" & Detalle_NLote & "','" & Detalle_Codigo_Proveedor &
            "','" & Detalle_NFactura &
            "','" & Detalle_Fecha & "','" & Compra_Precio & "','" & 'N')"
            Utilidades.In_Act_Bor(Sql)
        Endif

    Next
Endif
sql = "delete from lotedetalle where codigolote='" & Detalle_Codigo & "'"
Utilidades.In_Act_Bor(Sql)

End

Public Sub BtnSalirStock_Click()
    Me.Close
End

Public Sub GridLoteDetalle_DblClick()
    If Not Uso_Filtro Then
    Endif
End

```

Capítulo 2. CÓDIGO FUENTE

```
'***** FILTROS *****'
'
'*****
Public Sub ChkFechaCreacionLote_Click()

    If ChkFechaCreacionLote.value = True Then
        PanelFCreacion.Enabled = True
    Else
        PanelFCreacion.Enabled = False
    Endif

End

Public Sub ChkArticulo_Click()

    If ChkArticulo.value = True Then
        PanelArticulos.Enabled = True
    Else
        PanelArticulos.Enabled = False
    Endif

End

Public Sub BtnFiltrar_Click()

    Uso_Filtro = True
    Ejecutar_Filtros()

End

Public Sub BtnQuitarFiltro_Click()

    Uso_Filtro = False
    Quitar_Filtros()

End
'*****FIN FILTROS *****'
'
'*****

'***** Botones Manegar Datagrid Lotes *****'
'
'*****

Public Sub BtnLInicio_Click()
    Dim Sql As String
    If GridLote.Rows.Count > 0 And BtnAgregarLote.text = "NUEVO" And
BtnAgregarLote.Enabled = True Then
        GridLote.Rows.UnselectAll
        Linea_lote = 0
        GridLote.Rows.Select(Linea_lote, 1)
        sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
        "proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M on
L.materiaprime = M.codigomateriaprime where L.codigolote='" &
GridLote[Linea_Lote, 0].Text & "'"
        Llenar_Datagrid_LoteDetalle(sql)
    Endif
```

End

```
Public Sub BtnLPrevio_Click()  
    Dim Sql As String  
    If GridLote.Rows.Count > 0 And BtnAgregarLote.text = "NUEVO" And  
BtnAgregarLote.Enabled = True Then  
        If Linea_lote = 0 Then  
            GridLote.Rows.Select(Linea_Lote, 1)  
        Else  
            GridLote.Rows.UnselectAll  
            Linea_lote = Linea_lote - 1  
            GridLote.Rows.Select(Linea_lote, 1)  
        Endif  
        sql = "select  
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact  
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &  
        "proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M on  
L.materiaprime = M.codigomateriaprime where L.codigolote='" &  
GridLote[Linea_Lote, 0].Text & "'" &  
        Llenar_Datagrid_LoteDetalle(sql)  
    Endif
```

End

```
Public Sub BtnLSiguiente_Click()  
    Dim Sql As String  
    If GridLote.Rows.Count > 0 And BtnAgregarLote.text = "NUEVO" And  
BtnAgregarLote.Enabled = True Then  
        If Linea_lote = GridLote.Rows.Count - 1 Then  
            GridLote.Rows.Select(Linea_lote, 1)  
        Else  
            GridLote.Rows.UnselectAll  
            Linea_lote = Linea_lote + 1  
            GridLote.Rows.Select(Linea_lote, 1)  
        Endif  
        sql = "select  
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact  
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &  
        "proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M on  
L.materiaprime = M.codigomateriaprime where L.codigolote='" &  
GridLote[Linea_Lote, 0].Text & "'" &  
        Llenar_Datagrid_LoteDetalle(sql)  
    Endif
```

End

```
Public Sub BtnLFin_Click()  
    Dim Sql As String  
    If GridLote.Rows.Count > 0 And BtnAgregarLote.text = "NUEVO" And  
BtnAgregarLote.Enabled = True Then  
        GridLote.Rows.UnselectAll  
        Linea_lote = GridLote.Rows.Count - 1  
        GridLote.Rows.Select(Linea_lote, 1)  
        sql = "select  
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact  
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
```

Capítulo 2. CÓDIGO FUENTE

```
"proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M on
L.materiaprima = M.codigomateriaprima where L.codigolote=" &
GridLote[Linea_Lote, 0].Text & ""
    Llenar_Datagrid_LoteDetalle(sql)
Endif
End

'***** Lotes Detalle *****
'
'*****
Public Sub BtnLDInicio_Click()

    If GridLoteDetalle.Rows.Count > 0 Then
        GridLoteDetalle.Rows.UnselectAll
        Linea_lote_Detalle = 0
        GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
    Endif

End

Public Sub BtnLDPrevio_Click()

    If GridLoteDetalle.Rows.Count > 0 Then
        If Linea_lote_Detalle = 0 Then
            GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
        Else
            GridLoteDetalle.Rows.UnselectAll
            Linea_lote_Detalle = Linea_lote_Detalle - 1
            GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
        Endif
    Endif

End

Public Sub BtnLDSiguiente_Click()

    If GridLoteDetalle.Rows.Count > 0 Then
        If Linea_lote_Detalle = GridLoteDetalle.Rows.Count - 1 Then
            GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
        Else
            GridLoteDetalle.Rows.UnselectAll
            Linea_lote_Detalle = Linea_lote_Detalle + 1
            GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
        Endif
    Endif

End

Public Sub BtnLDFin_Click()

    If GridLoteDetalle.Rows.Count > 0 Then
        GridLoteDetalle.Rows.UnselectAll
        Linea_lote_Detalle = GridLoteDetalle.Rows.Count - 1
        GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub CmbArticulo_Click()
    Dim Sql As String
    Dim Codigo_Articulo As String
    Codigo_Articulo = Utilidades.Obtener_Campo_Tabla("select * from articulo where
nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    sql = "select * from articulomateriaprima where articulo='" & Codigo_Articulo
& "'"

    If CInt(Utilidades.Tabla_Vacia(sql)) > 0 Then ' Comprobamos si tiene
composicion este articulo
        VlbCantidadFabricada.Enabled = False
        VlbCosteLoteTotal.Enabled = False
        sql = "select * from materiaprima where codigomateriaprima in (select
materiaprima from articulomateriaprima where articulo='" & Codigo_Articulo & "'"
and materiaprima " &
        "in (select distinct materiaprima from stock))"

        If CInt(Utilidades.Tabla_Vacia(sql)) > 0 Then
            Utilidades.Llenar_Combobox(sql, CmbCLMateriaPrima, "nombre")
            CmbCLMateriaPrima.text = ""
        Else
            Articulo_Sin_stock = CmbArticulo.Text
            Balloon.Error("No hay Materias Primas en stock para Producir el Lote del
articulo: " & CmbArticulo.text, CmbArticulo)
        Endif

    Endif

End

End

Public Sub CmbCLMateriaPrima_Click()
    Dim sql As String
    Dim Codigo_MateriaPrima As String
    Dim Codigo_Proveedor As String
    If CmbCLMateriaPrima.text = "" Then

    Else
        TxtCLCoste.text = ""
        TxtCLNumeroFactura.Text = ""
        TxtCLProveedor.Text = ""
        TxtLStockMateriaPrima.Text = Format(0, "##0.00")
        Codigo_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select * from
materiaprima where nombre='" & CmbCLMateriaPrima.Text & "'",
"codigomateriaprima")
        Sql = "select distinct ffacturacompra from stock where materiaprima='" &
Codigo_MateriaPrima & "'" ORDER BY 1 DESC"
        Utilidades.Llenar_Combobox(sql, CmbCLFechaFactura, "ffacturacompra")
        If CmbCLFechaFactura.Count = 1 Then
            Codigo_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select * from
materiaprima where nombre='" & CmbCLMateriaPrima.Text & "'",
"codigomateriaprima")
            Sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima &
"'and ffacturacompra='" & CmbCLFechaFactura.Text & "'" and cantidad > 0"
            Utilidades.Llenar_Combobox(sql, CmbCLNumeroLote, "lotecompra")
            If CmbCLNumeroLote.Count = 1 Then
                CmbCLNumeroLote.text = CmbCLNumeroLote[0].Text
                sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima &
"'and ffacturacompra='" & CmbCLFechaFactura.Text &
                "' and lotecompra='" & CmbCLNumeroLote.text & "'"
            Endif
        Endif
    Endif
End
```



```
        Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla(sql, "proveedor")
        TxtCLProveedor.text = Utilidades.Obtener_Campo_Tabla("select * from
proveedor where codigoproveedor='" & Codigo_Proveedor & "'", "razonsocial")
        TxtCLNumeroFactura.Text = Utilidades.Obtener_Campo_Tabla(sql,
"nfacturacompra")
        TxtLStockMateriaPrima.Text = Format(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad"), "##0.00")
        VlbCLCantidad.Value = Format(Val(TxtLStockMateriaPrima.Text), "##0.00")
    Endif
    VlbCLCantidad.SetFocus
Else
    CmbCLFechaFactura.text = ""
    CmbCLFechaFactura.Sorted = True
Endif
Endif
End

Public Sub CmbCLFechaFactura_Click()

    Dim sql As String
    Dim Codigo_MateriaPrima As String
    Dim Codigo_Proveedor As String
    TxtCLCoste.text = ""
    TxtCLNumeroFactura.Text = ""
    TxtCLProveedor.Text = ""
    TxtLStockMateriaPrima.Text = Format(0, "##0.00")
    Codigo_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select * from
materiaprima where nombre='" & CmbCLMateriaPrima.Text & "'",
"codigomateriaprima")
    Sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima & "'and
ffacturacompra='" & CmbCLFechaFactura.Text & "' and cantidad > 0"
    Utilidades.Llenar_Combobox(sql, CmbCLNumeroLote, "lotecompra")
    If CmbCLNumeroLote.Count = 1 Then
        CmbCLNumeroLote.text = CmbCLNumeroLote[0].Text
        sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima &
"'and ffacturacompra='" & CmbCLFechaFactura.Text &
"' and lotecompra='" & CmbCLNumeroLote.text & "'"
        Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla(sql, "proveedor")
        TxtCLProveedor.text = Utilidades.Obtener_Campo_Tabla("select * from
proveedor where codigoproveedor='" & Codigo_Proveedor & "'", "razonsocial")
        TxtCLNumeroFactura.Text = Utilidades.Obtener_Campo_Tabla(sql,
"nfacturacompra")
        TxtLStockMateriaPrima.Text = Format(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad"), "##0.00")
        VlbCLCantidad.Value = Format(Val(TxtLStockMateriaPrima.Text), "##0.00")
        VlbCLCantidad.SetFocus
    Else
        CmbCLNumeroLote.text = ""
        CmbCLNumeroLote.Sorted = True
    Endif
End
End

Public Sub CmbCLNumeroLote_Click()

    Dim sql As String
    Dim Codigo_MateriaPrima As String
    Dim Codigo_Proveedor As String
    TxtCLCoste.text = ""
```

Capítulo 2. CÓDIGO FUENTE

```
TxtCLNumeroFactura.Text = ""
TxtCLProveedor.Text = ""
TxtLStockMateriaPrima.Text = Format(0, "##0.00")
Codigo_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select * from
materiaprima where nombre='" & CmbCLMateriaPrima.Text & "'",
"codigomateriaprima")
sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima & "'and
ffacturacompra='" & CmbCLFechaFactura.Text &
"' and lotecompra='" & CmbCLNumeroLote.text & "' and cantidad > 0"
Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla(sql, "proveedor")
TxtCLProveedor.text = Utilidades.Obtener_Campo_Tabla("select * from proveedor
where codigoproveedor='" & Codigo_Proveedor & "'", "razonsocial")
TxtCLNumeroFactura.Text = Utilidades.Obtener_Campo_Tabla(sql,
"nfacturacompra")
TxtLStockMateriaPrima.Text = Format(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad"), "##0.00")
VlbCLCantidad.Value = Val(TxtLStockMateriaPrima.Text)
VlbCLCantidad.SetFocus

End

Public Sub BtnAgregarLineaLote_Click()

If Comprobar_Datos_Lotes("DETALLES") Then
If TxtCLCoste.text = "" Then
Calculo_Costes_Linea_Detalle()
Endif
Calculo_Merma()
Realiza_Grabado_Linea_Lote()
limpiar_Datos_Lotes_Detalles()
Endif

End

Public Sub VlbCLCantidad_KeyRelease()

If Key.Code = Key.Return Or Key.Code = Key.Enter Then
If Comprobar_Datos_Lotes("DETALLES") Then
Calculo_Costes_Linea_Detalle()
Calculo_Merma()
Endif
Endif

End

Public Sub BtnBorrarLineaLote_Click()

Dim Sql As String
Dim Numero_fila As Integer
Dim Detalle_Cantidad As String
Dim Detalle_Coste As String
Dim Detalle_Codigo_Materia_Prima As String
Dim Detalle_Codigo_Proveedor As String
Dim Detalle_NLote As String
Dim Detalle_Fecha As String
Dim Detalle_NFactura As String
Dim Detalle_Linea As String
Dim Stock_Cantidad As String
Dim Compra_Id As String
Dim Compra_Precio As String
```

```

Dim Compra_Cantidad As String
If GridLoteDetalle.Rows.Selection.Count > 0 Then
    Numero_filas = GridLoteDetalle.Rows.Selection[0]
    Detalle_Cantidad = GridLoteDetalle[Numero_filas, 2].Text
    Detalle_Coste = GridLoteDetalle[Numero_filas, 3].Text
    Detalle_NLote = GridLoteDetalle[Numero_filas, 4].Text
    Detalle_Fecha = GridLoteDetalle[Numero_filas, 7].Text
    Detalle_NFactura = GridLoteDetalle[Numero_filas, 6].Text
    Detalle_Linea = GridLoteDetalle[Numero_filas, 1].Text
    sql = "select codigomateriaprima from materiaprima where nombre='" &
GridLoteDetalle[Numero_filas, 8].Text & "'"
    Detalle_Codigo_Materia_Prima = Utilidades.Obtener_Campo_Tabla(sql,
"codigomateriaprima")
    sql = "select codigoproveedor from proveedor where razonsocial='" &
GridLoteDetalle[Numero_filas, 5].Text & "'"
    Detalle_Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla(sql,
"codigoproveedor")
    'Deshacemos la operacion.
    VlbCantidadBruta.Value = Format(CFloat(VlbCantidadBruta.Value) -
CFloat(Detalle_Cantidad), "##0.00")
    Calculo_Merma()
    VlbCosteLoteTotal.value = Format(CFloat(VlbCosteLoteTotal.Value) -
CFloat(Detalle_Coste), "##0.00")
    sql = "select * from stock where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and ffacturacompra='" & Detalle_Fecha &
"' and lotecompra='" & Detalle_NLote & "' and proveedor='" &
Detalle_Codigo_Proveedor & "' and nfacturacompra='" & Detalle_NFactura & "'"
    ' Comprobamos que tenemos stock de esta materia prima.
    If Utilidades.Tabla_Vacia(sql) = 1 Then
        ' Si tenemos Stock actualizamos la cantidad en Stock
        Stock_Cantidad = Format(CFloat(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad")) + CFloat(Detalle_Cantidad), "##0.00")
        sql = "update stock set cantidad=" & Stock_Cantidad & " where
materiaprima='" & Detalle_Codigo_Materia_Prima & "'and ffacturacompra='" &
Detalle_Fecha &
"' and lotecompra='" & Detalle_NLote & "' and proveedor='" &
Detalle_Codigo_Proveedor & "' and nfacturacompra='" & Detalle_NFactura & "'"
        Utilidades.In_Act_Bor(Sql)

        ' Obtenemos la cantidad de compras y si es igual no se ha usado el Stock
y actualizamos Usado a N
        sql = "select * from compras where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and fechafacturacompra='" & Detalle_Fecha &
"' and lotecompra='" & Detalle_NLote & "' and proveedor='" &
Detalle_Codigo_Proveedor & "' and nfacturacompra='" & Detalle_NFactura & "'"
        Compra_Cantidad = Format(Utilidades.Obtener_Campo_Tabla(sql,
"cantidad"), "##0.00")
        If CFloat(Stock_Cantidad) = CFloat(Compra_Cantidad) Then
            sql = "update stock set usado='N' where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and ffacturacompra='" & Detalle_Fecha &
"' and lotecompra='" & Detalle_NLote & "' and proveedor='" &
Detalle_Codigo_Proveedor & "' and nfacturacompra='" & Detalle_NFactura & "'"
            Utilidades.In_Act_Bor(Sql)
        Endif
    Else
        ' Si no tenemos stock lo añadimos a Stock
        sql = "select * from compras where materiaprima='" &
Detalle_Codigo_Materia_Prima & "'and fechafacturacompra='" & Detalle_Fecha &

```

```

        "' and lotecompra='" & Detalle_NLote & "' and proveedor='" &
Detalle_Codigo_Proveedor & "' and nfacturacompra='" & Detalle_NFactura & "'"
        Compra_Id = Utilidades.Obtener_Campo_Tabla(sql, "codigocompra")
        Compra_Precio = Utilidades.Obtener_Campo_Tabla(sql, "precio")
        sql = "insert into stock
(codigostock,materiaprima,cantidad,lotecompra,proveedor,nfacturacompra,ffacturac
ompra,precio,usado)" &
        " values('" & Compra_Id & "',''" & Detalle_Codigo_Materia_Prima & "',''" &
Detalle_Cantidad & "',''" & Detalle_NLote & "',''" & Detalle_Codigo_Proveedor &
"',''" & Detalle_NFactura &
        "',''" & Detalle_Fecha & "',''" & Compra_Precio & "','N')'"
        Utilidades.In_Act_Bor(Sql)
    Endif
    sql = "delete from lotedetalle where linea='" & Detalle_Linea & "'"
    Utilidades.In_Act_Bor(Sql)
    sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
        "proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M
on L.materiaprima = M.codigomateriaprima where L.codigolote='" &
UCase(TxtLote.text) & "'"
        Llenar_Datagrid_LoteDetalle(sql)
    Endif

End

'***** Lotes *****
'
'*****
Public Sub BtnAgregarLote_Click()
    Dim Sql As String
    If Comprobar_Datos_Lotes("NUEVO") Then
        If BtnAgregarLote.Text = "NUEVO" Then
            BtnAgregarLote.Text = "GRABAR"
            BtnBorrarLote.Enabled = False
            BtnEditarLote.Enabled = False
            BtnConsultarLotes.Enabled = False
            FrameFiltrar.Enabled = False
            TxtLote.Enabled = False
            CmbArticulo.Enabled = False
            PanelComposicionLote.Enabled = True
            'BtnSalirStock.Enabled = False
            sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
                "proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M
on L.materiaprima = M.codigomateriaprima where L.codigolote='" &
UCase(TxtLote.text) & "'"
                Llenar_Datagrid_LoteDetalle(sql)
        Else
            If Comprobar_Datos_Lotes("GRABAR") Then
                BtnAgregarLote.Text = "NUEVO"
                BtnBorrarLote.Enabled = True
                BtnEditarLote.Enabled = True
                BtnConsultarLotes.Enabled = True
                FrameFiltrar.Enabled = True
                TxtLote.Enabled = True
                CmbArticulo.Enabled = True
                PanelComposicionLote.Enabled = False
            End If
        End If
    End If
End Sub

```

```

        ' BtnSalirStock.Enabled = True
        Realiza_Grabado_Lote()
        limpiar_Datos_Lotes()
        limpiar_Datos_Lotes_Detalles()
    Endif
Endif

