

REPOSITORIO ACADÉMICO DIGITAL INSTITUCIONAL

Sistema de administración empresarial para comercializadoras de calzado

Autor: Bismarck Ivanovitch Méndez Heredia

**Tesis presentada para obtener el título de:
Lic. En Sistemas Computarizados [sic.]**

**Nombre del asesor:
Víctor Manuel Pineda Almanza**

Este documento está disponible para su consulta en el Repositorio Académico Digital Institucional de la Universidad Vasco de Quiroga, cuyo objetivo es integrar organizar, almacenar, preservar y difundir en formato digital la producción intelectual resultante de la actividad académica, científica e investigadora de los diferentes campus de la universidad, para beneficio de la comunidad universitaria.

Esta iniciativa está a cargo del Centro de Información y Documentación "Dr. Silvio Zavala" que lleva adelante las tareas de gestión y coordinación para la concreción de los objetivos planteados.

Esta Tesis se publica bajo licencia Creative Commons de tipo "Reconocimiento-NoComercial-SinObraDerivada", se permite su consulta siempre y cuando se mantenga el reconocimiento de sus autores, no se haga uso comercial de las obras derivadas.





**UNIVERSIDAD
VASCO DE QUIROGA**

ESCUELA DE SISTEMAS COMPUTARIZADOS

**SISTEMA DE ADMINISTRACION
EMPRESARIAL PARA
COMERCIALIZADORAS DE CALZADO**

TESIS

Para Obtener el Título de:
LICENCIADO EN SISTEMAS COMPUTARIZADOS

Presenta:
BISMARCK IVANOVITCH MENDEZ HEREDIA

Asesor :
ING. VICTOR MANUEL PINEDA ALMANZA

Clave : 16PSU0014Q

No. Acuerdo : 952006

Noviembre / 2002

AGRADECIMIENTOS

A MI MADRE.

Por su profundo amor, interés y confianza que siempre ha depositado en mi.

Por el apoyo que me brindó durante los años de mi carrera.

A LA MEMORIA DE MI PADRE.

Que espiritualmente siempre me ha acompañado.

A MIS ABUELITOS.

Por su ejemplo honradez y cariño.

A MIS TIOS.

Con estimación.

A MI ASESOR.

Quién no escatimando tiempo,
tuvo la paciencia de asesorar el
presente trabajo.

A MI DIRECTOR.

Por su valiosa ayuda para la
elaboración de este trabajo.

A LA UNIVERSIDAD VASCO DE QUIROGA.

Fuente inagotable del
saber.

INDICE

INTRODUCCIÓN	1
CAPITULO 1 ANTECEDENTES	4
CAPITULO 2 OBJETIVO GENERAL	5
2.1 Objetivos Específicos	6
2.2 Metas	7
CAPITULO 3 ANALISIS DEL PROCESO ACTUAL	8
3.1 Proveedores	8
3.2 Compras	9
3.3 Ventas	13
3.4 Devoluciones	14
3.5 Cambios	14
3.6 Ofertas	15
3.7 Facturación	15
CAPITULO 4 MARCO TEORICO	17
4.1 Teoría General de Sistemas	17

4.2	Metodología de aplicación de la Teoría General de Sistemas para el análisis y diseño de sistemas	20
4.2.1	El sistema de control	22
4.2.2	Método de control	22
4.2.3	El sistema de control en las organizaciones	23
4.2.4	Gráfico del sistema ó proceso de control	24
4.3	Análisis y diseño de sistemas	25
4.3.1	Requerimientos del sistema	25
4.3.2	Requerimientos de entrada	26
4.3.3	Requerimientos de salida	26
4.3.4	Requerimientos de almacenamiento	27
4.4	Ciclo de vida clásico de un sistema	29
4.4.1	Investigación preliminar	30
4.4.2	Determinación de requerimientos	30
4.4.3	Diseño del sistema	30
4.4.4	Desarrollo del software	30
4.4.5	Prueba de los sistemas	31
4.4.6	Implantación y evaluación	31
4.5	Base de Datos	31
4.5.1	Características de las Bases de Datos	32

4.5.2	Ventajas en el uso de Bases de Datos	33
4.5.3	Diseño de una Base de Datos	34
4.5.3.1	Modelo jerárquico de datos	34
4.5.3.2	Modelo de datos en red	35
4.5.3.3	Modelo relacional de datos	36
4.6	Visual Basic	39
4.6.1	Características de Visual Basic	40
4.6.2	Tipos de declaración de variables	41
4.6.3	Tipos de Variables	42
4.7	Microsoft Access	43
4.8	ActiveReports Versión 2 Professional	45
CAPITULO 5	ESTRUCTURA DE LA BASE DE DATOS	46
5.1	Relaciones de las Tablas	53
CAPITULO 6	MANUAL DE USUARIO	55
6.1	Características y requerimientos	56
6.2	Botones más usados y sus funciones	57
6.3	Instalación	58
6.4	Como iniciar	61
6.5	Menú Principal	62
6.6	Opciones del menú y sus funciones	64

6.6.1	Archivo	64
6.6.1.1	Inicializar Archivos	64
6.6.1.2	Respaldo de información	65
6.6.1.3	Recuperación de información	65
6.6.1.4	Ruta de Base de Datos	66
6.6.1.5	Salir	66
6.6.2	Parámetros	67
6.6.3	Seguridad	67
6.6.4	Ayuda	68
6.6.5	Proveedores	69
6.6.6	Compras	71
6.6.7	Inventarios	77
6.6.8	Ventas	79
6.6.9	Facturas	83
6.6.10	Clientes	85
6.6.11	Empleados	87
6.6.12	Devoluciones	89
6.6.13	Catálogos	92
6.6.14	Históricos	94
CONCLUSIONES		99
BIBLIOGRAFÍA		102
CODIGO FUENTE		104
ANEXO		205

INTRODUCCION

INTRODUCCIÓN

En los últimos años el mundo ha sufrido una serie de cambios, especialmente en lo que respecta a las prácticas de los negocios.

El cliente se ha convertido en una fuente de información estratégica sobre la calidad del producto y del servicio en lugar de ser únicamente el objetivo de las campañas publicitarias de empresas productivas y de servicios. En las empresas de calzado por ejemplo, se elaboran planes de amplio alcance para satisfacer las necesidades del cliente. La misión ya no consiste en ser mejores que la competencia. La misión ahora es la Excelencia.

En consecuencia, las formas clásicas de dirigir a las organizaciones basadas en el análisis y la optimización de cada una de sus áreas funcionales van perdiendo vigencia a favor de enfoques de dirección sistémicos que abogan por conseguir un funcionamiento del sistema suficientemente satisfactorio para cada una de sus partes.

El Control Total de la Calidad, la Dirección por Objetivos, el Marketing y la Logística, entre otros constituyen bajo formas y campos diferentes, métodos de gestión inspirados en un enfoque integrador y sistémico.

Hoy en día todos los negocios ya sean grandes o pequeños están más relacionados con el uso de las computadoras que con los métodos manuales.

El éxito de un sistema basado en computadoras depende en gran medida de la definición concisa, y de la documentación, de la relación entre él y los sistemas manuales con los que debe de tener contacto.

La responsabilidad de los programadores es el diseño de sistemas, pero para la efectiva realización de esta amplia función, tales personas deben de tener el control centralizado o al menos tomar en cuenta las siguientes áreas de cualquier empresa:

1. Diseño y control de formas.
2. Estructura de procesamientos y control del manual de procedimientos.
3. Administración de los registros.
4. Distribución de las oficinas.
5. Estudios de simplificación de trabajos.

Es complicado definir qué es y qué no es un lenguaje de programación. Se asume generalmente que la traducción de las instrucciones a un código que comprende la computadora debe ser completamente sistemática. Normalmente es la computadora la que realiza la traducción.

Debido a esto el programador tiene la importante decisión de seleccionar el lenguaje de programación más adecuado y en el que tenga mayor experiencia.

Por todo lo anterior en el presente trabajo se mostrará detallada y gráficamente como a partir de la necesidad de un mejor servicio al cliente y agilización de procesos de la Zapatería Anylú surge un sistema computacional en el cual se usa como lenguaje de programación el Visual Basic debido a la experiencia obtenida en su manejo y por sus cualidades que se explicaran más adelante.

CAPITULO 1

ANTECEDENTES

1 . ANTECEDENTES

La zapatería actualmente conocida como ANYLU se remonta hacia la fecha del 18 de Diciembre de 1954 cuando se inauguró, estando situada en el portal mayor de la plaza principal de la ciudad de Zacapu Michoacán y que al paso del tiempo fue creciendo y convirtiéndose en una tradición familiar que continúa hasta hoy en día, pero ya establecida en un local propio y con el tiempo fue enriqueciéndose en cuanto a calidad y cantidad de calzado, actualmente dicha zapatería se encuentra situada en la Calle Juárez # 648, Colonia centro en la ciudad de Zacapu Michoacán y tiene un horario laboral de Lunes a Sábado de 9 AM a 7 PM.

El objetivo principal de establecer este negocio fue la necesidad de abastecimiento a la ciudad de Zacapu Michoacán de calzado en todas sus líneas, tanto de hombre como de mujer ya que haciendo un estudio de mercado previo acerca de estos artículos se detectó que las zapaterías existentes no daban abasto a la población. Así que además de la venta se hizo un compromiso de dar un servicio eficiente y amable.

A pesar del buen funcionamiento de la zapatería hasta ahora en forma manual, el dueño se ha convencido que los procesos pueden ser agilizados y optimizados por medio de un sistema de computo y además este sistema sería útil incluso en ahorros de tiempo y dinero.

CAPITULO 2

OBJETIVO GENERAL

2 . OBJETIVO GENERAL

La realización del Sistema de Administración Empresarial para Comercializadoras de Calzado (Que en lo consecutivo dentro de éste trabajo se definirá con las siglas **SAECC**) que permita apoyar la actualización y la superación en un sentido técnico en el que se espera obtener:

Rapidez.- Hacer la selección en base a un registro en el momento de la entrega de documentos.

Confiabilidad.- Tener el registro en un sistema, para que este no pueda ser modificado ni alterado.

Calidad.- De esta forma se tendría una total Organización, Dirección y Control del personal a ser contratado.

2.1 OBJETIVOS ESPECIFICOS

- Agilización en la búsqueda de calzado en la bodega, ya que actualmente existen problemas y tardanzas.
- Llevar un mejor control en cuanto al registro de ventas, bajas en inventarios, devoluciones, reportes de venta mensuales y diarios.
- Mayor rapidez en los cortes de cajas diarios.
- Mejor servicio a clientes y proveedores en aspectos tales como : pedidos, devoluciones, facturas, agenda de proveedores y agenda de clientes.
- Incrementar medidas de seguridad para un mejor control de la zapatería.
- Control de todo el personal que interviene en la operación del sistema.
- Creación de mayores utilidades económicas al dueño de la zapatería, ya que se dará mejor atención a los clientes y esto aumentará la clientela.

2.2 METAS

- En el sistema se enfocará principalmente en la Teoría de Sistemas y en la Teoría General de la Administración.
- El presente sistema se realizará con el objetivo de agilizar los métodos administrativos y contables.
- Se pretenderá obtener el sistema en el tiempo contemplado con el cliente.
- Generar un sistema amigable en el manejo y a la vista del usuario.
- Elaborar el sistema lo más completo posible, dándole al usuario mayores opciones en caso de crecimiento de la empresa.
- Lograr un sistema de calidad.

CAPITULO 3

ANALISIS DEL PROCESO ACTUAL

3 . ANÁLISIS DEL PROCESO ACTUAL

Desde su fundación y hasta la fecha la Zapatería Anylú ha desarrollado todas sus actividades de manera manual y de la manera que se describe a continuación:

3.1 PROVEEDORES

El contacto con los proveedores se realiza por medio de recomendaciones de otras zapaterías, otros proveedores son contactados para probar su calzado y a los ya conocidos se les visita directamente cuando hay necesidad de surtir.

El surtido del zapato es variable, entre 8 a 22 días, también varía de acuerdo a la temporada, como por ejemplo para el 20 de noviembre se surte 70% de tenis y 30% guarache.

Los pagos se hacen en abonos de 4 pagos semanales o si se surte cada mes se expide un cheque, si se es cliente del proveedor, éste nos proporciona el número de su cuenta para el deposito.

En los casos de que el proveedor ofrezca mercancía nueva se deja a prueba sin compromiso, si no es aceptada en el mercado se regresa.

La mercancía puede llegar por medio de los proveedores, por paquetería mandando un previo aviso para recogerla y otra

por auto-transporte. No todos los proveedores traen muestras físicas, algunos traen catálogos.

En las notas expedidas por parte de los proveedores aparece el precio unitario, el número de pares la suma total de estos, el crédito (que es variable dependiendo de cada proveedor), por ejemplo cuando son proveedores nuevos dan un crédito de una semana y los ya conocidos dan hasta un mes.

Los precios de mayoreo por parte de los proveedores se hacen dependiendo del proveedor, ya que ellos estipulan a partir de qué cantidad de docenas hacen un descuento, que no está definido, sino que también dependerá del proveedor.

3.2 COMPRAS

Al recibir el calzado se hace una revisión minuciosa de que éste sea el modelo correcto, que el material esté en buenas condiciones, la numeración sea la correcta, que no estén revueltos con otros estilos, que no vengan las cajas vacías, que el color sea el solicitado.

Si el zapato tiene defecto se separa y se hacen anotaciones de en que docena llegó, la fecha, el defecto que tiene y se guarda para cuando regrese el proveedor.

Si se presenta cualquier caso anterior los proveedores recogen la mercancía sin costo alguno.

Posteriormente se re-etiqueta la numeración para una mejor búsqueda (esto debido a que a veces la numeración no es muy visible en las cajas) además se les coloca a las cajas el precio, y éste ya lleva el porcentaje de ganancia.

También se hace una lista de precios que están bajo inventario.

Todos los pares entrantes se asientan en un libro de la siguiente manera:

- 1) Marca
- 2) Modelo
- 3) No. de pares
- 4) Sexo
- 5) Color
- 6) Precio al mayoreo y menudeo
- 7) Línea

Ejemplo:

Marca	Modelo	No. De pares	Sexo	Color	Precio Mayoreo Y menudeo	Línea
Ozono	7080	36 pares	Mujer	Negro	180.00 - 190.00	2/5.5
Flexi	5846	12 pares	Hombre	Negro	450.00 – 580.00	5/8.5
NIKE	5479	12 pares	Joven	Blanco	1124.00-1139.00	2/4.5

El zapato se coloca en anaqueles por línea y sexo, por ejemplo si es zapato de 18-21 de niño se colocará en el sexo de niño y en la línea del 18-21, y si es un zapato de mujer del 2-5 deberá ir ubicado en el sexo de mujer y línea del 2-5.

Se pone una muestra del nuevo modelo en el aparador, procurando que sea el número más solicitado.

Al zapato se le pone un margen de precio para que en un momento dado sea expuesto a rebajas, por ejemplo cuando se vende por mayoreo (de 3 pares en adelante).

Los estilos (casual, deportivo, formal, fútbol, etc.) y las líneas (el tamaño del calzado) existentes son:

CASUALES

Hombre		Mujer		Bebe	
Hombre	5/8.5	Mujer	2/5	Niño	10/12
Joven	2/4.5	Niña	18/21	Niña	10/12
Niño	18/21	Niña	15/17		
Niño	15/17	Niña	12/14		
Niño	12/14				

Guarache (Números completos)

Hombre	5/8
Joven	2/5
Mujer	2/5

Tenis

Hombre		Mujer		Bebe	
Hombre	5/8.5	Mujer	2/5	Niño	10/12
Joven	2/4.5	Niña	18/21	Niña	10/12
Niño	18/21	Niña	15/17		
Niño	15/17	Niña	12/14		
Niño	12/14				

Zapato de trabajo

Hombre	5/8.5
Joven	2/4.5
Niño	18/21

Bota de hule

Hombre	5/8.5
Joven	2/4.5
Niño	18/21
Niño	15/17
Niño	12/14

Bota vaquera

Hombre	5/8.5
Joven	2/4.5
Niño	18/21
Niño	15/17
Niño	12/14

Sandalias

Hombre		Mujer		Bebe	
Hombre	5/8.5	Mujer	2/5	Niño	10/12
Joven	2/4.5	Niña	18/21	Niña	10/12
Niño	18/21	Niña	15/17		
Niño	15/17	Niña	12/14		
Niño	12/14				

3.3 VENTAS

Al inicio de la operación diaria se toma en cuenta el dinero en caja que es variable todos los días.

También se lleva un registro de las ventas donde se anota el costo, precio, fecha, marca, número, línea. Se expiden notas especificando la fecha de venta, cantidad de pares, precio unitario, marca, modelo, total. Al final del día se hace el corte de caja.

El precio del calzado aumenta un porcentaje respecto al precio que nos proporciona el proveedor, para obtener una ganancia.

En cuanto al precio de mayoreo éste se da cuando se compran de tres pares en adelante y a la suma total se le rebaja el porcentaje que sea definido por el propietario.

3.4 DEVOLUCIONES

Las devoluciones se aceptan durante los primeros 22 días de garantía. La garantía cubre al calzado, solo si: está despegado, descosido, tiene suelas partidas, que tenga defectos de fabricación.

No se aceptan devoluciones cuando no les gusta el modelo, el zapato ha sido usado, si no les gusta la suavidad del calzado, si las suelas cambiaron de color, en términos generales por no tener el cuidado de tratar adecuadamente el calzado en general.

Cuando hay alguna devolución si el cliente no encuentra un zapato igual al que le salió defectuoso, entonces se le da un vale para que pase en un tiempo determinado por el par, que habrá repuesto el proveedor o en su defecto si el daño es menor se manda arreglar sin costo alguno para el cliente, pero de ninguna forma se le regresa el dinero.

3.5 CAMBIOS

Los cambios se efectúan cuando el número no le queda al cliente, si no es color deseado, si no es el modelo deseado.

Cuando en el caso de que el cliente desee cambiar su calzado por uno más caro se regresa el zapato y se paga la diferencia.

Cuando el cliente quiere cambiar algún zapato por cualquiera de los motivos anteriores se le da la opción de cambiarlo por el mismo modelo, cambiarlo por un modelo más caro pagando la diferencia, o en su defecto por uno de menor costo reembolsando la diferencia.

3.6 OFERTAS

Para el manejo de ofertas se toman en cuenta los siguientes aspectos:

- Cuando hay algún calzado que haya pasado de moda este se pone en oferta.
- Cuando empiezan a quedarse números no comerciales y dichas numeraciones que por lo regular son últimos pares de las docenas se ponen en oferta.
- Se tiene un límite para desplazar el zapato y cuando no ocurre esto se empieza a bajar un porcentaje y así hasta llegar al precio de costo y si aún así no sale el calzado se baja un poco más del costo para recuperar parte de la inversión.
- También esto se hace cuando los modelos no funcionan muy bien.

3.7 FACTURACIÓN

Por lo general los clientes no piden facturas , pero si se diera el caso se le piden los datos al cliente y se le adiciona al calzado el 15%

de IVA por lo que aumentaría su costo, de hecho el calzado que se vende usualmente no se le aumenta el IVA ya que la zapatería está dada de alta ante Hacienda como un pequeño contribuyente. Al calzado sólo se le aumenta el IVA cuando el cliente lo pide.

CAPITULO 4

MARCO TEORICO

4 . MARCO TEORICO

4.1 TEORIA GENERAL DE SISTEMAS

La teoría de la organización y la práctica administrativa han experimentado cambios sustanciales en años recientes.

La información proporcionada por las ciencias de la administración y la conducta ha enriquecido a la teoría tradicional.

Estos esfuerzos de investigación y de conceptualización a veces han llevado a descubrimientos divergentes. Sin embargo, surgió un enfoque que puede servir como base para lograr la convergencia, el enfoque de sistemas, que facilita la unificación de muchos campos del conocimiento. Dicho enfoque ha sido usado por las ciencias físicas, biológicas y sociales, como marco de referencia para la integración de la teoría organizacional moderna.

En el sistema se enfocará a la buena administración de la zapatería en todos los sentidos y es por esto que en cada módulo de éste principalmente estarán fundamentados en la Teoría de Sistemas y en la Teoría General de la Administración y que al interactuar dan como resultado un buen sistema.

La teoría de sistemas es consecuencia de la teoría General de Sistemas desarrollados por Von Bertalanffy y que se expandió hacia todas las ciencias, influyendo notablemente sobre la administración.

Su carácter integrador y abstracto y la posibilidad de comprensión de los efectos sinérgicos de la organización, son realmente sorprendentes. La visión del hombre funcional dentro de las organizaciones es la consecuencia principal sobre la concepción de la naturaleza humana. A pesar del enorme impulso de la teoría de sistemas aún carece de una mejor sistematización y detalle.

La teoría general de la administración pasó por una gradual y creciente ampliación desde el enfoque clásico – pasando por la humanística neoclásica, estructuralista, y behaviorista - hasta el enfoque sistemático.

En su época el enfoque clásico había sido profundamente influido por tres principios intelectuales dominantes en casi todas las ciencias en el inicio de este siglo: el reduccionismo, el pensamiento analítico y el mecanicismo.

Con la llegada de la Teoría General de los Sistemas, los principios de reduccionismo, del pensamiento analítico y del mecanicismo se sustituyeron totalmente por los principios opuestos de expansionismo, pensamiento sintético y de la teología.

Con estos tres principios la Teoría General de Sistemas proporcionó el surgimiento de la cibernética, de la investigación operacional y desembocó en la Teoría General de la Administración, dando una nueva dimensión a sus concepciones.

El concepto de sistema pasó a dominar las ciencias y principalmente la administración.

La empresa se presenta como una estructura autónoma con capacidad de reproducirse y puede ser visualizada a través de una teoría de sistemas capaz de proporcionar la visión de un sistema de sistemas de toma de decisiones, tanto del punto de vista individual como colectivo, o sea, de la organización como un conjunto. El enfoque sistemático dentro de la administración tiene por objeto representar, de forma comprensiva y objetiva el medio en que tiene lugar la toma de decisiones, toda vez que la toma de decisión sea mucho más fácil si se encuentra con una descripción concreta y objetiva del sistema dentro del cual va a ser tomada.

La teoría de sistemas penetró rápidamente en la teoría administrativa por dos razones básicas:

- a) Por un lado, frente a la necesidad de una síntesis y una integración mayor de las teorías que la precedieron, esfuerzo intentado con bastante éxito en la aplicación de las ciencias del comportamiento de la organización desarrollado por behavioristas.
- b) Por otro lado, la matemática , la cibernética, de un modo general y la tecnología de la información, de un modo especial, trajeron inmensas posibilidades y desarrollo y operatividad de las ideas que convergían hacia una teoría de sistemas más aplicada a la administración.

4.2 METODOLOGIA DE APLICACION DE LA TEORIA GENERAL DE SISTEMAS PARA EL ANALISIS Y DISEÑO DE SISTEMAS

Desde el punto de vista de la administración está compuesta de las siguientes etapas:

a) **Análisis de situación:** es la etapa en que el analista toma conocimiento del sistema, se ubica en cuanto a su origen, objetivo y trayectoria.

1. Definición de objetivo: el analista trata de determinar para que ha sido requerido ya que en general se le plantean los efectos pero no las causas.

2. Formulación del plan de trabajo: el analista fija los límites de interés del estudio a realizar, la metodología a seguir, los recursos materiales y humanos que necesitará, el tiempo que consumirá el trabajo y el costo del mismo. Esta etapa se conoce como propuesta de servicio y a partir de su aprobación se continúa con la metodología.

3. Relevamiento: el analista recopila toda la información referida al sistema en estudio, como así también toda la información que hace al límite de interés.

4. Diagnóstico: el analista mide la eficacia y la eficiencia del sistema en estudio. Eficacia es cuando el sistema logra los objetivos y eficiencia es cuando el sistema logra los objetivos con una relación costo beneficio positiva. Si un sistema es eficaz

pero no eficiente el analista deberá cambiar los métodos del sistema, si un sistema no es eficaz el analista deberá cambiar el sistema y si un sistema es eficiente el analista sólo podrá optimizarlo.

5. Diseño: el analista diseña el nuevo sistema.

a) Diseño global: en el determina la salida, los archivos, las entradas del sistema, hace un cálculo de costos y enumera los procedimientos. El diseño global debe ser presentado para su aprobación, aprobado el diseño global pasamos al siguiente paso.

b) Diseño detallado: el analista desarrolla en detalle la totalidad de los procedimientos enumerados en el diseño global y formula la estructura de organización la cual se aplicara sobre dichos procedimientos.

6. Implementación: la implementación del sistema diseñado significa llevar a la práctica al mismo, esta puesta en marcha puede hacerse de tres formas.

- a) Global.
- b) En fases.
- c) En paralelo.

7. Seguimiento y control: El analista debe verificar los resultados del sistema implementado y aplicar las acciones correctivas que considere necesarias para ajustar el problema.

4.2.1 EL SISTEMA DE CONTROL

Un sistema de control estudia la conducta del sistema con el fin de regularla de un modo conveniente para su supervivencia. Una de sus características es que sus elementos deben ser lo suficientemente sensitivas y rápidas como para satisfacer los requisitos para cada función del control.

4.2.2 METODO DE CONTROL.

Es una alternativa para reducir la cantidad de información recibida por quienes toman decisiones, sin dejar de aumentar su contenido informativo. Las tres formas básicas de implementar el método de control son:

1.- Reporte de variación: esta forma de variación requiere que los datos que representan los hechos reales sean comparados con otros que representan los hechos planeados, con el fin de determinar la diferencia. La variación se controla luego con el valor de control, para determinar si el hecho se debe o no informar. El resultado del procedimiento, es que únicamente se informa a quién toma las decisiones acerca de los eventos o actividades que se apartan de modo significativo de los planes, para que tomen las medidas necesarias.

2.- Decisiones programadas: otra aplicación de sistema de control implica el desarrollo y la implantación de decisiones programadas. Una parte apreciable de las decisiones de carácter

técnico y una parte pequeña de las decisiones tácticas abarcan decisiones repetitivas y rutinarias. Diseñando el sistema de información de manera que ejecute esas decisiones de rutina, el analista proporciona a los administradores más tiempo para dedicarse a otras decisiones menos estructuradas.

Si se procura que el sistema vigile las órdenes pendientes y se programa las decisiones de cuáles pedidos necesitan mayor atención, se logrará un significativo ahorro de tiempo y esfuerzo.

3.- Notificación automática: en este caso, el sistema como tal, no toma decisiones pero como vigila el flujo general de información puede proporcionar datos, cuando sea preciso y en el momento determinado.

Las notificaciones automáticas se hacen en algunos criterios predeterminados, pero solo quienes toman las decisiones deben decir si es necesario o no emprender alguna acción.

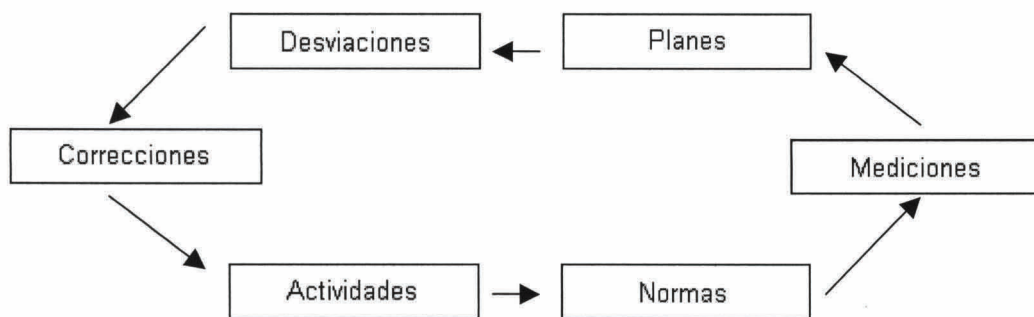
4.2.3 EL SISTEMA DE CONTROL EN LAS ORGANIZACIONES.

El control es uno de los cinco subsistemas corporativos (organización, planificación, coordinación y dirección son los restante) los cuales son muy difíciles de separar con respecto al de control. De ello se desprende todo el proceso administrativo, debe considerarse como un movimiento circular, en el cual todos los subsistemas están ligados intrincadamente, la relación entre la planificación y el control es

muy estrecha ya que el directivo fija el objetivo y las normas, ante las cuales se contrastan y evalúan acciones.

Es necesario ver al control para determinar si las asignaciones y las relaciones en la organización están siendo cumplimentadas tal como se las había previsto.

4.2.4 GRAFICO DEL SISTEMA O PROCESO DE CONTROL



Este gráfico representa el proceso de control como un sistema cerrado, es decir que posee la característica de la retroalimentación o autorregulación. El movimiento es circular y continuo, produciéndose de la siguiente manera: se parte de la actividad o realidad a la cual debemos medir, con el auxilio o utilización de normas, efectuada la decisión comparamos los resultados de los planes, de esta manera la realidad quedará ajustada para el futuro. Se nota en este punto que no sólo la realidad puede ser ajustada, otras veces son los planes los que necesitan corrección por estar sensiblemente alejado de las actividades.