Endif

End

Public Sub BtnBorrarLote_Click()
    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Dim Texto As String
    If GridLote.Rows.Selection.Count > 0 Then
        Numero_fila = GridLote.Rows.Selection[0]
        Texto = "Desea borrar el LoteCodigo : <b>" & GridLote[Numero_fila, 0].Text
        & "</b> y Artículo: <b>" & GridLote[Numero_fila, 6].Text & "</b>"
        Respuesta = Message.Delete(Texto, "Si", "No")
        If respuesta = 1 Then
            ' Borramos Lote
            sql = "delete from lotes where codigolote='" & GridLote[Numero_fila,
0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            ' Borramos Detalle de lote.
            sql = "delete from lotedetalle where codigolote='" & GridLote[Numero_fila,
0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            ' Borramos las temperaturas y los tiempos
            sql = "select * from tclote where codigolote='" & GridLote[Numero_fila,
0].Text & "'"
            If utilidades.Tabla_Vacia(sql) > 0 Then
                sql = "delete from tclote where codigolote='" & GridLote[Numero_fila,
0].Text & "'"
                Utilidades.In_Act_Bor(Sql)
            Endif

            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
"articulo A on L.articulo = A.codigoarticulo"
            Llenar_Datagrid_Lote(Sql)
            If GridLote.Rows.Count > 0 Then
                sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
"proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M
on L.materiaprima = M.codigomateriaprima where L.codigolote='" & GridLote[0,
0].Text & "'"
                Llenar_Datagrid_LoteDetalle(sql)
            Endif
        Endif
    Endif

End

```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnEditarLote_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Dia As String
    Dim Mes As String
    Dim Ano As String
    If BtnEditarLote.text = "EDITAR" And GridLote.Rows.Count > 0 Then ' Cargamos
los datos del lote seleccionado.
        Numero_fila = GridLote.Rows.Selection[0]
        TxtLote.text = GridLote[Numero_fila, 0].Text
        Dia = Mid(GridLote[Numero_fila, 1].Text, 1, 2)
        Mes = Mid(GridLote[Numero_fila, 1].Text, 4, 2)
        Ano = Mid(GridLote[Numero_fila, 1].Text, 7, 4)
        DteFechaCreacion.Value = Date(ano, mes, dia)
        VlbCantidadFabricada.Value = GridLote[Numero_fila, 2].Text
        VlbCosteLoteTotal.Value = GridLote[Numero_fila, 3].Text
        VlbCantidadBruta.Value = GridLote[Numero_fila, 4].Text
        VlbMerma.Value = GridLote[Numero_fila, 5].Text
        CmbArticulo.Text = GridLote[Numero_fila, 6].Text
        sql = "select * from tclote where codigolote='" & GridLote[Numero_fila,
0].Text & "'"
        If utilidades.Tabla_Vacia(sql) > 0 Then
            PanelTeTi.Enabled = True
            VlbTiempo1.Value = CInt(utilidades.Obtener_Campo_Tabla(sql, "tiempo1"))
            VlbTiempo2.Value = CInt(utilidades.Obtener_Campo_Tabla(sql, "tiempo2"))
            VlbTemperatural.Value = CInt(utilidades.Obtener_Campo_Tabla(sql,
"temperatural"))
            VlbTemperatura2.Value = CInt(utilidades.Obtener_Campo_Tabla(sql,
"temperatura2"))
        Endif
        Calculo_Merma()
        BtnEditarLote.text = "ACTUALIZAR"
        CmbArticulo.Enabled = False
        BtnAgregarLote.Enabled = False
        BtnBorrarLote.Enabled = False
        BtnConsultarLotes.Enabled = False
        TxtLote.Enabled = False
        'BtnSalirStock.Enabled = False
    Else ' Actualizamos el lote si los datos son correctos.
        If Comprobar_Datos_Lotes("GRABAR") Then
            Actualizar_Lote()
            limpiar_Datos_Lotes()
            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
            "articulo A on L.articulo = A.codigoarticulo"
            Llenar_Datagrid_Lote(Sql)
            If GridLote.Rows.Count > 0 Then
                sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
                "proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M
on L.materiaprime = M.codigomateriaprime where L.codigolote='" & GridLote[0,
0].Text & "'"
                Llenar_Datagrid_LoteDetalle(sql)
            Endif
            BtnAgregarLote.Enabled = True
            BtnBorrarLote.Enabled = True
        Endif
    End If
End Sub
```

```
        BtnConsultarLotes.Enabled = True
        TxtLote.Enabled = True
        CmbArticulo.Enabled = True
        BtnSalirStock.Enabled = True
        BtnEditarLote.text = "EDITAR"
    Endif
Endif
End

Public Sub ChkTiempoTemp_Click()

    If ChkTiempoTemp.Value = True Then
        PanelTeTi.Enabled = True
    Else
        PanelTeTi.Enabled = False
    Endif

End

Public Sub VlbMerma_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        Calculo_Merma()
    Endif
End

Public Sub GridLote_DblClick()

    Dim sql As String
    Dim Columna_Seleccionada As Integer
    If Not Uso_Filtro Then
        Columna_Seleccionada = GridLote.Column
        Select Columna_Seleccionada
        Case 0
            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
            "articulo A on L.articulo = A.codigoarticulo order by 1"
        Case 1
            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
            "articulo A on L.articulo = A.codigoarticulo order by 2"
        Case 2
            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
            "articulo A on L.articulo = A.codigoarticulo order by 3"
        Case 3
            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
            "articulo A on L.articulo = A.codigoarticulo order by 4"
        Case 4
            sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
            "articulo A on L.articulo = A.codigoarticulo order by 5"
        Case 5
```

```
        sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
        "articulo A on L.articulo = A.codigoarticulo order by 6"
    Case 6
        sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
        "articulo A on L.articulo = A.codigoarticulo order by 7"
    End Select
    Llenar_Datagrid_Lote(sql)
Endif

End

Public Sub BtnConsultarLotes_Click()
    Frm_ConsultaLotes.Showmodal()
End

''***** Funciones privadas*****
''
''*****

'' Configuramos los Objetos de los Lotes
Private Sub Configurar_Objeto_Lotes()
    Dim Error_Tablas_Vacias As String
    Dim Error_V As Boolean
    Error_V = False
    Uso_Filtro = False
    If Utilidades.Tabla_Vacia("select * from articulo") = 0 Then
        Error_Tablas_Vacias = "Deber Cear Articulos"
        Error_V = True
    Else
        CmbArticulo.clear
        Utilidades.Llenar_Combobox("select * from articulo", CmbArticulo,
"nombre")
        CmbArticulo.Index = -1
        CmbfArticulo.clear
        Utilidades.Llenar_Combobox("select * from articulo", CmbfArticulo,
"nombre")
        CmbfArticulo.Index = -1
    Endif
    If Error_V Then
        TxtErrorLotes.Visible = True
        TxtErrorLotes.text = Error_Tablas_Vacias
    Endif
    Contador_Linea_Detalle_lote = 1
    Linea_Lote = 0
    Linea_lote_Detalle = 0
    TxtLote.SetFocus()
    TxtLKilosMerma.text = Format(0, "##0.00")
    TxtLStockMateriaPrima.Text = Format(0, "##0.00")
    DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFICreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFFCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    Configurar_Datagrid_Lotes()
    Configurar_Datagrid_LotesDetalles()
End
```



```
' ' Configuramos Cabeceras de DataGridViewLotes
Private Sub Configurar_Datagrid_Lotes()
    Dim Sql As String
    GridLote.Columns.Count = 7
    GridLote.Header = GridLote.Both
    GridLote.Columns[0].Title = "Codigo"
    GridLote.Columns[1].Title = "Fecha"
    GridLote.Columns[2].Title = "Cantidad"
    GridLote.Columns[3].Title = "Coste"
    GridLote.Columns[4].Title = "Cantidad Bruta"
    GridLote.Columns[5].Title = "Merma"
    GridLote.Columns[6].Title = "Articulo"
    GridLote.Columns[0].Width = 100
    GridLote.Columns[1].Width = 100
    GridLote.Columns[2].Width = 100
    GridLote.Columns[3].Width = 90
    GridLote.Columns[4].Width = 150
    GridLote.Columns[5].Width = 150
    GridLote.Columns[6].Width = 150
    sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
        "articulo A on L.articulo = A.codigoarticulo"
    Llenar_Datagrid_Lote(Sql)
End

' ' Llenamos DataGridView Lote <br>
' ' Parametros:<br>
' '     Sql --> Sentencia para cargar datos en el Dataset
Private Sub Llenar_Datagrid_Lote(sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridLote.Rows.Count > 0 Then
        GridLote.Rows.Remove(0, GridLote.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    While Result_Trabajo.Available
        Numero_fila = GridLote.Rows.Count
        GridLote.Rows.Insert(Numero_fila, 1)
        GridLote[Numero_fila, 0].Text = Result_Trabajo["L.codigolote"]
        GridLote[Numero_fila, 1].Text = Result_Trabajo["L.fecha"]
        GridLote[Numero_fila, 2].Text = Result_Trabajo["L.cantidad"]
        GridLote[Numero_fila, 3].Text = Result_Trabajo["L.coste"]
        GridLote[Numero_fila, 4].Text = Result_Trabajo["L.cantidadbruta"]
        GridLote[Numero_fila, 5].Text = Result_Trabajo["L.merma"]
        GridLote[Numero_fila, 6].Text = Result_Trabajo["A.nombre"]
        Result_Trabajo.MoveNext
    Wend
    FMain.Conexion_Base_Datos.Close
    GridLote.Rows.Select(Linea_lote, 1)
End
```

```

'' Configuramos Cabeceras de DataGridViewLotes
Private Sub Configurar_Datagrid_LotesDetalles()
    Dim Sql As String
    GridLoteDetalle.Columns.Count = 9
    GridLoteDetalle.Header = GridLoteDetalle.Both
    GridLoteDetalle.Columns[0].Title = "Codigo"
    GridLoteDetalle.Columns[1].Title = "Linea"
    GridLoteDetalle.Columns[2].Title = "Cantidad"
    GridLoteDetalle.Columns[3].Title = "Coste"
    GridLoteDetalle.Columns[4].Title = "N Lote Compra"
    GridLoteDetalle.Columns[5].Title = "Proveedor"
    GridLoteDetalle.Columns[6].Title = "N Factura Compra"
    GridLoteDetalle.Columns[7].Title = "F Factura Compra"
    GridLoteDetalle.Columns[8].Title = "Materia Prima"
    GridLoteDetalle.Columns[0].Width = 100
    GridLoteDetalle.Columns[1].Width = 50
    GridLoteDetalle.Columns[2].Width = 100
    GridLoteDetalle.Columns[3].Width = 100
    GridLoteDetalle.Columns[4].Width = 150
    GridLoteDetalle.Columns[5].Width = 150
    GridLoteDetalle.Columns[6].Width = 150
    GridLoteDetalle.Columns[7].Width = 150
    GridLoteDetalle.Columns[8].Width = 200
    If GridLote.Rows.Count > 0 Then
        sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
        "proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M on
L.materiaprime = M.codigomateriaprime where L.codigolote='" & GridLote[0,
0].Text & "'"
        Llenar_Datagrid_LoteDetalle(sql)
    Endif
End

'' Llenamos DataGridView Lote Detalle<br>
'' Parametros:<br>
''     Sql --> Sentencia para cargar datos en el Dataset
Private Sub Llenar_Datagrid_LoteDetalle(sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridLoteDetalle.Rows.Count > 0 Then
        GridLoteDetalle.Rows.Remove(0, GridLoteDetalle.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    While Result_Trabajo.Available
        Numero_fila = GridLoteDetalle.Rows.Count
        GridLoteDetalle.Rows.Insert(Numero_fila, 1)
        GridLoteDetalle[Numero_fila, 0].Text = Result_Trabajo["L.codigolote"]
        GridLoteDetalle[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
        GridLoteDetalle[Numero_fila, 2].Text = Result_Trabajo["L.cantidad"]
        GridLoteDetalle[Numero_fila, 3].Text = Result_Trabajo["L.coste"]
        GridLoteDetalle[Numero_fila, 4].Text = Result_Trabajo["L.numerolotecompra"]
        GridLoteDetalle[Numero_fila, 5].Text = Result_Trabajo["P.razonsocial"]
    End While
End Sub

```

```

        GridLoteDetalle[Numero_fila, 6].Text = Result_Trabajo["L.nfacturacompra"]
        GridLoteDetalle[Numero_fila, 7].Text =
Result_Trabajo["L.fechafacturacompra"]
        GridLoteDetalle[Numero_fila, 8].Text = Result_Trabajo["M.nombre"]
        Result_Trabajo.MoveNext
    Wend
    FMain.Conexion_Base_Datos.Close
    GridLoteDetalle.Rows.Select(Linea_lote_Detalle, 1)
End

'' Funcion para comprobar los datos de los lotes y las lineas de lotes.<br>
'' Parametros:<br>
''     Accion Tomara los siguientes valores<br>
''         NUEVO -- Comprobar los datos para crear nuevo Lote.<br>
''         DETALLES -- Comprobar los datos antes de añadir linea detalle.
'' Devuelve:<br>
''     TRUE -- Datos corecctos<br>
''     FALSE -- Datos erroneos
Private Function Comprobar_Datos_Lotes(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
    If Accion = "NUEVO" Then
        Sql = "select * from lotes where codigolote='" & UCase(TxtLote.Text) & "'"
        If TxtLote.text = "" Then
            Balloon.Error("Debe introducir un codigo de lote", TxtLote)
            Validar = False
            TxtLote.SetFocus
        Else If Utilidades.Comprobar_existe_dato(Sql)
            Balloon.Error("El codigo del lote ya existe", TxtLote)
            Validar = False
            TxtLote.SetFocus
        Else If CmbArticulo.text = ""
            Balloon.Error("Debe elegir un articulo para crear el lote", CmbArticulo)
            Validar = False
        Else If CmbArticulo.text = Articulo_Sin_stock
            Balloon.Error("No hay Materias Primas en stock para Producir el Lote del
articulo: " & CmbArticulo.text, CmbArticulo)
            validar = False
        Endif
    Else If Accion = "DETALLES"
        If CmbCLMateriaPrima.text = "" Then
            Balloon.Error("Debe elegir Materia Prima", CmbCLMateriaPrima)
            Validar = False
        Else If CmbCLFechaFactura.text = ""
            Balloon.Error("Debe elegir Fecha Factura", CmbCLFechaFactura)
            Validar = False
        Else If CmbCLNumeroLote.text = ""
            Balloon.Error("Debe elegir Lote", CmbCLNumeroLote)
            Validar = False
        Else If VlbCLCantidad.text = "" Or VlbCLCantidad.Text = "0"
            Balloon.Error("Debe introducir una cantidad mayor que 0 ", VlbCLCantidad)
            Validar = False
            VlbCLCantidad.SetFocus
        Else If Val(VlbCLcantidad.value) > Val(TxtLStockMateriaPrima.Text)
            Balloon.Error("La cantidad de Materia Prima es superior al Stock",
VlbCLCantidad)
            Validar = False
    End If
End Function

```

```

        VlbCLCantidad.SetFocus
    Endif
Else If Accion = "GRABAR"
    If VlbCantidadFabricada.Value = "" Or VlbCantidadFabricada.Value = "0"
        Balloon.Error("No puede crear lote sin composicion de materias primas",
GridLoteDetalle)
        Validar = False
    Else If VlbCosteLoteTotal.Value = "" Or VlbCosteLoteTotal.Value = "0"
        Balloon.Error("El coste del lote no puede ser 0, introduzca cantidad y
pulse INTRO para calcular coste.", VlbCosteLoteTotal)
        Validar = False
        VlbCosteLoteTotal.SetFocus
    Endif
Endif
Return Validar
End

'' Procedimiento que limpia los datos para poder añadir otra linea de detalle.
Private Sub limpiar_Datos_Lotes_Detalles()

    TxtCLCoste.Text = ""
    VlbCLCantidad.text = ""
    TxtCLProveedor.text = ""
    TxtCLNumeroFactura.text = ""
    CmbCLFechaFactura.Clear
    CmbCLNumeroLote.Clear
    CmbCLMateriaPrima.Index = -1
    TxtLStockMateriaPrima.Text = Format(0, "##0.00")
End
'' Procedimiento que limpia los datos del lote.
Private Sub limpiar_Datos_Lotes()

    TxtLote.Text = ""
    VlbCantidadFabricada.Value = ""
    VlbCosteLoteTotal.Value = ""
    ChkTiempoTemp.Value = False
    PanelTeTi.Enabled = False
    VlbCantidadBruta.text = ""
    VlbMerma.text = ""
    TxtLKilosMerma.text = Format(0, "##0.00")
    TxtLStockMateriaPrima.Text = Format(0, "##0.00")
    VlbTiempo1.Value = ""
    VlbTemperatural.Value = ""
    VlbTiempo2.Value = ""
    VlbTemperatura2.Value = ""
    DteFechaCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    VlbCantidadFabricada.Enabled = False
    VlbCosteLoteTotal.Enabled = False
    CmbArticulo.Index = -1
End
'' Realiza el calculo del coste al añadir una linea de detalle. Añade los kilos
al total.
Private Sub Calculo_Costes_Linea_Detalle()
    Dim sql As String
    Dim Codigo_MateriaPrima As String
    Dim Codigo_Proveedor As String
    Codigo_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select * from
materiaprima where nombre='" & CmbCLMateriaPrima.Text & "'",
"codigomateriaprima")

```

Capítulo 2. CÓDIGO FUENTE

```
Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla("select * from proveedor
where razonsocial='" & TxtCLProveedor.Text & "'", "codigoproveedor")
sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima & "'and
ffacturacompra='" & CmbCLFechaFactura.Text &
"' and lotecompra='" & CmbCLNumeroLote.text & "' and proveedor='" &
Codigo_Proveedor & "' and nfacturacompra='" & TxtCLNumeroFactura.text & "'"
TxtLStockMateriaPrima.Text = Format(Val(TxtLStockMateriaPrima.text) -
Val(VlbCLCantidad.Value), "##0.00")
VlbCantidadBruta.Value = Format(Val(VlbCantidadBruta.value) +
Val(VlbCLCantidad.Value), "##0.00")
TxtCLCoste.text = Format(Val(VlbCLCantidad.value) *
Val(Utilidades.Obtener_Campo_Tabla(sql, "precio")), "##0.00")
VlbCosteLoteTotal.Value = Format(Val(VlbCosteLoteTotal.Value) +
Val(TxtCLCoste.Text), "##0.00")

End

'' Procedimiento para el calculo de la merma
Private Sub Calculo_Merma()
    Dim Merma As Float
    Dim Aux As Float
    If CInt(VlbMerma.value) > 0 Then
        Merma = (CFloat(VlbCantidadBruta.Value) * CFloat(VlbMerma.Value)) / 100

        Aux = CFloat(VlbCantidadBruta.Value) - Merma
        VlbCantidadFabricada.Value = Format(Aux, "##0.00")

        TxtLKilosMerma.text = Format(Merma, "##0.00")
    Else
        VlbCantidadFabricada.Value = Format(VlbCantidadBruta.Value, "##0.00")
        TxtLKilosMerma.text = Format(0, "##0.00")
    Endif
End

'' Procedimiento que realiza el almacenamiento del registro correspondiente al
lote que estamos creando.

Private Sub Realiza_Grabado_Lote()
    Dim sql As String
    Dim Codigo_Articulo As String
    Codigo_Articulo = Utilidades.Obtener_Campo_Tabla("select * from articulo where
nombre='" & CmbArticulo.Text & "'", "codigoarticulo")
    Calculo_Merma()
    If VlbCantidadBruta.Value = 0 Then
        VlbCantidadBruta.Value = VlbCantidadFabricada.Value
    Endif
    sql = "insert into
lotes(codigolote,fecha,cantidad,coste,cantidadbruta,merma,articulo) values('" &
    UCase(TxtLote.Text) & "','" &
    Utilidades.Fecha_Correcta(DteFechaCreacion.Value) & "','" &
    CFloat(VlbCantidadFabricada.value) & "','" &
    CFloat(VlbCosteLoteTotal.Value) & "','" & CFloat(VlbCantidadBruta.Value) & "','" &
    CFloat(VlbMerma.Value) & "','" & Codigo_Articulo & "'"
    Utilidades.In_Act_Bor(sql)
    ' Se se ha activado las temperatura.
    If ChkTiempoTemp.Value = True Then
        sql = "insert into
tclote(codigolote,temperatural1,tiempo1,temperatura2,tiempo2) values('" &
```