4.3 ANÁLISIS Y DISEÑO DE SISTEMAS

En la actualidad para muchas organizaciones, los sistemas de información basados en computadoras son el corazón de las actividades cotidianas y objeto de gran consideración en la toma de decisiones, las empresas consideran con mucho cuidado las capacidades de sus sistemas de información cuando deciden ingresar o no en nuevos mercados o cuando planean la respuesta que darán a la competencia.

4.3.1 REQUERIMIENTOS DEL SISTEMA

Los Sistemas de Información por computadora normalmente están integrados por muchos componentes. En la mayor parte de los casos, es difícil para los analistas entender todos estos componentes aún mismo tiempo; por lo tanto los investigadores tienen que comenzar con preguntas de tipo general con relación al propósito del sistema sus entradas y salidas de los procesos incluidos.

En los grandes proyectos de sistema varios analistas llevan a cabo una investigación en forma seccionada que la distribuyen entre ellos mismos, de manera que cada uno pueda trabajar en forma independiente.

Existen dos estrategias ampliamente utilizadas para determinar los requerimientos de información. Se clasifican en dos tipos:

1. - Flujo de datos.
2. - Estrategias de análisis de decisión para el conocimiento para los sistema de información.

4.3.2 REQUERIMIENTOS DE ENTRADA

Es el enlace que une al sistema de información con el mundo y sus usuarios, en esta existen aspectos generales que todos los analistas deben tener en cuenta estos son:

- Objetivos del diseño de entrada.
- Captura de datos para la entrada.

4.3.3 REQUERIMIENTOS DE SALIDA

A menudo, para los usuarios la característica más importante de un sistema de información es la salida que produce. Si la salida no es de calidad, se pueden convencer de que todo el sistema es tan innecesario que eviten su utilización y, por lo tanto, posiblemente ocasionen errores y que el sistema falle.

Los sistemas de información ya sean que se desarrollen sobre sistemas pequeños de escritorio o sobre grandes sistemas, utilizan tres métodos principales para la salida los cuales se clasifican en:

- Impresión
- Pantalla

- Despliegue y audio

Salida Impresa

Este tipo de salida es la que se encarga de producir grandes volúmenes de informes impresos. Un informe bien diseñado puede reemplazar a otro elaborado pobremente, proporcionando detalles innecesarios la cual no ayuda nada. Las opciones de salida impresa más comunes en las empresas son en papel.

Objetivos de la Salida

- Expresar la información relacionada con actividades pasadas, estado actual o proyecciones para el futuro.
- Señalar eventos importantes, oportunidades, problemas ó advertencia.
- Iniciar una acción
- Confirmar una acción.

4.3.4 REQUERIMIENTOS DE ALMACENAMIENTO

Los siguientes constituyen los tipos más comunes de dispositivos de almacenamiento:

Unidades de:

- Disco duro
- Disquete
- Compresión ZIP
- CD

➤ DVD

La función del Análisis puede ser dar soporte a las actividades de un negocio, o desarrollar un producto que pueda venderse para generar beneficios. Para conseguir este objetivo, un Sistema basado en computadoras hace uso de seis elementos fundamentales:

- Software, que son programas de computadora, con estructuras de datos y su documentación que hacen efectiva la logística, metodología o controles de requerimientos del programa.
- Hardware, dispositivos electrónicos y electromecánicos, que proporcionan capacidad de cálculos y funciones rápidas, exactas y efectivas (Computadoras, sensores, maquinarias, bombas, lectores, etc.), que proporcionan una función externa dentro de los sistemas.
- Personal, son los operadores o usuarios directos de las herramientas del sistema.
- Base de Datos, una gran colección de informaciones organizadas y enlazadas al sistema a las que se accede por medio del software.
- Documentación, manuales, formularios, y otra información descriptiva que detalla o da instrucciones sobre el empleo y operación del programa.

- Procedimientos, o pasos que definen el uso específico de cada uno de los elementos o componentes del sistema y las reglas de su manejo y mantenimiento.

4.4 CICLO DE VIDA CLÁSICO DE UN SISTEMA

Para el desarrollo del presente sistema se usarán como base el modelo de vida clásico.

El modelo de ciclo de vida clásico puede considerarse el más antiguo y el más utilizado en el desarrollo de sistemas, en ocasiones denominado modelo de cascada. Este modelo proporciona un enfoque tradicional en el desarrollo de sistemas, James Senn sugiere las siguientes actividades representadas en la figura 1.1., que constituyen el modelo de ciclo de vida clásico.



Figura 1.1. Ciclo de vida clásico

4.4.1 INVESTIGACIÓN PRELIMINAR.

Esta actividad tiene su punto de partida en una solicitud por la parte que requiere del sistema. Se lleva a cabo un estudio de factibilidad y a partir de este último se aprueba o no el desarrollo del sistema.

4.4.2 DETERMINACIÓN DE REQUERIMIENTOS.

Una vez aprobada la solicitud, se deben de realizar las investigaciones correspondientes para determinar los requisitos que deberá cumplir el sistema. Esto puede llevarse a cabo utilizando técnicas de recopilación de información como son cuestionarios, entrevistas, análisis de grupos, observación, muestreo y recopilación de documentos, etc.. Es decir en esta parte del modelo se lleva a cabo el análisis del sistema.

4.4.3 DISEÑO DEL SISTEMA.

El diseño del sistema se encuentra orientado a la solución del problema, representando mediante gráficas, tabulaciones u otros elementos, los atributos con los que deberá contar el sistema en proyecto. Estos atributos parten desde la estructura de los datos que se encuentran involucrados hasta el proporcionar una caracterización de las interfases con las que contará el sistema, que cumplan y faciliten su uso.

4.4.4 DESARROLLO DEL SOFTWARE.

Durante esta fase se llevará a cabo la codificación del sistema.

4.4.5 PRUEBA DE LOS SISTEMAS.

Una vez que se ha realizado la generación de código, el siguiente paso consiste en realizarle una prueba al sistema generado, para asegurar que éste no tenga fallas y produzca los resultados esperados a partir de las entradas definidas. En esta fase del sistema se llega a involucrar al usuario.

4.4.6 IMPLANTACIÓN Y EVALUACIÓN.

El proceso de implantación y evaluación consiste desde proporcionar la capacitación necesaria al usuario para el manejo del sistema hasta llevar a cabo una evaluación del sistema para identificar la calidad del producto. También durante este proceso se llevan a cabo tareas de planificación de mantenimiento, ya que posiblemente el sistema sufrirá de cambios a lo largo de su vida útil.

4.5 BASE DE DATOS

El término de bases de datos fue escuchado por primera vez en 1963, en un simposio celebrado en California – USA.

Una base de datos se puede definir como un conjunto de información relacionada que se encuentra agrupada ó estructurada.

Desde el punto de vista de la informática, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos.

Este trabajo se realizara con la finalidad de aprender un poco más sobre una base de datos, sus características, usos, estructuras, diseños, entre otros.

Una base de datos tiene mucha importancia en el ritmo de vida que llevamos en los actuales momentos, ya que, ésta acelera el ritmo en el momento de realizar una búsqueda de información.

Algunos conceptos de bases de datos:

Base de Datos: es la colección de datos aparentes usados por el sistema de aplicaciones de una determinada empresa.

Base de Datos: es un conjunto de información relacionada que se encuentra agrupada o estructurada. Un archivo por sí mismo no constituye una base de datos, sino más bien la forma en que está organizada la información, es la que da origen a la base de datos.

Base de Datos: colección de datos organizada para dar servicio a muchas aplicaciones al mismo tiempo al combinar los datos de manera que aparezcan estar en una sola ubicación.

4.5.1 CARACTERISTICAS DE LAS BASES DE DATOS.

Una base de datos contiene entidades de información que están relacionadas vía organización y asociación. La arquitectura lógica de una base de datos se define mediante un esquema que representa las definiciones de las relaciones entre las entidades de información. La arquitectura física de una base de datos depende de la configuración

del hardware residente. Sin embargo, tanto el esquema (descripción lógica), como la organización (descripción física) deben adecuarse para satisfacer los requerimientos funcionales y de comportamiento para el acceso al análisis y creación de informes.

4.5.2 VENTAJAS EN EL USO DE BASES DE DATOS.

La utilización de bases de datos como plataforma para el desarrollo de Sistemas de Aplicación en las organizaciones se ha incrementado notablemente en los últimos años, se debe a las ventajas que ofrece su utilización, algunas de las cuales se comentarán a continuación:

- Globalización de la información: permite a los diferentes usuarios considerar la información como un recurso corporativo que carece de dueños específicos.
- Eliminación de información inconsistente: si existen dos o más tablas con la misma información, los cambios que se hagan a éstos deberán hacerse a todas las copias de la tabla.
- Permite compartir información.
- Permite mantener la integridad en la información: la integridad de la información es una de sus cualidades altamente deseable y tiene por objetivo que sólo se almacena la información correcta.

- Independencia de datos: el concepto de independencia de datos es quizás el que más ha ayudado a la rápida proliferación del desarrollo de Sistemas de Bases de Datos. La independencia de datos implica un divorcio entre programas y datos.

4.5.3 DISEÑO DE UNA BASE DE DATOS.

Existen distintos modos de organizar la información y representar las relaciones entre los datos en una base de datos. Los Sistemas administradores de bases de datos convencionales usan uno de los tres modelos lógicos de bases de datos para hacer seguimiento de las entidades, atributos y relaciones. Los tres modelos lógicos principalmente de bases de datos son el jerárquico, de redes y el relacional. Cada modelo lógico tiene ciertas ventajas de procesamiento y también ciertas ventajas de negocios.

4.5.3.1 MODELO JERARQUICO DE DATOS.

Una clase de modelo lógico de bases de datos que tiene una estructura arborescente. Un registro subdivide en segmentos que se interconectan en relaciones padre e hijo y muchos más. Los primeros sistemas administradores de bases de datos eran jerárquicos. Puede representar dos tipos de relaciones entre los datos: relaciones de uno a uno y relaciones de uno a muchos. En la Figura 1.1 se muestra un ejemplo del modelo jerárquico.

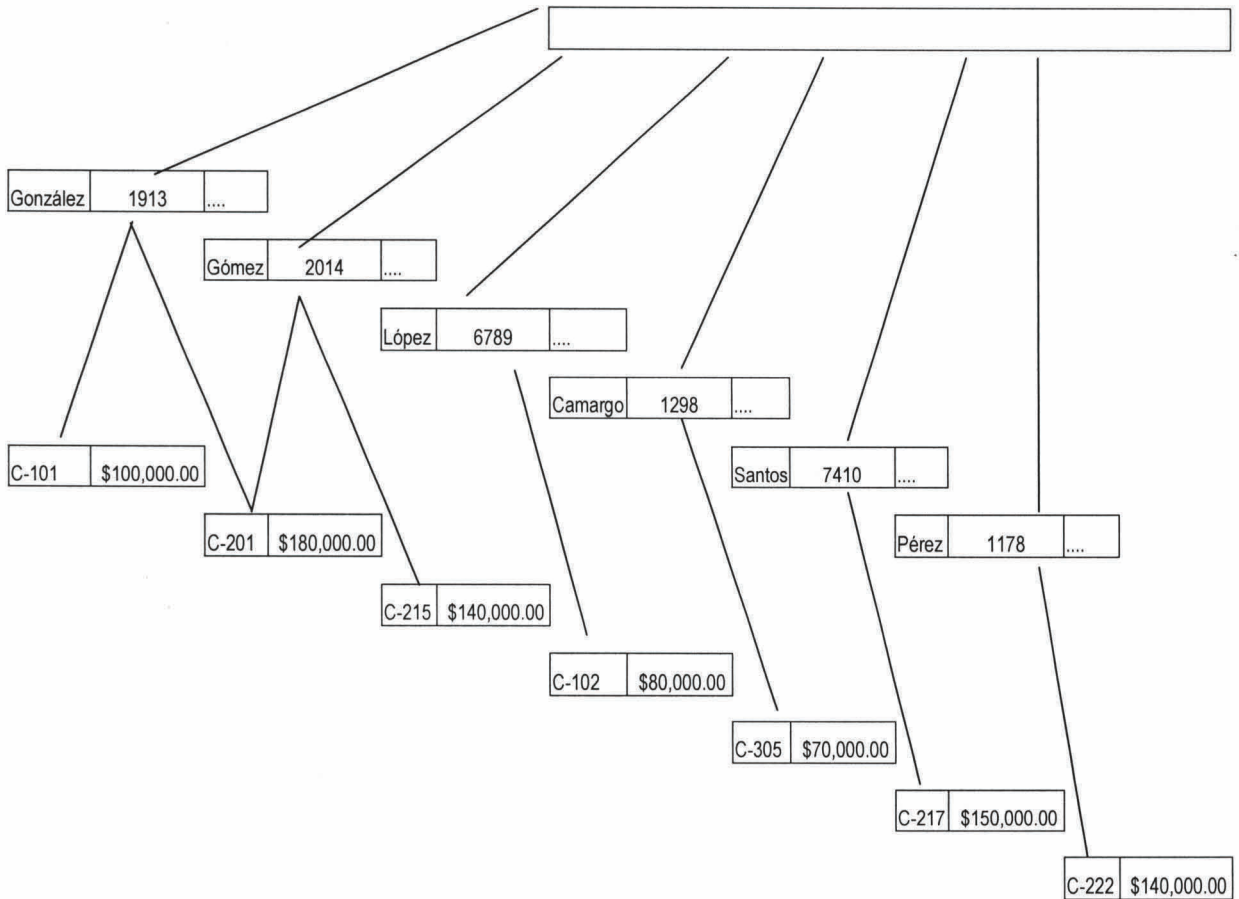


Figura 1.1 Ejemplo del modelo jerárquico.

4.5.3.2 MODELO DE DATOS EN RED.

Es una variación del modelo de datos jerárquico. De hecho las bases de datos pueden traducirse de jerárquicas a redes y viceversa con el objeto de optimizar la velocidad y la conveniencia del procesamiento. En la Figura 1.2 se muestra un ejemplo del modelo en red.

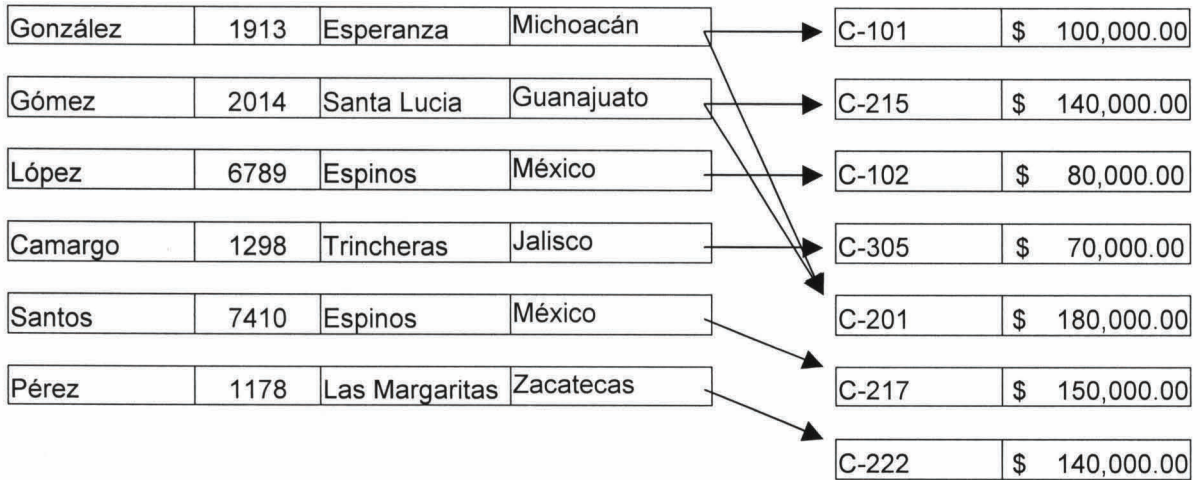


Figura 1.2 Ejemplo del modelo en red.

4.5.3.3 MODELO RELACIONAL DE DATOS.

Es el más reciente de estos modelos, supera algunas de las limitaciones de los otros dos anteriores. El modelo relacional de datos representa todos los datos en la base de datos como sencillas tablas de dos dimensiones llamadas relaciones. Las tablas son semejantes a los archivos planos, pero la información en más de un archivo puede ser fácilmente extraída y combinada.

Debido a esto es tomado este modelo en la realización del presente sistema.

El proceso de formulación de las entidades y sus relaciones en formatos de tabla usando los conceptos relacionales se llama proceso de normalización. La normalización se realiza ya que este proceso ataca la inconsistencia dentro de las bases de datos, evita las redundancias y mejora el mantenimiento.

Una vez normalizada una tabla los registros no contienen grupos repetitivos, todos los datos son completamente dependientes de la llave primaria e independientes uno del otro, es un registro lógico.

Las características más importantes del modelo relacional son:

- Las bases de datos relacionales son conceptualmente muy simples y fáciles de entender.
- Las relaciones no requieren ser predefinidas, la base de datos puede transformarse para satisfacer las condiciones cambiantes.
- Las relaciones quedan implícitas en las relaciones de los datos.

Un modelo relacional es lo que el usuario ve, y no necesariamente lo que se implantará físicamente.

Debido a que no hay una dependencia de posición entre las relaciones, las consultas no tienen que reflejar ninguna estructura en especial y por lo tanto pueden ser de tipo de procedimientos no estándar de consulta.

Este modelo proporciona un grado de independencia de los datos relativamente más alta. Sin embargo, para poder usar esta propiedad del modelo relacional, el diseño de las relaciones debe de ser completo y exacto. En la Figura 1.3 se muestra un ejemplo del modelo relacional.

Nombre Cliente	NIP	Calle	Ciudad	No. Cuenta
González	1913	Esperanza	Michoacán	C-101
Gómez	2014	Santa Lucia	Guanajuato	C-215
López	6789	Espinos	México	C-102
Camargo	1298	Trincheras	Jalisco	C-305
González	1913	Esperanza	Michoacán	C-201
Santos	7410	Espinos	México	C-217
Pérez	1178	Las Margaritas	Zacatecas	C-222
Gómez	2014	Santa Lucia	Guanajuato	C-201

No. Cuenta	Saldo
C-101	\$ 100,000.00
C-215	\$ 140,000.00
C-102	\$ 80,000.00
C-305	\$ 70,000.00
C-201	\$ 180,000.00
C-217	\$ 150,000.00
C-222	\$ 140,000.00

Figura 1.3 Ejemplo del modelo relacional.

4.6 VISUAL BASIC

Visual Basic es uno de los tantos lenguajes de programación que podemos encontrar hoy en día. Dicho lenguaje nace del BASIC (Beginner's All-purpose Symbolic Instruction Code) que fue creado en su versión original en el Dartmouth College, con el propósito de servir a aquellas personas que estaban interesadas en iniciarse en algún lenguaje de programación. Luego de sufrir varias modificaciones, en el año 1978 se estableció el BASIC estándar. La sencillez del lenguaje ganó el desprecio de los programadores avanzados por considerarlo "un lenguaje para principiantes".

Primero fue GW-BASIC, luego se transformó en QuickBASIC y actualmente se lo conoce como Visual Basic y la versión más reciente es la 6 que se incluye en el paquete Visual Studio 6 de Microsoft. Esta versión combina la sencillez del BASIC con un poderoso lenguaje de programación Visual que juntos permiten desarrollar robustos programas de 32 bits para Windows. Esta fusión de sencillez y la estética permitió ampliar mucho más el monopolio de Microsoft, ya que el lenguaje sólo es compatible con Windows, un sistema operativo de la misma empresa.

Visual Basic ya no es más "un lenguaje para principiantes" sino que es una perfecta alternativa para los programadores de cualquier nivel que deseen desarrollar aplicaciones compatibles con Windows.

En este trabajo se explicarán algunos términos y/o características del mismo con la finalidad de aprender más sobre este programa y manejarlo con facilidad.

Es un lenguaje de programación que se ha diseñado para facilitar el desarrollo de aplicaciones en un entorno grafico (GUI-GRAPHICAL USER INTERFACE) Como Windows 98, Windows NT o superior.

4.6.1 CARACTERISTICAS DE VISUAL BASIC.

Diseñador de entorno de datos: Es posible generar, de manera automática, conectividad entre controles y datos mediante la acción de arrastrar y colocar sobre formularios o informes.

Los Objetos Activos son una nueva tecnología de acceso a datos mediante la acción de arrastrar y colocar sobre formularios o informes.

Asistente para formularios: Sirve para generar de manera automática formularios que administran registros de tablas o consultas pertenecientes a una base de datos, hoja de calculo u objeto (ADO-ACTIVE DATA OBJECT)

Asistente para barras de herramientas: es factible incluir barra de herramientas personalizada, donde el usuario selecciona los botones que desea visualizar durante la ejecución.

En las aplicaciones HTML: Se combinan instrucciones de Visual Basic con código HTML para controlar los eventos que se realizan con frecuencia en una pagina web.

La Ventana de Vista de datos proporciona acceso a la estructura de una base de datos. Desde esta también acceso al Diseñador de Consultas y diseñador de Base de datos para administrar los registros.

4.6.2 TIPOS DE DECLARACIÓN DE VARIABLES

Dim: Al declarar una variable con esta palabra estamos diciendo que la variable sea local al ámbito en que se declara. Puede ser dentro de un procedimiento o dentro de un formulario, de esta forma no sería accesible desde los demás procedimientos o formularios.

Public: Las variables declaradas serán públicas y podrán estar accesibles desde todos los formularios de la aplicación. Para conseguirlo tendremos que declararlas en un módulo de código, no en la sección declarations de cualquier formulario de los que conste la aplicación. Para crear un módulo de código en el menú principal de Visual Basic marcamos en INSERT/MODULE y aparecerá junto a los demás formularios de la ventana de proyecto aunque con un icono distinto indicando que se trata de un módulo de código.

Static: Con esta forma de declarar variables conseguiremos que las variables locales no se creen y se destruyan al entrar y salir de los procedimientos donde fueron declaradas sino que se mantenga su valor durante todo el periodo de ejecución de la aplicación. De esta forma al entrar en algún procedimiento las variables recuerdan el valor que tenían cuando se salió de él.

4.6.3 TIPOS DE VARIABLES

TIPO	COMENTARIO
BOOLEAN	Sólo admite 2 valores TRUE o FALSE
BYTE	admite valores entre 0 y 255
INTEGER	admite valores entre -32768 y 32767
LONG	admite valores entre -2.147.483.648 y 2.147.483.647
SINGLE	admite valores decimales con precisión simple
DOUBLE	admite valores decimales de doble precisión
CURRENCY	válido para valores de tipo moneda
STRING	cadenas de caracteres
DATE	fechas, permite operar con ellas

Constante: Declaración de constantes que pueden ser usadas en cualquier punto en lugar de su valor, permitiendo cambiarlo cuando sea necesario, sin tener que cambiarlo en todos los sitios en que se utiliza. La expresión no puede utilizar llamadas a funciones, pues la constante se calcula en tiempo de compilación, no en tiempo de ejecución.

4.7 MICROSOFT ACCESS

Es un programa comercial de la empresa Microsoft Corporation, el cual es un sistema gestor de bases de datos, diseñado para ser utilizado en computadoras personales tipo PC.

Access, es una base de datos visual. Como todas las modernas bases de datos que trabajan en el entorno Windows, puede manejarse ejecutando unos cuantos clic de mouse sobre la pantalla. Access contiene herramientas de diseño y programación reservadas a los usuarios con mayor experiencia, aunque incluye bases de datos listas para ser usadas; están preparadas para tareas muy comunes, que cualquiera puede realizar en un momento determinado –ordenar libros, archivar documentación, etc.-.

El modelo de datos relacional, es el modelo utilizado por Visual Basic, Access de Microsoft y muchas otras bases de datos programables, es por esto que se decidió por usar el Access como sistema gestor de la base de datos del SAECC, además de que se encuentra implícito en el Visual Basic.

En la moderna terminología de las bases de datos, las preguntas que se hacen se denominan consultas, y la tabla que se presenta es una tabla o una vista. La diferencia entre una tabla y una vista es que la tabla está construida en la estructura de la base de datos y la vista es una forma de mirar la información que puede estar contenida en muchas tablas.

El lenguaje usual para realizar las solicitudes de las bases de datos relacionales es denominado SQL (que significa Structured Query Language o Lenguaje de Consultas Estructurado).

Access cuenta con diversas aplicaciones, tales como:

- El proceso de crear formularios, informes, páginas de acceso a datos, macros.
- Se pueden crear, ver, modificar y eliminar tablas.
- Se pueden crear, ver, modificar y eliminar procedimientos almacenados y diagramas de base de datos.
- Crear aplicaciones de cliente/servidor.
- Actualizar datos y objetos mediante el empleo del asistente para convertir a SQL Server.

4.8 ACTIVEREPORTS VERSION 2 PROFESSIONAL

A pesar de que Visual Basic cuenta con gran cantidad de controles e incluso su propio diseñador de reportes llamado CrystalReports a veces son bastante pobres, es por esto que la mayoría de las veces los programadores solucionan esas carencias desarrollando sus propios controles o usando diseñadores de reportes externos que sean compatibles con Visual Basic.

Por esta razón en la elaboración de los reportes del SAECC se ha implementado el uso del ActiveReports Versión 2 Professional.

ActiveReports combina el poder y comodidad de Microsoft Visual Basic con la avanzada tecnología el componente ActiveX Designer para dar un avanzado reporteador para desarrolladores de Visual Basic.

ActiveReports está completamente integrado al ambiente de programación del Visual Basic. Se siente y trabaja como VB.

Provee una completa y abierta arquitectura que le permite usar el código VB y el componente ActiveX en sus reportes, por eso usted puede lograr la elaboración de reportes sin límites.

También incluye un Wizard que le permite guiarlo en la creación de reportes sencillos sin escribir código alguno. El Wizard está integrado al ambiente de Visual Basic en forma de un módulo. Usted podrá adicionar cualquier tipo de control, gráfico o documentos a sus reportes e interactuar directamente con los datos de las base de datos.