Capítulo 2. CÓDIGO FUENTE

```
    UCase(TxtLote.Text) & "','" & VlbTemperatura1.Value & "','" & VlbTiempo1.Value
& "','" & VlbTemperatura2.Value & "','" & VlbTiempo2.Value & ")"
    Utilidades.In_Act_Bor(sql)
Endif

Contador_Linea_Detalle_lote = 1
sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
    "articulo A on L.articulo = A.codigoarticulo"
Llenar_Datagrid_Lote(Sql)
sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
    "proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M on
L.materiaprima = M.codigomateriaprima where L.codigolote='" & GridLote[0,
0].Text & "'"
Llenar_Datagrid_LoteDetalle(sql)
End
'' Procedimiento para la actualizacion de los datos del lote cuando se ha
editado.
Private Sub Actualizar_Lote()

    Dim Sql As String
    sql = "select * from tclote where codigolote='" & TxtLote.Text & "'"
    ' Se se ha activado las temperatura.
    If ChkTiempoTemp.Value = True Then ' Si tenemos marcado las temperaturas y
tiempos
        If utilidades.Tabla_Vacia(sql) > 0 Then ' Si existe actualizamos los datos
        Sql = "update tclote set temperatur1=" & VlbTemperatura1.Value &
",tiemp1=" & VlbTiempo1.Value & ",temperatura2=" & VlbTemperatura2.Value &
",tiempo2=" & VlbTiempo2.Value & " where codigolote='" & TxtLote.Text &
""
        Else ' si no existe registro lo añadimos
            sql = "insert into
tclote(codigolote,temperatur1,tiemp1,temperatura2,tiempo2) values('" &
UCase(TxtLote.Text) & "','" & VlbTemperatura1.Value & "','" &
VlbTiempo1.Value & "','" & VlbTemperatura2.Value & "','" & VlbTiempo2.Value & ")"
            Endif
            Utilidades.In_Act_Bor(sql)
        Else ' si no tenemos marcado comprobamos si existe temperatura para ese lote y
si existe lo borramos
            sql = "select * from tclote where codigolote='" & TxtLote.Text & "'"
            If utilidades.Tabla_Vacia(sql) > 0 Then
                sql = "delete from tclote where codigolote='" & TxtLote.Text & "'"
                Utilidades.In_Act_Bor(sql)
            Endif
        Endif
    End

'' Procedimiento que realiza el almacenamiento del registro correspondiente a la
linea de lote que estamos creando.

Private Sub Realiza_Grabado_Linea_Lote()

    Dim sql As String
    Dim Codigo_MateriaPrima As String
    Dim Codigo_Proveedor As String
    Dim Codigo_Stock As String
```

```

Codigo_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select * from
materiaprima where nombre='" & CmbCLMateriaPrima.Text & "'",
"codigomateriaprima")
Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla("select * from proveedor
where razonsocial='" & TxtCLProveedor.Text & "'", "codigoproveedor")
' Insertamos linea detalle
sql = "insert into
lotedetalle(codigolote,linea,cantidad,coste,numerolotecompra,proveedor,nfacturac
ompra,fechafacturacompra,materiaprima) values('" &
UCase(TxtLote.text) & "','" & Contador_Linea_Detalle_lote & "','" &
CFloat(VlbCLCantidad.value) & "','" & CFloat(TxtCLCoste.Text) & "','" &
CmbCLNumeroLote.Text & "','" &
Codigo_Proveedor & "','" & TxtCLNumeroFactura.Text & "','" &
CmbCLFechaFactura.text & "','" & Codigo_MateriaPrima & "')"
Utilidades.In_Act_Bor(sql)
' Actualizamos Stock
sql = "select * from stock where materiaprima='" & Codigo_MateriaPrima & "'and
ffacturacompra='" & CmbCLFechaFactura.Text &
"' and lotecompra='" & CmbCLNumeroLote.text & "' and proveedor='" &
Codigo_Proveedor & "' and nfacturacompra='" & TxtCLNumeroFactura.text & "'"
Codigo_Stock = Utilidades.Obtener_Campo_Tabla(sql, "codigostock")
' Actualizamos la cantidad en stock
sql = "update stock set cantidad=" & TxtLStockMateriaPrima.Text & ", usado='S'
where codigostock='" & Codigo_Stock & "'"
Llenar_Datagrid_LoteDetalle(sql)
' Recargamos datos en el DataGridView
Contador_Linea_Detalle_lote = Contador_Linea_Detalle_lote + 1
sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
"proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M on
L.materiaprima = M.codigomateriaprima where L.codigolote='" &
UCase(TxtLote.text) & "'"
Llenar_Datagrid_LoteDetalle(sql)

End

' Procedimiento para quitar los filtros de los lotes.
Private Sub Quitar_Filtros()
Dim Sql As String
CmbfArticulo.Index = -1
DteFICreacion.Value = Date(Year(Now), Month(Now), Day(Now))
DteFFCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
PanelArticulos.Enabled = False
PanelFCreacion.Enabled = False
ChkFechaCreacionLote.Value = False
ChkArticulo.Value = False
sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
"articulo A on L.articulo = A.codigoarticulo"
Llenar_Datagrid_Lote(Sql)
sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
"proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M on
L.materiaprima = M.codigomateriaprima where L.codigolote='" & GridLote[0,
0].Text & "'"
Llenar_Datagrid_LoteDetalle(sql)

```

```
End

'' Procedimiento para ejecutar los filtros seleccionados.
Private Sub Ejecutar_Filtros()

    Dim Sql As String
    sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
        "articulo A on L.articulo = A.codigoarticulo where "
    ' Filtro para la fecha de creacion.
    If ChkFechaCreacionLote.Value = True Then
        If DteFICreacion.Value = DteFFCreacion.value Then
            sql &= "L.fecha='" & Utilidades.Fecha_Correcta(DteFICreacion.value) &
""
        Else
            sql &= "L.fecha between '" &
utilidades.Fecha_Correcta(DteFICreacion.value) & "' and '" &
utilidades.Fecha_Correcta(DteFFCreacion.value) & "'"
        Endif
    Endif

    ' Filtro para Articulos
    If ChkArticulo.Value = True Then
        If ChkFechaCreacionLote.value = True Then
            sql &= " and A.nombre='" & CmbfArticulo.text & "'"
        Else
            sql &= "A.nombre='" & CmbfArticulo.text & "'"
        Endif
    Endif
    Llenar_Datagrid_Lote(Sql)
    If GridLote.Rows.Count > 0 Then
        sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
        "proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M on
L.materiaprime = M.codigomateriaprime where L.codigolote='" & GridLote[0,
0].Text & "'"
        Llenar_Datagrid_LoteDetalle(sql)
    Endif
End
```

2.2.5.2 Frm_ConsultaLotes.class

```
Public Linea_lote_Consulta As Integer ' Variable para controlar la linea de
posicion del DatagridLoteDetalle y poder movernos con flechas.
Public Lote_Generar As String
Public Sub Form_Open()
    Me.Title = "CONSULTA DE LOTES"
    Configurar_Objetos_Consulat_Lotes()
End

Public Sub BtnSalirStock_Click()
    Me.Close
End
```


Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnLDInicio_Click()

    If GridLoteDetalleConsulta.Rows.Count > 0 Then
        GridLoteDetalleConsulta.Rows.UnselectAll
        Linea_lote_Consulta = 0
        GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
    Endif

End

Public Sub BtnLDPrevio_Click()

    If GridLoteDetalleConsulta.Rows.Count > 0 Then
        If Linea_lote_Consulta = 0 Then
            GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
        Else
            GridLoteDetalleConsulta.Rows.UnselectAll
            Linea_lote_Consulta = Linea_lote_Consulta - 1
            GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
        Endif
    Endif

End

Public Sub BtnLDSiguiente_Click()

    If GridLoteDetalleConsulta.Rows.Count > 0 Then
        If Linea_lote_Consulta = GridLoteDetalleConsulta.Rows.Count - 1 Then
            GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
        Else
            GridLoteDetalleConsulta.Rows.UnselectAll
            Linea_lote_Consulta = Linea_lote_Consulta + 1
            GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
        Endif
    Endif

End

Public Sub BtnLDFin_Click()

    If GridLoteDetalleConsulta.Rows.Count > 0 Then
        GridLoteDetalleConsulta.Rows.UnselectAll
        Linea_lote_Consulta = GridLoteDetalleConsulta.Rows.Count - 1
        GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
    Endif

End

'''***** FILTROS *****'''
'''*****'''

Public Sub ChkFechaCreacionLote_Click()

    If ChkFechaCreacionLote.value = True Then
        PanelFCreacion.Enabled = True
    Else
        PanelFCreacion.Enabled = False
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub ChkArticulo_Click()
    Dim sql As String
    If ChkArticulo.value = True Then
        PanelArticulos.Enabled = True
        sql = "select nombre from articulo where codigoarticulo in(select distinct
articulo from lotes)"
        Utilidades.Llenar_Combobox(sql, CmbfArticulo, "nombre")
        CmbfArticulo.Index = -1
    Else
        PanelArticulos.Enabled = False
    Endif
End

Public Sub BtnFiltrar_Click()

    Dim Sql As String
    Dim Sql_Filtro As String
    Dim Codigo_Articulo As String
    sql_filtro = "select codigolote from lotes where"
    If ChkArticulo.Value = True Then
        sql = "select codigoarticulo from articulo  where nombre='" &
CmbfArticulo.Text & "'"
        Codigo_Articulo = Utilidades.Obtener_Campo_Tabla(sql, "codigoarticulo")
        sql_filtro &= " articulo='" & Codigo_Articulo & "'"
    Endif
    Utilidades.Llenar_Combobox(sql_filtro, CmbLote, "codigolote")
    CmbLote.Text = ""
    CmbLote.Index = -1
End

Public Sub BtnQuitarFiltro_Click()
    ChkArticulo.Value = False
    PanelArticulos.Enabled = False
    ChkFechaCreacionLote.Value = False
    PanelFCreacion.Enabled = False
    utilidades.Llenar_Combobox("select * from lotes", CmbLote, "codigolote")
    CmbLote.Index = -1
End

'''*****FIN FILTROS *****
'''
'''*****
Public Sub BtnGenerarLote_Click()

    If CmbLote.Text = "" Then
        Balloon.Error("Lote no existe.", CmbLote)
    Else If utilidades.Tabla_Vacia("select * from lotes where codigolote='" &
UCase(CmbLote.Text) & "'") > 0
        Lote_Generar = UCase(CmbLote.Text)
        Frm_Visualiza_Lote.Show
    Else
        Balloon.Error("Lote no existe.", CmbLote)
    Endif
End

End
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnConsultarLote_Click()

    Dim Sql As String
    If CmbLote.Text = "" Then
        If GridLoteDetalleConsulta.Rows.Count > 0 Then
            GridLoteDetalleConsulta.Rows.Remove(0, GridLoteDetalleConsulta.Rows.Count)
        Endif
        Limpiar_Campos()
    Else If utilidades.Tabla_Vacia("select * from lotes where codigolote='" &
    UCase(CmbLote.Text) & "'") > 0
        Visualizar_Campos_Lote(UCase(CmbLote.Text))
        If GridLoteDetalleConsulta.Rows.Count > 0 Then
            GridLoteDetalleConsulta.Rows.Remove(0, GridLoteDetalleConsulta.Rows.Count)
        Endif
        If utilidades.Tabla_Vacia("select * from lotedetalle where codigolote='" &
    UCase(CmbLote.Text) & "'") > 0 Then
            sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
            "proveedor P on L.proveedor = P.codigoproveedor left join materiaprima M
on L.materiaprima = M.codigomateriaprima where L.codigolote='" & CmbLote.Text &
            "'"
            Llenar_Datagrid_LoteDetalle(sql)
        Endif
    Else
        Balloon.Error("Lote no existe.", CmbLote)
    Endif
End

''***** Funciones privadas*****
''
''*****
Private Sub Configurar_Objetos_Consulat_Lotes()
    'Utilidades.Llenar_Combobox("select * from articulo", CmbfArticulo, "nombre")
    'CmbfArticulo.Index = -1
    Linea_lote_Consulta = 0
    Utilidades.Llenar_Combobox("select * from lotes", CmbLote, "codigolote")
    CmbLote.Index = -1
    DteFICreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    DteFFCreacion.Value = Date(Year(Now), Month(Now), Day(Now))
    Configurar_Datagrid_LotesDetalles()
End

'' Configuramos Cabeceras de DataGridViewLotes
Private Sub Configurar_Datagrid_LotesDetalles()
    GridLoteDetalleConsulta.Columns.Count = 9
    GridLoteDetalleConsulta.Header = GridLoteDetalleConsulta.Both
    GridLoteDetalleConsulta.Columns[0].Title = "Codigo"
    GridLoteDetalleConsulta.Columns[1].Title = "Linea"
    GridLoteDetalleConsulta.Columns[2].Title = "Cantidad"
    GridLoteDetalleConsulta.Columns[3].Title = "Coste"
    GridLoteDetalleConsulta.Columns[4].Title = "N Lote Compra"
    GridLoteDetalleConsulta.Columns[5].Title = "Proveedor"
    GridLoteDetalleConsulta.Columns[6].Title = "N Factura Compra"
    GridLoteDetalleConsulta.Columns[7].Title = "F Factura Compra"
    GridLoteDetalleConsulta.Columns[8].Title = "Materia Prima"
    GridLoteDetalleConsulta.Columns[0].Width = 100
    GridLoteDetalleConsulta.Columns[1].Width = 50
    GridLoteDetalleConsulta.Columns[2].Width = 100
```

```
GridLoteDetalleConsulta.Columns[3].Width = 100
GridLoteDetalleConsulta.Columns[4].Width = 150
GridLoteDetalleConsulta.Columns[5].Width = 150
GridLoteDetalleConsulta.Columns[6].Width = 150
GridLoteDetalleConsulta.Columns[7].Width = 150
GridLoteDetalleConsulta.Columns[8].Width = 200

End

'' Llenamos DataGridView Lote Detalle<br>
'' Parametros:<br>
''     Sql --> Sentencia para cargar datos en el Dataset
Private Sub Llenar_Datagrid_LoteDetalle(sql As String)
    Dim Result_Trabajo As Result
    Dim Numero_fila As Integer
    If GridLoteDetalleConsulta.Rows.Count > 0 Then
        GridLoteDetalleConsulta.Rows.Remove(0, GridLoteDetalleConsulta.Rows.Count)
    Endif
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    While Result_Trabajo.Available
        Numero_fila = GridLoteDetalleConsulta.Rows.Count
        GridLoteDetalleConsulta.Rows.Insert(Numero_fila, 1)
        GridLoteDetalleConsulta[Numero_fila, 0].Text =
Result_Trabajo["L.codigolote"]
        GridLoteDetalleConsulta[Numero_fila, 1].Text = Result_Trabajo["L.linea"]
        GridLoteDetalleConsulta[Numero_fila, 2].Text = Result_Trabajo["L.cantidad"]
        GridLoteDetalleConsulta[Numero_fila, 3].Text = Result_Trabajo["L.coste"]
        GridLoteDetalleConsulta[Numero_fila, 4].Text =
Result_Trabajo["L.numerolotecompra"]
        GridLoteDetalleConsulta[Numero_fila, 5].Text =
Result_Trabajo["P.razonsocial"]
        GridLoteDetalleConsulta[Numero_fila, 6].Text =
Result_Trabajo["L.nfacturacompra"]
        GridLoteDetalleConsulta[Numero_fila, 7].Text =
Result_Trabajo["L.fechafacturacompra"]
        GridLoteDetalleConsulta[Numero_fila, 8].Text = Result_Trabajo["M.nombre"]
        Result_Trabajo.MoveNext
    Wend
    GridLoteDetalleConsulta.Rows.Select(Linea_lote_Consulta, 1)
    FMain.Conexion_Base_Datos.Close
End

'' Nos muestra los datos del lote elegido<br>
'' Parametros:<br>
''     Codigo --> Codigo del lote
Private Sub Visualizar_Campos_Lote(Codigo As String)

    Dim Result_Trabajo As Result
    Dim Sql As String
    sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
    "articulo A on L.articulo = A.codigoarticulo where L.codigolote='" & codigo &
""
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
```

```
Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
Result_Trabajo.MoveFirst
TxtCodigoLote.Text = Result_Trabajo["L.codigolote"]
TxtFechaCreacion.Text = Result_Trabajo["L.fecha"]
TxtCantidad.Text = Result_Trabajo["L.cantidad"]
TxtCoste.Text = Result_Trabajo["L.coste"]
TxtCantidadBruta.Text = Result_Trabajo["L.cantidadbruta"]
TxtMerma.Text = Result_Trabajo["L.merma"]
TxtArticulo.Text = Result_Trabajo["A.nombre"]
FMain.Conexion_Base_Datos.Close
```

End

' Limpiamos los datos

```
Private Sub Limpiar_Campos()
```

```
    TxtCodigoLote.Text = ""
    TxtFechaCreacion.Text = ""
    TxtCantidad.Text = ""
    TxtCoste.Text = ""
    TxtCantidadBruta.Text = ""
    TxtMerma.Text = ""
    TxtArticulo.Text = ""
```

End

2.2.5.3 *Frm_Visualiza_Lote.class*

```
Public Lote_Actual As String
```

```
Public Sub Form_Open()
```

```
    Me.Title = "Lote :" & Frm_ConsultaLotes.Lote_Generar
    Lote_Actual = Frm_ConsultaLotes.Lote_Generar
    Generar_Lote()
```

End

```
Public Sub BtnSalirLote_Click()
```

```
    Me.Close
```

End

```
Public Sub BtnGenerarPdf_Click()
```

```
    Exec ["wkhtmltopdf", FMain.CarpetaTrazabilidad & "/" & Lote_Actual &
".html", FMain.CarpetaTrazabilidad & "/" & Lote_Actual & ".pdf"] Wait
    Exec ["evince", FMain.CarpetaTrazabilidad & "/" & Lote_Actual & ".pdf"]
```

End

```
Private Sub Generar_Lote()
```

```
    Dim cadena As String
    Dim Sql As String
    Dim Linea_Detalles As Result
    Dim Lote As String
    Dim Aux As String
    Dim NombreListado As String
    Dim RutaListado As String
    ' Cabecera Lote
    Dim Fecha As String
```

[illegible]

```
Linea_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select nombre from
materiaprima where codigomateriaprima='" & aux & "'", "nombre")
Linea_Cantidad = Linea_Detalles["cantidad"]
Linea_Coste = Linea_Detalles["coste"]
Linea_LoteCompra = Linea_Detalles["numerolote compra"]
aux = Linea_Detalles["proveedor"]
Linea_Proveedor = Utilidades.Obtener_Campo_Tabla("select razonsocial from
proveedor where codigoproveedor='" & aux & "'", "razonsocial")
Linea_NfacturaCompra = Linea_Detalles["nfactura compra"]
Linea_FechaFacturaCompra = Linea_Detalles["fecha factura compra"]
cadena &= " <tr style=\"border:1px solid black;\"><td>" & Linea_MateriaPrima
& "</td><td>" & Linea_Cantidad & "</td><td>" & Linea_Coste & "</td><td>"
cadena &= Linea_Proveedor & "</td><td>" & Linea_LoteCompra & "</td><td>" &
Linea_NfacturaCompra & "</td><td>" & Linea_FechaFacturaCompra & "</td></tr>"
Linea_Detalles.MoveNext
Wend

cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
WebLote.Url = RutaListado & ".html"
End
```

2.2.5.4 *Frm_Trazabilidad.class*

```
Public Factura_Albaran As String

Public Sub Form_Open()

    Dim Sql As String
    Me.Title = "GESTION DE TRAZABILIDAD"
    sql = "select codigolote from lotes"
    Utilidades.Llenar_Combobox(sql, cmblotos, "codigolote")
    cmblotos.Index = -1

End

Public Sub BtnSalirTrazabilidad_Click()
    Me.Close
End

Public Sub BtnVisualizar_Click()
    Dim Lote As Form
    If ComprobarLote() Then
        Factura_Albaran = cmblotos.Text
        lote = New Frm_Visualiza_Lote_Trazabilidad
        activar(lote, cmblotos.Text)
        cmblotos.Index = -1
    Endif
End
```

Capítulo 2. CÓDIGO FUENTE

```
''***** Funciones privadas*****  
''  
''*****  
  
'' Procedimiento para Generar mostrar en el Workspace el lote seleccionado <br>  
'' Parametros:<br>  
''     elform --> formulario abrir  
Public Procedure activar(elform As Form, lote As String)  
  
    Dim cont As Form  
    Dim lAbierto As Boolean  
    Dim Titulo As String  
    Titulo = "Lote :" & lote  
    For Each cont In wrktrazabilidad.windows 'buscamos si ya hay habierto un  
formulario con el mismo nombre del que se quiere abrir, o sea, si ya está  
abierto.  
        'If cont.name = elform.Name And cont.text = elform.text Then  
  
            If cont.title = Titulo Then  
                lAbierto = True  
                'Break  
            Endif  
        Next  
  
        If lAbierto Then  
            wrktrazabilidad.ActiveWindow = cont 'Si ya está lo hacemos activo y  
eliminamos el que se quería abrir por ser un duplicado  
            elform.Delete  
        Else  
            wrktrazabilidad.Add(elform) 'lo añadimos al workspace  
        Endif  
  
End  
  
'' comprobamos que se ha introducido un lote correcto  
Private Function ComprobarLote() As Boolean  
  
    Dim Validar As Boolean  
    Dim Sql As String  
    validar = True  
    sql = "select codigolote from lotes where codigolote='" & cmblotestext & "'"<br>  
    If cmblotestext = "" Then  
        Balloon.Error("Introduzca Lote para ver Trazabilidad", cmblotestext)  
        validar = False  
        cmblotestext.SetFocus  
    Else If Not Utilidades.Comprobar_existe_dato(Sql)  
        Balloon.Error("Lote no existe", cmblotestext)  
        Validar = False  
        cmblotestext.SetFocus  
    Endif  
    Return validar  
  
End
```


2.2.5.5 *Frm_Visualiza_Lote_Trazabilidad.class*

```
Public Lote_Actual As String

Public Sub Form_Open()
    Me.Title = "Lote :" & Frm_Trazabilidad.Factura_Albaran
    Lote_Actual = Frm_Trazabilidad.Factura_Albaran
    Generar_Trazabilidad()
End

'''***** Funciones privadas*****'''

''' Procedimiento para generar un fichero html con la trazabilidad del lote
elegido.

Public Sub Generar_Trazabilidad()

    Dim Sql As String
    Dim Cadena As String
    Dim Ruta_Generar_Trazabilidad As String
    ' Result para obtener los datos.
    Dim Lote As Result
    Dim Composicion_Lote As Result
    Dim Clientes_lote As Result
    'Variables de los datos del Lote.
    Dim Lote_Articulo As String
    Dim Lote_Fecha As String
    Dim Lote_Cantidad As String
    Dim Lote_Coste As String
    Dim Lote_Cantidadbruta As String
    Dim Lote_Merma As String
    'Variables para las lineas detalle lote.
    Dim Lote_Detalle_Cantidad As String
    Dim Lote_Detalle_Coste As String
    Dim Lote_Detalle_NumeroLoteCompra As String
    Dim Lote_Detalle_Proveedor As String
    Dim Lote_Detalle_NumeroFacturaCompra As String
    Dim Lote_Detalle_FechaFacturaCompra As String
    Dim Lote_Detalle_Materiaprima As String
    ' Variables Cliente Factura
    Dim Cliente As String
    Dim Factura As String

    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    sql = "select
L.codigolote,L.fecha,L.cantidad,L.coste,L.cantidadbruta,L.merma,A.nombre from
lotes L left join " &
        "articulo A on L.articulo = A.codigoarticulo where L.codigolote='" &
Lote_Actual & "'"
    Lote = FMain.Conexion_Base_Datos.Exec(Sql)
    Lote_Articulo = Lote["A.nombre"]
    Lote_Fecha = Lote["L.fecha"]
    Lote_Cantidad = Lote["L.cantidad"]
    Lote_Coste = Lote["L.coste"]
    Lote_Cantidadbruta = Lote["L.cantidadbruta"]
    Lote_Merma = Lote["L.merma"]
```

```

Ruta_Generar_Trazabilidad = FMain.CarpetaTrazabilidad & "/" & Lote_Actual
cadena = cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html; charset=utf-8\"/></head><body>"
cadena &= "<div><img src=\"\" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\" width:900px;height:200px;\"> </div><br><br>"
cadena &= "<div style=\"border:1px solid black;
width:300px;float:left;\"><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&TRAZABILIDAD DE LOTES
<br>Lote Numero : </strong> " & Lote_Actual & "<br>"
cadena &= "<strong>Articulo :</strong> " & Lote_Articulo & " <br>"
cadena &= "<strong>Coste :</strong> " & Lote_Coste & " <br>"
cadena &= "<strong>Cantidad Bruta :</strong> " & Lote_Cantidadbruta & " <br>"
cadena &= "<strong>Merma :</strong> " & Lote_Merma & " <br>"
cadena &= "<strong>Cantidad Neta :</strong> " & Lote_Cantidad & " <br>"
cadena &= "<strong>Fecha :</strong> " & Lote_Fecha & "</div><br><br><br><br>"
cadena &= "<div style=\"width:900px;\"><br><br><br><br><center>
<h1>COMPOSICION DEL LOTE</h1></center></div>"
cadena &= "<br><br><div style=\" width:500px;clear:both;\">"
cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse: collapse;\"><tr><th>Materia Prima</th><th>Cantidad</th><th>Coste</th>"
cadena &= "<th> Proveedor </th> <th>N Factura Compra </th> <th>N Lote Compra
</th><th>Fecha Compra </th></tr>"
FMain.Conexion_Base_Datos.Close
FMain.Conexion_Base_Datos.Open
sql = "select
L.codigolote,L.linea,L.cantidad,L.coste,L.numerolotecompra,P.razonsocial,L.nfact
uracompra,L.fechafacturacompra,M.nombre from lotedetalle L left join " &
"proveedor P on L.proveedor = P.codigoproveedor left join materiaprime M
on L.materiaprime = M.codigomateriaprime where L.codigolote='" & Lote_Actual &
""
Composicion_Lote = FMain.Conexion_Base_Datos.Exec (Sql)
Composicion_Lote.MoveFirst
While Composicion_Lote.Available
Lote_Detalle_Cantidad = Composicion_Lote["L.cantidad"]
Lote_Detalle_Coste = Composicion_Lote["L.coste"]
Lote_Detalle_NumeroLoteCompra = Composicion_Lote["L.numerolotecompra"]
Lote_Detalle_Proveedor = Composicion_Lote["P.razonsocial"]
Lote_Detalle_NumeroFacturaCompra = Composicion_Lote["L.nfacturacompra"]
Lote_Detalle_FechaFacturaCompra = Composicion_Lote["L.fechafacturacompra"]
Lote_Detalle_Materiaprime = Composicion_Lote["M.nombre"]
cadena &= " <tr style=\"border:1px solid black;\"><td>" &
Lote_Detalle_Materiaprime & "</td><td>" & Lote_Detalle_Cantidad & "</td><td>" &
Lote_Detalle_Coste & "</td><td>"
cadena &= Lote_Detalle_Proveedor & "</td><td>" &
Lote_Detalle_NumeroFacturaCompra & "</td><td>" & Lote_Detalle_NumeroLoteCompra &
"</td><td>" & Lote_Detalle_FechaFacturaCompra & "</td></tr>"
Composicion_Lote.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<div style=\"width:900px;float:left;\"><br><br><br><br><center>
<h1>CLIENTES</h1></center></div>"
cadena &= "<div style=\"width:900px;float:left;\"><center>
<h2>FACTURAS</h2></center></div>"
FMain.Conexion_Base_Datos.Close
FMain.Conexion_Base_Datos.Open
'sql = "select C.razonsocial,F.numerofactura from facturas F,facturasdetalles
D left join clientes C on F.cliente = C.codigocliente" &
" " where D.lote='" & Lote_Actual & "' and
F.numerofactura=D.numerofactura"

```

```

    sql = "select distinct C.razonsocial, F.numerofactura from facturas
F,facturasdetalles D, clientes C where C.codigocliente = F.cliente and " &
    "F.numerofactura=D.numerofactura and D.lote='" & Lote_Actual & "'"
    Clientes_lote = FMain.Conexion_Base_Datos.Exec(Sql)
    Clientes_lote.MoveFirst
    cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse: collapse;\><tr><th>Cliente</th><th>N Factura</th></tr>"
    While Clientes_lote.Available
        Cliente = Clientes_lote["C.razonsocial"]
        Factura = Clientes_lote["F.numerofactura"]
        cadena &= " <tr style=\"border:1px solid black;\><td>" & Cliente &
"</td><td>" & Factura & "</td></tr>"
        Clientes_lote.MoveNext
    Wend
    cadena &= " </table></div>"
    cadena &= "<div style=\"width:900px;float:left;\><center>
<h2>ALBARANES</h2></center></div>"
    FMain.Conexion_Base_Datos.Close
    FMain.Conexion_Base_Datos.Open
    cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse: collapse;\><tr><th>Cliente</th><th>N Albaran</th></tr>"
    sql = "select distinct C.razonsocial, F.numeroalbaran from albaranes
F,albaranesdetalle D, clientes C where C.codigocliente = F.cliente and " &
    "F.numeroalbaran=D.numeroalbaran and D.lote='" & Lote_Actual & "'"
    Clientes_lote = FMain.Conexion_Base_Datos.Exec(Sql)
    Clientes_lote.MoveFirst
    While Clientes_lote.Available
        Cliente = Clientes_lote["C.razonsocial"]
        Factura = Clientes_lote["F.numeroalbaran"]
        cadena &= " <tr style=\"border:1px solid black;\><td>" & Cliente &
"</td><td>" & Factura & "</td></tr>"
        Clientes_lote.MoveNext
    Wend
    cadena &= " </table></div>"
    cadena &= "<br><br><br><br><br><br><br><br><br><br><br><br></body></html>"

    File.Save(Ruta_Generar_Trazabilidad & ".html", cadena)
    WebTrazabilidad.Url = Ruta_Generar_Trazabilidad & ".html"
End

Public Sub Form_Close()

    Kill FMain.CarpetaTrazabilidad & "/" & Lote_Actual & ".html"
End

Public Sub BtnSalirTrazabilidad_Click()

    Me.close

End

Public Sub BtnGenerarPdf_Click()

    Exec ["wkhtmltopdf", FMain.CarpetaTrazabilidad & "/" & Lote_Actual & ".html",
FMain.CarpetaTrazabilidad & "/" & Lote_Actual & ".pdf"] Wait
    Exec ["evince", FMain.CarpetaTrazabilidad & "/" & Lote_Actual & ".pdf"]

End

```

2.2.6 Modulo Utilidades.

2.2.6.1 *Frm_Conexionbasedatos.class*

```
Public Sub Form_Open()  
    Me.Center  
    Me.title = "Conexion a la base de datos"  
    TxtHost.Text = "localhost"  
    TxtPuerto.text = "3306"  
    If Exist(fmain.DirConexion & "/conexion.conf") Then  
        CargarDatosConexion()  
    Endif  
End  
  
Private Sub CargarDatosConexion()  
  
    TxtHost.text = FMain.FicheroConexionBD["ConexionBD/Host"]  
    TxtPuerto.text = FMain.FicheroConexionBD["ConexionBD/Puerto"]  
    TxtUsuario.text = FMain.FicheroConexionBD["ConexionBD/Usuario"]  
    TxtPassword.Text = Utilidades.cripto(2,  
FMain.FicheroConexionBD["ConexionBD/Password"])  
    TxtBaseDatos.text = FMain.FicheroConexionBD["ConexionBD/BaseDatos"]  
  
End  
  
Public Sub BtnSalir_Click()  
    Me.Close  
End  
  
Public Sub BtnAceptar_Click()  
    Dim Validar As Boolean = True  
    Dim FicheroConexionBD As Settings  
    FicheroConexionBD = New Settings(fmain.DirConexion & "/conexion.conf")  
    If TxtHost.text = "" Then  
        Balloon.Error("Debe introducir direccion IP, si es el mismo equipo  
LOCALHOST", TxtHost)  
        validar = False  
    Else  
        Balloon.Hide(TxtHost)  
    Endif  
  
    If TxtPuerto.text = "" Then  
        Balloon.Error("Debe introducir puerto normalmente es 3306", TxtPuerto)  
        validar = False  
    Else  
        Balloon.Hide(TxtPuerto)  
    Endif  
  
    If TxtUsuario.text = "" Then  
        Balloon.Error("Debe introducir usuario de acceso a la base de datos",  
TxtUsuario)  
        validar = False  
    Else  
        Balloon.Hide(TxtUsuario)  
    Endif  
  
    If TxtPassword.text = "" Then
```

```
Balloon.Error("Debe introducir contraseña de usuario", TxtPassword)
validar = False
Else
    Balloon.Hide(TxtPassword)
Endif

If TxtBaseDatos.text = "" Then
    Balloon.Error("Debe introducir nombre de la base de datos", TxtBaseDatos)
    validar = False
Else
    Balloon.Hide(TxtBaseDatos)
Endif

If validar Then

    fmain.Conexion_Base_Datos.Type = "mysql"
    fmain.Conexion_Base_Datos.Host = TxtHost.Text
    fmain.Conexion_Base_Datos.Port = TxtPuerto.Text
    fmain.Conexion_Base_Datos.User = TxtUsuario.Text
    fmain.Conexion_Base_Datos.Password = TxtPassword.Text
    fmain.Conexion_Base_Datos.Name = TxtBaseDatos.Text
    Try fmain.Conexion_Base_Datos.Open
        fmain.Conexion_Base_Datos.Close
    If Error Then
        Message.Error("Error en datos de conexion. Revise la configuracion.")
    Else
        LblDatosActualizados.Visible = True
        FicheroConexionBD["ConexionBD/Host"] = TxtHost.Text
        FicheroConexionBD["ConexionBD/Puerto"] = TxtPuerto.text
        FicheroConexionBD["ConexionBD/Usuario"] = TxtUsuario.text
        FicheroConexionBD["ConexionBD/Password"] = Utilidades.cripto(1,
TxtPassword.text)
        FicheroConexionBD["ConexionBD/BaseDatos"] = TxtBaseDatos.text
        FicheroConexionBD.Save

        Me.Close
    Endif
Endif

End

Public Sub TxtHost_KeyPress()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtPuerto.SetFocus
    Endif

End

Public Sub TxtPuerto_KeyPress()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtUsuario.SetFocus
    Endif

End
```

```
Public Sub TxtUsuario_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtPassword.SetFocus  
    Endif  
  
End
```

```
Public Sub TxtPassword_KeyRelease()  
  
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then  
        TxtBaseDatos.SetFocus  
    Endif  
  
End
```

End

2.2.6.2 *Frm_CopiasSeguridad.class*

```
Public Sub Form_Open()  
    Me.Title = "COPIAS DE SEGURIDAD"  
    DircopiaMysql.Value = FMain.CarpetaCopiasSeguridad  
    Txtfichero.Text = FMain.Conexion_Base_Datos.Name & Format(Now,  
"yyyymmddhhnnss") & ".sql"  
    dircopiatar.value = User.Home  
    lblrutatrabajo.Text = FMain.CarpetaCarnea  
    TxtHost.Text = FMain.Conexion_Base_Datos.Host  
    Txtpuerto.Text = FMain.Conexion_Base_Datos.Port  
    TxtUsuario.Text = FMain.Conexion_Base_Datos.User  
    TxtPassword.Text = FMain.Conexion_Base_Datos.Password  
    TxtBaseDatos.Text = FMain.Conexion_Base_Datos.Name  
    Txtficheroruta.Text = TxtBaseDatos.Text & Format(Now, "yyyymmddhhnnss") &  
".tar.gz"  
    FileChooser.Root = FMain.CarpetaCarnea  
    FileChooser.Filter = ["*.sql", "Ficheros SQL"]  
End  
  
Public Sub BtnSalirTrazabilidad_Click()  
    Me.Close  
End  
  
Public Sub BtnRealizaCopia_Click()  
  
    Dim Copia_Mysql As String  
    Dim Comando As String  
    Dim Opciones As String  
    Dim Conexion As New Connection  
    Dim validar As Boolean  
    validar = True  
    'mysqldump -h localhost -P 3306 -u gambas -pgambas --databases carnea --add-  
drop-database --add-drop-table >  
/home/rescobar/carnea/copiasseguridad/carnea.sql  
    Opciones = " --add-drop-database --add-drop-table"  
    Copia_Mysql = DircopiaMysql.Value & "/" & Txtfichero.Text  
    comando = "mysqldump -h " & TxtHost.Text & " -P " & Txtpuerto.Text & " -u " &  
TxtUsuario.Text & " -p" & TxtPassword.Text & " --databases " & TxtBaseDatos.Text  
&  
    " --add-drop-database --add-drop-table > " & Copia_Mysql
```

```
Conexion.Type = "mysql"
Conexion.Host = TxtHost.Text
Conexion.Port = Txtpuerto.Text
Conexion.User = TxtUsuario.Text
Conexion.Password = TxtPassword.Text
Conexion.Name = TxtBaseDatos.Text
If DirCopiaMysql.Value = "" Then
    Balloon.Error("Debe introducir Ruta para generar el fichero", DirCopiaMysql)
    validar = False
Else If TxtFichero.Text = ""
    Balloon.Error("Debe introducir nombre de fichero", Txtfichero)
    validar = False
Else If DirCopiatar.Value = ""
    Balloon.Error("Debe introducir Ruta para generar el fichero", DirCopiatar)
    validar = False
Else If TxtFicheroRuta.Text = ""
    Balloon.Error("Debe introducir nombre de fichero", Txtficheroruta)
    validar = False