CAPITULO 5

ESTRUCTURA DE LA BASE DE DATOS

5. ESTRUCTURA DE LA BASE DE DATOS

Las siguientes son las tablas de la Base de Datos del Sistema Integral de Calzado:

BANCOS		
CAMPO	TIPO	TAMAÑO
Bancos	Text	50
Bancos_ID	Long	4

ESTADOS		
CAMPO	TIPO	TAMAÑO
Estados	Text	30
Estados_ID	Long	4

FORMA DE PAGO		
CAMPO	TIPO	TAMAÑO
FormaPag	Text	15
FormaPag_ID	Long	4

IDS		
CAMPO	TIPO	TAMAÑO
TbTablas	Text	30
Ids	Long	4

COLORES		
CAMPO	TIPO	TAMAÑO
Colores	Text	25
Col_ID	Long	4

ESTILOS		
CAMPO	TIPO	TAMAÑO
Estilos	Long	4
Estilos_ID	Text	30

LINEAS		
CAMPO	TIPO	TAMAÑO
Lineas	Currency	8
Lineas_ID	Long	4

MARCAS		
CAMPO	TIPO	TAMAÑO
Marcas	Text	25
Marcas_ID	Long	4
Iniciales	Text	5

MATERIAL		
CAMPO	TIPO	TAMAÑO
Material	Text	50
Material_ID	Long	4

SEXO		
CAMPO	TIPO	TAMAÑO
Sexo	Text	10
Sexo_ID	Long	4
Iniciales	Text	5

VALORES DEFAULT		
CAMPO	TIPO	TAMAÑO
Identificador	Long	4
Descripcion	Text	50
Valor	Currency	8

FORMA DE CREDITO		
CAMPO	TIPO	TAMAÑO
Semanal	Text	7
Mensual	Text	7

CLIENTES		
CAMPO	TIPO	TAMAÑO
CveClientes	Text	4
Folio	Text	4
Fecha	Date	8
Ciudad	Text	20
RFC	Text	20
Colonia	Text	30
TelePart	Text	20
TeleNeg	Text	20
Pais	Text	15
Observaciones	Memo	0
NomCliente	Text	50
Calle	Text	30
CP	Text	15
Estados_ID	Long	4
Clientes_ID	Single	4
Estados	Text	30

EMPLEADOS		
CAMPO	TIPO	TAMAÑO
FechaNacimiento	Date	8
Ciudad	Text	15
Pais	Text	15
Sueldo	Currency	8
Puesto	Text	15
Observaciones	Memo	0
FechaIngreso	Date	8
CveEmpleado	Text	8
Calle	Text	30
Colonia	Text	30
Telefono	Text	15
CP	Text	15
Nombre	Text	50
Empleado_ID	Single	4
TelEmergencia	Text	20
TelRef1	Text	20
TelRef2	Text	20
Referencia1	Text	50
Referencia2	Text	50
Estado	Long	4

INVENTARIO		
CAMPO	TIPO	TAMAÑO
Existencias	Single	4
CostUnit	Currency	8
CostMen	Currency	8
Ubicación	Text	30
FechaIngreso	Date	8
StockMin	Single	4
StockMax	Single	4
CostMay	Currency	8
Observaciones	Memo	0
NvaEntrada	Double	8
MedidaNumero	Currency	8
DIIMarcas_ID	Long	4
DIICol_ID	Long	4
DIIEstilos_ID	Long	4
DIILineas_ID	Long	4
DIIMaterial_ID	Long	4
DIISexo_ID	Long	4
DIIModelo_ID	Double	8
Ventas_ID	Single	4

SEGURIDAD		
CAMPO	TIPO	TAMAÑO
ID	Long	4
Nombre	Text	50
Usuario	Text	20
Contraseña	Text	20
Contraseña1	Text	21
BotProv	Boolean	1
BtInv	Boolean	1
BtCompras	Boolean	1
BtFac	Boolean	1
BtCli	Boolean	1
BtVts	Boolean	1
BtCat	Boolean	1
BtHist	Boolean	1
BtSeg	Boolean	1
BtDev	Boolean	1
BtEmp	Boolean	1
BtIniArch	Boolean	1
BtResplnf	Boolean	1
BtReclnf	Boolean	1
BtEdicion	Boolean	1
BtRuta	Boolean	1
TxExist	Boolean	1

COMPRAS		
CAMPO	TIPO	TAMAÑO
Observaciones	Memo	0
CveProveedores	Text	4
FechCompra	Date	8
Folio	Text	4
NoFac	Double	8
CondHist	Byte	1

DETALLE COMPRAS		
CAMPO	TIPO	TAMAÑO
DIINoPares	Double	8
DIIPrecUni	Currency	8
DIIPecMen	Currency	8
DIIPerMay	Currency	8
DIIImporte	Currency	8
DIIModelo	Double	8
DII_ID	Long	4
DIIFoIFac	Text	4
DIIDescripcion	Text	50
DIIMarcas_ID	Long	4
DIICol_ID	Long	4
DIIEstilos_ID	Long	4
DIILineas_ID	Long	4
DIIMaterial_ID	Long	4
DIISexo_ID	Long	4
DIIClaveCalzado	Text	20
Ventas_ID	Single	4

VENTAS		
CAMPO	TIPO	TAMAÑO
FechaVenta	Date	8
Observaciones	Memo	0
Descuento	Currency	8
NoVentas	Text	4
CveProveedores	Text	4
CveEmpleado	Text	8
FormaPago	Text	15
Empleado_ID	Single	4
NombEmp	Text	50
CheckCancel	Boolean	1
CheckFact	Boolean	1
IVA	Currency	8
ChackIVA	Boolean	1
CondHist	Byte	1
TipoCancel	Text	25

DETALLE VENTAS		
CAMPO	TIPO	TAMAÑO
DII_NoPares	Single	4
DII_costUnit	Currency	8
DII_costoPublico	Currency	8
DII_ID	Single	4
DII_FolFac	Text	4
DII_Importe	Currency	8
DII_Descripcion	Text	50
DII_Medidas	Currency	8
DII_ClaveCalzado	Text	20
DII_ClaveProveedor	Text	8

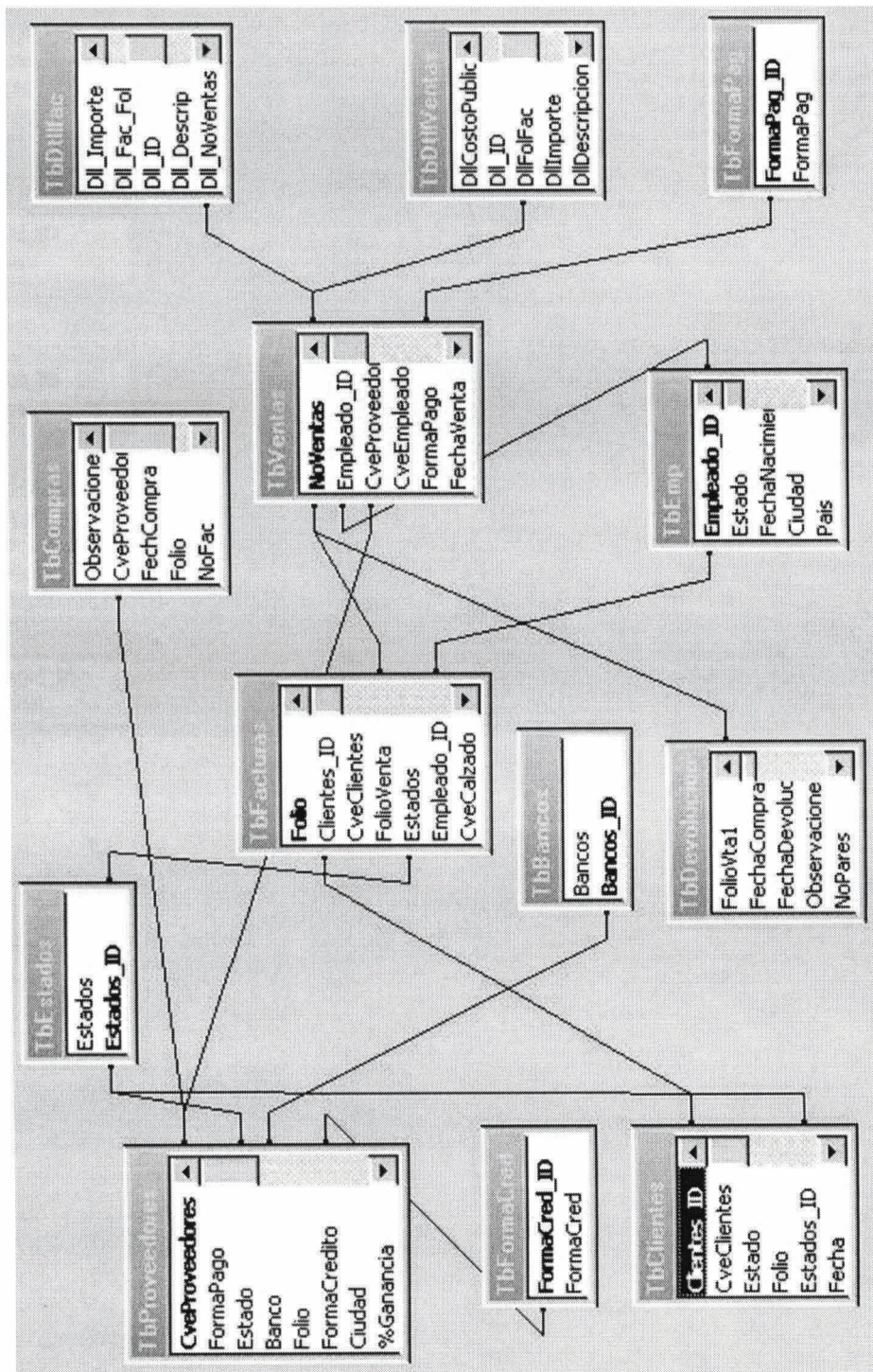
FACTURAS		
CAMPO	TIPO	TAMAÑO
Folio	Text	4
CveClientes	Text	4
Fecha	Date	8
Ciudad	Text	20
RFC	Text	20
Referencia	Double	8
PedidoNo	Single	4
Colonia	Text	30
TelePart	Text	20
TeleNeg	Text	20
Pais	Text	15
Atendido	Text	15
Observaciones	Memo	0
NombCliente	Text	50
Calle	Text	30
CP	Text	15
Estados	Text	30
Empleados_ID	Single	4
Clientes_ID	Single	4
CveCalzado	Text	20
NombEmp	Text	50
FolioVenta	Text	4
CheckCancel	Boolean	1
CondHist	Byte	1
TipoCancel	Text	25

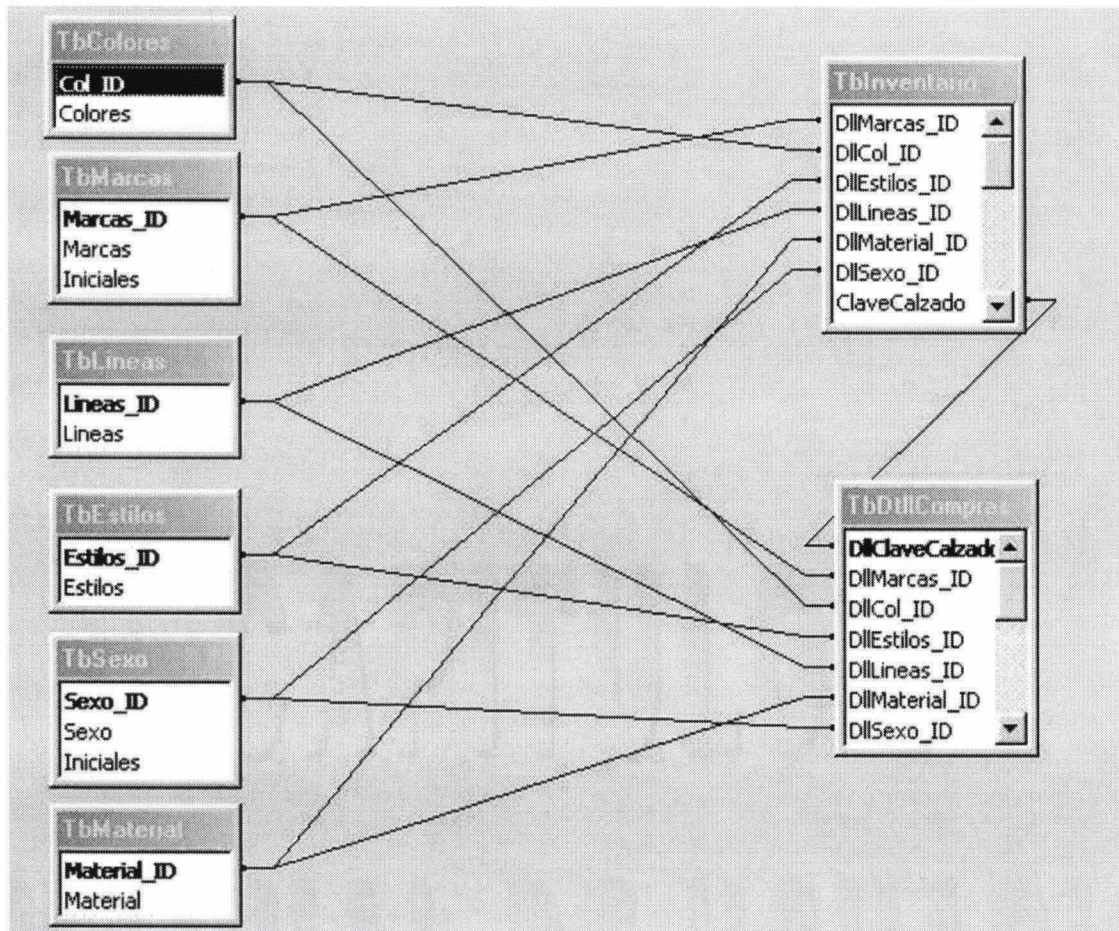
DETALLE FACTURAS		
CAMPO	TIPO	TAMAÑO
DIICantidad	Single	4
DIIPrecUnit	Currency	8
DIIImporte	Currency	8
DIIFac_Fol	Text	4
DII_ID	Long	4
DIIDescrip	Text	50
DIINoVentas	Text	4
DIIVentas_ID	Single	4
DIIClaveCalzado	Text	20
DIICostPub	Currency	8

DEVOLUCION		
CAMPO	TIPO	TAMAÑO
FechaCompra	Date	8
FechaDevolucion	Date	8
Observacion	Memo	0
NoPares	Long	4
DLL_ID	Single	4
FolioVta1	Text	10
ClaveCalz1	Text	20
ParesCamb1	Long	4
DescCalz1	Text	50
FolioVta2	Text	10
ClaveCalz2	Text	20
ParesCamb2	Long	4
DescCalz2	Text	50
CostUni2	Currency	8
CostPub	Currency	8
Importe2	Currency	8
FolioVta3	Text	10
ClaveCalz3	Text	20
ParesDevol3	Long	4
DescCalz3	Text	50
NoPars3	Long	4
ClaveProv2	Text	10
Cambio	Boolean	1
Devolucion	Boolean	1
FolioDevolID	Long	4
CondHist	Byte	1

PROVEEDORES		
CAMPO	TIPO	TAMAÑO
CveProveedores	Text	4
Folio	Text	4
FormaPago	Long	4
Ciudad	Text	15
FormaCredito	Long	15
%Ganancia	Currency	8
DescProv	Currency	8
Observaciones	Memo	0
RFC	Text	20
TelParticular	Text	15
NombreProveedor	Text	50
RazonSocial	Text	50
Calle	Text	30
Colonia	Text	30
TelNegocio	Text	15
Fax	Text	15
Pais	Text	15
Email	Text	30
Fecha	Date	8
NoCuenta	Text	15
Estado	Long	4
Banco	single	4

5.1 RELACIONES DE LAS TABLAS





CAPITULO 6

MANUAL DE USUARIO

6. MANUAL DE USUARIO

El SAECC fue diseñado con la finalidad de proporcionar al usuario una herramienta flexible e integrada que permita el control de todas las operaciones realizadas dentro de la zapatería Anylú , contemplando los siguientes módulos :

- Entradas
- Ventas
- Personal
- Devoluciones
- Muestrarios
- Archivos Históricos

Y los aspectos relacionados con estos.

Este manual tiene como propósito el de explicar de manera simple y concisa, la forma correcta de operar el SAECC. El presente documento está diseñado, pensando en que pueda ser utilizado por usuarios que tengan poca experiencia con sistemas informáticos, sin embargo siempre es recomendable que se tengan conocimientos de Windows, ya que las formas o pantallas manejadas tienen un ambiente del sistema operativo Windows.

6.1 CARACTERÍSTICAS Y REQUERIMIENTOS

El presente sistema fue diseñado pensando primordialmente en aprovechar la facilidad que en la actualidad nos brindan los equipos de cómputo para el manejo de información en forma empresarial de manera de que sea fácil, rápida y confiable. Algunas de las principales características del programa son:

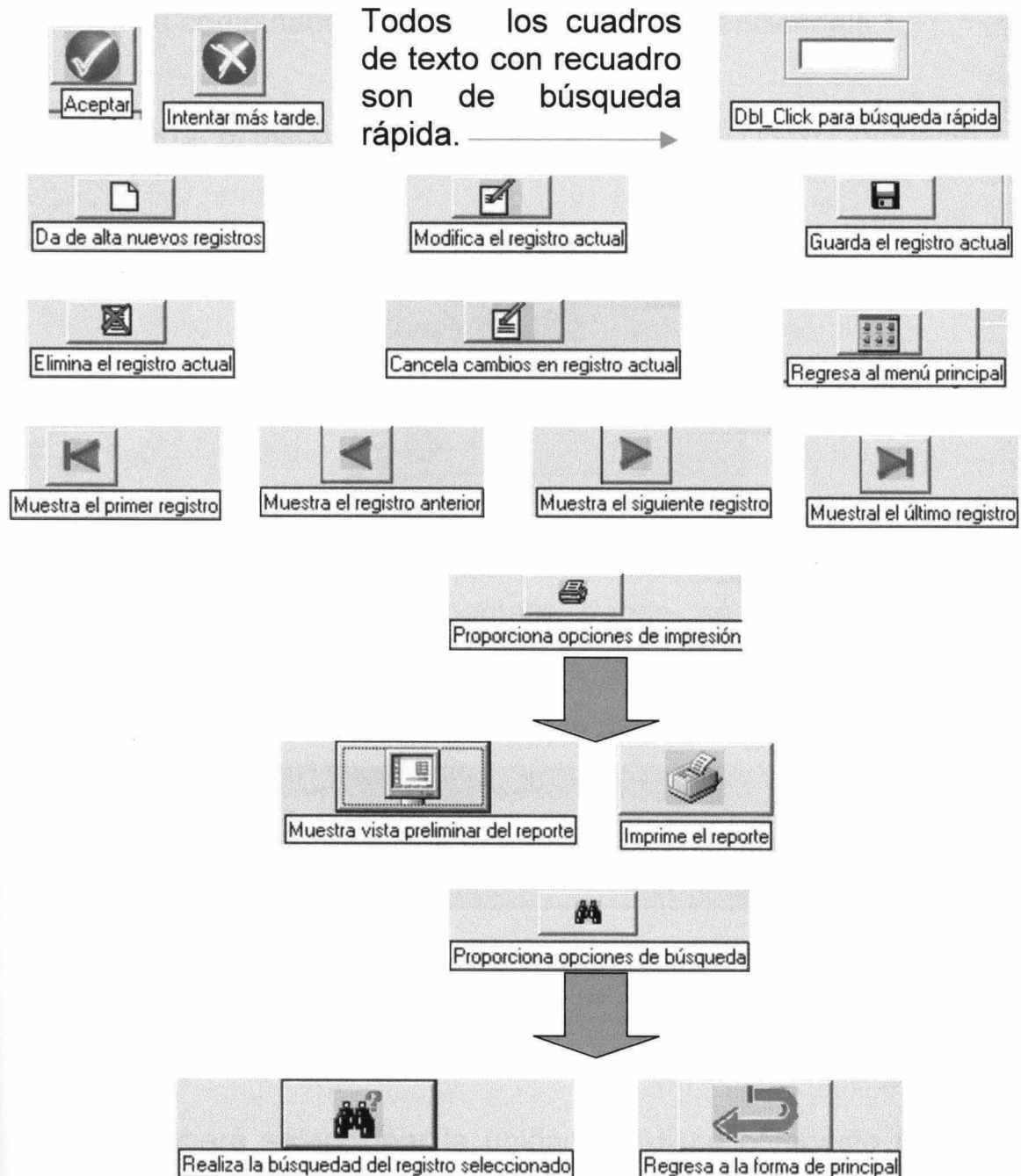
- El programa realizado fue en Visual Basic lo que permite un ambiente grafico y de simple manejo.
- Existe facilidad en saber el funcionamiento de los botones, ya que cuentan con un texto sensible al puntero del mouse y despliegan un mensaje que indica su función al posicionarse sobre ellos.
- También incluye un sistema de protección lo cual evita que el personal no autorizado tenga acceso a la información.
- Contiene un administrador del sistema que puede restringir accesos dentro del mismo sistema.

En realidad los requerimientos de hardware necesarios para el SAECC son los mínimos:

COMPONENTE	DESCRIPCION
Computadora	Procesador Pentium, Pentium I, II, III ó IV.
	Unidad de disco compacto.
	233 MHz o superior.
	32 Mb en RAM o superior.
	Monitor SVGA o de mayor resolución.
Sistema Operativo	Windows 95, 98, 2000, ME o XP.
Memoria	El sistema ocupa 20.6 Mb de espacio en HD.
Espacio Requerido	Dependerá de la capacidad de información que se almacene.

6.2 BOTONES MÁS USADOS Y SUS FUNCIONES

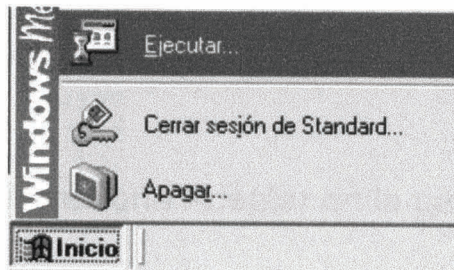
Se mostrarán los botones más usados y sus funciones contenidos a lo largo del presente sistema:



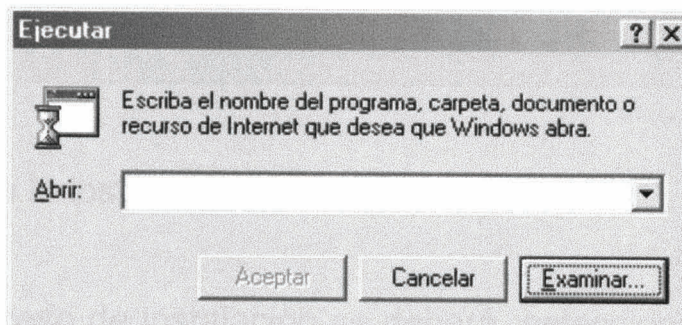
6.3 INSTALACIÓN

A continuación se presentarán los pasos de como instalar el SAECC.

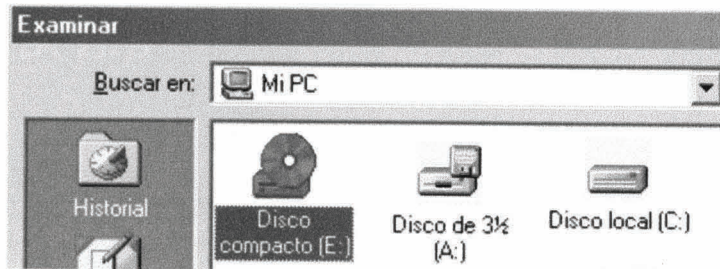
- Si su computadora se encuentra apagada enciéndala.
- Insertar el disco de SAECC en la unidad de Disco Compacto.
- Seleccione Ejecutar de la barra de Menú de inicio.



- Se desplegará una pantalla donde se pondrá el archivo ejecutable del programa (setup). Dicho archivo se encontrará mediante la opción examinar de la pantalla.



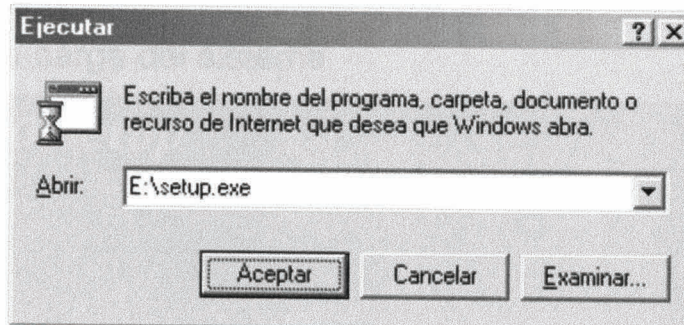
- Se deberá seleccionar la unidad de disco compacto donde se encuentra contenido el sistema y se da doble clic.



- De los cuatro íconos que aparecen seleccionar el que dice setup y presionar el botón de Abrir.

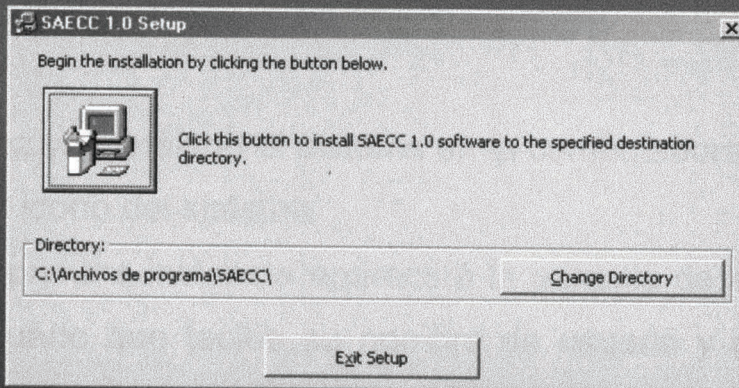


- A continuación se presiona Aceptar en la pantalla desplegada.



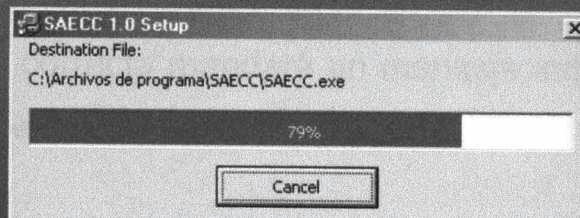
- Se iniciará la instalación del sistema en su PC.
- En el proceso de instalación se deberá seleccionar la carpeta y subcarpeta donde desee que se instale su sistema, que por default será : C:\ Archivos de programa\SAECC\ . Es opcional cambiar o no el subdirectorio.

SAECC 1.0 Setup



- Prosigue la carga del sistema.

SAECC 1.0 Setup



- El sistema lo podrá encontrar en los programas del menú de inicio de Windows.



6.4 COMO INICIAR

Una vez ya instalado el sistema en la computadora se procede a dar clic en el icono del sistema.

Como primera instancia aparecerá la pantalla de acceso y se le pedirá al usuario que teclee su nombre de usuario y su contraseña como se muestra a continuación:



Si el usuario no se encuentra dado de alta en el sistema o si la contraseña no es la correcta mandará un mensaje indicándole que es incorrecto y que debe tratar otra vez.



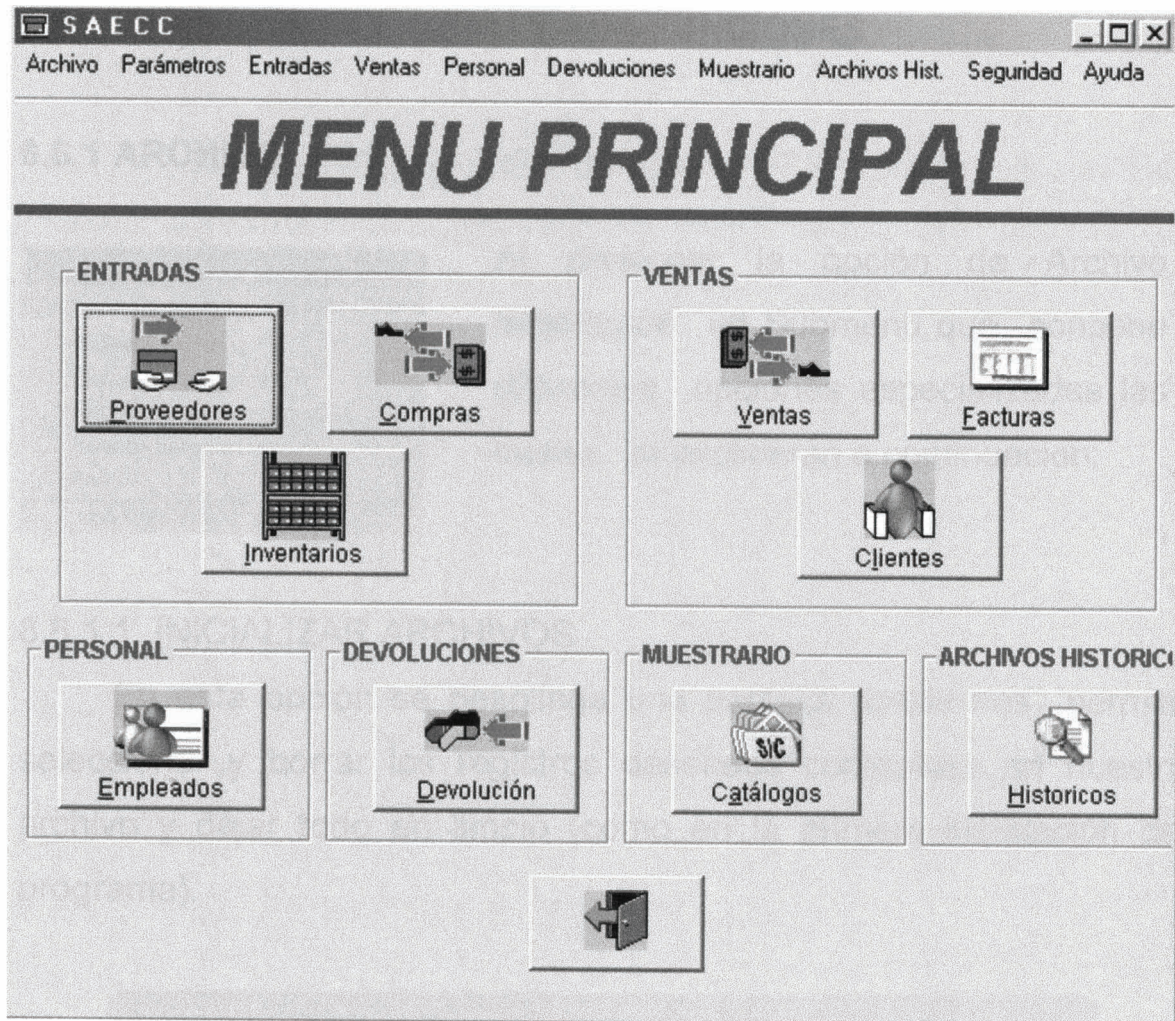
De lo contrario puede acceder de manera automáticamente al programa mostrando la pantalla de presentación del programa.

Ejemplo:



6.5 MENU PRINCIPAL

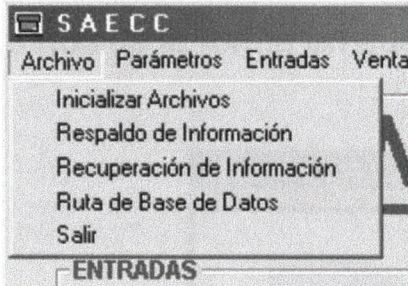
Una vez que se ha logrado tener acceso al sistema y en caso de que se tenga nivel de administrador (que será el usado a lo largo de los ejemplos) la primera pantalla mostrada al usuario es la de Menú Principal, que se mostrará a continuación así como la descripción de la forma en como se organiza la pantalla.



El Menú Principal cuenta con una barra de herramientas en la cual se encuentran opciones que tienen la misma función que los botones contenidos ahí y sólo son accesos rápidos (como Entradas, Ventas, Personal, etc.), y debido a esto sólo se mostrarán y explicarán a continuación las más importantes.

6.6 OPCIONES DEL MENU Y SUS FUNCIONES

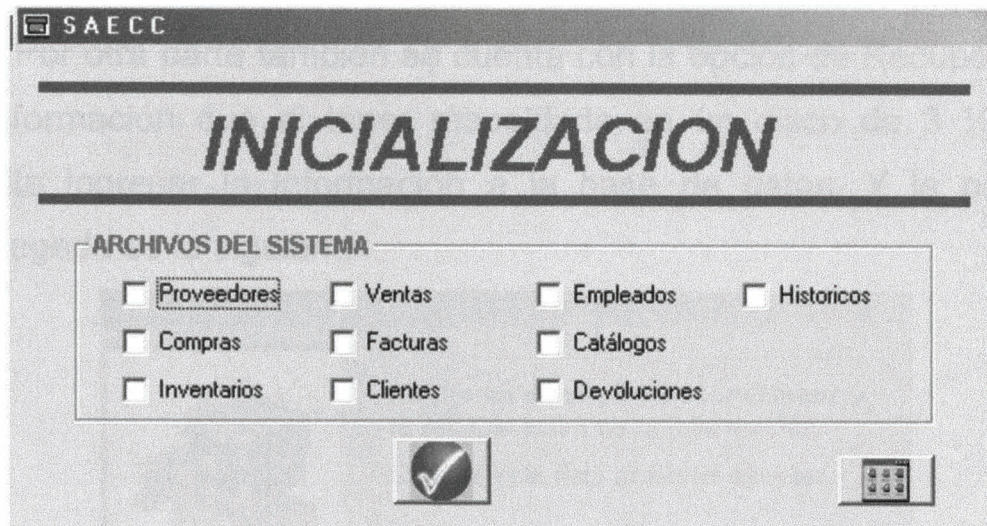
6.6.1 ARCHIVO



Al presionar la opción de Archivo desplegará un Submenú que contiene diferentes opciones especializadas las cuales se explicarán a continuación:

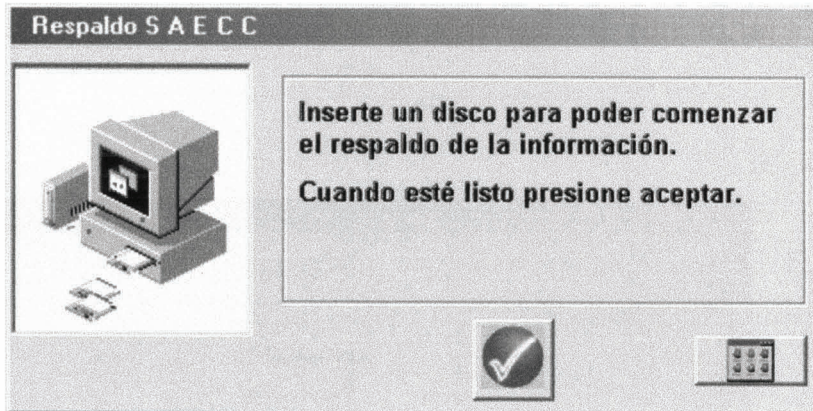
6.6.1.1 INICIALIZAR ARCHIVOS

En esta opción se despliega una pantalla donde nos permite seleccionar y borrar los registros deseados contenidos en nuestro archivo y dejar todo en limpio (como en la primera instalación del programa).



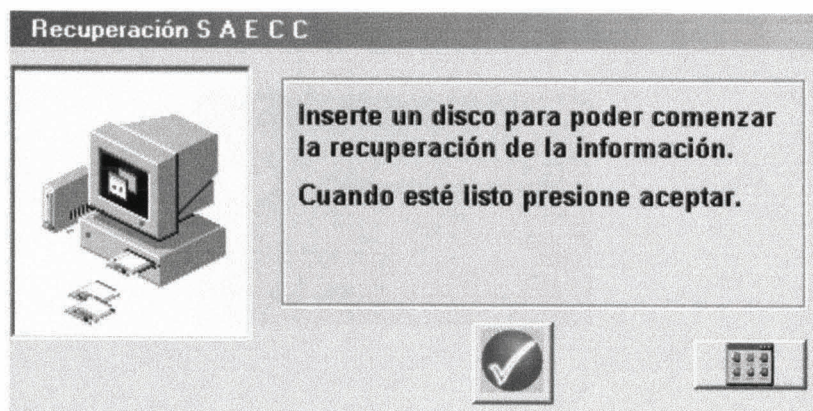
6.6.1.2 RESPALDO DE INFORMACIÓN

En esta opción nos despliega una forma donde nos permite pasar a un disco de 3 ½ toda la información contenida en la base de datos de la forma en que se presenta a continuación:



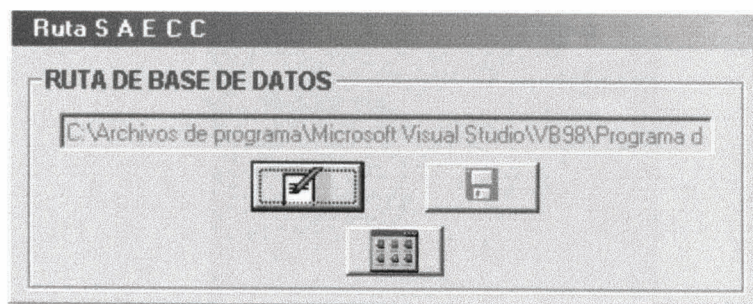
6.6.1.3 RECUPERACIÓN DE INFORMACIÓN

Por otra parte también se cuenta con la opción de Recuperación de Información que al tener respaldada en un disco de 3 ½ nos permite ingresar la información a la base de datos. Y la pantalla desplegada es la siguiente:



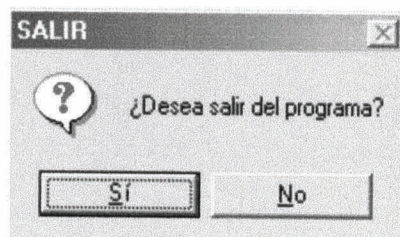
6.6.1.4 RUTA DE BASE DE DATOS

En la opción de Ruta de Base de Datos como su nombre lo indica, se proporciona la ruta donde está contenida la Base de Datos de nuestro sistema, pero se debe de tener cuidado y si es posible sólo se deberá dejar accesible al administrador del sistema, ya que si se hace algún cambio indebido el programa no funcionara de manera correcta.

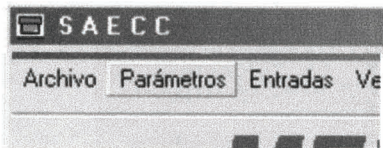


6.6.1.5 SALIR

Por último en los Submenús del Menú de Archivo tenemos la opción de salir, que al presionarla nos pregunta si se desea salir del sistema.



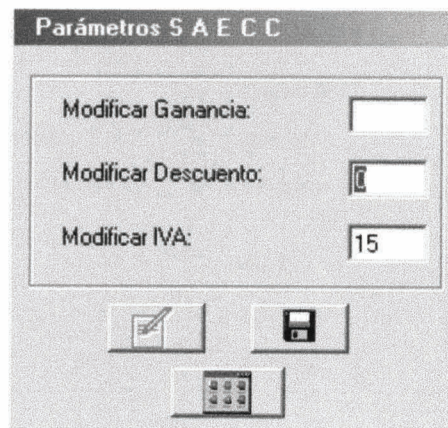
6.6.2 PARÁMETROS



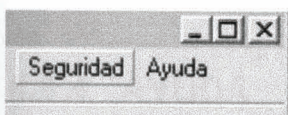
En la opción de Parámetros están contenidas precisamente algunos parámetros que actualmente se pueden

variar y es por esto que se le da la opción al usuario de modificarlos.

Parámetros tales como:



6.6.3 SEGURIDAD



Esta opción es una parte muy importante de nuestro sistema, ya que aquí es la parte

donde se dan de alta a los usuarios que van a manejar al sistema y además de que se les proporcionan o restringen derechos en el mismo programa.

SAECC

SEGURIDAD

DATOS DEL USUARIO

Nombre :

Clave de Usuario:

Contraseña :

Confirmar Contraseña:

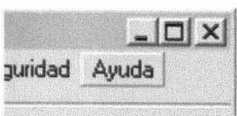
SEGURIDAD DE BOTONES

<input checked="" type="checkbox"/> Inicializar Archivos	<input checked="" type="checkbox"/> Proveedores	<input checked="" type="checkbox"/> Ventas
<input checked="" type="checkbox"/> Respaldo Información	<input checked="" type="checkbox"/> Compras	<input checked="" type="checkbox"/> Facturas
<input checked="" type="checkbox"/> Recuperación Información	<input checked="" type="checkbox"/> Inventarios	<input checked="" type="checkbox"/> Clientes
<input checked="" type="checkbox"/> Empleados	<input checked="" type="checkbox"/> Historico	<input checked="" type="checkbox"/> Ruta Base Datos
<input checked="" type="checkbox"/> Catálogos	<input checked="" type="checkbox"/> Seguridad	<input checked="" type="checkbox"/> Existencias
<input checked="" type="checkbox"/> Devoluciones	<input checked="" type="checkbox"/> Parámetros	

Navigation and System Icons:

- Navigation: Home, Previous, Next, End
- System: New File, Edit, Save, Print, Help, Exit, Refresh, Settings

6.6.4 AYUDA

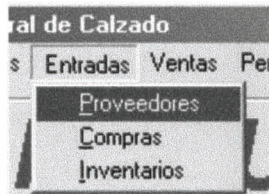


Proporcionará al usuario lo que se conoce como el Manual del Usuario. En caso de dudas en el manejo

del Sistema o alguna duda en cuanto a las funciones de las pantallas.

6.6.5 PROVEEDORES

La pantalla de Proveedores se puede acceder desde el Menú Principal mediante dos formas:



En esta pantalla de altas, bajas y cambios se concentran los proveedores y sus datos.

SAECC

PROVEEDORES

Clave Proveedor: 0001	Fecha de alta: 27/08/2002
Razon Social: <input type="text" value="DISTRIBUIDORA DE CALZADO SAN MATEO"/>	
Nombre del Proveedor: <input type="text" value="Eric Huante Heredia"/>	
Colonia: <input type="text" value="Santa Elena"/>	Tel. Negocio: <input type="text" value="017221498530"/>
Dirección: <input type="text" value="Hacienda Pasteje # 30"/>	Tel. Particular: <input type="text" value="017282851016"/>
E-mail: <input type="text" value="e_huante@hotmail.com"/>	Fax: <input type="text"/>
C.P.: <input type="text" value="5211"/>	Forma de Pago: <input type="text" value="Crédito"/>
RFC: <input type="text" value="12345EHHH"/>	Forma de Credito: <input type="text" value="Mensual"/>
Estado: <input type="text" value="México"/>	Banco: <input type="text" value="Banamex"/>
Ciudad: <input type="text" value="Toluca"/>	No. de Cuenta: <input type="text" value="54774578875676"/>
Pais: <input type="text" value="México"/>	Descuento del Prov.: % <input type="text" value="5"/>
	Porcentaje de Ganancia: % <input type="text" value="10"/>

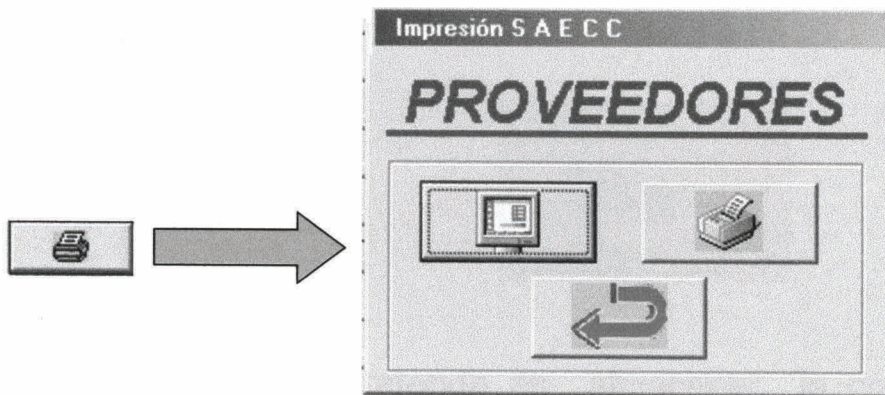
Observaciones

⏪
⏩
⏴
⏵

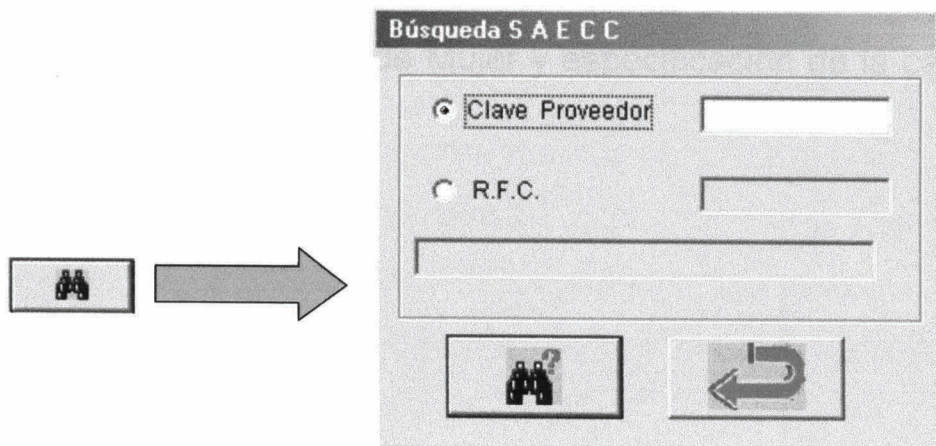
📄
✍️
💾
📄

✉️
🖨️
🔍
📱

La pantalla de Proveedores cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización.

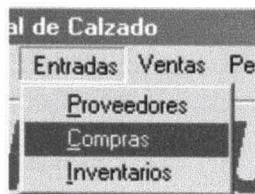


También cuenta con Búsquedas personalizadas dando opción a buscar por la Clave del Proveedor o por su R.F.C. Agilizando las búsquedas cuando se cuenta con un gran número de proveedores.



6.6.6 COMPRAS

A la pantalla de Compras se puede acceder desde el Menú Principal mediante dos formas:



Esta pantalla lo que pretende es almacenar la compra del calzado hecha al proveedor de acuerdo a la factura que nos expide y especificando el proveedor de dicha compra.

En el siguiente ejemplo ilustra que se realizó una compra de media docena del calzado (Marca: Alfa, Sexo: Mujer, Estilo: Zapatilla, Modelo: 369, Material: Charol, Color: Negro, Estilo: Números contenidos dentro de la línea de Mujer y especificación de la cantidad de cada número).

SAECC

COMPRAS





Fecha de compra: 09/09/2002


No. Factura: Clave Proveedor:


Descripción	Precio Menenudeo	Precio Mayoreo	Pares	Precio Unitario	Importe
Alfa, Mujer, Zapatilla, Charol, Negro, 22	\$494.00	\$469.30	1	\$380.00	\$380
Alfa, Mujer, Zapatilla, Charol, Negro, 22.5	\$494.00	\$469.30	1	\$380.00	\$380
Alfa, Mujer, Zapatilla, Charol, Negro, 23	\$494.00	\$469.30	1	\$380.00	\$380
Alfa, Mujer, Zapatilla, Charol, Negro, 23.5	\$494.00	\$469.30	1	\$380.00	\$380


SubTotal :
 I.V.A. :
 Total :


Importe total con letra:


























Una vez que se ha presionado el botón de registro nuevo, para dar de alta un nuevo calzado sólo se da doble clic sobre el grid o según lo que se vaya a realizar se puede posicionar sobre el grid y aparecerá un texto sensible al mouse que indicará la manera de acceder según sea la necesidad (Altas, Modificar, Eliminar).

Descripción	Precio Menenudeo	Precio Mayoreo	Pares	Precio Unitario	Importe

Altas Dbl_Click en registro vacío, Modificar Dbl_Click en registro lleno, Eliminar presionar Supr

Una vez que se ha dado el doble clic sobre el registro vacío aparecerá la siguiente pantalla.

SAECC

CAPTURA DE CALZADO

Clave Existente Clave Nueva

Clave del Calzado: Modelo:

Marca: ALFMER36922
ALFMER36922.5
ALFMER36923
Sexo: ALFMER36923.5
ALFMER36924
Estilo: ALFMER36924
ALFMER36924.5

Material:

Color:

Linea:

Cantidad de Pares: Precio de Compra:

Precio Sugerido de Menudeo: Total a Pagar:

Precio Sugerido de Mayoreo:

Descripción del Calzado :

Generar Clave

Aquí se captura el nuevo calzado que se acaba de surtir.

Esta pantalla presenta dos opciones:

- **Clave Existente:** Que es la que se maneja por default y ésta se usará cuando el calzado que se ingresa ya existe en nuestro almacén, por lo que solo se escoge la clave del calzado correcta. Se utiliza cuando se desea surtir calzado en nuestro inventario.

Se estipulará sólo la cantidad a surtir y cambiar el precio de compra en caso de que sea necesario (cualquier calzado que ya exista en inventario, a la hora de guardar aumentará las existencias en inventario).

SAECC

CAPTURA DE CALZADO

Clave Existente
 Clave Nueva

Clave del Calzado: <input type="text" value="ALFMER36922"/>	Modelo: <input type="text" value="369"/>
Marca: <input type="text" value="Alfa"/>	Material: <input type="text" value="Charol"/>
Sexo: <input type="text" value="Mujer"/>	Color: <input type="text" value="Negro"/>
Estilo: <input type="text" value="Zapatilla"/>	Linea: <input type="text" value="22"/>

Cantidad de Pares: <input type="text"/>	Precio de Compra: <input type="text" value="\$380.00"/>
Precio Sugerido de Menudeo: <input type="text" value="\$494.00"/>	Total a Pagar: <input type="text" value="\$0.00"/>
Precio Sugerido de Mayoreo: <input type="text" value="\$469.30"/>	

Descripción del Calzado :

Alfa, Mujer, Zapatilla, Charol, Negro, 22

- **Clave Nueva:** Se utiliza cuando el calzado que se compra aún no se encuentra en el inventario y por lo tanto no cuenta con una Clave de Calzado (cualquier calzado nuevo que se dé de alta en las compras se dará de alta en el inventario al guardar).

SAECC

CAPTURA DE CALZADO

Clave Existente Clave Nueva

Clave del Calzado: Modelo:

Marca: Material:

Sexo: Color:

Estilo: Línea:

Cantidad de Pares: Precio de C:

Precio Sugerido de Menudeo: Total a Pagar:

Precio Sugerido de Mayoreo:

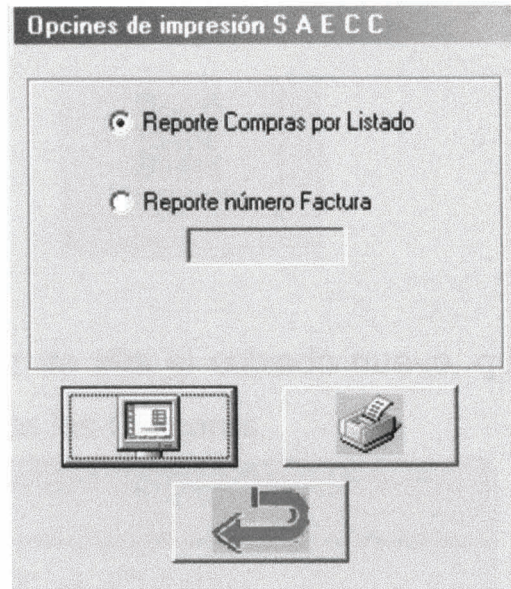
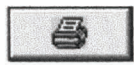
Descripción del Calzado :

22
22.5
23
23.5
24
24.5
25
25.5

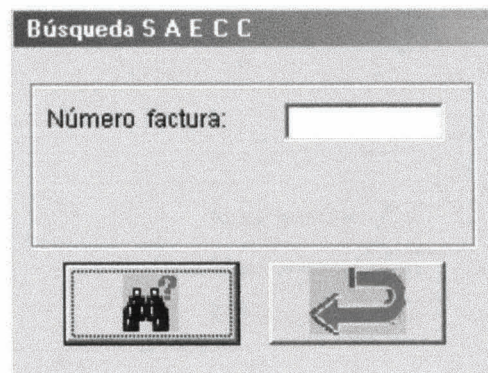
Una vez que se ha capturado todo el calzado se da aceptar y se procederá a guardar o cancelar las operaciones realizadas en la pantalla principal de Compras.

La pantalla de Compras cuenta con opciones de impresión que proporcionan al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización, además cuenta con opción a dos tipos de reporte: Reporte por Listado

(presenta un concentrado de las Compras hechas) y Reporte por Número de Factura (para imprimir una Compra específica).



También cuenta con Búsquedas personalizadas dando opción a buscar por el número de Factura, agilizando las búsquedas cuando se cuenta con un gran número de Facturas.



6.6.7 INVENTARIO

A la pantalla de Inventario se puede acceder desde el Menú Principal mediante dos formas:



En esta pantalla podemos dar de alta el calzado nuevo, que ya tenemos y que no entra por medio de las Compras.

SAECC

INVENTARIO

Clave Calzado: Fecha de Ingreso: 09/09/2002

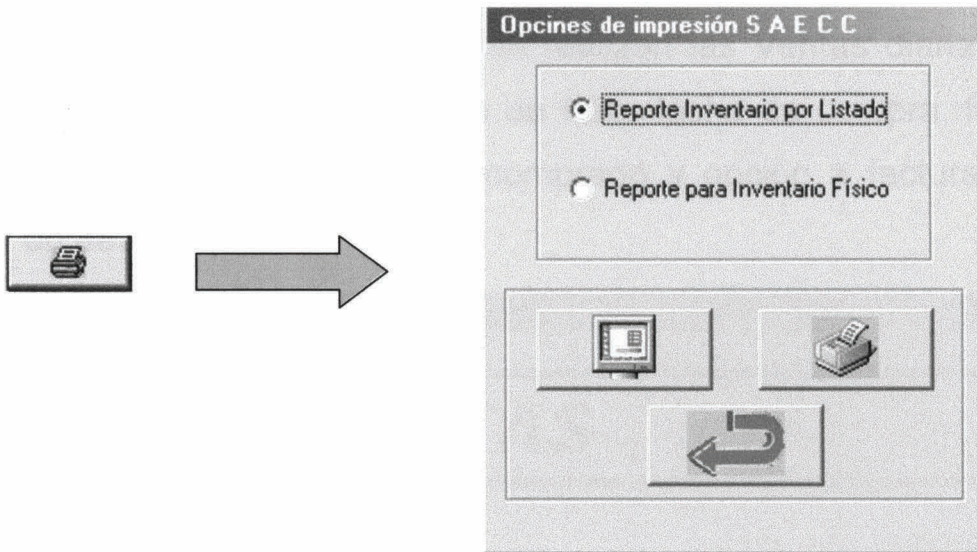
GENERADOR DE CLAVES

Marca: Modelo: Linea:
 Sexo: Material:
 Estilo: Color:

Costo Unitario: \$ Stock Mínimo:
 Costo de Menudeo: \$ Stock Máximo:
 Costo de Mayoreo: \$ Existencias:
 Ubicación:

Descripción del Calzado :

La pantalla de Inventario cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización, además cuenta con opción a dos tipos de reporte: Reporte por Listado (presenta un concentrado Inventario) y Reporte para Inventario Físico (para cotejar en Inventarios Físicos).



También cuenta con Búsquedas personalizadas dando opción a buscar por la Clave de Calzado. Agilizando las búsquedas cuando se cuenta con un gran Inventario.



Una vez que se ha presionado el botón de registro nuevo, para vender un calzado sólo se da doble clic sobre el grid o según lo que se vaya a realizar se puede posicionar sobre el grid y aparecerá un texto sensible al mouse que indicará la manera de acceder según la necesidad (Altas, Eliminar).

Clave Proveedor	Descripción	Costo Unitario	Pares	Costo Publico	Importe

Altas Dbl_Click en registro vacio, Eliminar presionar Supr

Una vez que se ha dado el doble clic sobre el registro vacío aparecerá la siguiente pantalla.

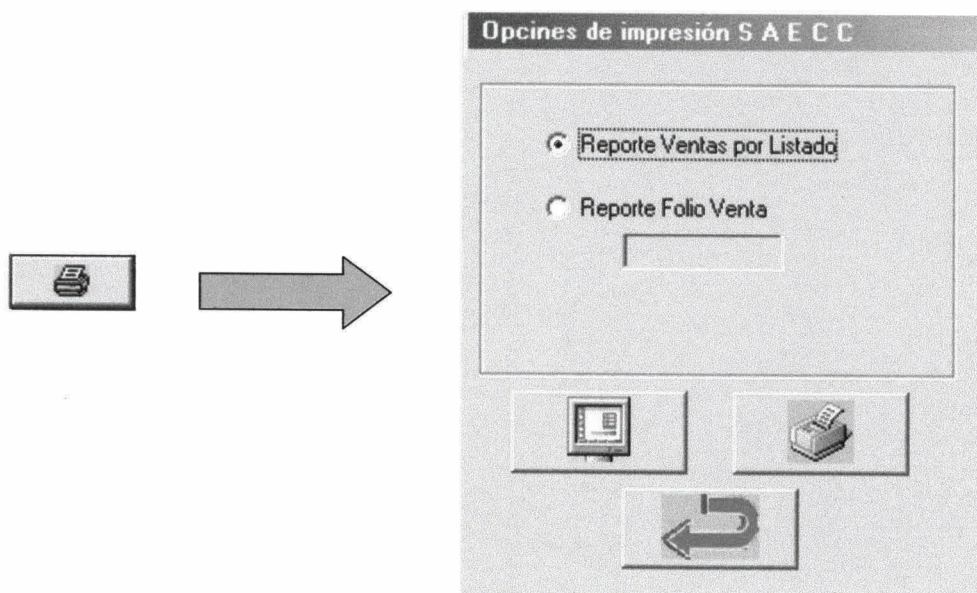
SAECC

CAPTURA VENTAS

Clave Calzado:	<input type="text" value="FLEHRE34625"/>	Costo Unitario:	<input type="text" value="\$250.00"/>
Cantidad de Pares:	<input type="text" value="1"/>	Costo al Publico:	<input type="text" value="\$300.00"/>
Clave Proveedor:	<input type="text"/>	Importe	<input type="text" value="300"/>
Descripción del Calzado:	<input type="text" value="0001"/> <input type="text" value="0002"/>		
<input type="text" value="Flexi, Hombre, Casual, Piel, Gato, 25.5"/>			

Una vez que se ha capturado todo el calzado se da aceptar y se procederá a guardar o cancelar las operaciones realizadas en la pantalla principal de Ventas.

La pantalla de Ventas cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización, además cuenta con opción a dos tipos de reporte: Reporte por Listado (presenta un concentrado de las ventas hechas) y Reporte por Folio de Venta (para imprimir una Venta específica).





También cuenta con búsquedas personalizadas dando opción a buscar por el número de Factura, agilizando las búsquedas cuando se cuenta con un gran número de ventas.



Búsqueda S A E C C

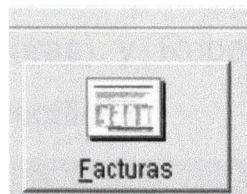
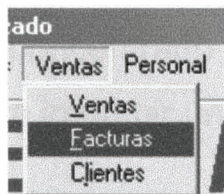
Folio venta:

A screenshot of a software window titled "Búsqueda S A E C C". The window has a light grey background. At the top, the title is in a dark grey bar. Below the title, there is a label "Folio venta:" followed by a white rectangular text input field. At the bottom of the window, there are two square buttons. The left button contains a black icon of two people with a question mark above them. The right button contains a black icon of a curved arrow pointing to the left.

6.6.9 FACTURAS

A la pantalla de facturas se puede acceder desde el Menú Principal mediante dos formas:



Por medio de esta pantalla se realizan las facturas diarias del calzado, previamente estipuladas para facturar en las ventas.