Endif

If validar Then
    Try Conexion.Open
    Conexion.Close
    If Error Then
        Message.Error("Error en datos de conexion. Revise la configuracion.")
    Else
        Shell Comando Wait
    Endif
    'tar czfv carneaerp.tar.gz /home/rescobar/carneaerp
    comando = "tar czf " & dircopiatar.Value & "/" & Txtficheroruta.Text & " " &
FMain.CarpetaCarnea
    Shell Comando Wait
    If (Not Exist(Copia_Mysql)) Or (Not Exist(dircopiatar.Value & "/" &
Txtficheroruta.Text)) Then
        textmensaje.Visible = True
        textmensaje.Foreground = Color.Red
        textmensaje.Text = "ERROR AL CREAR COPIA"
    Else
        textmensaje.Visible = True
        textmensaje.Foreground = Color.Green
        textmensaje.Text = "COPIA REALIZADA CON EXITO"
    Endif
End If

End

Public Sub BtnRestaurar_Click()
    Dim Comando As String
    Dim Conexion As New Connection
    Conexion.Type = "mysql"
    Conexion.Host = TxtHost.Text
    Conexion.Port = Txtpuerto.Text
    Conexion.User = TxtUsuario.Text
    Conexion.Password = TxtPassword.Text
    Conexion.Name = TxtBaseDatos.Text
    If FileChooser.SelectedPath = "" Then
        Balloon.Error("Debe elegir fichro para restaurar", FileChooser)
```

```
Else
    Try Conexion.Open
    Conexion.Close
    If Error Then
        Message.Error("Error en datos de conexion. Revise la configuracion.")
    Else
        comando = "mysql -h " & TxtHost.Text & " -P " & Txtpuerto.Text & " -u " &
TxtUsuario.Text & " -p" & TxtPassword.Text & " < " & FileChooser.SelectedPath
        Shell Comando Wait
        TextInfoRestaurar.Enabled = True
        TextInfoRestaurar.Text = "Restauracion realizada con exito."
    Endif
Endif

Endif

End
```

2.2.6.3 *Frm_Informes.class*

```
Public Accion_Realizar As String
Public Articulo_MateriaPrima As String
Public Cliente_FacturaAlbaran As String
Public Proveedor_MateriaPrima As String
Public Vendedor_Clientes As String
Public Lote_Generar As String
Public Cliente_Averia As String
Public Articulo_Averia As String

Public Sub BtnSalirListados_Click()
    Me.Close
End

Public Sub Form_Open()
    Me.Title = "GENERACION DE INFORMES"
    Configurar_Objeto_Informmes()
End

'''***** Funciones Articulos*****'''
'''*****'''
Public Sub RdbArticuloMateriaprima_Click()

    cmbarticulos.Enabled = True

End

Public Sub RdbGeneralArticulos_Click()

    cmbarticulos.Enabled = False

End

Public Sub BtnArticulosGenerar_Click()

    If RdbGeneralArticulos.Value = True Then
        GenerarListadoArticulos()
    Else If RdbArticuloMateriaprima.Value = True
        If cmbarticulos.Text = "" Then
```



```
Balloon("Debe elegir Articulo para ver su composicion.", cmbarticulos)
Else
    Articulo_MateriaPrima = cmbarticulos.Text
    GenerarListadoArticulosMateriaprima()
Endif

Endif
End
Public Sub Form_Close()

    'Kill FMain.CarpetaListados & "/*.html"
    Dim nombreadchivo As String
    For Each nombreadchivo In Dir(FMain.CarpetaListados, "*.*", GB.File).Sort()
        Kill FMain.CarpetaListados & "/" & nombreadchivo
    Next

End

''***** Funciones Clientes*****
''
''*****

Public Sub BtnClientesGenerar_Click()

    If rdbClientes.value = True Then
        GenerarListadoClientes()
    Else If RdbClienteFacturas.Value = True
        If cmbclientes.Text = "" Then
            Balloon("Debe elegir Cliente para ver las Factuas/Albaranes.",
cmbclientes)
        Else
            Cliente_FacturaAlbaran = cmbclientes.Text
            GenerarListadoClientesFacturasAlbaranes()
        Endif
    Endif

End

End

Public Sub RdbClienteFacturas_Click()

    cmbclientes.Enabled = True

End

Public Sub RdbClientes_Click()

    cmbclientes.Enabled = False

End

''***** Funciones Proveedores*****
''
''*****

Public Sub BtnProveedorGenerar_Click()

    If RdbProveedor.Value = True Then
        GenerarListadoProveedores()
```

Capítulo 2. CÓDIGO FUENTE

```
Else If RdbProveedorMateriaPrima.Value = True
    If cmbproveedor.Text = "" Then
        Balloon("Debe elegir Proveedor para ver Materias Primas.", cmbproveedor)
    Else
        Proveedor_MateriaPrima = cmbproveedor.Text
        GenerarListadoProveedoresMateriaPrima()
    Endif
Endif

End

Public Sub RdbProveedor_Click()

    cmbproveedor.Enabled = False

End

Public Sub RdbProveedorMateriaPrima_Click()

    cmbproveedor.Enabled = True

End

'''***** Funciones Vendedores*****'''

Public Sub BtnVendedoresGenerar_Click()

    If RdbVendedor.Value = True Then
        GenerarListadoVendedores()
    Else If RdbVendedorClientes.Value = True
        If cmbvendedor.Text = "" Then
            Balloon("Debe elegir Vendedor para ver Clientes", cmbvendedor)
        Else
            Vendedor_Clientes = cmbvendedor.Text
            GenerarListadoVendedorClientes()
        Endif
    Endif

End

Public Sub RdbVendedor_Click()

    cmbvendedor.Enabled = False

End

Public Sub RdbVendedorClientes_Click()

    cmbvendedor.Enabled = True

End

'''***** Funciones Lotes *****'''
```

Capítulo 2. CÓDIGO FUENTE

```
Public Sub BtnLotesGenerar_Click()

    If cmblote.text = "" Then
        Balloon("Debe elgir Lote", cmblote)
    Else If Not Utilidades.Comprobar_existe_dato("select * from lotes where
codigolote='" & cmblote.Text & "'")
        Balloon("Lote no existe", cmblote)
    Else
        Lote_Generar = UCase(cmbLote.Text)
        GenerarLote()
    Endif

End

'''***** Funciones Averias *****'''

Public Sub RdbTodasAverias_Click()

    CmbClienteAveria.Enabled = False
    CmbArticuloAveria.Enabled = False
    cmbarticuloaveria.Index = -1
    CmbClienteAveria.Index = -1
End

Public Sub RdbclienteAveria_Click()

    CmbClienteAveria.Enabled = True
    CmbArticuloAveria.Enabled = False
    cmbarticuloaveria.Index = -1
End

Public Sub RdbArticuloAveria_Click()

    CmbClienteAveria.Enabled = False
    CmbArticuloAveria.Enabled = True
    CmbClienteAveria.Index = -1
End

Public Sub BtnAveriasGenerar_Click()

    If RdbTodasAverias.Value = True Then
        GenerarListadoAveriasTodas()
    Else If RdbclienteAveria.Value = True
        If CmbClienteAveria.Text = "" Then
            Balloon("Debe elegir Cliente para ver Averias", CmbClienteAveria)
        Else
            Cliente_Averia = CmbClienteAveria.Text
            GenerarListadoAveriasClientes()
        Endif
    Else If RdbArticuloAveria.value = True
        If CmbArticuloAveria.Text = "" Then
            Balloon("Debe elegir Articulo para ver Averias", Cmbarticuloaveria)
        Else
            Articulo_Averia = cmbarticuloaveria.Text
            GenerarListadoAveriasArticulos()
        Endif
    Endif

End
```

```
Endif

End

''***** Funciones privadas*****
''
''*****

'' Procedimiento para Generar mostrar en el Workspace el lote seleccionado <br>
'' Parametros:<br>
''     elform --> formulario abrir
Public Procedure activar(elform As Form)

    Dim cont As Form
    Dim lAbierto As Boolean

    For Each cont In WrkInfome.windows 'buscamos si ya hay habierto un
formulario con el mismo nombre del que se quiere abrir, o sea, si ya está
abierto.
        'If cont.name = elform.Name And cont.text = elform.text Then
        If cont.title = elform.Title Then
            lAbierto = True
            'Break
        Endif
    Next

    If lAbierto Then
        WrkInfome.ActiveWindow = cont 'Si ya está lo hacemos activo y
eliminamos el que se quería abrir por ser un duplicidado
        elform.Delete
    Else
        WrkInfome.Add(elform) 'lo añadimos al workspace
    Endif

End

'' Configuracion de los objetos del formulario
Private Sub Configurar_Objeto_Informmes()
    Dim sql As String
    RdbGeneralArticulos.Value = True
    rdbClientes.Value = True
    RdbProveedor.Value = True
    RdbVendedor.Value = True
    RdbTodasAverias.Value = True
    Utilidades.Llenar_Combobox("select nombre from articulo", cmbarticulos,
"nombre")
    Utilidades.Llenar_Combobox("select razonsocial from clientes", cmbclientes,
"razonsocial")
    Utilidades.Llenar_Combobox("select razonsocial from proveedor", cmbproveedor,
"razonsocial")
    Utilidades.Llenar_Combobox("select nombre from vendedor", cmbvendedor,
"nombre")
    Utilidades.Llenar_Combobox("select codigolote from lotes", cmb lote,
"codigolote")
    sql = "select distinct C.razonsocial,T.nombre,A.peso,A.lote,A.descripcion
from averia A" &
    " left join vendedor V on A.vendedor=V.codigovendedor left join clientes C on
A.cliente=C.codigocliente left join " &
```

Capítulo 2. CÓDIGO FUENTE

```
" articulo T on A.articulo=T.codigoarticulo"
' Utilidades.Llenar_Combobox("select nombre from articulo", cmbarticuloaveria,
"nombre")
Utilidades.Llenar_Combobox(sql, cmbarticuloaveria, "T.nombre")
Utilidades.Llenar_Combobox(sql, CmbClienteAveria, "C.razonsocial")
cmbclientes.Index = -1
cmbarticulos.Index = -1
cmbproveedor.Index = -1
cmbvendedor.Index = -1
cmbblote.Index = -1
cmbarticuloaveria.Index = -1
CmbClienteAveria.Index = -1
```

End

```
' ' Procedimiento para generar el listado de los articulos
Private Sub GenerarListadoArticulos()
```

```
Dim Listado As Form
Accion_Realizar = "RdbGeneralArticulos"
listado = New Frm_Listado
activar(listado)
```

End

```
' ' Procedimiento para generar el listado de las materias primas de un articulo.
Private Sub GenerarListadoArticulosMateriaprima()
```

```
Dim Listado As Form
Accion_Realizar = "RdbArticuloMateriaprima"
listado = New Frm_Listado
activar(listado)
```

End

```
' ' Procedimiento para generar el listado de los Clientes.
Private Sub GenerarListadoClientes()
```

```
Dim Listado As Form
Accion_Realizar = "rdbClientes"
listado = New Frm_Listado
activar(listado)
```

End

```
' ' Procedimiento para generar el listado de las Facturas/Albaranes por Cliente.
Private Sub GenerarListadoClientesFacturasAlbaranes()
```

```
Dim Listado As Form
Accion_Realizar = "RdbClienteFacturas"
listado = New Frm_Listado
activar(listado)
```

End

```
' ' Procedimiento para generar el Listado de Proveedores
```

```
Private Sub GenerarListadoProveedores()
```

```
Dim Listado As Form
```

Capítulo 2. CÓDIGO FUENTE

```
Accion_Realizar = "RdbProveedor"
listado = New Frm_Listado
activar(listado)

End
'' Procedimiento para generar el Listado de las Materias Primas del Proveedor

Private Sub GenerarListadoProveedoresMateriaPrima()

    Dim Listado As Form
    Accion_Realizar = "RdbProveedorMateriaPrima"
    listado = New Frm_Listado
    activar(listado)

End
'' Procedimiento para generar el Listado de Vendedores
Private Sub GenerarListadoVendedores()

    Dim Listado As Form
    Accion_Realizar = "RdbVendedor"
    listado = New Frm_Listado
    activar(listado)

End

'' Procedimiento para generar el Listado de los Clientes del Vendedor
Private Sub GenerarListadoVendedorClientes()

    Dim Listado As Form
    Accion_Realizar = "RdbVendedorClientes"
    listado = New Frm_Listado
    activar(listado)

End
'' Procedimiento para generar El Lote
Private Sub GenerarLote()

    Dim Listado As Form
    Accion_Realizar = "GenerarLote"
    listado = New Frm_Listado
    activar(listado)

End

'' Procedimiento para generar el Listado de todas las Averias
Private Sub GenerarListadoAveriasTodas()

    Dim Listado As Form
    Accion_Realizar = "RdbTodasAverias"
    listado = New Frm_Listado
    activar(listado)

End

'' Procedimiento para generar el Listado de las Averias dado un Cliente
Private Sub GenerarListadoAveriasClientes()

    Dim Listado As Form
```

```
Accion_Realizar = "RdbClientesAverias"
listado = New Frm_Listado
activar(listado)

End

'' Procedimiento para generar el Listado de las Averias dado un Articulo
Private Sub GenerarListadoAveriasArticulos()

    Dim Listado As Form
    Accion_Realizar = "RdbArticulosAverias"
    listado = New Frm_Listado
    activar(listado)

End
```

2.2.6.4 *Frm_Listado.class*

```
Public NombreListado As String
Public RutaListado As String

Public Sub BtnSalirListado_Click()

    Me.Close

End

Public Sub BtnGenerarPdf_Click()

    Exec ["wkhtmltopdf", FMain.CarpetaListados & "/" & NombreListado & ".html",
FMain.CarpetaListados & "/" & NombreListado & ".pdf"] Wait
    Exec ["evince", FMain.CarpetaListados & "/" & NombreListado & ".pdf"]

End

Public Sub Form_Open()

    NombreListado = Frm_Informes.Accion_Realizar
    RutaListado = FMain.CarpetaListados
    If NombreListado = "RdbGeneralArticulos" Then
        Me.Title = "LISTADO ARTICULOS"
        GenberarInformeArticulos()
    Else If NombreListado = "RdbArticuloMateriaprima"
        Me.Title = "LISTADO MATERIAS PRIMAS ARTICULO"
        GenberarInformeArticuloMateriasPrimas()
    Else If NombreListado = "rdbClientes"
        Me.Title = "LISTADO DE CLIENTES"
        GenerarInformeClientes()
    Else If NombreListado = "RdbClienteFacturas"
        Me.Title = "LISTADO DEL FACTURAS Y ALBARANES"
        GenerarInformeClientesFacturasAlbaranes()
    Else If NombreListado = "RdbProveedor"
        Me.Title = "LISTADO DE PROVEEDORES"
        GenerarInformeProveedores()
    Else If NombreListado = "RdbProveedorMateriaPrima"
        Me.Title = "LISTADO MATERIAS PRIMAS PROVEEDOR"
        GenerarInformeProveedorMateriaPrima()
    Else If NombreListado = "RdbVendedor"
```

208

Capítulo 2. CÓDIGO FUENTE

```
cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse: collapse;\"><tr><th>Codigo Materia Prima</th><th>Nombre Materia
Prima</th></tr>"

While MateriasPrimas.Available
    Codigo_MateriaPrima = MateriasPrimas["M.codigomateriaprima"]
    Nombre_MateriaPrima = MateriasPrimas["M.nombre"]

    cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo_MateriaPrima
& "</td><td>" & Nombre_MateriaPrima & "</td></tr>"
    MateriasPrimas.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
Weblistado.Url = RutaListado & ".html"

End

Private Sub GenberarInformeArticulos()
    Dim Sql As String
    Dim Cadena As String
    Dim Articulos As Result
    ' Datos de los articulos
    Dim Codigo_Articulo As String
    Dim Nombre_Articulo As String
    Dim Familia_Articulo As String
    NombreListado = "ListadoArticulos"
    RutaListado = RutaListado & "/ListadoArticulos"
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    sql = "select * from articulo"
    Articulos = FMain.Conexion_Base_Datos.Exec(Sql)

    cadena = cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html;charset=utf-8\"/></head><body>"
    cadena &= "<style> table td {text-align:center;} </style>"

    cadena &= "<div><img src=\"\" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\" width:900px;height:200px;\"> </div><br><br>"
    cadena &= "<div
style=\"width:900px;float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LIS
TADO ARTICULOS </strong></h1></center></div>"
    cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse: collapse;\"><tr><th>Codigo</th><th>Nombre</th><th>Familia</th></tr>"
    While Articulos.Available
        Codigo_Articulo = Articulos["codigoarticulo"]
        Nombre_Articulo = Articulos["nombre"]
        Familia_Articulo = Articulos["familia"]
        cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo_Articulo &
"</td><td>" & Nombre_Articulo & "</td><td>" & Familia_Articulo & "</td></tr>"
        Articulos.MoveNext
    Wend
    cadena &= " </table></div>"
    cadena &= "<br><br><br><br><br><br><br><br><br><br><br><br><br></body></html>"
    File.Save(RutaListado & ".html", cadena)
    Weblistado.Url = RutaListado & ".html"
```

210

```
Dim Sql As String
Dim Cadena As String
Dim Facturas As Result
Dim Albaranes As Result
' Datos de los clientes
Dim Codigo_Cliente As String
Dim Nombre_Cliente As String
'Datos Factura/Albaran
Dim Codigo As String
Dim Fecha As String
Dim Total As String
Dim Pagado As String
Dim Contabilizada As String

NombreListado = "ListadoClienteFacturaAlbaran"
RutaListado = RutaListado & "/ListadoClienteFacturaAlbaran"
If Not FMain.Conexion_Base_Datos.Opened Then
    FMain.Conexion_Base_Datos.Open
Endif
Nombre_Cliente = Frm_Informes.Cliente_FacturaAlbaran
Codigo_Cliente = Utilidades.Obtener_Campo_Tabla("select codigocliente from
clientes where razonsocial='" & Nombre_Cliente & "'", "codigocliente")
sql = "select * from facturas where cliente='" & Codigo_Cliente & "'"
Facturas = FMain.Conexion_Base_Datos.Exec(Sql)

cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html;charset=utf-8\"/></head><body>"
cadena &= "<style> table td {text-align:center;} </style>"
cadena &= "<div><img src=\"" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\"" width:900px;height:200px;\> </div><br><br>"
cadena &= "<div
style=\""width:900px;float:left\""><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LIS
TADO FACTURAS/ALBARANES DEL CLIENTE : " & Nombre_Cliente & "
</strong></h1></center></div>"
cadena &= "<div style=\""width:900px;float:left;\><center>
<h2>FACTURAS</h2></center></div>"
cadena &= " <table style=\""width:900px;border: 1px solid black;border-
collapse:"
collapse;\><tr><th>Codigo</th><th>Fecha</th><th>Total</th><th>Cobrada</th>"
cadena &= "<th>Contabilizada</th></tr> "
While Facturas.Available
    Codigo = Facturas["numerofactura"]
    Fecha = Facturas["fecha"]
    Total = Facturas["total"]
    Pagado = Facturas["cobrada"]
    Contabilizada = Facturas["contabilizada"]
    cadena &= " <tr style=\""border:1px solid black;\><td>" & Codigo &
"</td><td>" & Fecha & "</td><td>"
    cadena &= Total & "</td><td>" & Pagado & "</td><td>" & Contabilizada &
"</td></tr>"
    Facturas.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<div style=\""width:900px;float:left;\><center>
<h2>ALBARANES</h2></center></div>"
cadena &= " <table style=\""width:900px;border: 1px solid black;border-
collapse:"
collapse;\><tr><th>Codigo</th><th>Fecha</th><th>Total</th><th>Pagado</th>
```