SAECC

FACTURAS

Folio Factura: 0001 Fecha: 09/09/2002

Atendido por: JAV01 Javier Garfias Heredia

Clave Cliente: 0001 Pedido No.: 5477

Nombre Cliente: Edgar Rafael Contreras Heredia

Colonia: Félix Ireta Tel. Negocio: 014433164633

Calle: Tanganxoan # 887 Tel. Particular: 014433164633

C.P.: 58070 Estado: Michoacán

RFC: EGHM827824 Ciudad: Morelia

Folio Venta: 0001 País: México

Descripción	Pares	Precio Publico	Importe
Flexi, Hombre, Casual, Piel, Cafe, 25.5	1	\$300.00	\$300.00

SubTotal : \$300.00

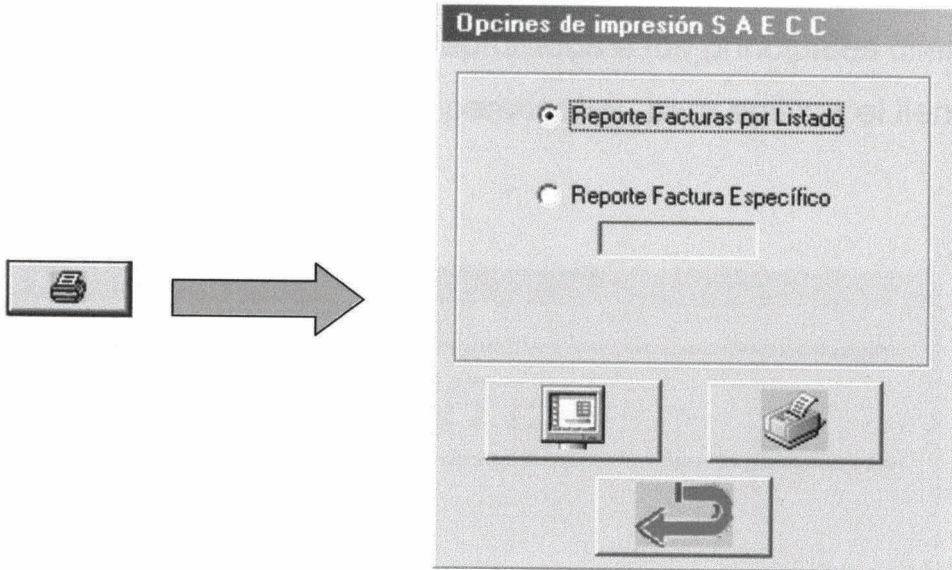
Descuento: \$ 5

Subtotal Descuento: \$295.00

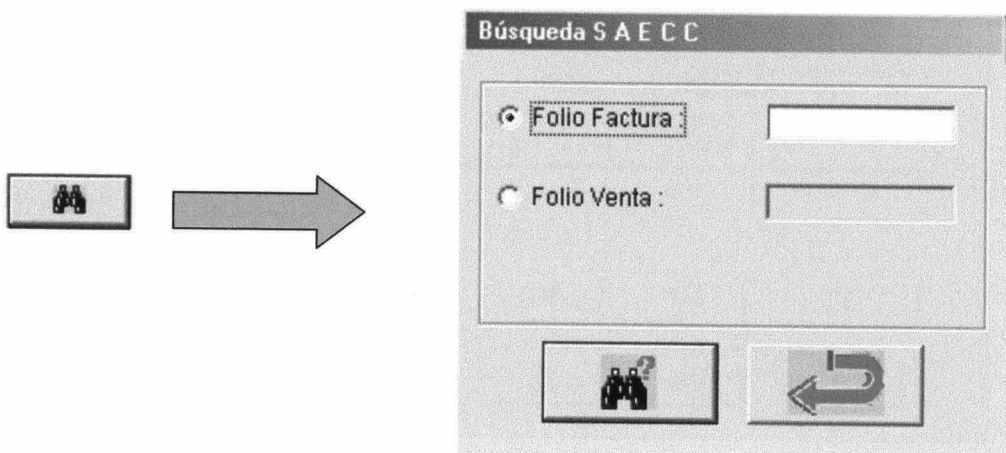
Importe total con letra I.V.A. : \$44.25

Trescientos treinta y nueve pesos 25/100 M.N. Total : \$339.25

La pantalla de facturas cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización, además cuenta con opción a dos tipos de reporte: Reporte por Listado (Presenta un concentrado del las facturas) y Reporte de Factura Específico (para imprimir una factura específica).

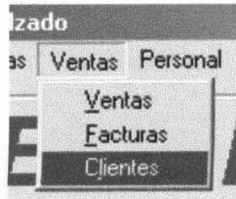


También cuenta con búsquedas personalizadas dando opción a buscar por el folio de Factura o por folio de venta. Agilizando las búsquedas cuando se cuenta con un gran número de facturas.



6.6.10 CLIENTES

A la pantalla de clientes se puede acceder desde el Menú Principal mediante dos formas:



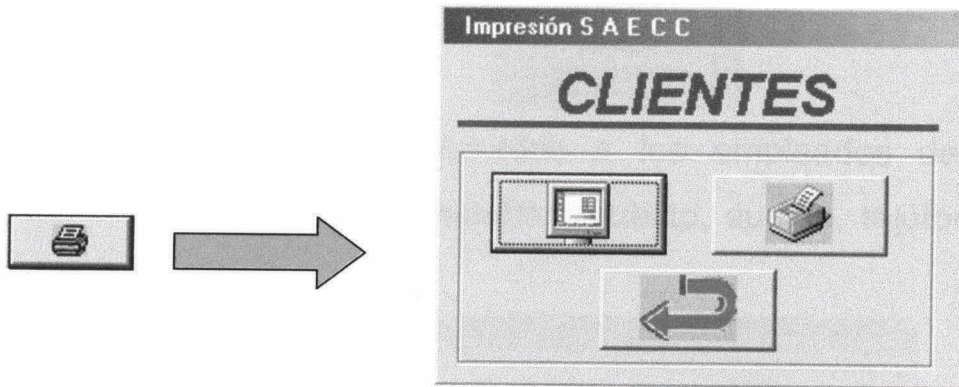
En esta pantalla se tendrán capturados a los principales clientes de la zapatería, así como sus datos personales, para agilizar el llenado de las Facturas.

CLIENTES

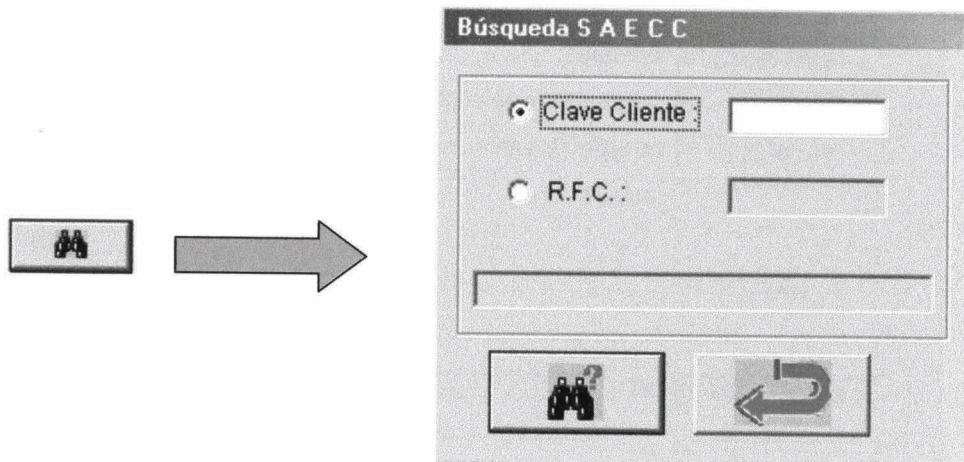
Clave Cliente:	0001	Fecha:	09/09/2002
Nombre del Cliente:	Edgar Rafael Contreras Heredia		
Colonia:	Félix Ireta	Tel. Negocio:	014433164633
Dirección:	Tanganxoan # 887	Tel. Particular:	014433164633
C.P.:	58070	Estado:	Michoacán ▾
RFC:	EGHM827824	Ciudad:	Morelia
		Pais:	México

Observaciones

La pantalla de clientes cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización.

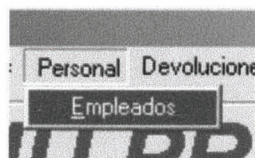


También cuenta con búsquedas personalizadas dando opción a buscar por la clave del cliente o por su R.F.C. . Agilizando las búsquedas cuando se cuenta con un gran número de clientes.



6.6.11 EMPLEADOS

A la pantalla de empleados se puede acceder desde el Menú Principal mediante dos formas:



En esta pantalla se tendrán capturados a los empleados de la zapatería, así como sus datos personales, puesto, sueldo, teléfonos de emergencia, etc.

SAECC

EMPLEADOS

DATOS PERSONALES

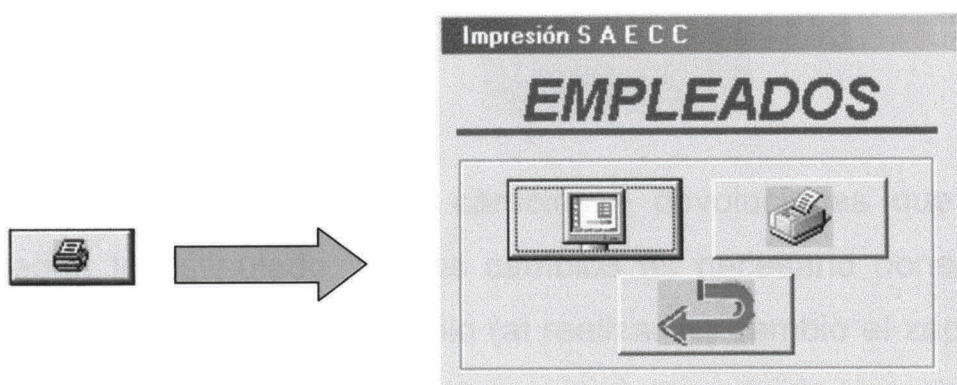
Clave Empleado: <input type="text" value="JAV01"/>	Fecha de Contratación: 9/09/2002
Nombre: <input type="text" value="Javier Garfias Heredia"/>	
Dirección: <input type="text" value="Arriaga Rivera # 58"/>	Puesto: <input type="text" value="Mostrador"/>
Colonia: <input type="text" value="Col. Centro"/>	Telefono: <input type="text" value="014363632733"/>
C.P. <input type="text" value="58600"/>	Telefono de emergencia: <input type="text" value="014363636150"/>
Estado: <input type="text" value="Michoacán"/>	Fecha de Nacimiento: <input type="text" value="07/04/1982"/>
Ciudad: <input type="text" value="Morelia"/>	Sueldo: <input type="text" value="700"/>
Pais: <input type="text" value="México"/>	

REFERENCIAS

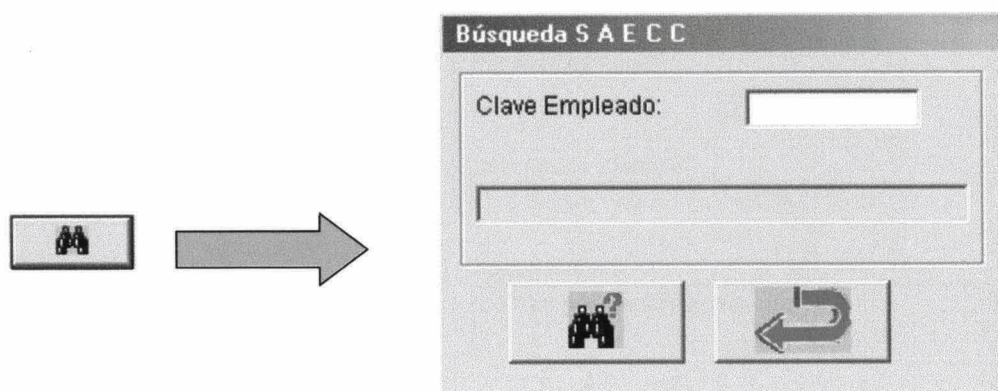
Referencia 1 <input type="text" value="Jose Luis Navarrete Arevalo"/>	Tel: <input type="text" value="014363635289"/>
Referencia 2 <input type="text" value="Alvaro Martinez Martinez"/>	Tel: <input type="text" value="014363631685"/>

Observaciones:

La pantalla de empleados cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización.

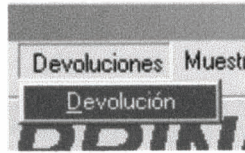


También cuenta con búsquedas personalizadas dando opción a buscar por la clave del empleado. Agilizando las búsquedas cuando se cuenta con un gran número de Empleados.



6.6.12 DEVOLUCIONES

A la pantalla de devolución se puede acceder desde el Menú Principal mediante dos formas:



Aquí se realizan todos los cambios y devoluciones que se realizan según lo estipulado. En los cambios es necesario poner el calzado por el cual se hará el cambio (al realizar un cambio el zapato que se cambió se aumenta en inventario y se descuenta de la venta).

SAECC

DEVOLUCIONES

Fecha Devolución: 09/09/2002

Fecha Compra: 09/09/2002

Calzado a cambiar:

Folio Venta:

Clave Calzado:

Pares a Cambiar:

Número de Pares:

Descripción Calzado:

Calzado a cambio:

Folio Venta:

Clave Calzado:

Cantidad de Pares:

Clave Proveedor:

Descripción Calzado:

Devolución

Folio Venta:

Clave Calzado:

Pares a Devolver:

Número de Pares:

Descripción Calzado:

Observaciones:

En caso de que se haga una devolución, entonces se selecciona la opción de devolución (en una devolución el zapato se descuenta de la venta, pero ya no entra en inventario, ya que es devuelto por defecto.)

SAECC

DEVOLUCIONES

Cambio

Fecha Devolución: 09/09/2002

Fecha Compra: 09/09/2002

Calzado a cambiar:

Folio Venta:

Clave Calzado:

Pares a Cambiar:

Número de Pares:

Descripción Calzado:

Calzado a cambio:

Folio Venta:

Clave Calzado:

Cantidad de Pares:

Clave Proveedor:

Descripción Calzado:

Devolución

Folio Venta:

Clave Calzado:





Pares a Devolver:





Número de Pares:





Descripción Calzado:

Observaciones:

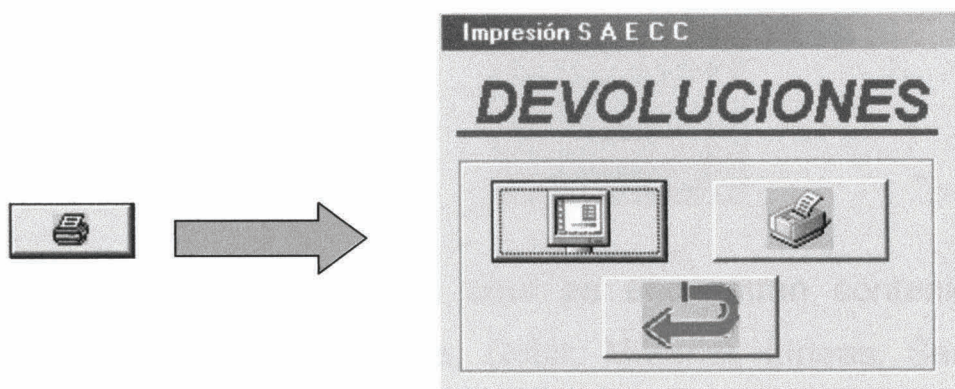
El charol estaba partido.

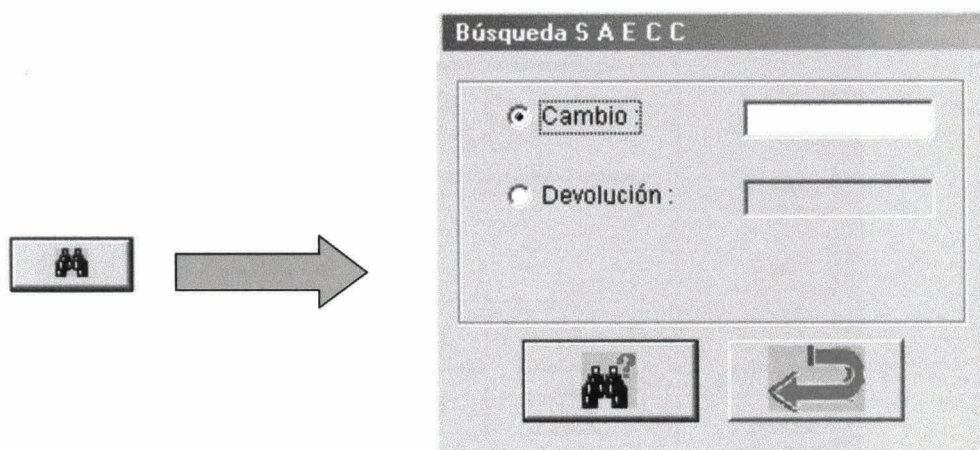





La pantalla de devoluciones cuenta con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización.

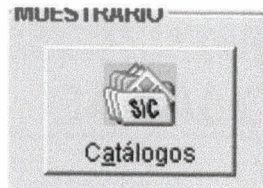


También cuenta con búsquedas personalizadas dando opción a buscar por cambio o devolución. Agilizando las búsquedas cuando se cuenta con un gran número de cambios o devoluciones.



6.6.13 CATALOGOS

A la pantalla de Catálogos se puede acceder desde el Menú Principal mediante dos formas:

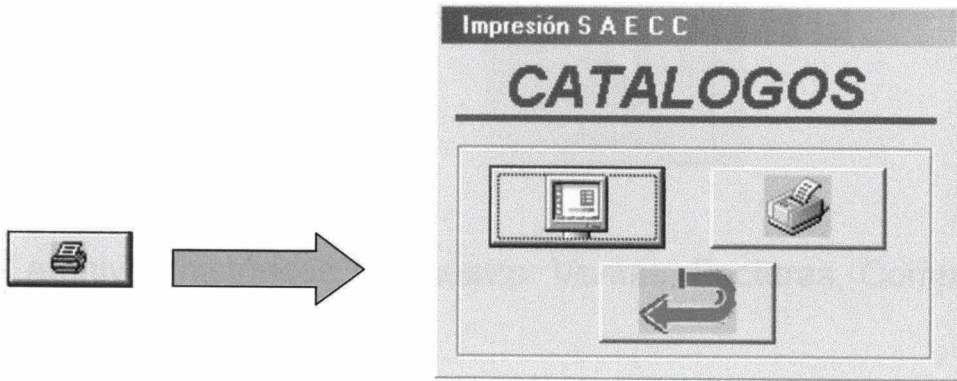


Como su nombre lo indica aquí se encuentran contenidos diversos catálogos (Marcas, Estilos, Color, Material, Líneas, Sexo), que en su conjunto conforman la descripción del calzado.

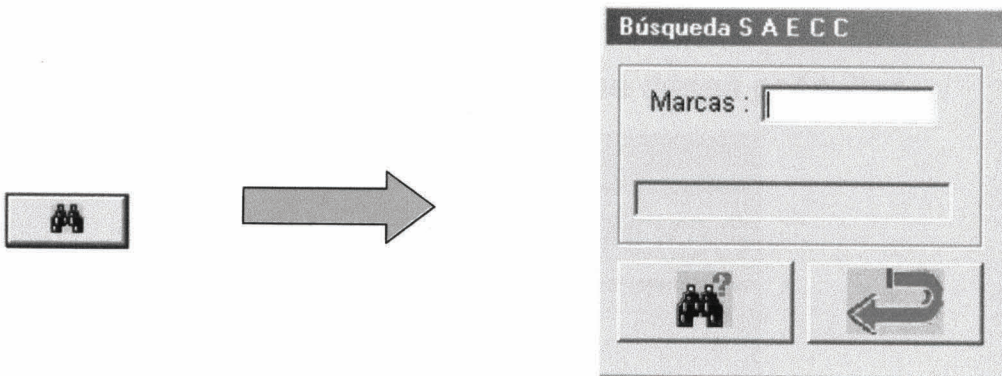
En este caso se mostrará el catálogo de marcas, que es el que aparece por default. Los demás catálogos pueden ser accedidos presionando el deseado.



Todas las pantallas de catálogos cuentan con opciones de impresión que proporciona al usuario la oportunidad de visualizar el reporte antes de imprimirlo o mandar a impresión directa sin previsualización.

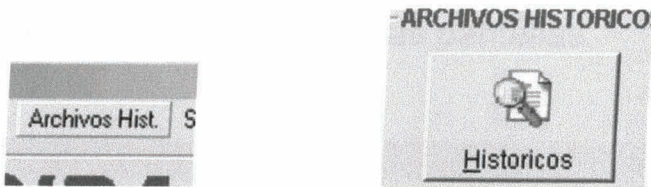


También cuentan con búsquedas personalizadas dando opción a buscar por cada uno de los catálogos existentes. Agilizando las búsquedas .



6.6.14 HISTORICOS

A la pantalla de históricos se puede acceder desde el Menú Principal mediante dos formas:

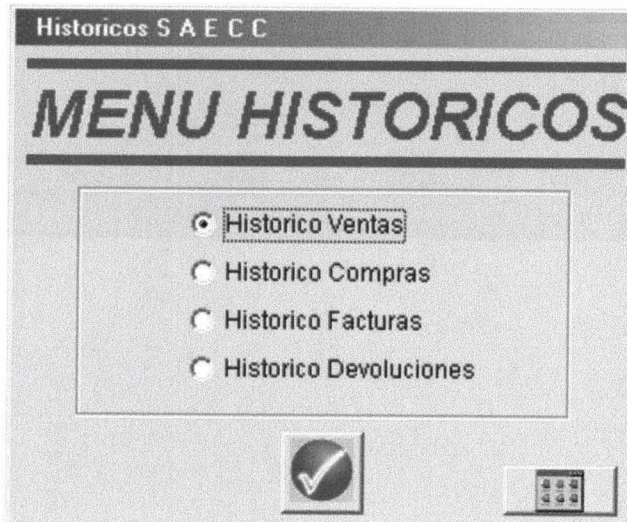


Los archivos históricos son cuatro: Ventas, Facturas, Compras y Devoluciones.

Estas pantallas tienen dos funciones, la primera que pueden ser archivos de consulta en caso de que el usuario desee ver las Ventas (que también podrían ser Compras, Facturas o Devoluciones) contenidas en un periodo determinado de tiempo.

La segunda para sacar de las tablas principales registros ya muy viejos y así evitar el acumulamiento masivo de información.

Al acceder a los archivos históricos, se desplegará una pantalla de opción donde se escogerá cuál histórico se desea consultar.



En realidad son pantallas de consultas, por lo que no cuentan con altas, bajas, cambios u opciones de impresión o búsqueda. En cambio sólo se pueden filtrar fechas si se quiere consultar o pasar la información de una tabla a otra.

A continuación se presentarán las pantallas:

SAECC

FILTRADOR DE VENTAS
 Fecha del al

HISTORIAL VENTAS

Folio Venta:	Fecha de Venta:
Clave Empleado: <input style="width: 150px;" type="text"/>	<input style="width: 200px;" type="text"/>
Forma de Pago: <input style="width: 100px;" type="text"/>	Venta a Facturar <input type="checkbox"/>

Clave Proveedor	Descripción	Costo Unitario	Pares	Costo Publico	Importe

SubTotal :	\$0.00
Descuento:	\$
Subtotal Descuento:	\$0.00
I.V.A. : <input type="checkbox"/>	0
Total :	\$0.00

SAECC

FILTRADOR DE COMPRAS

Fecha del al

HISTORIAL COMPRAS

Fecha de compra:

No. Factura: Clave Proveedor:

Descripción	Precio Menenudeo	Precio Mayoreo	Pares	Precio Unitario	Importe

SubTotal :

I.V.A. :

Total :

Importe total con letra:

⏪
⏩
⏴
⏵

☐
SAECC

FILTRADOR DE FACTURAS

Fecha del al

HISTORIAL DE FACTURAS

Folio Factura: Fecha:

Atendido por:

Clave Cliente: Pedido No.:

Nombre Cliente:

Colonia: Tel. Negocio:

Calle: Tel. Particular:

C.P.: Estado:

RFC: Ciudad:

Folio Venta: Pais:

Descripción	Pares	Precio Publico	Importe

SubTotal :

Descuento: \$

Subtotal Descuento:

I.V.A. :

Total :

Importe total con letra

A diferencia de las demás pantallas, el formulario de históricos devoluciones no contiene un filtrador de fechas, debido a que al transportar las ventas a históricos, todas las devoluciones relacionadas con alguna venta se pasarán automáticamente, por lo que sólo es un pantalla de consulta.

SAECC

HISTORICO DEVOLUCIONES

Fecha de devolución:

Fecha de compra:

Cambio

Calzado a cambiar:	Calzado a cambio:
Folio de Venta: <input type="text"/>	Folio de Venta: <input type="text"/>
Clave Calzado: <input type="text"/>	Clave Calzado: <input type="text"/>
Pares a Cambiar: <input type="text"/>	Cantidad de Pares: <input type="text"/>
Número de Pares: <input type="text"/>	Clave Proveedor: <input type="text"/>
Descripcion del Calzado: <input type="text"/>	Descripcion del Calzado: <input type="text"/>

Devolución

Folio de Venta: <input type="text"/>	Observaciones: <input type="text"/>
Clave Calzado: <input type="text"/>	
Pares a Cambiar: <input type="text"/>	
Número de Pares: <input type="text"/>	
Descripcion del Calzado: <input type="text"/>	

CONCLUSIONES

CONCLUSIONES

Antes de comenzar con el desarrollo de cualquier proyecto, se conduce un estudio de sistemas para detectar todos los detalles de la situación actual de la empresa. La información reunida con este estudio sirve como base para crear varias estrategias de Diseño.

Es importante considerar que con la importancia del Sistema de Administración Empresarial para Comercializadoras de Calzado, se lograron disminuir en una forma importante muchos de los procedimientos manuales que anteriormente ocasionaban consumo de tiempo y descuido en la atención al cliente.

El Sistema de Administración Empresarial para Comercializadoras de Calzado fue diseñado pensando sobre todo en que fuera un sistema fácil de usar y que además contará con los elementos de seguridad necesarios.

Con respecto a la accesibilidad, esto se consigue al diseñar menús y pantallas sencillas que en todo momento muestran mensajes informativos de las operaciones que se realizan y los botones de rápida función.

Retomando el aspecto de la seguridad, se cuenta con dos elementos sumamente útiles y efectivos para lograrla, estos son: en primer lugar, las contraseñas para el acceso y las restricciones de

pantallas. En segundo lugar se tienen las opciones de respaldo y recuperación de la información, disponibles dentro del sistema.

Aunque el sistema fue diseñado para el uso de personas con poco conocimiento de computación, es recomendable la capacitación del personal que estará a cargo del sistema, sobre todo para evitar el mal uso del mismo.

Asignar a los usuarios claves individuales, preferentemente asignadas por el dueño del sistema, así se obtendrá un mejor control y además que éstas no sean fáciles de deducir por personas ajenas al sistema.

Es recomendable el hacer un respaldo semanal (Preferentemente) de la información contenida en nuestra base de datos, y así evitar la pérdida de información o el borrado accidental.

Lo más importante: que el usuario quede siempre satisfecho con cualquier sistema que desee, cubriendo así sus necesidades.

Como se ha visto a lo largo de este trabajo, hoy día se comienza a imponer la obligación de normas de calidad del software donde un fallo en la información, o en el tratamiento de ésta puede llevar a fallos catastróficos y de consecuencias imprevisibles.

Por ello las organizaciones están exigiendo controles de calidad más rigurosos en la confección de su software.

Hoy día el tener implantados sistemas de calidad en la empresa, debe llevar no solo el tener que instalar la metodología del sistema de calidad sino también sistemas de información que controlen y coordinen el sistema, sistemas automáticos, sistemas documentales, etc. Por todo ello la implantación de sistemas de calidad en cualquier empresa u organización debe implicar que también el software que empleen los posea, y ello repercute en la obligación de que sus proveedores de software los hayan empleado en la elaboración de sus productos.

BIBLIOGRAFIA

BIBLIOGRAFÍA

- <http://www.pegasosoft.com/curso/introduccion.htm>
- <http://www.infosistemas.com.mx/soto10.htm>
- <http://www.monografias.com>
- <http://www.activereports.com>

- HENRY F. KORTH & ABRAHAM SILBERSCHATZ, "Análisis y Diseño de Sistemas", Segunda Edición, Editorial Mc Graw Hill.

- ROGER S. PRESSMAN, "Ingeniería del Software", Cuarta Edición, Editorial Mc Graw Hill.

- LINDA GAIL & JOHN CHRISTIE, "Enciclopedia de Términos de Computación", PHH, Pentice may.

- ROGER S. PRESSMAN, "Ingeniería del Software: Un enfoque práctico", Segunda edición, Editorial McGraw Hill, 1990.

- KENDALL & KENDALL, “Análisis y Diseño de Sistemas”, Tercera Edición; Pearson Educación.

- JAMES A SENN, “Análisis y Diseño de Sistema de Información”, Editorial Mc Graw Hill, Enero 1990.

- JAMES A. SENN, “Análisis y Diseño de Sistemas de Información”, Segunda Edición, Editorial Mc Graw Hill, Abril 2000.