```

cadena &= "<th>Facturada</th></tr> "
sql = "select * from albaranes where cliente='" & Codigo_Cliente & "'"
FMain.Conexion_Base_Datos.Close
FMain.Conexion_Base_Datos.Open
Albaranes = FMain.Conexion_Base_Datos.Exec(Sql)
Albaranes.MoveFirst
While Albaranes.Available
    Codigo = Albaranes["numeroalbaran"]
    Fecha = Albaranes["fecha"]
    Total = Albaranes["total"]
    Pagado = Albaranes["pagado"]
    Contabilizada = Albaranes["facturado"]
    cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo &
"</td><td>" & Fecha & "</td><td>"
    cadena &= Total & "</td><td>" & Pagado & "</td><td>" & Contabilizada &
"</td></tr>"
    Albaranes.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
Weblistado.Url = RutaListado & ".html"

End

Private Sub GenerarInformeProveedores()

    Dim Sql As String
    Dim Cadena As String
    Dim Proveedor As Result
    ' Datos de los Proveedores
    Dim Codigo_Proveedor As String
    Dim Nombre_Proveedor As String
    Dim Direccion_Proveedor As String
    Dim Nif_Proveedor As String
    Dim Localidad_Proveedor As String
    Dim Provincia_Proveedor As String
    NombreListado = "ListadoProveedores"
    RutaListado = RutaListado & "/ListadoProveedores"
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    sql = "select * from proveedor"
    Proveedor = FMain.Conexion_Base_Datos.Exec(Sql)
    cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html; charset=utf-8\"/></head><body>"
    cadena &= "<style> table td {text-align:center;} </style>"
    cadena &= "<div><img src=\"\" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\" width:900px;height:200px;\"> </div><br><br>"
    cadena &= "<div
style=\"width:900px;float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LIS
TADO PROVEEDORES </strong></h1></center></div>"
    cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse:
collapse;\"><tr><th>Codigo</th><th>Nombre</th><th>Nif</th><th>Direccion</th>"
    cadena &= "<th>Localidad</th><th>Provincia</th></tr> "
    While Proveedor.Available
        Codigo_Proveedor = Proveedor["codigoproveedor"]
        Nombre_Proveedor = Proveedor["razonsocial"]

```

```

    Direccion_Proveedor = Proveedor["direccion"]
    Nif_Proveedor = Proveedor["nif"]
    Localidad_Proveedor = Proveedor["localidad"]
    Provincia_Proveedor = Proveedor["provincia"]
    cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo_Proveedor &
"</td><td>" & Nombre_Proveedor & "</td><td>"
    cadena &= Nif_Proveedor & "</td><td>" & Direccion_Proveedor & "</td><td>" &
Localidad_Proveedor & "</td><td>" & Provincia_Proveedor & "</td></tr>"
    Proveedor.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
Weblistado.Url = RutaListado & ".html"

End

```

```

Private Sub GenerarInformeProveedorMateriaPrima()

```

```

    Dim Sql As String
    Dim Cadena As String
    Dim MateriasPrimas As Result
    ' Datos de los articulos y materias primas
    Dim Proveedor As String
    Dim Codigo_Proveedor As String
    Dim Nombre_MateriaPrima As String
    Dim Codigo_MateriaPrima As String
    NombreListado = "ListadoProveedorMateriaPrima"
    RutaListado = RutaListado & "/ListadoProveedorMateriaPrima"
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif

    Proveedor = Frm_Informes.Proveedor_MateriaPrima
    Codigo_Proveedor = Utilidades.Obtener_Campo_Tabla("select codigoproveedor from
proveedor where razonsocial='" & Proveedor & "'", "codigoproveedor")
    sql = "select M.codigomateriaprima,M.nombre from materiaprima M left join
proveedormateriaprima A on M.codigomateriaprima=A.materiaprima where
A.proveedor='" & Codigo_Proveedor & "'"
    MateriasPrimas = FMain.Conexion_Base_Datos.Exec(Sql)
    cadena = cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html; charset=utf-8\"/></head><body>"
    cadena &= "<style> table td {text-align:center;} </style>"

    cadena &= "<div><img src=\"'" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\" width:900px;height:200px;\"> </div><br><br>"
    cadena &= "<div
style=\"width:900px;float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
LIS
TADO MATERIAS PRIMAS PROVEEDOR: " & Proveedor & " </strong></h1></center></div>"
    cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse: collapse;\"><tr><th>Codigo Materia Prima</th><th>Nombre Materia
Prima</th></tr>"

    While MateriasPrimas.Available
        Codigo_MateriaPrima = MateriasPrimas["M.codigomateriaprima"]
        Nombre_MateriaPrima = MateriasPrimas["M.nombre"]
    
```

```

    cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo_MateriaPrima
& "</td><td>" & Nombre_MateriaPrima & "</td></tr>"
    MateriasPrimas.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
Weblistado.Url = RutaListado & ".html"

```

End

Private Sub GenerarInformeVendedores()

```

    Dim Sql As String
    Dim Cadena As String
    Dim Vendedor As Result
    ' Datos del Vendedor
    Dim Codigo_Vendedor As String
    Dim Nombre_Vendedor As String
    Dim Direccion_Vendedor As String
    Dim Nif_Vendedor As String
    Dim Localidad_Vendedor As String
    Dim Provincia_Vendedor As String
    NombreListado = "ListadoVendedores"
    RutaListado = RutaListado & "/ListadoVendedores"
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    sql = "select * from vendedor"
    Vendedor = FMain.Conexion_Base_Datos.Exec(Sql)
    cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html; charset=utf-8\"/></head><body>"
    cadena &= "<style> table td {text-align:center;} </style>"
    cadena &= "<div><img src=\"\" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\" width:900px;height:200px;\"> </div><br><br>"
    cadena &= "<div
style=\"width:900px;float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LIS
TADO VENDEDORES </strong></h1></center></div>"
    cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse:
collapse;\"><tr><th>Codigo</th><th>Nombre</th><th>Nif</th><th>Direccion</th>"
    cadena &= "<th>Localidad</th><th>Provincia</th></tr> "
    While Vendedor.Available
        Codigo_Vendedor = Vendedor["codigovendedor"]
        Nombre_Vendedor = Vendedor["nombre"]
        Direccion_Vendedor = Vendedor["direccion"]
        Nif_Vendedor = Vendedor["nif"]
        Localidad_Vendedor = Vendedor["localidad"]
        Provincia_Vendedor = Vendedor["provincia"]
        cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo_Vendedor &
"</td><td>" & Nombre_Vendedor & "</td><td>"
        cadena &= Nif_Vendedor & "</td><td>" & Direccion_Vendedor & "</td><td>" &
Localidad_Vendedor & "</td><td>" & Provincia_Vendedor & "</td></tr>"
        Vendedor.MoveNext
    Wend
    cadena &= " </table></div>"
    cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
    File.Save(RutaListado & ".html", cadena)
    Weblistado.Url = RutaListado & ".html"

```



```
Private Sub GenerarLote()

    Dim cadena As String
    Dim Sql As String
    Dim Linea_Detalles As Result
    Dim Aux As String
    ' Cabecera Lote
    Dim Lote As String
    Dim Fecha As String
    Dim Coste As String
    Dim Cantidad As String
    Dim Merma As String
    Dim Artículo_Codigo As String
    Dim Artículo_Nombre As String
    ' Línea Detalle Lote
    Dim Linea_MateriaPrima As String
    Dim Linea_Cantidad As String
    Dim Linea_Coste As String
    Dim Linea_Lotecompra As String
    Dim Linea_Proveedor As String
    Dim Linea_NfacturaCompra As String
    Dim Linea_FechaFacturaCompra As String


    NombreListado = "ListadoLote"
    RutaListado = RutaListado & "/ListadoLote"
    Lote = Frm_Informes.Lote_Generar
    Fecha = Utilidades.Obtener_Campo_Tabla("select fecha from lotes where codigolote='" & Lote & "'", "fecha")
    Coste = Utilidades.Obtener_Campo_Tabla("select coste from lotes where codigolote='" & Lote & "'", "coste")
    Cantidad = Utilidades.Obtener_Campo_Tabla("select cantidad from lotes where codigolote='" & Lote & "'", "cantidad")
    Merma = Utilidades.Obtener_Campo_Tabla("select merma from lotes where codigolote='" & Lote & "'", "merma")
    Artículo_Codigo = Utilidades.Obtener_Campo_Tabla("select articulo from lotes where codigolote='" & Lote & "'", "articulo")
    Artículo_Nombre = Utilidades.Obtener_Campo_Tabla("select nombre from articulo where codigoarticulo='" & Artículo_Codigo & "'", "nombre")
    cadena = cadena = "<html><head><meta http-equiv=\"Content-Type\" content=\"text/html;charset=utf-8\"/></head><body>"
    cadena &= "<style> table td {text-align:center;} </style>"
    cadena &= "<div><img src=\"" & FMain.CarpetaCarnea & "/logo/logo.png\" style=\"" width:900px;height:200px;\> </div><br><br>"
    cadena &= "<div style=\""border:1px solid black;width:300px,float:left;\><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LOTE <br>Lote Numero : </strong> " & Lote & "<br>"
    cadena &= "<strong>Fecha :</strong> " & fecha & " <br>"
    cadena &= "<strong>Coste :</strong> " & Coste & "<br>"
    cadena &= "<strong>Cantidad :</strong> " & Cantidad & "<br>"
    cadena &= "<strong>Merma :</strong> " & Merma & " <br>"
    cadena &= "<strong>Artículo :</strong> " & Artículo_Nombre & "<br>"


    cadena &= " <br></div><br><br><br><br><br>"
    cadena &= "<br><br><div style=\"" width:500px;clear:both;margin-top:.5em;\>"
    cadena &= " <table style=\""width:900px;border: 1px solid black;border-collapse: collapse;\><tr><th>Materia Prima</th><th>Cantidad</th><th>Precio</th><th>Proveedor</th>"
        cadena &= "<th>Lote Compra</th><th>Factura Compra</th><th>Fecha Compra</th></tr>"
```



```

sql = "select * from lotedetalle where codigolote='" & Lote & "'"
If Not FMain.Conexion_Base_Datos.Opened Then
    FMain.Conexion_Base_Datos.Open
Endif
Linea_Detalles = FMain.Conexion_Base_Datos.Exec(Sql)
Linea_Detalles.MoveFirst
While Linea_Detalles.Available
    aux = Linea_Detalles["materiaprima"]
    Linea_MateriaPrima = Utilidades.Obtener_Campo_Tabla("select nombre from
materiaprima where codigomateriaprima='" & aux & "'", "nombre")
    Linea_Cantidad = Linea_Detalles["cantidad"]
    Linea_Coste = Linea_Detalles["coste"]
    Linea_Lotecompra = Linea_Detalles["numerolotecompra"]
    aux = Linea_Detalles["proveedor"]
    Linea_Proveedor = Utilidades.Obtener_Campo_Tabla("select razonsocial from
proveedor where codigoproveedor='" & aux & "'", "razonsocial")
    Linea_NfacturaCompra = Linea_Detalles["nfacturacompra"]
    Linea_FechaFacturaCompra = Linea_Detalles["fechafacturacompra"]
    cadena &= " <tr style=\"border:1px solid black;\"><td>" & Linea_MateriaPrima
& "</td><td>" & Linea_Cantidad & "</td><td>" & Linea_Coste & "</td><td>"
    cadena &= Linea_Proveedor & "</td><td>" & Linea_Lotecompra & "</td><td>" &
Linea_NfacturaCompra & "</td><td>" & Linea_FechaFacturaCompra & "</td></tr>"
    Linea_Detalles.MoveNext
Wend

cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
Weblistado.Url = RutaListado & ".html"

End

Private Sub GenerarInformeTodasAverias()

    Dim Sql As String
    Dim Cadena As String
    Dim Averia As Result
    ' Datos de las Averias
    Dim Codigo_Averia As String
    Dim Fecha_Averia As String
    Dim Articulo_Averia As String
    Dim Cliente_Averia As String
    Dim Lote_Averia As String
    Dim Descripcion_Averia As String
    NombreListado = "ListadoTodasAverias"
    RutaListado = RutaListado & "/ListadoTodasAverias"
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripci
on from averia A" &
    " left join vendedor V on A.vendedor=V.codigovendedor left join clientes C
on A.cliente=C.codigocliente left join " &
    " articulo T on A.articulo=T.codigoarticulo"
    Averia = FMain.Conexion_Base_Datos.Exec(Sql)
    cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html;charset=utf-8\"/></head><body>"
    cadena &= "<style> table td {text-align:center;} </style>"

```

```
cadena &= "<div><img src=\"\" &FMain.CarpetaCarnea & "/logo/logo.png\"  
style=\" width:900px;height:200px;\">> </div><br><br>"  
cadena &= "<div  
style=\"width:900px,float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LIS  
TADO AVERIAS </strong></h1></center></div>"  
cadena &= " <table style=\"width:900px;border: 1px solid black;border-  
collapse:  
collapse;\">><tr><th>Codigo</th><th>Fecha</th><th>Articulo</th><th>Cliente</th>"  
cadena &= "<th>Lote</th><th>Descripcion</th></tr> "  
While averia.Available  
    Codigo_Averia = Averia["A.codigoaveria"]  
    Fecha_Averia = Averia["A.fecha"]  
    Articulo_Averia = Averia["T.nombre"]  
    Cliente_Averia = Averia["C.razonsocial"]  
    Lote_Averia = Averia["A.lote"]  
    Descripcion_Averia = Averia["A.descripcion"]  
    cadena &= " <tr style=\"border:1px solid black;\">><td>" & Codigo_Averia &  
"</td><td>" & Fecha_Averia & "</td><td>"  
    cadena &= Articulo_Averia & "</td><td>" & Cliente_Averia & "</td><td>" &  
Lote_Averia & "</td><td>" & Descripcion_Averia & "</td></tr>"  
    averaia.MoveNext  
Wend  
cadena &= " </table></div>"  
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"  
File.Save(RutaListado & ".html", cadena)  
Weblistado.Url = RutaListado & ".html"
```

End

```
Private Sub GenerarInformeClienteAverias()
```

```

Dim Sql As String
Dim Cadena As String
Dim Averia As Result
' Datos de las Averias
Dim Codigo_Averia As String
Dim Fecha_Averia As String
Dim Artículo_Averia As String
Dim Cliente_Averia As String
Dim Lote_Averia As String
Dim Descripcion_Averia As String
Dim Peso_Averia As String
NombreListado = "ListadoClienteAverias"
RutaListado = RutaListado & "/ListadoClienteAverias"
If Not FMain.Conexion_Base_Datos.Opened Then
    FMain.Conexion_Base_Datos.Open
Endif
Cliente_Averia = Frm_Informes.Cliente_Averia
sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripcion from averia A" &
" left join vendedor V on A.vendedor=V.codigovendedor left join clientes C on
A.cliente=C.codigocliente left join " &
" articulo T on A.articulo=T.codigoarticulo where C.razonsocial ='" &
Cliente_Averia & "'"
Averia = FMain.Conexion_Base_Datos.Exec(Sql)
cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html; charset=utf-8\"/></head><body>"
cadena &= "<style> table td {text-align:center;} </style>"

```

```
cadena &= "<div><img src=\"\" & Main.CarpetaCarnea & \"/logo/logo.png\"  
style=\" width:900px;height:200px;\">cadena &= "<div  
style=\"width:900px;float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;& LIS  
TADO AVERIAS DEL CLIENTE : " & Cliente_Averia & " </strong></h1></center></div>"  
cadena &= " <table style=\"width:900px;border: 1px solid black;border-  
collapse:  
collapse;\">cadena &= "<th>Lote</th><th>Descripcion</th></tr> "  
While averia.Available  
Codigo_Averia = Averia["A.codigoaveria"]  
Fecha_Averia = Averia["A.fecha"]  
Articulo_Averia = Averia["T.nombre"]  
Peso_Averia = Averia["A.peso"]  
Lote_Averia = Averia["A.lote"]  
Descripcion_Averia = Averia["A.descripcion"]  
cadena &= " <tr style=\"border:1px solid black;\">"</td><td>" & Fecha_Averia & "</td><td>"  
cadena &= Articulo_Averia & "</td><td>" & Peso_Averia & "</td><td>" &  
Lote_Averia & "</td><td>" & Descripcion_Averia & "</td></tr>"  
averia.MoveNext  
Wend  
cadena &= " </table></div>"  
cadena &= "<br><br><br><br><br><br><br><br><br><br><br><br><br><br></body></html>"  
File.Save(RutaListado & ".html", cadena)  
Weblistado.Url = RutaListado & ".html"
```

End

```
Private Sub GenerarInfomreArticuloAverias()
```

```

Dim Sql As String
Dim Cadena As String
Dim Averia As Result
' Datos de las Averias
Dim Codigo_Averia As String
Dim Fecha_Averia As String
Dim Artículo_Averia As String
Dim Cliente_Averia As String
Dim Lote_Averia As String
Dim Descripcion_Averia As String
Dim Peso_Averia As String
NombreListado = "ListadoArticuloAverias"
RutaListado = RutaListado & "/ListadoArticuloAverias"
If Not FMain.Conexion_Base_Datos.Opened Then
    FMain.Conexion_Base_Datos.Open
Endif
Artículo_Averia = Frm_Informes.Artículo_Averia
sql = "select
A.codigoaveria,A.fecha,V.nombre,C.razonsocial,T.nombre,A.peso,A.lote,A.descripcion
on from averia A" &
" left join vendedor V on A.vendedor=V.codigovendedor left join clientes C
on A.cliente=C.codigocliente left join " &
" articulo T on A.articulo=T.codigoarticulo where T.nombre='" &
Artículo_Averia & "'"
Averia = FMain.Conexion_Base_Datos.Exec(Sql)
cadena = "<html><head><meta http-equiv=\"Content-Type\"
content=\"text/html; charset=utf-8\"/></head><body>\"
cadena &= "<style> table td {text-align:center;} </style>\"

```

```
cadena &= "<div><img src=\"\" & FMain.CarpetaCarnea & "/logo/logo.png\"
style=\" width:900px;height:200px;\"> </div><br><br>"
cadena &= "<div
style=\"width:900px;float:left\"><center><h1><strong>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&LIS
TADO AVERIAS DEL ARTICULO : \" & Articulo_Averia & \"
</strong></h1></center></div>"
cadena &= " <table style=\"width:900px;border: 1px solid black;border-
collapse:
collapse;\"><tr><th>Codigo</th><th>Fecha</th><th>Cliente</th><th>Peso</th>"
cadena &= "<th>Lote</th><th>Descripcion</th></tr> "
While averia.Available
Codigo_Averia = Averia["A.codigoaveria"]
Fecha_Averia = Averia["A.fecha"]
Cliente_Averia = Averia["C.razonsocial"]
Peso_Averia = Averia["A.peso"]
Lote_Averia = Averia["A.lote"]
Descripcion_Averia = Averia["A.descripcion"]
cadena &= " <tr style=\"border:1px solid black;\"><td>" & Codigo_Averia &
"</td><td>" & Fecha_Averia & "</td><td>"
cadena &= Cliente_Averia & "</td><td>" & Peso_Averia & "</td><td>" &
Lote_Averia & "</td><td>" & Descripcion_Averia & "</td></tr>"
averia.MoveNext
Wend
cadena &= " </table></div>"
cadena &= "<br><br><br><br><br><br><br><br><br><br><br></body></html>"
File.Save(RutaListado & ".html", cadena)
Weblistado.Url = RutaListado & ".html"
```

End

2.2.6.5 Frm_Utilidades.class

```
Public Sub Form_Open()
```

```
Me.Title = "MODULO DE UTILIDADES"
' Configuramos IconPanel de Configuracion de base de datos
TxtHost.Text = "localhost"
TxtPuerto.text = "3306"
If Exist(fmain.DirConexion & "/conexion.conf") Then
' Message.Info(fMain.DirConexion)
CargarDatosConexion()
Endif
TxtNombreFormaPago.SetFocus
Configurar_Tipos_Iva()
Configurar_Cuentas_Contabilidad()
Configurar_Forma_Pago()
Configurar_Dispositivo_Movil()
```

End

```
Public Sub BtnSalirUtilidades_Click()
```

```
Me.Close
```

End

Capítulo 2. CÓDIGO FUENTE

```
''*****
''                               INICIO CODIGO  CONFIGURACION BASE DE DATOS
''*****
Private Sub CargarDatosConexion()

    TxtHost.text = FMain.FicheroConexionBD["ConexionBD/Host"]
    TxtPuerto.text = FMain.FicheroConexionBD["ConexionBD/Puerto"]
    TxtUsuario.text = FMain.FicheroConexionBD["ConexionBD/Usuario"]
    TxtPassword.Text = Utilidades.cripto(2,
FMain.FicheroConexionBD["ConexionBD/Password"])
    TxtBaseDatos.text = FMain.FicheroConexionBD["ConexionBD/BaseDatos"]

End

Public Sub BtnAceptar_Click()
    Dim Validar As Boolean = True

    If TxtHost.text = "" Then
        Balloon.Error("Debe introducir direccion IP, si es el mismo equipo
LOCALHOST", TxtHost)
        validar = False
    Else
        Balloon.Hide(TxtHost)
    Endif

    If TxtPuerto.text = "" Then
        Balloon.Error("Debe introducir puerto normalmente es 3306", TxtPuerto)
        validar = False
    Else
        Balloon.Hide(TxtPuerto)
    Endif

    If TxtUsuario.text = "" Then
        Balloon.Error("Debe introducir usuario de acceso a la base de datos",
TxtUsuario)
        validar = False
    Else
        Balloon.Hide(TxtUsuario)
    Endif

    If TxtPassword.text = "" Then
        Balloon.Error("Debe introducir contraseña de usuario", TxtPassword)
        validar = False
    Else
        Balloon.Hide(TxtPassword)
    Endif

    If TxtBaseDatos.text = "" Then
        Balloon.Error("Debe introducir nombre de la base de datos", TxtBaseDatos)
        validar = False
    Else
        Balloon.Hide(TxtBaseDatos)
    Endif

    If validar Then
        fmain.Conexion_Base_Datos.Host = TxtHost.Text
        fmain.Conexion_Base_Datos.Port = TxtPuerto.Text
        fmain.Conexion_Base_Datos.User = TxtUsuario.Text
        fmain.Conexion_Base_Datos.Password = TxtPassword.Text
    Endif
End Sub
```

```
fmain.Conexion_Base_Datos.Name = TxtBaseDatos.Text
Try fmain.Conexion_Base_Datos.Open
    fmain.Conexion_Base_Datos.Close
If Error Then
    Message.Error("Error en datos de conexion. Revise la configuracion.")
Else
    lblDatosActualizados.Visible = True
    fmain.FicheroConexionBD["ConexionBD/Host"] = TxtHost.Text
    fmain.FicheroConexionBD["ConexionBD/Puerto"] = TxtPuerto.text
    fmain.FicheroConexionBD["ConexionBD/Usuario"] = TxtUsuario.text
    fmain.FicheroConexionBD["ConexionBD/Password"] = Utilidades.cripto(1,
TxtPassword.text)
    fmain.FicheroConexionBD["ConexionBD/BaseDatos"] = TxtBaseDatos.text
    fmain.FicheroConexionBD.Save

    Endif
Endif

End

Public Sub TxtHost_KeyPress()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtPuerto.SetFocus
    Endif

End

Public Sub TxtPuerto_KeyPress()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtUsuario.SetFocus
    Endif

End

Public Sub TxtUsuario_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtPassword.SetFocus
    Endif

End

Public Sub TxtPassword_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtBaseDatos.SetFocus
    Endif

End

'*****
''
''          FIN      CODIGO  CONFIGURACION BASE DE DATOS
''*****
```

Capítulo 2. CÓDIGO FUENTE

```
''*****
''                               INICIO CODIGO  CONFIGURACION FORMA DE PAGO
''*****

Private Sub Configurar_Forma_Pago()
    Dim Cabeceras As New String[]
    Dim Sql As String

    Cabeceras.Add("Nombre")
    Cabeceras.Add("Dias")
    Cabeceras.Add("Porcentaje")

    DtsFormaPago.Connection = FMain.Conexion_Base_Datos
    sql = "select * from formapago order by nombreformapago"
    DtsFormaPago.Table = sql
    DtbFormaPago.Labels = Cabeceras
    DtbFormaPago.view.Columns[0].Width = 200
    DtbFormaPago.view.Columns[1].Width = 100
    DtbFormaPago.view.Columns[2].Width = 100

End

Public Sub BtnAgregarFormaPago_Click()
    Dim Sql As String

    If Comprobar_Datos_Forma_Pago("AGREGAR") Then
        Sql = "insert into formapago (nombreformapago,dias,porcentaje) values ('" &
        UCase(TxtNombreFormaPago.Text) & "'," & Val(VlBDias.Text) & "," &
        Val(vLBPorcentaje.Text) & ")"
        Utilidades.In_Act_Bor(Sql)
        TxtNombreFormaPago.Text = ""
        VlBDias.text = ""
        vLBPorcentaje.text = ""
        Configurar_Forma_Pago()

    Endif

End

'' Funcion que comprueba que los datos introducidos son correctos de las Formas
de Pago.<br>
'' Parametros:<br>
'' Accion --> Accion que vamos a realizar puede tomar los siguientes
Valores:<br>
'' AGREGAR --> Cuando estamos agregando Forma de Pago.<br>
'' Devuelve:<br>
'' TRUE --> Datos son correctos<br>
'' FALSE --> Datos erroneos.
Private Function Comprobar_Datos_Forma_Pago(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True

    If TxtNombreFormaPago.text = "" Then
        Balloon.Error("Debe introducir un Nombre para la Forma de Pago.",
        TxtNombreFormaPago)
        Validar = False
        TxtNombreFormaPago.SetFocus
    Endif
End Function
```

```
If Accion = "AGREGAR" Then
    sql = "select * from formapago where nombreformapago='" &
UCase(TxtNombreFormaPago.text) & "'"
    If Utilidades.Comprobar_existe_dato(Sql) Then
        Balloon.Error("La Forma de Pago ya existe", TxtNombreFormaPago)
        Validar = False
        TxtNombreFormaPago.SetFocus
    Endif
Endif
Return Validar
End

Public Sub BtnBorrarFormaPago_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbFormaPago.View.Row
    If DtbFormaPago.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar la Forma de Pago : " &
DtbFormaPago.View[Numero_fila, 0].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from formapago where nombreformapago='" &
DtbFormaPago.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_Forma_Pago()
        Endif
    Endif

End

Public Sub BtnModificarFormaPago_Click()

    Dim Sql As String
    Dim Numero_fila As Integer

    Numero_fila = DtbFormaPago.View.Row
    If LblModificarFormaPago.Text = "EDITAR" Then
        LblModificarFormaPago.Text = "ACTUALIZAR"
        TxtNombreFormaPago.Text = DtbFormaPago.View[Numero_fila, 0].Text
        VlBDias.Text = DtbFormaPago.View[Numero_fila, 1].Text
        vLBPorcentaje.Text = DtbFormaPago.View[Numero_fila, 2].Text
        TxtNombreFormaPago.Enabled = False
        BtnAgregarFormaPago.Enabled = False
        BtnBorrarFormaPago.Enabled = False
        VlBDias.SetFocus
    Else
        sql = "update formapago set dias=" & Val(VlBDias.Text) & ",porcentaje=" &
Val(vLBPorcentaje.text) & " where nombreformapago='" &
UCase(TxtNombreFormaPago.Text) & "'"
        Utilidades.In_Act_Bor(sql)
        LblModificarFormaPago.Text = "EDITAR"
        TxtNombreFormaPago.Text = ""
        VlBDias.Text = ""
        vLBPorcentaje.Text = ""
        TxtNombreFormaPago.Enabled = True
        BtnAgregarFormaPago.Enabled = True
    End If
End Sub
```


Capítulo 2. CÓDIGO FUENTE

```
        BtnBorrarFormaPago.Enabled = True
        TxtNombreFormaPago.SetFocus
        Configurar_Forma_Pago()
    Endif

End

Public Sub TxtNombreFormaPago_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlBDias.SetFocus
    Endif

End

Public Sub VlBDias_KeyPress()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        vLBPorcentaje.SetFocus
    Endif

End

'*****
'                               FIN CODIGO  CONFIGURACION FORMA DE PAGO
'*****

'*****
'                               INICIO CODIGO  CONFIGURACION DISPOSITIVO MOVIL
'*****

Private Sub Configurar_Dispositivo_Movil()
    Dim Cabeceras As New String[]
    Dim Sql As String

    Cabeceras.Add("Codigo")
    Cabeceras.Add("Nombre")
    Cabeceras.Add("Especificaciones")

    DtsDispositivoMovil.Connection = FMain.Conexion_Base_Datos
    sql = "select * from dispositivomovil order by codigodispositivomovil"
    DtsDispositivoMovil.Table = sql
    DtbDispositivoMovil.Labels = Cabeceras
    DtbDispositivoMovil.view.Columns[0].Width = 100
    DtbDispositivoMovil.view.Columns[1].Width = 200
    DtbDispositivoMovil.view.Columns[2].Width = 500

End

Public Sub BtnAgregarDM_Click()
    Dim Sql As String

    If Comprobar_Datos_Dispositivo_Movil("AGREGAR") Then
        Sql = "insert into dispositivomovil
(codigodispositivomovil,nombre,especificaciones) values ('" &
UCase(TxtCodigoDM.Text) & "',''" & UCase(TxtNombreDM.Text) & "',''" &
TxtEspecificacionesDM.Text & "')"
        Utilidades.In_Act_Bor(Sql)
```

```
TxtCodigoDM.Text = ""
TxtNombreDM.Text = ""
TxtEspecificacionesDM.Text = ""
Configurar_Dispositivo_Movil()
Endif
End

Public Sub BtnBorrarDM_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbDispositivoMovil.View.Row
    If DtbDispositivoMovil.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar el Dispositivo Movil (Si lo borra
se eliminara del vendedor asociado) : " & DtbDispositivoMovil.View[Numero_fila,
0].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from dispositivomovil where codigodispositivomovil='" &
DtbDispositivoMovil.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            sql = "delete from dispositivomovilvendedor where dispositivomovil='" &
DtbDispositivoMovil.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_Dispositivo_Movil()
        Endif
    Endif

End

Public Sub BtnModificarDM_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Numero_fila = DtbDispositivoMovil.View.Row
    If LblModificarDM.Text = "EDITAR" Then
        LblModificarDM.Text = "ACTUALIZAR"
        TxtCodigoDM.Text = DtbDispositivoMovil.View[Numero_fila, 0].Text
        TxtNombreDM.Text = DtbDispositivoMovil.View[Numero_fila, 1].Text
        TxtEspecificacionesDM.Text = DtbDispositivoMovil.View[Numero_fila, 2].Text
        TxtCodigoDM.Enabled = False
        BtnAgregarDM.Enabled = False
        BtnBorrardm.Enabled = False
        TxtNombreDM.SetFocus
    Else
        If Comprobar_Datos_Dispositivo_Movil("MODIFICAR") Then
            sql = "update dispositivomovil set nombre='" & UCase(TxtNombreDM.Text) &
"',especificaciones='" & UCase(TxtEspecificacionesDM.Text) & "'" where
codigodispositivomovil='" & UCase(TxtCodigoDM.Text) & "'"
            Utilidades.In_Act_Bor(sql)
            LblModificarFormaPago.Text = "EDITAR"
            TxtCodigoDM.Text = ""
            TxtNombreDM.Text = ""
            TxtEspecificacionesDM.Text = ""
            TxtCodigoDM.Enabled = True
            BtnAgregarDM.Enabled = True
            BtnBorrarDM.Enabled = True
            TxtCodigoDM.SetFocus
            Configurar_Dispositivo_Movil()
        End If
    End If

End
```

```
        Endif
    Endif

End

'' Funcion que comprueba que los datos introducidos son correctos de los
Dispositivos Moviles.<br>
'' Parametros:<br>
'' Accion --> Accion que vamos a realizar puede tomar los siguientes
Valores:<br>
'' AGREGAR --> Cuando estamos agregando Forma de Pago.<br>
'' Devuelve:<br>
'' TRUE --> Datos son correctos<br>
'' FALSE --> Datos erroneos.
Private Function Comprobar_Datos_Dispositivo_Movil(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True

    If TxtCodigoDM.text = "" Then
        Balloon.Error("Debe introducir un Codigo para el Dispositivo Movil.",
TxtCodigoDM)
        Validar = False
        TxtCodigoDM.SetFocus
    Else If TxtNombreDM.text = ""
        Balloon.Error("Debe introducir un Nombre para el Dispositivo Movil.",
TxtNombreDM)
        Validar = False
        TxtNombreDM.SetFocus
    Endif

    If Accion = "AGREGAR" Then
        sql = "select * from dispositivomovil where codigodispositivomovil='" &
UCase(TxtCodigoDM.text) & "'"
        If Utilidades.Comprobar_existe_dato(Sql) Then
            Balloon.Error("Dispositivo Movil ya existe", TxtCodigoDM)
            Validar = False
            TxtCodigoDM.SetFocus
        Endif
    Endif
    Return Validar
End
Public Sub TxtCodigoDM_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtNombreDM.SetFocus
    Endif

End

Public Sub TxtNombreDM_KeyRelease()
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtEspecificacionesDM.SetFocus
    Endif
End
```

Capítulo 2. CÓDIGO FUENTE

```
'*****
''                               INIVIO CODIGO  CONFIGURACION DISPOSITIVO MOVIL
'*****

'*****
''                               INICIO CODIGO  CONFIGURACION CUENTAS CONTABLIDIDAD
'*****

Public Sub BtnActualizarCuentasContabilidad_Click()
    Dim Sql As String
    If Comprobar_Datos_Cuentas() Then
        sql = "update cuentascontabilidad set venta='" & TxtVenta.Text & "',iva='" &
Txtiva.text & "',ivare='" & Txtivare.Text & "',re='" & TxtRecargoE.text &
"' ,cobro='" & TxtCobro.Text & "' where id='1'"
        Utilidades.In_Act_Bor(sql)
        LblInfo.Visible = True
    Endif
End

'' Funcion que  comprueba que los datos son correctos antes de actualizar los
datos.
Private Function Comprobar_Datos_Cuentas() As Boolean

    Dim Validar As Boolean
    validar = True
    If TxtVenta.Text = "" Then
        Balloon.Error("Debe introducir un Numero de Cuenta de Venta.", TxtVenta)
        Validar = False
        TxtVenta.SetFocus
    Else If TxtVenta.Length < 7 Or (Not IsNumber(TxtVenta.Text))
        Balloon.Error("Formato erroneo, debe ser un dato numerico y de 7
caracteres.", TxtVenta)
        Validar = False
        TxtVenta.SetFocus
    Else If Txtiva.Text = "" Then
        Balloon.Error("Debe introducir un Numero de Cuenta de Iva.", Txtiva)
        Validar = False
        Txtiva.SetFocus
    Else If Txtiva.Length < 7 Or (Not IsNumber(Txtiva.Text))
        Balloon.Error("Formato erroneo, debe ser un dato numerico y de 7
caracteres.", Txtiva)
        Validar = False
        Txtiva.SetFocus
    Else If Txtivare.Text = "" Then
        Balloon.Error("Debe introducir un Numero de Cuenta de Iva Repercutido.",
Txtivare)
        Validar = False
        Txtivare.SetFocus
    Else If Txtivare.Length < 7 Or (Not IsNumber(Txtivare.Text))
        Balloon.Error("Formato erroneo, debe ser un dato numerico y de 7
caracteres.", Txtivare)
        Validar = False
        Txtivare.SetFocus
    Else If TxtRecargoE.Text = "" Then
        Balloon.Error("Debe introducir un Numero de Cuenta de Iva Recargo
Equivalente.", TxtRecargoE)
```

```
        Validar = False
        TxtRecargoE.SetFocus
    Else If TxtRecargoE.Length < 7 Or (Not IsNumber(TxtRecargoE.Text))
        Balloon.Error("Formato erroneo, debe ser un dato numerico y de 7
caracteres.", TxtRecargoE)
        Validar = False
        TxtRecargoE.SetFocus
    Else If TxtCobro.Text = "" Then
        Balloon.Error("Debe introducir un Numero de Cuenta de Iva Repercutido.",
TxtCobro)
        Validar = False
        TxtCobro.SetFocus
    Else If TxtCobro.Length < 7 Or (Not IsNumber(TxtCobro.Text))
        Balloon.Error("Formato erroneo, debe ser un dato numerico y de 7
caracteres.", TxtCobro)
        Validar = False
        Txtcobro.SetFocus
    Endif
    Return validar
End

Private Sub Configurar_Cuentas_Contabilidad()
    TxtVenta.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "venta")
    TxtIva.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "iva")
    TxtIvaRe.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "ivare")
    TxtRecargoE.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "re")
    TxtCobro.Text = Utilidades.Obtener_Campo_Tabla("select * from
cuentascontabilidad", "cobro")
End

Public Sub TxtVenta_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtIva.SetFocus
    Endif

End

Public Sub TxtIva_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtIvaRe.SetFocus
    Endif

End

Public Sub TxtIvaRe_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtRecargoE.SetFocus
    Endif

End
```

Capítulo 2. CÓDIGO FUENTE

```
End

Public Sub TxtRecargoE_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        TxtCobro.SetFocus
    Endif

End

'''*****
'''                               FIN CODIGO  CONFIGURACION CUENTAS CONTABILIDAD
'''*****

'''*****
'''                               INICIO CODIGO  CONFIGURACION TIPOS DE IVA
'''*****

Public Sub BtnAgregarTipoIva_Click()
    Dim Sql As String
    If Comprobar_Datos_Tipo_Iva("AGREGAR") Then
        Sql = "insert into tipoiva (nombretipoiva,iva,recargo) values ('" &
        UCase(TxtNombreTipoIva.Text) & "'," & Val(VlBTIiva.Text) & "," &
        Val(vLbTIRecargo.Text) & ")"
        Utilidades.In_Act_Bor(Sql)
        TxtNombreTipoIva.Text = ""
        VlBTIiva.text = ""
        vLbTIRecargo.text = ""
        Configurar_Tipos_Iva()
    Endif
End

Public Sub BtnBorrarTipoIva_Click()

    Dim Sql As String
    Dim Numero_fila As Integer
    Dim Respuesta As Integer
    Numero_fila = DtbTipoIva.View.Row
    If DtbTipoIva.View.Rows.Count > 0 Then
        Respuesta = Message.Delete("Desea Borrar la Forma de Pago : " &
        DtbTipoIva.View[Numero_fila, 0].Text, "Si", "No")
        If respuesta = 1 Then
            sql = "delete from tipoiva where nombretipoiva='" &
            DtbTipoIva.View[Numero_fila, 0].Text & "'"
            Utilidades.In_Act_Bor(Sql)
            Configurar_Tipos_Iva()
        Endif
    Endif
End