- MICHAEL HALVORSON, “Visual Basic 6”, Tercera Edición, Editorial Mc Graw Hill, 2000.

- L.S.C. JORGE HUMBERTO REYNAGA PEÑA, “Apuntes de Análisis y Diseño de Sistemas ”, Séptimo Semestre, UVAQ.

- Ing. SHAHROKH BAHADOR, “Apuntes de Visual Basic Avanzado”, Séptimo Semestre, UVAQ.

- HENRY F. KORTH & ABRAHAM SILBERSCHATZ, “Fundamentos de Bases de Datos”, Segunda Edición, Editorial Mc Graw Hill.

CODIGO FUENTE

CONTRASEÑA

Option Explicit

```
Private Sub BotAceptar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT ID
From TbSeg WHERE Usuario ='" & TxG(0) & "' AND
Contrasena = '" & TxG(1) & "'", dbOpenDynaset)
```

```
If Not MiRegistro.EOF Then
    MiID = MiRegistro!ID
    Contrasena = TxG(1)
    FoPortada.Show
    Pausar 0.03
    Unload FoPortada
    FoMenu.Show
    Unload Me
End If
```

```
MiRegistro.Close
MiBaseDatos.Close
End Sub
```

```
Private Sub BtAceptar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT ID
From TbSeg WHERE Usuario ='" & TxG(0) & "' AND
Contrasena = '" & TxG(1) & "'", dbOpenDynaset)
```

```
If MiRegistro.EOF Then
    MsgBox "La contraseña no es la correcta, favor de
verificar", vbCritical, "Contraseña"
    TxG(0) = ""
    TxG(1) = ""
    TxG(0).SetFocus
```

```
    TexMensaje.ForeColor = &HFF& 'Color rojo
    TexMensaje = "ACCESO DENEGADO"
```

```
    BtAceptar.Enabled = True
```

```
End If
```

```
MiRegistro.Close
MiBaseDatos.Close
End Sub
```

```
Private Sub BtCancelar_Click()
End
End Sub
```

```
Private Sub Form_Activate()
```

```
    BotAccept.Visible = False
    BotAccept.Enabled = False
    BtAceptar.Visible = True
    BtAceptar.Enabled = True
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
    BotAccept.Visible = False
    BotAccept.Enabled = False
    BtAceptar.Visible = True
    BtAceptar.Enabled = True
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
Set FoContrana = Nothing
End Sub
```

```
Private Sub TxG_GotFocus(Index As Integer)
HighLight
End Sub
```

```
Private Sub TxG_KeyPress(Index As Integer, KeyAscii As
Integer)
If KeyAscii = 13 Then
    SendKeys "{tab}"
End If
End Sub
```

```
Private Sub TxG_LostFocus(Index As Integer)
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT ID
From TbSeg WHERE Usuario ='" & TxG(0) & "' AND
Contrasena = '" & TxG(1) & "'", dbOpenDynaset) 'Costante
para abrir en modo solo lectura y si fuera dbOpendynaset era
para lectura/escritura'
```

```
If Not MiRegistro.EOF Then
    TexMensaje.ForeColor = &HFF00& 'Color verde
    TexMensaje = "ACCESO CONCEDIDO"
    BotAccept.Visible = True
    BotAccept.Enabled = True
    BotAccept.SetFocus
    BtAceptar.Enabled = False
    BtAceptar.Visible = False
End If
```

```
MiRegistro.Close
MiBaseDatos.Close
```

```
End Sub
```

BUSQUEDA CLIENTE

```
Private Sub BotBusqueda_Click()
If OptionRFC.Value = True Then
```

```
    Dim MiBD As Object
    Dim MiTabla As Object
```

```
If TexBusqueda <> "" Then
```

```
    Set MiBD = OpenDatabase(MiBDStr)
```

```

Set MiTabla = MiBD.OpenRecordset("SELECT
TbClientes.* FROM TbClientes WHERE RFC = " &
TexBusqueda & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then 'Con este EOF quiere decir que la
clausula de arriba no se encontro un registro duplicado.'

```

```

MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusqueda = ""
TexBusqueda.SetFocus

```

```

Else
FoClientes.DataClientes.Recordset.FindFirst "RFC = "
& LstClave & ""
Unload Me
End If

```

```

End If

```

```

Else

```

```

If TexCveCli <> "" Then

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbClientes.* FROM TbClientes WHERE CveClientes = " &
TexCveCli & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then 'Con este EOF quiere decir que la
clausula de arriba no se encontro un registro duplicado.'

```

```

MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCveCli = ""
TexCveCli.SetFocus

```

```

Else
FoClientes.DataClientes.Recordset.FindFirst
"CveClientes = " & LstClave2 & ""
Unload Me
End If

```

```

End If

```

```

End If

```

```

End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```

Private Sub BotRegresar_Click()
Unload Me
Set FoBuscClie = Nothing
End Sub

```

```

Private Sub BotRegresar_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```

Private Sub Form_Activate()
If OptionCveCli.Value = True Then
TexCveCli.Enabled = True
TexBusqueda.Enabled = False
TexBusqueda.BackColor = ColorGris
Else
TexCveCli.Enabled = False

```

```

TexBusqueda.Enabled = True
End If
End Sub

```

```

Private Sub Form_Load()
LlenarCombo LstClave, "TbClientes", "RFC", "Clientes_ID"
LlenarCombo LstClave2, "TbClientes", "CveClientes",
"Clientes_ID"
LstClave.Visible = False
LstClave2.Visible = False
End Sub

```

```

Private Sub LstClave_DblClick()
TexBusqueda = LstClave
LstClave.Visible = False
TexBusqueda.SetFocus
End Sub

```

```

Private Sub LstClave2_DblClick()
TexCveCli = LstClave2
LstClave2.Visible = False
TexCveCli.SetFocus
End Sub

```

```

Private Sub OptionCveCli_Click()
If OptionRFC.Value = True Then
TexBusqueda.Enabled = True
TexCveCli.Enabled = False
TexCveCli.BackColor = ColorGris
TexBusqueda.BackColor = vbWhite
Else
TexBusqueda.Enabled = False
TexCveCli.Enabled = True
TexCveCli.BackColor = vbWhite
TexBusqueda.BackColor = ColorGris
TexBusqueda = ""
TexResult = ""

```

```

End If

```

```

End Sub

```

```

Private Sub OptionCveCli_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```

Private Sub OptionRFC_Click()
If OptionRFC.Value = True Then
TexBusqueda.Enabled = True
TexCveCli.Enabled = False
TexCveCli.BackColor = ColorGris
TexBusqueda.BackColor = vbWhite
TexCveCli = ""
TexResult = ""

```

```

Else
TexBusqueda.Enabled = False
TexCveCli.Enabled = True
TexCveCli.BackColor = vbWhite
TexBusqueda.BackColor = ColorGris
End If
End Sub

```

```
Private Sub OptionRFC_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub TexBusqueda_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexBusqueda.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True

End Sub
```

```
Private Sub TexBusqueda_Db1Click()
LstClave.Visible = True
End Sub
```

```
Private Sub TexBusqueda_GotFocus()
LstClave2.Visible = False
HighLight
If TexBusqueda <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbClientes.* FROM TbClientes WHERE RFC = "" &
TexBusqueda & """, dbOpenSnapshot)
```

```
    If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusqueda = ""
TexBusqueda.SetFocus
Else
TexResult = MiTabla.NombCliente
End If
End If
End Sub
```

```
Private Sub TexBusqueda_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexBusqueda = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub
```

```
Private Sub TexCveCli_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexCveCli.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True

End Sub
```

```
Private Sub TexCveCli_Db1Click()
LstClave2.Visible = True
End Sub
```

```
Private Sub TexCveCli_GotFocus()
```

```
LstClave.Visible = False
HighLight
```

```
If TexCveCli <> "" Then
Dim MiBD As Object
Dim MiTabla As Object
```

```
Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbClientes.* FROM TbClientes WHERE CveClientes = "" &
TexCveCli & """, dbOpenSnapshot)
```

```
    If MiTabla.EOF
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCveCli = ""
TexCveCli.SetFocus
Else
TexResult = MiTabla.NombCliente
End If
End If
End Sub
```

```
Private Sub TexCveCli_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexCveCli = LstClave2
SendKeys "{tab}"
LstClave2.Visible = False
End If
End Sub
```

BUSQUEDA COLOR

```
Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexCol <> "" Then
```

```
Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbColores.* FROM TbColores WHERE Colores = "" &
TexCol & """, dbOpenSnapshot)
```

```
    If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCol = ""
TexCol.SetFocus
Else
FoColores.DataColores.Recordset.FindFirst " Colores
= "" & LstClave & ""
Unload Me
End If
End If
End Sub
```

```
Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotRegresar_Click()
Unload Me
Set FoBuscColor = Nothing
End Sub
```

```
Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Load()
LlenarCombo LstClave, "TbColores", "Colores"
LstClave.Visible = False
End Sub
```

```
Private Sub LstClave_DbClick()
TexCol = LstClave
LstClave.Visible = False
TexCol.SetFocus
End Sub
```

```
Private Sub TexCol_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCol.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True

End Sub
```

```
Private Sub TexCol_DbClick()
LstClave.Visible = True
End Sub
```

```
Private Sub TexCol_GotFocus()
HighLight
If TexCol <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbColores.* FROM TbColores WHERE Colores = " &
TexCol & """, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCol = ""
TexCol.SetFocus
Else
TexResult = MiTabla.Colores
End If
End If
End Sub
```

```
Private Sub TexCol_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexCol = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub
```

BUSQUEDA COMPRAS

```
Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexNoFac <> "" Then
```

```
Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbCompras.* FROM TbCompras WHERE NoFac = " &
TexNoFac, dbOpenSnapshot)
```

```
If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexNoFac = ""
TexNoFac.SetFocus
Else
FoCompras.DataCompras.Recordset.FindFirst "NoFac
=" & LstClave
Unload Me
End If
End If
```

```
End Sub
```

```
Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotRegresar_Click()
Unload Me
Set FoBuscComp = Nothing
End Sub
```

```
Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Load()
LlenarCombo LstClave, "TbCompras", "NoFac", "Folio"
LstClave.Visible = False
End Sub
```

```
Private Sub LstClave_DbClick()
TexNoFac = LstClave
LstClave.Visible = False
TexNoFac.SetFocus
End Sub
```

```
Private Sub TexNoFac_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexNoFac.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
If Not IsNumeric(TexNoFac) And TexNoFac <> "" Then
MsgBox "Poner solo números", vbExclamation
TexNoFac = ""
TexNoFac.SetFocus
End If
End Sub
```

```
Private Sub TexNoFac_DbClick()
LstClave.Visible = True
End Sub
```

```
Private Sub TexNoFac_GotFocus()
LstClave.Visible = False
HighLight
If TexNoFac <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbCompras.* FROM TbCompras WHERE NoFac = " &
TexNoFac, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexNoFac = ""
TexNoFac.SetFocus
End If
End If
End Sub
```

```
Private Sub TexNoFac_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexNoFac = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub
```

BUSQUEDA DEVOLUCION

```
Private Sub BotBusqueda_Click()
If OptionDevol.Value = True Then
Dim MiBD As Object
Dim MiTabla As Object
If TexBusDevol <> "" Then

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbDevolucion.* FROM TbDevolucion WHERE FolioVta3 =
"" & TexBusDevol & """, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusDevol = ""
TexBusDevol.SetFocus
Else
FoDevolucion.DataDevolucion.Recordset.FindFirst
"FolioVta3 = "" & LstClave & """
Unload Me
End If
End If
Else
If TexBusCamb <> "" Then

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbDevolucion.* FROM TbDevolucion WHERE FolioVta1 =
"" & TexBusCamb & """, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusCamb = ""
```

```
TexBusCamb.SetFocus
Else
FoDevolucion.DataDevolucion.Recordset.FindFirst
"FolioVta1 = "" & LstClave2 & """
Unload Me
End If
End If

End If

End Sub
```

```
Private Sub BotBusqueda_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub BotRegresar_Click()
Unload Me
Set FoBuscDevol = Nothing
End Sub
```

```
Private Sub BotRegresar_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Activate()
If OptionCambio.Value = True Then
TexBusDevol.Enabled = False
TexBusCamb.Enabled = True
TexBusDevol.BackColor = ColorGris
Else
TexBusCamb.Enabled = False
TexBusDevol.Enabled = True
End If
End Sub
```

```
Private Sub Form_Load()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
Dim MiRegistro2 As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
```

```
Set MiRegistro2 = MiBaseDatos.OpenRecordset("SELECT
FolioVta1,FolioDevolID From TbDevolucion WHERE
CondHist = 0 ", dbOpenSnapshot)
```

```
Do Until MiRegistro2.EOF
LstClave2.AddItem IIf(IsNull(MiRegistro2!FolioVta1), "",
MiRegistro2!FolioVta1)
LstClave2.ItemData(LstClave2.NewIndex) =
MiRegistro2!FolioDevolID
MiRegistro2.MoveNext
Loop
```

```
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
FolioVta3,FolioDevolID From TbDevolucion WHERE
CondHist = 0 ", dbOpenSnapshot)
```

```
Do Until MiRegistro.EOF
LstClave.AddItem IIf(IsNull(MiRegistro!FolioVta3), "",
MiRegistro!FolioVta3)
```

```

LstClave.ItemData(LstClave.NewIndex) =
MiRegistro!FolioDevolID
MiRegistro.MoveNext
Loop

```

```

LstClave.Visible = False
LstClave2.Visible = False
End Sub

```

```

Private Sub LstClave_DbIClick()
TexBusDevol = LstClave
LstClave.Visible = False
TexBusDevol.SetFocus
End Sub

```

```

Private Sub LstClave2_DbIClick()
TexBusCamb = LstClave2
LstClave2.Visible = False
TexBusCamb.SetFocus
End Sub

```

```

Private Sub OptionCambio_Click()
If OptionDevol.Value = True Then
TexBusDevol.Enabled = True
TexBusCamb.Enabled = False
TexBusCamb.BackColor = ColorGris
TexBusDevol.BackColor = vbWhite
Else
TexBusDevol.Enabled = False
TexBusCamb.Enabled = True
TexBusCamb.BackColor = vbWhite
TexBusDevol.BackColor = ColorGris
TexBusDevol = ""
TexResult = ""

```

```

End If
End Sub

```

```

Private Sub OptionCambio_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```

Private Sub OptionDevol_Click()
If OptionDevol.Value = True Then
TexBusDevol.Enabled = True
TexBusCamb.Enabled = False
TexBusCamb.BackColor = ColorGris
TexBusDevol.BackColor = vbWhite
TexBusCamb = ""
TexResult = ""

```

```

Else
TexBusDevol.Enabled = False
TexBusCamb.Enabled = True
TexBusCamb.BackColor = vbWhite
TexBusDevol.BackColor = ColorGris

```

```

End If
End Sub

```

```

Private Sub OptionDevol_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```

Private Sub TexBusDevol_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexBusDevol.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True

```

```

End Sub

```

```

Private Sub TexBusDevol_DbIClick()
LstClave.Visible = True
End Sub

```

```

Private Sub TexBusDevol_GotFocus()
LstClave2.Visible = False
HighLight
If TexBusDevol <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbDevolucion.* FROM TbDevolucion WHERE FolioVta3 =
" & TexBusDevol & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusDevol = ""
TexBusDevol.SetFocus
End If
End If
End Sub

```

```

Private Sub TexBusDevol_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexBusDevol = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub

```

```

Private Sub TexBusCamb_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexBusCamb.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True

```

```

End Sub

```

```

Private Sub TexBusCamb_DbIClick()
LstClave2.Visible = True
End Sub

```

```

Private Sub TexBusCamb_GotFocus()
LstClave.Visible = False
HighLight
If TexBusCamb <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)

```

```

Set MiTabla = MiBD.OpenRecordset("SELECT
TbDevolucion.* FROM TbDevolucion WHERE FolioVta1 =
" & TexBusCamb & """, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusCamb = ""
TexBusCamb.SetFocus
End If
End If
End Sub

```

```

Private Sub TexBusCamb_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexBusCamb = LstClave2
SendKeys "{tab}"
LstClave2.Visible = False
End If
End Sub

```

BUSQUEDA EMPLEADOS

```

Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexCveEmp <> "" Then

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbEmp.* FROM TbEmp WHERE CveEmpleado = " &
TexCveEmp & """, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCveEmp = ""
TexCveEmp.SetFocus
Else
FoEmp.DataEmp.Recordset.FindFirst " CveEmpleado
= " & LstClave & ""
Unload Me
End If
End If
End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub BotRegresar_Click()
Unload Me
Set FoBuscEmp = Nothing
End Sub

```

```

Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub Form_Load()
LlenarCombo LstClave, "TbEmp", "CveEmpleado"
LstClave.Visible = False
End Sub

```

```

Private Sub LstClave_DblClick()
TexCveEmp = LstClave
LstClave.Visible = False
TexCveEmp.SetFocus
End Sub

```

```

Private Sub TexCveEmp_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCveEmp.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End Sub

```

```

Private Sub TexCveEmp_DblClick()
LstClave.Visible = True
End Sub

```

```

Private Sub TexCveEmp_GotFocus()
HighLight
If TexCveEmp <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbEmp.* FROM TbEmp WHERE CveEmpleado = " &
TexCveEmp & """, dbOpenSnapshot)

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCveEmp = ""
TexCveEmp.SetFocus
Else
TexResult = MiTabla.Nombre
End If
End If
End Sub

```

```

Private Sub TexCveEmp_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexCveEmp = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub

```

BUSQUEDA ESTILOS

```

Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexEst <> "" Then

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbEstilos.* FROM TbEstilos WHERE Estilos = " & TexEst
& """, dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexEst = ""
TexEst.SetFocus

```



```

Else
  FoEstilos.DataEstilos.Recordset.FindFirst " Estilos = ""
& LstClave & ""
  Unload Me
End If
End If
End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub BotRegresar_Click()
Unload Me
Set FoBuscEstil = Nothing
End Sub

```

```

Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub Form_Load()
LlenarCombo LstClave, "TbEstilos", "Estilos"
LstClave.Visible = False
End Sub

```

```

Private Sub LstClave_DbClick()
TexEst = LstClave
LstClave.Visible = False
TexEst.SetFocus
End Sub

```

```

Private Sub TexEst_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexEst.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End Sub

```

```

Private Sub TexEst_DbClick()
LstClave.Visible = True
End Sub

```

```

Private Sub TexEst_GotFocus()
HighLight
If TexEst <> "" Then
  Dim MiBD As Object
  Dim MiTabla As Object

  Set MiBD = OpenDatabase(MiBDStr)
  Set MiTabla = MiBD.OpenRecordset("SELECT
TbEstilos.* FROM TbEstilos WHERE Estilos = "" & TexEst
& """, dbOpenSnapshot)

  If MiTabla.EOF Then
    MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
    TexEst = ""
    TexEst.SetFocus
  Else
    TexResult = MiTabla.Estilos
  End If
End If
End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```

End If
End If
End Sub

```

```

Private Sub TexEst_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
  TexEst = LstClave
  SendKeys "{tab}"
  LstClave.Visible = False
End If
End Sub

```

BUSQUEDA FACTURA

```

Private Sub BotBusqueda_Click()
If OptionVenta.Value = True Then
  Dim MiBD As Object
  Dim MiTabla As Object
  If TexBusFolVta <> "" Then

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbFacturas.* FROM TbFacturas WHERE FolioVenta = "" &
TexBusFolVta & """, dbOpenSnapshot)

    If MiTabla.EOF Then
      MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
      TexBusFolVta = ""
      TexBusFolVta.SetFocus
    Else
      FoFacturas.DataFacturas.Recordset.FindFirst
"FolioVenta = "" & LstClave & ""
      Unload Me
    End If
  End If
End Sub

```

```

Else
  If TexBusFolFac <> "" Then

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbFacturas.* FROM TbFacturas WHERE Folio = "" &
TexBusFolFac & """, dbOpenSnapshot)

    If MiTabla.EOF Then
      MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
      TexBusFolFac = ""
      TexBusFolFac.SetFocus
    Else
      FoFacturas.DataFacturas.Recordset.FindFirst "Folio =
"" & LstClave2 & ""
      Unload Me
    End If
  End If
End Sub

```

```

End Sub

```

```

End Sub

```

```

End Sub

```

```

End If

```

```

End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

```
Private Sub BotRegresar_Click()
Unload Me
Set FoBuscFac = Nothing
End Sub
```

```
Private Sub BotRegresar_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Activate()
If OptionFactura.Value = True Then
    TexBusFolVta.Enabled = False
    TexBusFolFac.Enabled = True
    TexBusFolVta.BackColor = ColorGris
Else
    TexBusFolFac.Enabled = False
    TexBusFolVta.Enabled = True
End If
End Sub
```

```
Private Sub Form_Load()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
Dim MiRegistro2 As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
FolioVenta,Folio From TbFacturas WHERE CondHist = 0 ",
dbOpenSnapshot)
```

```
Do Until MiRegistro.EOF
    LstClave.AddItem IIf(IsNull(MiRegistro!FolioVenta), "",
MiRegistro!FolioVenta)
    LstClave.ItemData(LstClave.NewIndex) = MiRegistro!Folio
    MiRegistro.MoveNext
Loop
```

```
Set MiRegistro2 = MiBaseDatos.OpenRecordset("SELECT
Folio,Folio From TbFacturas WHERE CondHist = 0 ",
dbOpenSnapshot)
```

```
Do Until MiRegistro2.EOF
    LstClave2.AddItem IIf(IsNull(MiRegistro2!Folio), "",
MiRegistro2!Folio)
    LstClave2.ItemData(LstClave2.NewIndex) =
MiRegistro2!Folio
    MiRegistro2.MoveNext
Loop
```

```
LstClave.Visible = False
LstClave2.Visible = False
End Sub
```

```
Private Sub LstClave_DblClick()
TexBusFolVta = LstClave
LstClave.Visible = False
TexBusFolVta.SetFocus
End Sub
```

```
Private Sub LstClave2_DblClick()
TexBusFolFac = LstClave2
LstClave2.Visible = False
TexBusFolFac.SetFocus
End Sub
```

```
Private Sub OptionFactura_Click()
If OptionVenta.Value = True Then
    TexBusFolVta.Enabled = True
    TexBusFolFac.Enabled = False
    TexBusFolFac.BackColor = ColorGris
    TexBusFolVta.BackColor = vbWhite
Else
    TexBusFolVta.Enabled = False
    TexBusFolFac.Enabled = True
    TexBusFolFac.BackColor = vbWhite
    TexBusFolVta.BackColor = ColorGris
    TexBusFolVta = ""
    TexResult = ""
```

```
End If
End Sub
```

```
Private Sub OptionFactura_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub OptionVenta_Click()
If OptionVenta.Value = True Then
    TexBusFolVta.Enabled = True
    TexBusFolFac.Enabled = False
    TexBusFolFac.BackColor = ColorGris
    TexBusFolVta.BackColor = vbWhite
    TexBusFolFac = ""
    TexResult = ""
```

```
Else
    TexBusFolVta.Enabled = False
    TexBusFolFac.Enabled = True
    TexBusFolFac.BackColor = vbWhite
    TexBusFolVta.BackColor = ColorGris
```

```
End If
End Sub
```

```
Private Sub OptionVenta_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub TexBusFolVta_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexBusFolVta.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
```

```
End Sub
```

```
Private Sub TexBusFolVta_DblClick()
LstClave.Visible = True
End Sub
```

```
Private Sub TexBusFolVta_GotFocus()
LstClave2.Visible = False
HighLight
If TexBusFolVta <> "" Then
    Dim MiBD As Object
    Dim MiTabla As Object
```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbFacturas.* FROM TbFacturas WHERE FolioVenta = " &
TexBusFolVta & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusFolVta = ""
TexBusFolVta.SetFocus
End If
End If
End Sub

```

```

Private Sub TexBusFolVta_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexBusFolVta = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub

```

```

Private Sub TexBusFolFac_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexBusFolFac.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True
End Sub

```

```

Private Sub TexBusFolFac_DblClick()
LstClave2.Visible = True
End Sub

```

```

Private Sub TexBusFolFac_GotFocus()
LstClave.Visible = False
HighLight
If TexBusFolFac <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbFacturas.* FROM TbFacturas WHERE Folio = " &
TexBusFolFac & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusFolFac = ""
TexBusFolFac.SetFocus
End If
End If
End Sub

```

```

Private Sub TexBusFolFac_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexBusFolFac = LstClave2
SendKeys "{tab}"
LstClave2.Visible = False
End If
End Sub

```

BUSQUEDA INVENTARIO

```

Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexCveCalz <> "" Then

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCveCalz = ""
TexCveCalz.SetFocus
Else
FoInvent.DataInventario.Recordset.FindFirst "
ClaveCalzado = " & LstClave & ""
Unload Me
End If
End If
End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub BotRegresar_Click()
Unload Me
Set FoBuscInv = Nothing
End Sub

```

```

Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub Form_Load()
LlenarCombo LstClave, "TbInventario", "ClaveCalzado"
LstClave.Visible = False
End Sub

```

```

Private Sub LstClave_DblClick()
TexCveCalz = LstClave
LstClave.Visible = False
TexCveCalz.SetFocus
End Sub

```

```

Private Sub TexCveCalz_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCveCalz.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True

```

```

End Sub

```

```

Private Sub TexCveCalz_DblClick()
LstClave.Visible = True
End Sub

```

```

Private Sub TexCveCalz_GotFocus()
HighLight
If TexCveCalz <> "" Then
    Dim MiBD As Object
    Dim MiTabla As Object

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz & "", dbOpenSnapshot)

    If MiTabla.EOF Then
        MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
        TexCveCalz = ""
        TexCveCalz.SetFocus
    Else
        TexResult = MiTabla.Descripcion
    End If
End If
End Sub

```

```

Private Sub TexCveCalz_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    TexCveCalz = LstClave
    SendKeys "{tab}"
    LstClave.Visible = False
End If
End Sub

```

BUSQUEDA LINEAS

```

Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexLin <> "" Then

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbLineas.* FROM TbLineas WHERE Lineas = " & TexLin,
dbOpenSnapshot)

    If MiTabla.EOF Then
        MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
        TexLin = ""
        TexLin.SetFocus
    Else
        FoLineas.DataLines.Recordset.FindFirst " Lineas = "
& LstClave
        Unload Me
    End If
End If
End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub BotRegresar_Click()
Unload Me
Set FoBuscLineas = Nothing
End Sub

```

```

Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub Form_Load()
LlenarCombo LstClave, "TbLineas", "Lineas"
LstClave.Visible = False
End Sub

```

```

Private Sub LstClave_DbClick()
TexLin = LstClave
LstClave.Visible = False
TexLin.SetFocus
End Sub

```

```

Private Sub TexLin_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexLin.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End Sub

```

```

Private Sub TexLin_DbClick()
LstClave.Visible = True
End Sub

```

```

Private Sub TexLin_GotFocus()
HighLight
If TexLin <> "" Then
    Dim MiBD As Object
    Dim MiTabla As Object

```

```

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbLineas.* FROM TbLineas WHERE Lineas = " & TexLin,
dbOpenSnapshot)

```

```

    If MiTabla.EOF Then
        MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
        TexLin = ""
        TexLin.SetFocus
    Else
        TexResult = MiTabla.Lineas
    End If
End If
End Sub

```

```

Private Sub TexLin_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    TexLin = LstClave
    SendKeys "{tab}"
    LstClave.Visible = False
End If
End Sub

```

BUSQUEDA PROVEEDORES

```

Private Sub BotBusqueda_Click()

```

```

If OptionRFC.Value = True Then
    Dim MiBD As Object
    Dim MiTabla As Object
    If TexBusRFC <> "" Then

```

```

        Set MiBD = OpenDatabase(MiBDStr)

```

```
Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE RFC = " &
TexBusRFC & """, dbOpenSnapshot)
```

```
If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusRFC = ""
TexBusRFC.SetFocus
Else
FoProv.DataProvee.Recordset.FindFirst "RFC = " &
LstClave & ""
Unload Me
End If
End If
```

```
Else
If TexBusCveProv <> "" Then
```

```
Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE
CveProveedores = " & TexBusCveProv & """,
dbOpenSnapshot)
```

```
If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexBusCveProv = ""
TexBusCveProv.SetFocus
Else
FoProv.DataProvee.Recordset.FindFirst
"CveProveedores = " & LstClave2 & ""
Unload Me
End If
End If
```

```
End If
```

```
End Sub
```

```
Private Sub BotBusqueda_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub BotRegresar_Click()
Unload Me
Set FoBuscClie = Nothing
End Sub
```

```
Private Sub BotRegresar_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Activate()
If OptionCveProv.Value = True Then
TexBusRFC.Enabled = False
TexBusCveProv.Enabled = True
TexBusRFC.BackColor = ColorGris
Else
TexBusCveProv.Enabled = False
TexBusRFC.Enabled = True
End If
End Sub
```

```
Private Sub Form_Load()
LlenarCombo LstClave, "TbProveedores", "RFC", "Folio"
LlenarCombo LstClave2, "TbProveedores",
"CveProveedores", "Folio"
LstClave.Visible = False
LstClave2.Visible = False
End Sub
```

```
Private Sub LstClave_DblClick()
TexBusRFC = LstClave
LstClave.Visible = False
TexBusRFC.SetFocus
End Sub
```

```
Private Sub LstClave2_DblClick()
TexBusCveProv = LstClave2
LstClave2.Visible = False
TexBusCveProv.SetFocus
End Sub
```

```
Private Sub OptionCveProv_Click()
If OptionRFC.Value = True Then
TexBusRFC.Enabled = True
TexBusCveProv.Enabled = False
TexBusCveProv.BackColor = ColorGris
TexBusRFC.BackColor = vbWhite
Else
TexBusRFC.Enabled = False
TexBusCveProv.Enabled = True
TexBusCveProv.BackColor = vbWhite
TexBusRFC.BackColor = ColorGris
TexBusRFC = ""
TexResult = ""
```

```
End If
End Sub
```

```
Private Sub OptionCveProv_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```
Private Sub OptionRFC_Click()
If OptionRFC.Value = True Then
TexBusRFC.Enabled = True
TexBusCveProv.Enabled = False
TexBusCveProv.BackColor = ColorGris
TexBusRFC.BackColor = vbWhite
TexBusCveProv = ""
TexResult = ""
```

```
Else
TexBusRFC.Enabled = False
TexBusCveProv.Enabled = True
TexBusCveProv.BackColor = vbWhite
TexBusRFC.BackColor = ColorGris
End If
End Sub
```

```
Private Sub OptionRFC_GotFocus()
LstClave2.Visible = False
LstClave.Visible = False
End Sub
```

```

Private Sub TexBusRFC_Change()
Dim Index As Integer
  On Error Resume Next
  Index = FindFirstMatch(LstClave, TexBusRFC.Text, -1)
  LstClave.ListIndex = Index
  If Index >= 0 Then LstClave.Selected(Index) = True
End Sub

```

```

Private Sub TexBusRFC_Db1Click()
LstClave.Visible = True
End Sub

```

```

Private Sub TexBusRFC_GotFocus()
LstClave2.Visible = False
HighLight
If TexBusRFC <> "" Then
  Dim MiBD As Object
  Dim MiTabla As Object

  Set MiBD = OpenDatabase(MiBDStr)
  Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE RFC = "" &
TexBusRFC & """, dbOpenSnapshot)

```

```

  If MiTabla.EOF Then
    MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
    TexBusRFC = ""
    TexBusRFC.SetFocus
  Else
    TexResult = MiTabla.NombreProveedor
  End If
End If
End Sub

```

```

Private Sub TexBusRFC_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
  TexBusRFC = LstClave
  SendKeys "{tab}"
  LstClave.Visible = False
End If
End Sub

```

```

Private Sub TexBusCveProv_Change()
Dim Index As Integer
  On Error Resume Next
  Index = FindFirstMatch(LstClave2, TexBusCveProv.Text, -
1)
  LstClave2.ListIndex = Index
  If Index >= 0 Then LstClave2.Selected(Index) = True
End Sub

```

```

Private Sub TexBusCveProv_Db1Click()
LstClave2.Visible = True
End Sub

```

```

Private Sub TexBusCveProv_GotFocus()
LstClave.Visible = False
HighLight
If TexBusCveProv <> "" Then
  Dim MiBD As Object
  Dim MiTabla As Object

  Set MiBD = OpenDatabase(MiBDStr)

```

```

  Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE
CveProveedores = "" & TexBusCveProv & """,
dbOpenSnapshot)

```

```

  If MiTabla.EOF Then
    MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
    TexBusCveProv = ""
    TexBusCveProv.SetFocus
  Else
    TexResult = MiTabla.NombreProveedor
  End If
End If
End Sub

```

```

Private Sub TexBusCveProv_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
  TexBusCveProv = LstClave2
  SendKeys "{tab}"
  LstClave2.Visible = False
End If
End Sub

```

BUSQUEDA VENTAS

```

Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If TexBusFolVta <> "" Then

  Set MiBD = OpenDatabase(MiBDStr)
  Set MiTabla = MiBD.OpenRecordset("SELECT
TbVentas.* FROM TbVentas WHERE NoVentas = "" &
TexBusFolVta & """, dbOpenSnapshot)

```

```

  If MiTabla.EOF Then
    MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
    TexBusFolVta = ""
    TexBusFolVta.SetFocus
  Else
    FoVentas.DataVentas.Recordset.FindFirst "NoVentas
= "" & LstClave & ""
    Unload Me
  End If
End If
End Sub

```

```

Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub BotRegresar_Click()
Unload Me
Set FoBuscVent = Nothing
End Sub

```

```

Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub Form_Load()
Dim MiBaseDatos As Object

```

Dim MiRegistro As Object

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
NoVentas,Ventas_ID From TbVentas WHERE CondHist = 0
", dbOpenSnapshot)
```

```
Do Until MiRegistro.EOF
    LstClave.AddItem IIf(IsNull(MiRegistro.NoVentas), "",
MiRegistro.NoVentas)
    LstClave.ItemData(LstClave.NewIndex) =
MiRegistro.Ventas_ID
    MiRegistro.MoveNext
Loop
```

```
LstClave.Visible = False
End Sub
```

```
Private Sub LstClave_DblClick()
TexBusFolVta = LstClave
LstClave.Visible = False
TexBusFolVta.SetFocus
End Sub
```

```
Private Sub TexBusFolVta_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexBusFolVta.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End Sub
```

```
Private Sub TexBusFolVta_DblClick()
LstClave.Visible = True
End Sub
```

```
Private Sub TexBusFolVta_GotFocus()
LstClave.Visible = False
HighLight
If TexBusFolVta <> "" Then
    Dim MiBD As Object
    Dim MiTabla As Object

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbVentas.* FROM TbVentas WHERE NoVentas = " &
TexBusFolVta & """, dbOpenSnapshot)
```

```
    If MiTabla.EOF Then
        MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
        TexBusFolVta = ""
        TexBusFolVta.SetFocus
    End If
End Sub
```

```
Private Sub TexBusFolVta_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    TexBusFolVta = LstClave
    SendKeys "{tab}"
    LstClave.Visible = False
End If
End Sub
```

BUSQUEDA SEGURIDAD

Option Explicit

```
Private Sub BotBusqueda_Click()
Dim MiBD As Object
Dim MiTabla As Object
If LstClave <> "" Then

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT Nombre
FROM TbSeg WHERE ID = " &
LstClave.ItemData(LstClave.ListIndex), dbOpenSnapshot)
```

```
    If MiTabla.EOF Then
        MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
        TexBusFolVta = ""
        TexBusFolVta.SetFocus
    Else
        FoSeg.Data1.Recordset.FindFirst "ID = " &
LstClave.ItemData(LstClave.ListIndex)
        Unload Me
    End If
Else
    MsgBox "La clave proporcionada no existe,trate de nuevo
", vbExclamation
    TexBusFolVta = ""
    TexBusFolVta.SetFocus
End If
End Sub
```

```
Private Sub BotBusqueda_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotRegresar_Click()
Unload Me
Set FoBuseVent = Nothing
End Sub
```

```
Private Sub BotRegresar_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Load()
LlenarCombo LstClave, "TbSeg", "Nombre", "ID"
LstClave.Visible = False
End Sub
```

```
Private Sub LstClave_DblClick()
TexBusFolVta = LstClave
LstClave.Visible = False
TexBusFolVta.SetFocus
End Sub
```

```
Private Sub TexBusFolVta_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexBusFolVta.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End Sub
```

```
Private Sub TextBoxFolVta_DblClick()
LstClave.Visible = True
End Sub
```

```
Private Sub TextBoxFolVta_GotFocus()
LstClave.Visible = False
If TextBoxFolVta <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT Nombre
FROM TbSeg WHERE ID = " &
LstClave.ItemData(LstClave.ListIndex), dbOpenSnapshot)
```

```
If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TextBoxFolVta = ""
TextBoxFolVta.SetFocus
End If
End If
End Sub
```

```
Private Sub TextBoxFolVta_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TextBoxFolVta = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub
```

CLIENTES

```
Option Explicit
Dim MiEmp As String
```

```
Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataClientes.Recordset.EditMode = 0 Then
If Not DataClientes.Recordset.EOF Then
MiReg = TextCveCli
DataClientes.Recordset.MoveLast
End If

If DataClientes.Recordset.RecordCount <= 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = True
BotImprimir.Enabled = False
BotBuscar.Enabled = False
ElseIf DataClientes.Recordset.RecordCount <= 1 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
```

```
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
```

Else

```
DataClientes.Recordset.FindFirst " CveClientes =" &
MiReg & ""
```

```
If DataClientes.Recordset.AbsolutePosition = 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = True
BotUltimo.Enabled = True
```

```
ElseIf DataClientes.Recordset.AbsolutePosition =
DataClientes.Recordset.RecordCount - 1 Then
```

```
BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
```

Else

```
BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = True
BotUltimo.Enabled = True
```

End If

```
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
```

End If

Else

```
BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True
```

```
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
```

```
BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
```

End If

```
BloqueaControles
End Sub
```

```
Private Sub BotAgregar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
DataClientes.Recordset.AddNew
LabelFecha = Date
HabilitarBotones
```



```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(Folio) as Folio1 From TbClientes", dbOpenSnapshot)

If Not IsNull(MiRegistro!Folio1) Then
    TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
Else
    TexFolio = "0001"
End If

TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
MiRegistro.Close

Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(CveClientes) as Clave From TbClientes",
dbOpenSnapshot)

If Not IsNull(MiRegistro!CLAVE) Then
    TexCveCli = Format((MiRegistro!CLAVE) + 1, "0000")
Else
    TexCveCli = "0001"
End If

TexCveCli.BackColor = &H8000000F
TexCveCli.Locked = True
TexCliID = IncrementadorID("TbClientes")
TexPais = "México"
TexNomCli.SetFocus
End Sub

Private Sub BotAnterior_Click()
DataClientes.Recordset.MovePrevious
HabilitarBotones
End Sub

Private Sub BotBuscar_Click()
FoBuscCli.Show vbModal
End Sub

Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
    DataClientes.Recordset.CancelUpdate
    HabilitarBotones
    BotAgregar.SetFocus
End If

End Sub

Private Sub BotEliminar_Click()
Dim Eliminar As String
Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
If Eliminar = vbYes Then
    With DataClientes.Recordset
        .Delete
        .Requery
    End With
    If Not .EOF Then
        .MoveLast
    Else
        BotEliminar.Enabled = False
        .Requery
        DataClientes.Refresh
    End If
End Sub

```

```

End With
End If
HabilitarBotones
End Sub

```

```

Private Sub BotGuardar_Click()
Dim añadir As String
If TexRFC = "" Or TexNomCli = "" Or TexCol = "" Then
    añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
    TexCveCli.SetFocus
Else
    DataClientes.Recordset.Update
    DataClientes.Refresh
    HabilitarBotones
End If
End Sub

```

```

Private Sub BotImprimir_Click()
FoOpRpCli.Show
End Sub

```

```

Private Sub BotMenu_Click()
FoMenu.Show
Unload FoClientes
End Sub

```

```

Private Sub BotModificar_Click()
DataClientes.Recordset.Edit
HabilitarBotones
TexCveCli.BackColor = &H8000000F
TexCveCli.Locked = True
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
End Sub

```

```

Private Sub BotPrimero_Click()
DataClientes.Recordset.MoveFirst
HabilitarBotones
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataClientes.Recordset.EOF And
DataClientes.Recordset.RecordCount > 1 Then
    DataClientes.Recordset.MoveNext
End If
HabilitarBotones
End Sub

```

```

Private Sub BotUltimo_Click()
DataClientes.Recordset.MoveLast
HabilitarBotones
End Sub

```

```

Private Sub CombEstados_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Form_Activate()
HabilitarBotones

```

```
BotAgrega.SetFocus
End Sub
```

```
Private Sub Form_Load()
FoMenu.Hide
DataClientes.DatabaseName = MiBDStr
LlenarCombo CombEstados, "TbEstados", "Estados",
"Estados_ID"
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub
```

```
Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataClientes.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataClientes.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
Next
End With
```

```
End Sub
```

```
Private Sub TexCalle_GotFocus()
HighLight
End Sub
```

```
Private Sub TexCalle_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub TexCiud_GotFocus()
HighLight
```

```
End Sub
```

```
Private Sub TexCiud_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub TexCol_GotFocus()
HighLight
End Sub
```

```
Private Sub TexCol_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub TexCP_GotFocus()
HighLight
End Sub
```

```
Private Sub TexCP_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub TexEstadosID_Change()
BusquedaldCombo TexEstadosID, CombEstados
End Sub
Private Sub CombEstados_Click()
DesplegarEstados CombEstados, TexEstadosID
End Sub
```

```
Private Sub TexNomCli_GotFocus()
HighLight
End Sub
```

```
Private Sub TexNomCli_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub TexPais_GotFocus()
HighLight
End Sub
```

```
Private Sub TexPais_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
```

```
End If
End Sub
```

```
Private Sub TexRFC_GotFocus()
HighLight
End Sub
```

```
Private Sub TexRFC_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexRFC_LostFocus()
TexRFC = DameFormato(TexRFC, TexRFC.Tag)
End Sub
```

```
Private Sub Text1_GotFocus()
HighLight
End Sub
```

```
Private Sub Text1_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexTelNeg_GotFocus()
HighLight
End Sub
```

```
Private Sub TexTelNeg_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexTelPart_GotFocus()
HighLight
End Sub
```

```
Private Sub TexTelPart_KeyPress(KeyAscii As Integer)
If DataClientes.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

COMPRAS

```
Option Explicit
Dim MiModelo As String
Dim MiBol As Boolean
```

```
Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
```

```
If DataCompras.Recordset.EditMode = 0 Then
If Not DataCompras.Recordset.EOF Then
  MiReg = TexNoFac
  DataCompras.Recordset.MoveLast
End If
```

```
If DataCompras.Recordset.RecordCount <= 0 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotAgregar.Enabled = True
  BotGuardar.Enabled = False
  BotCancelar.Enabled = False
  BotEliminar.Enabled = False
  BotImprimir.Enabled = False
  BotBuscar.Enabled = False
  BotModificar.Enabled = False
  BotMenu.Enabled = True
ElseIf DataCompras.Recordset.RecordCount <= 1 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotAgregar.Enabled = True
  BotGuardar.Enabled = False
  BotCancelar.Enabled = False
  BotEliminar.Enabled = True
  BotImprimir.Enabled = True
  BotBuscar.Enabled = True
  BotModificar.Enabled = True
  BotMenu.Enabled = True
```

Else

```
DataCompras.Recordset.FindFirst " NoFac = " &
PonCero(MiReg)
```

```
If DataCompras.Recordset.AbsolutePosition = 0 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = True
  BotUltimo.Enabled = True
```

```
ElseIf DataCompras.Recordset.AbsolutePosition =
DataCompras.Recordset.RecordCount - 1 Then
```

```
  BotPrimero.Enabled = True
  BotAnterior.Enabled = True
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
```

Else

```
  BotPrimero.Enabled = True
  BotAnterior.Enabled = True
  BotSiguiente.Enabled = True
  BotUltimo.Enabled = True
```

End If

```
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
```

End If

Else

```
BotAgregar.Enabled = False
```

```

BotGuardar.Enabled = True
BotCancelar.Enabled = True

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False

BotEliminar.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False

End If
BloqueaControles
ActualizarGrid
End Sub

Private Sub BotAgregar_GotFocus()
LstClave.Visible = False
End Sub

Private Sub BotAnterior_Click()
DataCompras.Recordset.MovePrevious
HabilitarBotones
End Sub

Private Sub BotAnterior_GotFocus()
LstClave.Visible = False
End Sub

Private Sub BotBuscar_Click()
FoBuscComp.Show vbModal
End Sub

Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbQuestion, "CANCELAR")
If CANCELAR = vbYes Then
BorrarTab "TbTmpInventario"
DataCompras.Recordset.CancelUpdate
PasarDatos "TbDtllCompras", "TbTmpDtllCompras",
"DlIFolFac =" & TexFolio & ""
HabilitarBotones
BotAgregar.SetFocus
End If

MiModelo = ""

End Sub

Private Sub BotEliminar_Click()
Dim Eliminar As String
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
If Eliminar = vbYes Then
With DataCompras.Recordset
.Delete
Set MiBaseDatos = OpenDatabase(MiBDStr)

```

```

Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT DIINoPares,
DIIPrecUni, DIIPrecMen, DIIClaveCalzado, DIIPrecMay,
DIImporte, DIIDescripcion, DIIModelo, DIID_ID FROM
TbTmpDtllCompras WHERE DIIFolFac =" & TexFolio & "",
dbOpenDynaset) 'Costante para abrir en modo solo lectura y si
fuera dbOpenDynaset era para lectura/escritura'

Do Until MiRegistro.EOF
MiRegistro.Delete
MiRegistro.MoveNext
Loop

If Not MiRegistro.EOF Then
.MoveLast
Else
BotEliminar.Enabled = False
DataCompras.Refresh
End If
End With
End If
HabilitarBotones

End Sub

Private Sub BotGuardar_Click()
Dim añadir As String
Dim BuscarDatos
Dim Mult As Variant
If Trim(TexNoFac) = "" Or Trim(TexCvePro) = "" Then
añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
Else
Mult = LabelSubTotal * (15 / 100)
LabelIVA = FormatCurrency(Mult)
BorrarTab "TbDtllCompras", "DIIFolFac =" & TexFolio &
""
PasarDatos "TbTmpDtllCompras", "TbDtllCompras"
BorrarTab "TbTmpInventario"
DataCompras.Recordset.Update
DataCompras.Refresh
HabilitarBotones
BotAgregar.SetFocus
End If
End Sub

Private Sub BotAgregar_Click()

Dim MiBaseDatos As Object
Dim MiRegistro As Object

BorrarTab "TbTmpDtllCompras"
BorrarTab "TbTmpInventario"

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(Folio) as Folio1 From TbCompras", dbOpenSnapshot)
DataCompras.Refresh
DataCompras.Recordset.AddNew
HabilitarBotones
LabelFechCompr = Date
TexNoFac.SetFocus
If Not IsNull(MiRegistro!Folio1) Then
TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
Text1.BackColor = &H8000000F
Text1.Locked = True

```

```
Else
  TexFolio = "0001"
  TexFolio.BackColor = &H8000000F
  TexFolio.Locked = True
  Text1.BackColor = &H8000000F
  Text1.Locked = True
End If
End Sub
```

```
Private Sub BotImprimir_Click()
FoOpRpComp.Show
End Sub
```

```
Private Sub BotMenu_Click()
FoMenu.Show
Unload FoCompras
End Sub
```

```
Private Sub BotModificar_Click()
BorrarTab "TbTmpInventario"
DataCompras.Recordset.Edit
HabilitarBotones
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
Text1.BackColor = &H8000000F
Text1.Locked = True
TexNoFac.SetFocus
End Sub
```

```
Private Sub BotModificar_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotPrimero_Click()
DataCompras.Recordset.MoveFirst
HabilitarBotones
End Sub
```

```
Private Sub BotPrimero_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotSiguiente_Click()
```

```
If Not DataCompras.Recordset.EOF And
DataCompras.Recordset.RecordCount > 1 Then
  DataCompras.Recordset.MoveNext
End If
HabilitarBotones
End Sub
```

```
Private Sub BotSiguiente_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotUltimo_Click()
DataCompras.Recordset.MoveLast
HabilitarBotones
End Sub
```

```
Private Sub BotUltimo_GotFocus()
```

```
LstClave.Visible = False
End Sub
```

```
Private Sub Form_Activate()
HabilitarBotones
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
Text1.BackColor = &H8000000F
Text1.Locked = True
```

```
MiBol = True
End Sub
```

```
Private Sub Form_Load()
DataCompras.DatabaseName = MiBDStr
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
LlenarCombo LstClave, "TbProveedores", "CveProveedores",
"Folio"
MiBol = False
FoMenu.Hide
End Sub
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub
```

```
Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataCompras.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H8000000F
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataCompras.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H8000000F
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is MSHFlexGrid Then
If DataCompras.Recordset.EditMode = 0 Then
.Controls(i).BackColor = &H80000004
Else
.Controls(i).BackColor = vbWhite
End If
End If
Next
```

```

End With
End Sub

Private Sub GDComp_DbIcClick()
If DataCompras.Recordset.EditMode <> 0 Then
If GDComp.TextMatrix(GDComp.Row, 0) = "" Then
FoDetallesCompras.ModoEdit = 2
ElseIf GDComp.TextMatrix(GDComp.Row, 0) <> "" Then
FoDetallesCompras.ModoEdit = 1
End If
FoDetallesCompras.Show vbModal
End If
End Sub

```

```

Private Sub GDComp_GotFocus()
LstClave.Visible = False
HighLight
End Sub

```

```

Private Sub GDComp_KeyPress(KeyAscii As Integer)
If DataCompras.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub GDComp_KeyUp(KeyCode As Integer, Shift As Integer)
Dim MiBaseDatos As Object
Dim MiRegistro As Object
If KeyCode = 46 And GDComp.TextMatrix(GDComp.Row, 0) <> "" Then
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("select TbTmpDtlCompras.* FROM TbTmpDtlCompras WHERE Dtl_ID =" & GDComp.TextMatrix(GDComp.Row, 0), dbOpenDynaset)
If Not MiRegistro.EOF Then
MiRegistro.Delete
MiRegistro.Close
ActualizarGrid
End If
End If
End Sub

```

```

Private Sub LabelTotal_Change()
If MiBol = True Then
Text1 = ImporteLetra(LabelTotal)
End If
End Sub

```

```

Private Sub LstClave_DbIcClick()
TexCvePro = LstClave
LstClave.Visible = False
TexCvePro.SetFocus
End Sub

```

```

Private Sub LstClave_LostFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub TexCvePro_Change()

```

```

If DataCompras.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCvePro.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCvePro_DbIcClick()
If DataCompras.Recordset.EditMode <> 0 Then
LstClave.Visible = True
End If
End Sub

```

```

Private Sub TexCvePro_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCvePro_KeyPress(KeyAscii As Integer)
If DataCompras.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
TexCvePro = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End If
End Sub

```

```

Private Sub TexCvePro_LostFocus()
If DataCompras.Recordset.EditMode <> 0 And TexCvePro <> "" Then

```

```

Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT TbProveedores.*
From TbProveedores WHERE CveProveedores =" &
TexCvePro & "", dbOpenSnapshot) 'Costante para abrir en
modo solo lectura y si fuera dbOpenDynaset era para
lectura/escritura'

```

```

If MiRegistro.EOF Then
MsgBox "La clave proporcionada no existe, favor de
verificar", vbExclamation
TexCvePro = ""
TexCvePro.SetFocus
End If
End If

```

```

End Sub

```

```

Private Sub TexNoFac_GotFocus()
LstClave.Visible = False
HighLight
End Sub

```

```

Private Sub TexNoFac_KeyPress(KeyAscii As Integer)
If DataCompras.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If

```

```

End If
End Sub

Private Sub TexNoFac_LostFocus()
If Not IsNumeric(TexNoFac) And TexNoFac <> "" Then
MsgBox "Poner solo números", vbExclamation
TexNoFac = ""
TexNoFac.SetFocus
End If
End Sub

Public Sub ActualizarGrid()
Dim MiSubtotal As Currency
Dim i As Long
If DataCompras.EditMode = 0 Then
BorrarTab "TbTmpDtllCompras"
PasarDatos "TbDtllCompras", "TbTmpDtllCompras",
"DllFolFac="" & TexFolio & ""
End If
i = 1
With GDCComp
.Clear
.Cols = 15
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DllLineasII_ID, DllMarcas_ID, DllLineas_ID, DllSexo_ID,
DllMaterial_ID, DllEstilos_ID, DllCol_ID, DllNoPares,
DllPrecUni, DllPrecMen, DllClaveCalzado, DllPrecMay,
DllImporte, DllDescripcion, DllModelo, Dll_ID FROM
TbTmpDtllCompras WHERE DllFolFac="" & TexFolio & """,
dbOpenSnapshot)
If MiRegistro.EOF Then
GDCComp.Clear
GDCComp.Rows = 2
End If
Do Until MiRegistro.EOF
.Rows = MiRegistro.RecordCount + 2
.TextMatrix(i, 0) = MiRegistro!Dll_ID
.TextMatrix(i, 4) = MiRegistro!DllNoPares
.TextMatrix(i, 5) = Format(MiRegistro!DllPrecUni,
"currency")
.TextMatrix(i, 2) = Format(MiRegistro!DllPrecMen,
"currency")
.TextMatrix(i, 7) = MiRegistro!DllClaveCalzado
.TextMatrix(i, 3) = Format(MiRegistro!DllPrecMay,
"currency")
.TextMatrix(i, 6) = Format(MiRegistro!DllImporte,
"currency")
.TextMatrix(i, 8) = MiRegistro!DllModelo
.TextMatrix(i, 1) = MiRegistro!DllDescripcion
.TextMatrix(i, 9) = MiRegistro!DllMarcas_ID
.TextMatrix(i, 10) = MiRegistro!DllCol_ID
.TextMatrix(i, 11) = MiRegistro!DllEstilos_ID
.TextMatrix(i, 12) = MiRegistro!DllMaterial_ID
.TextMatrix(i, 13) = MiRegistro!DllSexo_ID
.TextMatrix(i, 14) = MiRegistro!DllLineas_ID
MiSubtotal = MiSubtotal + MiRegistro!DllImporte
i = i + 1

MiRegistro.MoveNext
Loop
LabelSubTotal = Format(MiSubtotal, "currency")
LabelIVA = Format(MiSubtotal * (IVA / 100), "currency")
LabelTotal = Format(CCur(LabelIVA) + CCur(MiSubtotal),
"currency")
MiRegistro.Close

```

```

.ColWidth(0) = 0
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 5) = "Precio Unitario"
.ColAlignment(5) = 7
.ColWidth(5) = 1050
.TextMatrix(0, 2) = "Precio Menenudeo"
.ColAlignment(2) = 7
.ColWidth(2) = 1300
.ColWidth(7) = 0
.TextMatrix(0, 3) = "Precio Mayoreo"
.ColAlignment(3) = 7
.ColWidth(3) = 1100
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000
.ColWidth(8) = 0
.TextMatrix(0, 1) = "Descripción"
.ColWidth(1) = 3150
.ColWidth(9) = 0
.ColWidth(10) = 0
.ColWidth(11) = 0
.ColWidth(12) = 0
.ColWidth(13) = 0
.ColWidth(14) = 0
.ColWidth(15) = 0
End With
End Sub

```

```

End With
End Sub

```

```

Public Sub PasarDatos(Tb1 As String, Tb2 As String, Optional
WH As String)
Dim MiBD As Object
Dim MiTab1 As Object, MiTab2 As Object, MiTab3 As
Object, MiTab4 As Object, MiTab5 As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTab1 = MiBD.OpenRecordset("SELECT " & Tb1 &
"* FROM " & Tb1 & IIf(WH = "", "", " WHERE " & WH),
dbOpenSnapshot)
Set MiTab2 = MiBD.OpenRecordset("SELECT " & Tb2 &
"* FROM " & Tb2, dbOpenDynaset)

```

```

Do Until MiTab1.EOF
MiTab2.AddNew
MiTab2!DllNoPares = MiTab1!DllNoPares
MiTab2!DllPrecUni = MiTab1!DllPrecUni
MiTab2!DllPrecMen = MiTab1!DllPrecMen
MiTab2!DllPrecMay = MiTab1!DllPrecMay
MiTab2!DllImporte = MiTab1!DllImporte
MiTab2!DllModelo = MiTab1!DllModelo
MiTab2!Dll_ID = MiTab1!Dll_ID
MiTab2!DllFolFac = MiTab1!DllFolFac
MiTab2!DllDescripcion = MiTab1!DllDescripcion
MiTab2!DllMarcas_ID = MiTab1!DllMarcas_ID
MiTab2!DllCol_ID = MiTab1!DllCol_ID
MiTab2!DllEstilos_ID = MiTab1!DllEstilos_ID
MiTab2!DllLineas_ID = MiTab1!DllLineas_ID
MiTab2!DllMaterial_ID = MiTab1!DllMaterial_ID
MiTab2!DllSexo_ID = MiTab1!DllSexo_ID
MiTab2!DllLineasII_ID = MiTab1!DllLineasII_ID
MiTab2!DllClaveCalzado = MiTab1!DllClaveCalzado
MiTab2!Ventas_ID = MiTab1!Ventas_ID
MiTab2.Update
MiTab1.MoveNext
Loop

```

```

Set MiTab3 = MiBD.OpenRecordset("SELECT
TbTmpInventario.* FROM TbTmpInventario ",
dbOpenDynaset)