Public Sub BtnModificarTipoIva_Click()

    Dim Sql As String
    Dim Numero_fila As Integer

    Numero_fila = DtbTipoIva.View.Row
```

```
If LblModificarTipoIva.Text = "EDITAR" Then
    LblModificarTipoIva.Text = "ACTUALIZAR"
    TxtNombreTipoIva.Text = DtbTipoIva.View[Numero_fila, 0].Text
    VlBTIiva.Text = DtbTipoIva.View[Numero_fila, 1].Text
    vLbTIRecargo.Text = DtbTipoIva.View[Numero_fila, 2].Text
    TxtNombreTipoIva.Enabled = False
    BtnAgregarTipoIva.Enabled = False
    BtnBorrarTipoIva.Enabled = False
    VlBTIiva.SetFocus
Else
    sql = "update tipoiva set iva=" & Val(VlBTIiva.Text) & ",recargo=" &
Val(vLbTIRecargo.text) & " where nombretipoiva='" & UCase(TxtNombreTipoIva.Text)
& "'"
    Utilidades.In_Act_Bor(sql)
    LblModificarTipoIva.Text = "EDITAR"
    TxtNombreTipoIva.Text = ""
    VlBTIiva.Text = ""
    vLbTIRecargo.Text = ""
    TxtNombreTipoIva.Enabled = True
    BtnAgregarTipoIva.Enabled = True
    BtnBorrarTipoIva.Enabled = True
    TxtNombreTipoIva.SetFocus
    Configurar_Tipos_Iva()
Endif

End

Private Sub Configurar_Tipos_Iva()

    Dim Cabeceras As New String[]
    Dim Sql As String

    Cabeceras.Add("Nombre")
    Cabeceras.Add("Iva")
    Cabeceras.Add("Recargo")
    DtsTipoIva.Connection = FMain.Conexion_Base_Datos
    sql = "select * from tipoiva"
    DtsTipoIva.Table = sql
    DtbTipoIva.Labels = Cabeceras
    DtbTipoIva.view.Columns[0].Width = 200
    DtbTipoIva.view.Columns[1].Width = 100
    DtbTipoIva.view.Columns[2].Width = 100

End

'' Funcion que comprueba que los datos introducidos son correctos de las Tipos
de Iva.<br>
'' Parametros:<br>
'' Accion --> Accion que vamos a realizar puede tomar los siguientes
Valores:<br>
'' AGREGAR --> Cuando estamos agregando Forma de Pago.<br>
'' Devuelve:<br>
'' TRUE --> Datos son correctos<br>
'' FALSE --> Datos erroneos.
Private Function Comprobar_Datos_Tipo_Iva(Accion As String) As Boolean
    Dim Validar As Boolean
    Dim Sql As String
    Validar = True
```

```
If TxtNombreTipoIva.text = "" Then
    Balloon.Error("Debe introducir un Nombre para el Tipo de Iva.",
TxtNombreTipoIva)
    Validar = False
    TxtNombreTipoIva.SetFocus
Else If VlBTIiva.Text = 0
    Balloon.Error("Debe introducir un valor para el Iva no puede ser 0.",
VlBTIiva)
    Validar = False
    VlBTIiva.SetFocus
Else If vLbTIRecargo.Text = 0
    Balloon.Error("Debe introducir un valor para el Recargo no puede ser 0.",
vLbTIRecargo)
    Validar = False
    vLbTIRecargo.SetFocus
Endif

If Accion = "AGREGAR" Then
    sql = "select * from tipoiva where nombretipoiva='" &
UCase(TxtNombreTipoIva.text) & "'"
    If Utilidades.Comprobar_existe_dato(Sql) Then
        Balloon.Error("El nombre de Tipo de Iva ya existe", TxtNombreTipoIva)
        Validar = False
        TxtNombreTipoIva.SetFocus
    Endif
Endif
Return Validar
End
Public Sub TxtNombreTipoIva_KeyRelease()
    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        VlBTIiva.SetFocus
    Endif
End

Public Sub VlBTIiva_KeyRelease()

    If Key.Code = Key.Return Or Key.Code = Key.Enter Then
        vLbTIRecargo.SetFocus
    Endif
End
```

```
' '*****
' '
' '                FIN CODIGO  CONFIGURACION TIPOS DE IVA
' '*****
```


2.2.7 Utilidades.

Funciones genéricas utilizadas a lo largo del desarrollo de la aplicación. Como pueden ser las funciones de acceso a la base de datos, llenar combobox, datagrid etc..

Cada función contiene documentación de los parámetros que recibe y los valores que devuelve si devuelve algún valor.

```
' ' Función para acciones de Introduccion, Borrado y Actualizacion de los datos
en la base de datos.<br>
' ' Parametros:<br>
' ' Sql --> donde introducimos la sentencia sql q ejecutar.
```

```
Public Sub In_Act_Bor(Sql As String)
```

```
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    FMain.Conexion_Base_Datos.Exec(Sql)
    'FMain.Conexion_Base_Datos.Close
End
```

```
' ' Funcion para comprobar que un dato existe en la base de datos.<br>
' ' Parametros:<br>
' '   Sql --> Sentencia select para comprobar la existencia
dehttps://github.com/belmotek/gambas-enhanced-controls1 dato.<br>
' ' Devuelve:<br>
' '   True --> Si existe el valor buscado<br>
' '   False --> Si no existe el valor buscado<br>
```

```
Public Function Comprobar_existe_dato(Sql As String) As Boolean
```

```
    Dim Resultado As Result
    Dim Validar As Boolean
    Validar = False
```

```
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
```

```
    Resultado = FMain.Conexion_Base_Datos.Exec(Sql)
    If Resultado.Count > 0 Then
        Validar = True
    Endif
```

```
    Return Validar
    'FMain.Conexion_Base_Datos.Close
End
```

```
' ' Procedimiento para rellenar con datos un combo box.<br>
' ' Parametros:<br>
' '   Sql --> donde introducimos la sentencia sql q ejecutar.<br>
' '   Combobox_Trabajo --> Combobox a rellenar.<br>
' '   Campo --> Campo con que rellenar el Combobox.
```

```
Public Sub Llenar_Combobox(Sql As String, ByRef Combobox_Trabajo As ComboBox,
Campo As String)
    Dim Result_Trabajo As Result

    Combobox_Trabajo.Clear
    'FMain.Conexion_Base_Datos.Close
    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    'Combobox_Trabajo.Add("")
    While Result_Trabajo.Available
        Combobox_Trabajo.Add(Result_Trabajo[Campo])
        Result_Trabajo.MoveNext
    Wend

    'Combobox_Trabajo.text = ""
    'FMain.Conexion_Base_Datos.Close

End

'' Procedimiento para rellenar con datos un combo box.<br>
'' Parametros:<br>
'' Sql --> donde introducimos la sentencia sql q ejecutar.<br>
'' Listobox_Trabajo --> Listbox a rellenar.<br>
'' Campo --> Campo con que rellenar el Combobox.

Public Sub Llenar_Listbox(Sql As String, ByRef Listbox_Trabajo As ListBox, Campo
As String)
    Dim Result_Trabajo As Result

    Listbox_Trabajo.Clear

    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif

    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveFirst
    While Result_Trabajo.Available
        Listbox_Trabajo.Add(Result_Trabajo[Campo])
        Result_Trabajo.MoveNext
    Wend

    'FMain.Conexion_Base_Datos.Close

End

'' Funcion para otener un dato de una tabla, devuelve el dato en un String.<br>
'' Parametros:<br>
'' Sql --> Sentencia select para comprobar la existencia del dato.<br>
'' Campo --> Campo para extraer su valor.<br>
'' Devuelve:<br>
'' El valor del campo<br>
```

```
''  
Public Function Obtener_Campo_Tabla(sql As String, Campo As String) As String  
    Dim Valor_A_Devolver As String  
    Dim Result_Trabajo As Result  
  
    If Not FMain.Conexion_Base_Datos.Opened Then  
        FMain.Conexion_Base_Datos.Open  
    Endif  
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)  
    Result_Trabajo.MoveFirst  
    Valor_A_Devolver = Result_Trabajo[Campo]  
  
    'FMain.Conexion_Base_Datos.Close  
    Return Valor_A_Devolver  
End  
  
'' Funcion para comprobar si una tabla tiene datos o esta vacia. Para ello vamos  
a devolver el numero de elementos.< br >  
'' Parametros:<br>  
''  Sql --> Sentencia select para comprobar la existencia de datos en la  
tabla.<br>  
'' Devuelve:<br>  
''  El numero de elementos.  
  
Public Function Tabla_Vacia(Sql As String) As Integer  
    Dim N_filas As Integer  
    Dim Result_Trabajo As Result  
    N_filas = 0  
  
    If Not FMain.Conexion_Base_Datos.Opened Then  
        FMain.Conexion_Base_Datos.Open  
    Endif  
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)  
    N_filas = Result_Trabajo.Count  
    Return N_filas  
    'FMain.Conexion_Base_Datos.Close  
End  
  
'' Procedimiento para eliminar un elemento de un combobox.<br>  
'' Parametros:<br>  
''  Combobox_Trabajo --> El combobox donde eliminar el elemento.<br>  
''  Valor_Elimiar --> Elemento a eliminar.  
Public Sub Eliminar_Elemento_ComboBox(ByRef Combobox_Trabajo As ComboBox,  
Valor_Eliminar As String)  
    Dim i As Integer  
    Dim Lista_Auxiliar As New String[]  
    Dim Elementos_Combobox As Integer  
    Dim Elementos_Lista As Integer  
    Elementos_Combobox = Combobox_Trabajo.Count  
    For i = 0 To Elementos_Combobox - 1  
        If Combobox_Trabajo[i].Text = Valor_Eliminar Then  
            Else  
                Lista_Auxiliar.Add(Combobox_Trabajo[i].Text)  
            Endif  
        Next  
        Combobox_Trabajo.Clear  
        Elementos_Lista = Lista_Auxiliar.Count
```

Capítulo 2. CÓDIGO FUENTE

```
For i = 0 To Elementos_Lista - 1
    Combobox_Trabajo.Add(Lista_Auxiliar[i])
Next
End

'' Funcion que nos devuelve el valor maximo de un campo en una tabla.<br>
'' Parametros:<br>
''   Sql --> Sentencia select que ordena la tabla por el un campo.<br>
''   Campo --> Campo que se desea el valor maximo.<br>
'' Devuelve:<br>
''   El valor maximo en formato String.
Public Function Valor_Maximo(Sql As String, Campo As String) As String
    Dim Valor_A_Devolver As String
    Dim Result_Trabajo As Result

    If Not FMain.Conexion_Base_Datos.Opened Then
        FMain.Conexion_Base_Datos.Open
    Endif
    Result_Trabajo = FMain.Conexion_Base_Datos.Exec(Sql)
    Result_Trabajo.MoveLast
    Valor_A_Devolver = Result_Trabajo[Campo]
    FMain.Conexion_Base_Datos.Close

    Return Valor_A_Devolver
End

'' Funcion para encriptar cadenas de caracteres:<br>
'' Parametros:<br>
''   modo --> Accion a realizar<br>
''           1 Encripta.<br>
''           2 Desencripta.<br>
''   skey --> Cadena a encriptar<br>
'' Devuelve : La cadena encriptado o desencriptada.
Public Function cripto(modo As Integer, skey As String) As String

    Dim cad, cad2 As String
    Dim $n, $i As Integer
    Dim ar As Variant[]
    Dim sus As String[]

    ar = [{".", "w"}, {"$", "s"}, {">", "S"}, {"*", "f"}, [{"", "A"}, {"-", "m"}, {"", "X"}], [{"<", "f"}, {";", "0"}] 'sustituimos algunos caracteres por otros al azar...¿?
    If modo = 1
        cad = Base64(skey)
        For $n = 0 To 3
            cad2 = ""
            For $i = Len(cad) To 1 Step -1
                cad2 &= Mid(cad, $i, 1)
            Next
            cad = Base64(cad2)
        Next
        For Each sus In ar
            cad = Replace(cad, sus[1], sus[0])
        Next
    Else If modo = 2
        For Each sus In ar
            skey = Replace(skey, sus[0], sus[1])
        Next
    End If
End Function
```

```
        Next
        cad = UnBase64(skey)
        For $n = 0 To 3
            cad2 = ""
            For $i = Len(cad) To 1 Step -1
                cad2 &= Mid(cad, $i, 1)
            Next
            cad = UnBase64(cad2)
        Next
    Endif
    Return cad
End

'' Funcion para poner una fecha en formato DD/MM/AAAA <br>
'' Parametros:<br>
'' Fecha_Trabajo --> Fecha que devuelve un Datebox. La fecha la devuelve
MM/DD/AAAA.<br>
'' Devuelve:<br>
'' Fecha en formato correcto DD/MM/AAAA

Public Function Fecha_Correcta(Fecha_Trabajo As String) As String
    Dim Fecha_Devuelta As String
    Dim Dia As String
    Dim Mes As String
    Dim Ano As String
    Dia = Mid(Fecha_Trabajo, 4, 2)
    Mes = Mid(Fecha_Trabajo, 1, 2)
    Ano = Mid(Fecha_Trabajo, 7, 4)
    Fecha_Devuelta = Dia & "/" & Mes & "/" & Ano
    Return Fecha_Devuelta
End

'' Funcion para calcular el numero de factura o albaran
'' Parametros:<br>
'' Tipo --> Tomara los siguientes valores<br>
'' A -- Albaran.<br>
'' F -- Factura.<br>
'' Devuelve: <br>
'' Numero De factura o albaran siguiente.

Public Function Calculo_Numero_FA(Tipo As String) As String

    Dim Ano As String
    Dim Numero As String
    Dim Aux As Integer
    Dim Nummero_Devuelto As String
    ano = Format(Now, "yy")

    If tipo = "F" Then
        If Utilidades.Tabla_Vacia("select * from facturas") > 0 Then
            numero = Utilidades.Valor_Maximo("select * from facturas order by 1",
"numerofactura")
            aux = CInt(Mid(numero, 2, 9))
            Nummero_Devuelto = "F" & CInt(aux + 1)
        Else
            Nummero_Devuelto = "F" & ano & "0000001"
        Endif
    Else
```

Capítulo 2. CÓDIGO FUENTE

```
If Utilidades.Tabla_Vacia("select * from albaranes") > 0 Then
    numero = Utilidades.Valor_Maximo("select * from albaranes order by 1",
"numeroalbaran")
    aux = CInt(Mid(numero, 2, 9))
    Nummero_Devuelto = "A" & CInt(aux + 1)
Else
    Nummero_Devuelto = "A" & ano & "0000001"
Endif
Endif
Return Nummero_Devuelto
End

'' Funcion que nos genera las facturas/albaranes para impresion<br>
'' Parametros:<br>
''     NumeroFactura --> Factura albaran que queremos imprimir.<br>
Public Sub Generar_Factura_Albaran(NumeroFactura As String)

    Dim Tipo As String
    Dim cadena As String
    Dim Sql As String
    Dim Sql_Cabecera As String
    Dim Sql_Linea As String
    Dim Sql_Familia As String
    Dim Cabecera As Result
    Dim Lineas As Result
    Dim Familias As Result
    ' Variables Cabecera.
    Dim Domicilio As String
    Dim Poblacion As String
    Dim Provincia As String
    DimCodigoPostal As String
    Dim RazonSocial As String
    Dim Fecha As String
    Dim Formapago As String
    Dim Nif As String
    Dim BaseImponible As String
    Dim DescuentoPP As String
    Dim DescuentoESP As String
    Dim ImporteDPP As String
    Dim ImporteDESP As String
    Dim Basemenosdescuento As Float
    Dim Iva As String
    Dim RE As String
    Dim ImporteIva As String
    Dim ImporteRE As String
    Dim Total As Float
    ' Variables lineas detalle
    Dim Articulo As String
    Dim Cantidad As String
    Dim Precio As String
    Dim Importe As String
    Dim Lote As String
    Dim Articulo_Familia As String
    Dim Recargo As String
    Dim Ruta_Generar_Factura_Albaran As String
    Tipo = Mid(NumeroFactura, 1, 1)
    If Tipo = "F" Then
```

```

    Sql_Cabecera = "select
F.numerofactura,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.cobrada,F.formapago,F.vencimiento,F.con
tabilizada,F.total,C.razonsocial,V.nombre from facturas F left join clientes C
on" &
    " F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor where F.numerofactura='" & NumeroFactura & "'"
    Sql_Linea = "select
L.numerofactura,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
facturasdetalles L left join articulo A on L.articulo=A.codigoarticulo" &
    " where L.numerofactura='" & NumeroFactura & "'"
    Else
    Sql_Cabecera = "select
F.numeroalbaran,F.fecha,F.hora,F.base,F.iva,F.recargo,F.importeiva,F.importereca
rgo,F.fechacobro,F.descuentopp,F.importedescuentopp" &

",F.descuentoesp,F.importedescuentoesp,F.pagado,F.formapago,F.vencimiento,F.fact
urado,F.total,C.razonsocial,V.nombre from albaranes F left join clientes C on" &
    " F.cliente=C.codigocliente left join vendedor V on
F.vendedor=V.codigovendedor where F.numeroalbaran='" & NumeroFactura & "'"
    Sql_Linea = "select
L.numeroalbaran,L.linea,L.precio,L.peso,L.importe,L.lote,A.nombre from
albaranesdetalle L left join articulo A on L.articulo=A.codigoarticulo" &
    " where L.numeroalbaran='" & NumeroFactura & "'"
    Endif

FMain.Conexion_Base_Datos.Close
' If Not FMain.Conexion_Base_Datos.Opened Then
FMain.Conexion_Base_Datos.Open
'Endif

Cabecera = FMain.Conexion_Base_Datos.Exec(Sql_Cabecera)
BaseImponible = Cabecera["F.base"]
DescuentoPP = Cabecera["F.descuentopp"]
DescuentoESP = Cabecera["F.descuentoesp"]
ImporteDPP = Cabecera["F.importedescuentopp"]
ImporteDESP = Cabecera["F.importedescuentoesp"]
Basemenosdescuento = CFloat(BaseImponible) - CFloat(ImporteDPP) -
CFloat(ImporteDESP)
Iva = Cabecera["F.iva"]
RE = Cabecera["F.recargo"]
ImporteIva = Cabecera["F.importeiva"]
ImporteRE = Cabecera["F.importerecarga"]
Total = Cabecera["F.total"]
RazonSocial = Cabecera["C.razonsocial"]
fecha = Cabecera["F.fecha"]
Formapago = Cabecera["F.formapago"]
sql = "select direccion from clientes where razonsocial='" & RazonSocial & "'"
Domicilio = Utilidades.Obtener_Campo_Tabla(sql, "direccion")
sql = "select localidad from clientes where razonsocial='" & RazonSocial & "'"
Poblacion = Utilidades.Obtener_Campo_Tabla(sql, "localidad")
sql = "select provincia from clientes where razonsocial='" & RazonSocial & "'"
provincia = Utilidades.Obtener_Campo_Tabla(sql, "provincia")
sql = "select codigopostal from clientes where razonsocial='" & RazonSocial &
""
CodigoPostal = Utilidades.Obtener_Campo_Tabla(sql, "codigopostal")
sql = "select nif from clientes where razonsocial='" & RazonSocial & "'"

```

240

241