```

```

Do Until MiTab3.EOF
  If MiTab3!ModoEdit = 1 Then
    Set MiTab4 = MiBD.OpenRecordset("SELECT
Existencias FROM TblInventario WHERE ClaveCalzado = "
& MiTab3!ClaveCalzado & """, dbOpenDynaset)
    MiTab4.Edit
    MiTab4!Existencias = MiTab4!Existencias +
MiTab3!Existencias
    MiTab4.Update
  ElseIf MiTab3!ModoEdit = 2 Then
    Set MiTab5 = MiBD.OpenRecordset("SELECT
TblInventario.* FROM TblInventario", dbOpenDynaset)
    MiTab5.AddNew
    MiTab5!Existencias = MiTab3!Existencias
    MiTab5!CostUnit = MiTab3!CostUnit
    MiTab5!CostMen = MiTab3!CostMen
    MiTab5!ClaveCalzado = MiTab3!ClaveCalzado
    MiTab5!CostMay = MiTab3!CostMay
    MiTab5!Descripcion = MiTab3!Descripcion
    MiTab5!FechaIngreso = MiTab3!FechaIngreso
    MiTab5!DIIModelo = MiTab3!DIIModelo
    MiTab5!DIIMarcas_ID = MiTab3!DIIMarcas_ID
    MiTab5!DIICol_ID = MiTab3!DIICol_ID
    MiTab5!DIIEstilos_ID = MiTab3!DIIEstilos_ID
    MiTab5!DIIMaterial_ID = MiTab3!DIIMaterial_ID
    MiTab5!DIISexo_ID = MiTab3!DIISexo_ID
    MiTab5!DIILineas_ID = MiTab3!DIILineas_ID
    MiTab5!Ventas_ID = MiTab3!Ventas_ID
    MiTab5.Update
  End If
  MiTab3.MoveNext
Loop
End Sub

```

CONDICION HISTORICO

```

Option Explicit
Dim MiModelo As String
Dim MiBol As Boolean

```

```

Private Sub HabilitarBotones()
Dim MiReg As String
  BloqueaControles
  If DataCompras.Recordset.EditMode = 0 Then
    If Not DataCompras.Recordset.EOF Then
      MiReg = TexNoFac
      DataCompras.Recordset.MoveLast
    End If

    If DataCompras.Recordset.RecordCount <= 0 Then
      BotPrimero.Enabled = False
      BotAnterior.Enabled = False
      BotSiguiente.Enabled = False
      BotUltimo.Enabled = False
      BotGuardar.Enabled = True
      BotMenu.Enabled = True
    ElseIf DataCompras.Recordset.RecordCount <= 1 Then
      BotPrimero.Enabled = False
      BotAnterior.Enabled = False
      BotSiguiente.Enabled = False
      BotUltimo.Enabled = False
      BotGuardar.Enabled = True
      BotMenu.Enabled = True

    Else

      DataCompras.Recordset.FindFirst " NoFac = " &
PonCero(MiReg)

```

```

If DataCompras.Recordset.AbsolutePosition = 0 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = True
  BotUltimo.Enabled = True

  ElseIf DataCompras.Recordset.AbsolutePosition =
DataCompras.Recordset.RecordCount - 1 Then

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
  Else
    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
  End If

  BotGuardar.Enabled = True
  BotMenu.Enabled = True
End If

Else

  BotGuardar.Enabled = True

  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False

  BotMenu.Enabled = False

End If
  BloqueaControles
  ActualizarGrid
  Text2.Locked = False
  Text2.BackColor = vbWhite
  Text3.Locked = False
  Text3.BackColor = vbWhite

End Sub

Private Sub BotAgregar_GotFocus()
  LstClave.Visible = False
End Sub

Private Sub BotAnterior_Click()
  DataCompras.Recordset.MovePrevious
  HabilitarBotones
End Sub

Private Sub BotAnterior_GotFocus()
  LstClave.Visible = False
End Sub

Private Sub BotBuscar_Click()
  FoBuscComp.Show vbModal
End Sub

Private Sub BotGuardar_Click()
  Dim MiBaseDatos As Object

```



```
Dim MiRegistro As Object
Dim Confirm As String
```

```
Confirm = MsgBox("Se dispone a pasar toda la información
contenida en la forma de Compras hacia Historico Compras, y
no se podrá regresar.¿Desea continuar?", vbYesNo +
vbCritical, "ADVERTENCIA")
If Confirm = vbYes Then
```

```
    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
CondHist FROM TbCompras WHERE (((FechCompra)
Between #" & Format(Text2, "mm/dd/yy") & "# And #" &
Format(Text3, "mm/dd/yy") & "#))", dbOpenDynaset)
```

```
    Do Until MiRegistro.EOF
        MiRegistro.Edit
        MiRegistro!CondHist = 1
        MiRegistro.Update
        MiRegistro.MoveNext
    Loop
End If
```

```
End Sub
```

```
Private Sub BotAgregar_Click()
```

```
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(Folio) as Folio1 From TbCompras", dbOpenSnapshot)
```

```
DataCompras.Recordset.AddNew
HabilitarBotones
LabelFechCompr = Date
TexNoFac.SetFocus
If Not IsNull(MiRegistro!Folio1) Then
    TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
    TexFolio.BackColor = &H8000000F
    TexFolio.Locked = True
    Text1.BackColor = &H8000000F
    Text1.Locked = True
Else
    TexFolio = "0001"
    TexFolio.BackColor = &H8000000F
    TexFolio.Locked = True
    Text1.BackColor = &H8000000F
    Text1.Locked = True
```

```
End If
End Sub
```

```
Private Sub BotImprimir_Click()
FoOpRpComp.Show
End Sub
```

```
Private Sub BotMenu_Click()
FoMenuHist.Show
Unload FoComprasHist
End Sub
```

```
Private Sub BotModificar_Click()
DataCompras.Recordset.Edit
HabilitarBotones
TexFolio.BackColor = &H8000000F
```

```
TexFolio.Locked = True
Text1.BackColor = &H8000000F
Text1.Locked = True
TexNoFac.SetFocus
End Sub
```

```
Private Sub BotModificar_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotPrimero_Click()
DataCompras.Recordset.MoveFirst
HabilitarBotones
End Sub
```

```
Private Sub BotPrimero_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotSiguiente_Click()
```

```
If Not DataCompras.Recordset.EOF And
DataCompras.Recordset.RecordCount > 1 Then
    DataCompras.Recordset.MoveNext
End If
HabilitarBotones
End Sub
```

```
Private Sub BotSiguiente_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BotUltimo_Click()
DataCompras.Recordset.MoveLast
HabilitarBotones
End Sub
```

```
Private Sub BotUltimo_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub BtFiltrar_Click()
If IsDate(Text2) And IsDate(Text3) Then
    DataCompras.RecordSource = "SELECT TbCompras.*
FROM TbCompras WHERE (((FechCompra) Between #" &
Format(Text2, "mm/dd/yy") & "# And #" & Format(Text3,
"mm/dd/yy") & "#))"
    DataCompras.Refresh
    HabilitarBotones
ElseIf Not IsDate(Text2) Then
    MsgBox "Fecha incorrecta, favor de verificar",
vbInformation
    Text2 = ""
ElseIf Not IsDate(Text3) Then
    MsgBox "Fecha incorrecta, favor de verificar",
vbInformation
    Text3 = ""
End If
End Sub
```

```
Private Sub Form_Activate()
HabilitarBotones
```

```
Text2 = "01" & Format(Date, "-mm-yyyy")
Text3 = DiaUltimoDelMes(Date) & Format(Date, "-mm-
yyyy")
Text2.Locked = False
Text2.BackColor = vbWhite
Text3.Locked = False
Text3.BackColor = vbWhite
```

```
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
Text1.BackColor = &H8000000F
Text1.Locked = True
```

```
MiBol = True
End Sub
```

```
Private Sub Form_Load()
DataCompras.DatabaseName = MiBDStr
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
LlenarCombo LstClave, "TbProveedores", "CveProveedores",
"Folio"
```

```
MiBol = False
FoMenu.Hide
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub
```

```
Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataCompras.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H8000000F
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataCompras.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H8000000F
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is MSHFlexGrid Then
If DataCompras.Recordset.EditMode = 0 Then
.Controls(i).BackColor = &H80000004
Else
.Controls(i).BackColor = vbWhite
```

```
End If
End If
Next
End With
```

```
End Sub
```

```
Private Sub GDComp_DblClick()
If DataCompras.Recordset.EditMode <> 0 Then
If GDComp.TextMatrix(GDComp.Row, 0) = "" Then
FoDetallesCompras.ModoEdit = 2
ElseIf GDComp.TextMatrix(GDComp.Row, 0) <> "" Then
FoDetallesCompras.ModoEdit = 1
End If
FoDetallesCompras.Show vbModal
End If
End Sub
```

```
Private Sub GDComp_GotFocus()
LstClave.Visible = False
HighLight
End Sub
```

```
Private Sub GDComp_KeyPress(KeyAscii As Integer)
If DataCompras.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub GDComp_KeyUp(KeyCode As Integer, Shift As
Integer)
Dim MiBaseDatos As Object
Dim MiRegistro As Object
If KeyCode = 46 And GDComp.TextMatrix(GDComp.Row, 0)
<> "" Then
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("select
TbTmpDtllCompras.* FROM TbTmpDtllCompras WHERE
Dtl_ID =" & GDComp.TextMatrix(GDComp.Row, 0),
dbOpenDynaset)
If Not MiRegistro.EOF Then
MiRegistro.Delete
MiRegistro.Close
ActualizarGrid
End If
End If
End Sub
```

```
Private Sub LabelTotal_Change()
If MiBol = True Then
Text1 = ImporteLetra(LabelTotal)
End If
End Sub
```

```
Private Sub LstClave_DblClick()
TexCvePro = LstClave
LstClave.Visible = False
TexCvePro.SetFocus
End Sub
```

```
Private Sub TexCvePro_Change()
If DataCompras.Recordset.EditMode <> 0 Then
```

```

Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCvePro.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End If
End Sub

Private Sub TexCvePro_DbClick()
If DataCompras.Recordset.EditMode <> 0 Then
LstClave.Visible = True
End If
End Sub

Private Sub TexCvePro_GotFocus()
If DataCompras.Recordset.EditMode <> 0 Then
LstClave.Visible = True
HighLight
End If
End Sub

Private Sub TexCvePro_KeyPress(KeyAscii As Integer)
If DataCompras.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
TexCvePro = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End If
End Sub

Private Sub TexCvePro_LostFocus()
If DataCompras.Recordset.EditMode <> 0 And TexCvePro <> "" Then

Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
MiBaseDatos.OpenRecordset("SELECT TbProveedores.*
From TbProveedores WHERE CveProveedores = " &
TexCvePro & " ", dbOpenSnapshot)

If MiRegistro.EOF Then
MsgBox "La clave proporcionada no existe, favor de
verificar", vbExclamation
TexCvePro = ""
TexCvePro.SetFocus
End If
End If
End Sub

Private Sub TexNoFac_GotFocus()
LstClave.Visible = False
HighLight
End Sub

Private Sub TexNoFac_KeyPress(KeyAscii As Integer)
If DataCompras.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexNoFac_LostFocus()
If Not IsNumeric(TexNoFac) And TexNoFac <> "" Then
MsgBox "Poner solo números", vbExclamation
TexNoFac = ""
TexNoFac.SetFocus
End If
End Sub

```

```

Public Sub ActualizarGrid()
Dim MiSubtotal As Currency
Dim i As Long

```

```

i = 1
With GDComp
.Clear
.Cols = 15
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIILineasII_ID, DIIMarcas_ID, DIILineas_ID, DIISexo_ID,
DIIMaterial_ID, DIIEstilos_ID, DIICol_ID, DIINoPares,
DIIPrecUni, DIIPrecMen, DIIClaveCalzado, DIIPrecMay,
DIIMpporte, DIIDescripcion, DIIModelo, DII_ID FROM
TbDtlCompras WHERE DIIFolFac = " & TexFolio & " ",
dbOpenSnapshot)
If MiRegistro.EOF Then
GDComp.Clear
GDComp.Rows = 2
End If
Do Until MiRegistro.EOF
.Rows = MiRegistro.RecordCount + 2
.TextMatrix(i, 0) = MiRegistro!DII_ID
.TextMatrix(i, 4) = MiRegistro!DIINoPares
.TextMatrix(i, 5) = Format(MiRegistro!DIIPrecUni,
"currency")
.TextMatrix(i, 2) = Format(MiRegistro!DIIPrecMen,
"currency")
.TextMatrix(i, 7) = MiRegistro!DIIClaveCalzado
.TextMatrix(i, 3) = Format(MiRegistro!DIIPrecMay,
"currency")
.TextMatrix(i, 6) = Format(MiRegistro!DIIMpporte,
"currency")
.TextMatrix(i, 8) = MiRegistro!DIIModelo
.TextMatrix(i, 1) = MiRegistro!DIIDescripcion
.TextMatrix(i, 9) = MiRegistro!DIIMarcas_ID
.TextMatrix(i, 10) = MiRegistro!DIICol_ID
.TextMatrix(i, 11) = MiRegistro!DIIEstilos_ID
.TextMatrix(i, 12) = MiRegistro!DIIMaterial_ID
.TextMatrix(i, 13) = MiRegistro!DIISexo_ID
.TextMatrix(i, 14) = MiRegistro!DIILineas_ID
MiSubtotal = MiSubtotal + MiRegistro!DIIMpporte
i = i + 1

MiRegistro.MoveNext
Loop
LabelSubTotal = Format(MiSubtotal, "currency")
LabelIVA = Format(MiSubtotal * (IVA / 100), "currency")
LabelTotal = Format(CCur(LabelIVA) + CCur(MiSubtotal),
"currency")
MiRegistro.Close
.ColWidth(0) = 0
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 5) = "Precio Unitario"
.ColAlignment(5) = 7
.ColWidth(5) = 1050

```

```
.TextMatrix(0, 2) = "Precio Menenudeo"
.ColAlignment(2) = 7
.ColWidth(2) = 1300
.ColWidth(7) = 0
.TextMatrix(0, 3) = "Precio Mayoreo"
.ColAlignment(3) = 7
.ColWidth(3) = 1100
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000
.ColWidth(8) = 0
.TextMatrix(0, 1) = "Descripción"
.ColWidth(1) = 3150
.ColWidth(9) = 0
.ColWidth(10) = 0
.ColWidth(11) = 0
.ColWidth(12) = 0
.ColWidth(13) = 0
.ColWidth(14) = 0
.ColWidth(15) = 0
End With
End Sub
```

DETALLES COMPRAS

Option Explicit

```
Private Sub BtClave_Click()
Dim añadir As String
```

```
If CombMarca = "" Or CombSexo = "" Or CombEstilo = "" Or
CombLinea = "" Or CombMater = "" Or CombColor = "" Or
Trim(TEXModel) = "" Or Trim(TEXNoPar) = "" Or
Trim(TEXPrecUni) = "" Then
añadir = MsgBox("Para poder generar la clave, debe llenar
los campos", vbInformation)
Else
```

```
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbMarcas WHERE Marcas_ID =" &
CombMarca.ItemData(CombMarca.ListIndex),
dbOpenDynaset)
```

```
TexCveCalz = MiRegistro!Iniciales
TexDescrip = CombMarca
MiRegistro.Close
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbSexo WHERE Sexo_ID =" &
CombSexo.ItemData(CombSexo.ListIndex), dbOpenDynaset)
```

```
TexCveCalz = TexCveCalz + MiRegistro!Iniciales
TexDescrip = TexDescrip & ", " & CombSexo
MiRegistro.Close
```

```
TexDescrip = TexDescrip & ", " & CombEstilo
```

```
TexCveCalz = TexCveCalz & TEXModel
```

```
TexDescrip = TexDescrip & ", " & CombMater
```

```
TexDescrip = TexDescrip & ", " & CombColor
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Lineas From TbLineas WHERE Lineas_ID =" &
CombLinea.ItemData(CombLinea.ListIndex),
dbOpenDynaset)
TexCveCalz = TexCveCalz + CStr(MiRegistro!Lineas)
TexDescrip = TexDescrip & ", " & CombLinea
MiRegistro.Close
```

```
TexCveCalz = UCase(TexCveCalz)
End If
End Sub
```

```
Private Sub CombColor_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub CombColor_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End Sub
```

```
Private Sub CombEstilo_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub CombEstilo_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End Sub
```

```
Private Sub CombLinea_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub CombLinea_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End Sub
```

```
Private Sub CombMarca_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub CombMarca_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End Sub
```

```
Private Sub CombMater_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub CombMater_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
```

```

End Sub

Private Sub CombSexo_GotFocus()
LstClave.Visible = False
End Sub

Private Sub CombSexo_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{tab}"
End If
End Sub

Private Sub Command1_Click()
Dim Mult As Variant
Dim Mult2 As Variant
Dim Mult3 As Variant
Dim Res As Variant
Dim Res2 As Variant
Dim añadir As String
Dim CLAVE As String

If TexCveCalz = "" Then
    CLAVE = MsgBox("Para poder guardar, debe generar la
Clave de Calzado", vbInformation)
Else
    If Trim(TexModel) = "" Or Trim(TexPrecUni) = "" Or
Trim(TexNoPar) = "" Then
        añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
    Else
        Mult = Format(TexPrecUni, "currency") * (Ganancia /
100)
        Res = Mult + Format(TexPrecUni, "currency")
        LabelPrecMen = Format((Res), "currency")

        Mult2 = Format(LabelPrecMen, "currency") * (Descuento
/ 100)
        Res2 = Format(LabelPrecMen, "currency") - Mult2
        LabelPrecMay = Format((Res2), "currency")

        Mult3 = Format(TexPrecUni, "currency") *
Format(TexNoPar, "currency")
        LabelTotPag = Format((Mult3), "currency")

        Dim MiBaseDatos As Object
        Dim MiRegistro As Object

        Set MiBaseDatos = OpenDatabase(MiBDStr)
        If ModoEdit = 1 Then

            Set                MiRegistro                =
MiBaseDatos.OpenRecordset("SELECT        DIILineasII_ID,
DIIMarcas_ID, DIILineas_ID, DIISexo_ID, DIIMaterial_ID,
DIIEstilos_ID, DIICol_ID, DIINoPares, DIIPrecUni,
DIIPrecMen, DIIClaveCalzado, DIIPrecMay, DIIImporte,
DIIDescripcion, DIIModelo FROM TbTmpDtIICompras
WHERE DII_ID =" & TexDetalleID, dbOpenDynaset)

            MiRegistro.Edit

            MiRegistro!DIINoPares = TexNoPar
            MiRegistro!DIIPrecUni = TexPrecUni
            MiRegistro!DIIPrecMen = LabelPrecMen
            MiRegistro!DIIClaveCalzado = TexCveCalz
            MiRegistro!DIIPrecMay = LabelPrecMay
            MiRegistro!DIIImporte = LabelTotPag
            MiRegistro!DIIDescripcion = TexDescrip
            MiRegistro!DIIModelo = TexModel
            MiRegistro!DIIMarcas_ID = TexMarcaID

```

```

MiRegistro!DIICol_ID = TexColoresID
MiRegistro!DIIEstilos_ID = TexEstilosID
MiRegistro!DIIMaterial_ID = TexMaterialID
MiRegistro!DIISexo_ID = TexSexoID
MiRegistro!DIILineas_ID = TexLineasID
MiRegistro.Update
Else

    Set                MiRegistro                =
MiBaseDatos.OpenRecordset("SELECT        DIILineasII_ID,
DIIMarcas_ID, DIILineas_ID, DIISexo_ID, DIIMaterial_ID,
DIIEstilos_ID, DIICol_ID, DIINoPares, DIIPrecUni,
DIIPrecMen, DIIClaveCalzado, DIIPrecMay, DIIImporte,
DIIDescripcion, DIIModelo, DII_ID, DIIFolFac FROM
TbTmpDtIICompras ", dbOpenDynaset)

    MiRegistro.AddNew
    MiRegistro!DIINoPares = TexNoPar
    MiRegistro!DIIPrecUni = TexPrecUni
    MiRegistro!DIIPrecMen = LabelPrecMen
    MiRegistro!DIIClaveCalzado = TexCveCalz
    MiRegistro!DIIPrecMay = LabelPrecMay
    MiRegistro!DIIImporte = LabelTotPag
    MiRegistro!DIIDescripcion = TexDescrip
    MiRegistro!DIIModelo = TexModel
    MiRegistro!DII_ID = TexDetalleID
    MiRegistro!DIIFolFac = TexFolioFac
    MiRegistro!DIIMarcas_ID = TexMarcaID
    MiRegistro!DIICol_ID = TexColoresID
    MiRegistro!DIIEstilos_ID = TexEstilosID
    MiRegistro!DIIMaterial_ID = TexMaterialID
    MiRegistro!DIISexo_ID = TexSexoID
    MiRegistro!DIILineas_ID = TexLineasID
    MiRegistro.Update

    Set                MiRegistro                =
MiBaseDatos.OpenRecordset("SELECT        Ventas_ID,
FechaIngreso, DIIMarcas_ID, DIILineas_ID, DIISexo_ID,
DIIMaterial_ID, DIIEstilos_ID, DIICol_ID, Existencias,
ModoEdit, CostUnit, CostMen, ClaveCalzado, CostMay,
Descripcion, DIIModelo FROM TbTmpInventario ",
dbOpenDynaset)

    MiRegistro.AddNew
    MiRegistro!Existencias = TexNoPar
    MiRegistro!CostUnit = TexPrecUni
    MiRegistro!CostMen = LabelPrecMen
    MiRegistro!ClaveCalzado = TexCveCalz
    MiRegistro!CostMay = LabelPrecMay
    MiRegistro!Descripcion = TexDescrip
    MiRegistro!FechaIngreso =

FoCompras.LabelFechCompr
    MiRegistro!DIIModelo = TexModel
    MiRegistro!DIIMarcas_ID = TexMarcaID
    MiRegistro!DIICol_ID = TexColoresID
    MiRegistro!DIIEstilos_ID = TexEstilosID
    MiRegistro!DIIMaterial_ID = TexMaterialID
    MiRegistro!DIISexo_ID = TexSexoID
    MiRegistro!DIILineas_ID = TexLineasID
    MiRegistro!Ventas_ID = TexInvenID
    MiRegistro!ModoEdit = EditMod
    MiRegistro.Update

End If
Unload Me
FoCompras.Show
End If
End Sub

Private Sub Command2_Click()
Unload Me

```

```

FoCompras.Show
End Sub

Private Sub Form_Activate()
Dim miVar As Variant
LlenarCombo CombColor, "TbColores", "Colores", "Col_ID"
LlenarCombo CombEstilo, "TbEstilos", "Estilos", "Estilos_ID"
LlenarCombo CombLinea, "TbLineas", "Lineas", "Lineas_ID"
LlenarCombo CombMarca, "TbMarcas", "Marcas",
"Marcas_ID"
LlenarCombo CombMater, "TbMaterial", "Material",
"Material_ID"
LlenarCombo CombSexo, "TbSexo", "Sexo", "Sexo_ID"

If ModoEdit = 1 Then

    TexDetalleID =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
0)
    TexNoPar =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
4)
    TexPrecUni =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
5)
    LabelPrecMen =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
2)
    TexCveCalz =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
7)
    LabelPrecMay =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
3)
    LabelTotPag =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
6)
    TexDescrip =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
1)
    TexModel =
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
8)
    TexFolioFac = FoCompras.TexFolio
    TexCveCalz.Enabled = False
    BusquedaIdCombo
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
9), CombMarca
    BusquedaIdCombo
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
10), CombColor
    BusquedaIdCombo
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
11), CombEstilo
    BusquedaIdCombo
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
12), CombMater
    BusquedaIdCombo
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
13), CombSexo
    BusquedaIdCombo
FoCompras.GDComp.TextMatrix(FoCompras.GDComp.Row,
14), CombLinea
    Option1_Click
    Option1.Enabled = False
    Option2.Enabled = False
    LstClave.Visible = False
Else
    TexFolioFac = FoCompras.TexFolio
    TexDetalleID = IncrementadorID("TbDtllCompras")

```

```

Option1_Click
End If
TexFolioFac.BackColor = &H8000000F
TexFolioFac.Locked = True
TexDescrip.BackColor = &H8000000F
TexDescrip.Locked = True
TexCveCalz.BackColor = &H8000000F
TexCveCalz.Locked = True

```

```
End Sub
```

```

Private Sub CombColor_Click()
If CombColor.ListIndex <> -1 Then
    TexColoresID =
CombColor.ItemData(CombColor.ListIndex)
End If
End Sub

```

```

Private Sub CombEstilo_Click()
If CombEstilo <> "" Then
    TexEstilosID =
CombEstilo.ItemData(CombEstilo.ListIndex)
End If
End Sub

```

```

Private Sub CombLinea_Click()
If CombLinea <> "" Then
    TexLineasID =
CombLinea.ItemData(CombLinea.ListIndex)
End If
End Sub

```

```

Private Sub CombMarca_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
If CombMarca <> "" Then
    TexMarcaID =
CombMarca.ItemData(CombMarca.ListIndex)
    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbMarcas WHERE Marcas_ID =" &
CombMarca.ItemData(CombMarca.ListIndex),
dbOpenDynaset)
End If
End Sub

```

```

Private Sub CombMater_Click()
If CombMater <> "" Then
    TexMaterialID =
CombMater.ItemData(CombMater.ListIndex)
End If
End Sub

```

```

Private Sub CombSexo_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
If CombSexo <> "" Then
    TexSexoID = CombSexo.ItemData(CombSexo.ListIndex)
    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbSexo WHERE Sexo_ID =" &
CombSexo.ItemData(CombSexo.ListIndex), dbOpenDynaset)
End If
End Sub

```

```
Private Sub Form_Load()
FoCompras.Hide
LlenarCombo LstClave, "TbInventario", "ClaveCalzado"
TexInvenID = IncrementadorID("TbVentas")
End Sub
```

```
Private Sub LstClave_DblClick()
TexCveCalz = LstClave
LstClave.Visible = False
TexCveCalz.SetFocus
TexPrecUni.BackColor = vbWhite
TexPrecUni.Enabled = True
TexPrecUni = ""
TexNoPar.BackColor = vbWhite
TexNoPar.Enabled = True
TexNoPar = ""
End Sub
```

```
Private Sub Option1_Click()
If Option1.Value = True Then
EditMod = "1"
LstClave.Visible = True
CombMarca.BackColor = ColorGris
CombColor.BackColor = ColorGris
CombEstilo.BackColor = ColorGris
CombMater.BackColor = ColorGris
CombSexo.BackColor = ColorGris
CombLinea.BackColor = ColorGris
```

```
TexPrecUni.BackColor = ColorGris
TexNoPar.BackColor = ColorGris
TexModel.BackColor = ColorGris
TexModel.Enabled = False
TexNoPar.Enabled = False
TexPrecUni.Enabled = False
```

```
CombMarca.Enabled = False
CombColor.Enabled = False
CombEstilo.Enabled = False
CombMater.Enabled = False
CombSexo.Enabled = False
CombLinea.Enabled = False
BtClave.Enabled = False
```

```
TexCveCalz.TabStop = True
```

```
Else
EditMod = "2"
CombMarca.ListIndex = -1
CombColor.ListIndex = -1
CombEstilo.ListIndex = -1
CombMater.ListIndex = -1
CombSexo.ListIndex = -1
CombLinea.ListIndex = -1
```

```
CombMarca.BackColor = vbWhite
CombColor.BackColor = vbWhite
CombEstilo.BackColor = vbWhite
CombMater.BackColor = vbWhite
CombSexo.BackColor = vbWhite
CombLinea.BackColor = vbWhite
```

```
TexCveCalz.BackColor = ColorGris
TexCveCalz.Enabled = True
TexCveCalz = ""
TexModel.BackColor = vbWhite
TexModel.Enabled = True
TexModel = ""
```

```
TexNoPar.BackColor = vbWhite
TexNoPar.Enabled = True
TexNoPar = ""
TexPrecUni.BackColor = vbWhite
TexPrecUni.Enabled = True
TexPrecUni = ""
TexDescrip.Enabled = True
TexDescrip = ""
TexPrecUni.BackColor = vbWhite
TexPrecUni.Enabled = True
TexPrecUni = ""
TexNoPar.BackColor = vbWhite
TexNoPar.Enabled = True
TexNoPar = ""
```

```
CombMarca.Enabled = True
CombColor.Enabled = True
CombEstilo.Enabled = True
CombMater.Enabled = True
CombSexo.Enabled = True
CombLinea.Enabled = True
BtClave.Enabled = True
```

```
TexCveCalz.TabStop = False
TexModel.SetFocus
```

```
End If
```

```
End Sub
```

```
Private Sub Option2_Click()
Option1_Click
End Sub
```

```
Private Sub TexCveCalz_Change()
```

```
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCveCalz.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
```

```
End Sub
```

```
Private Sub TexCveCalz_DblClick()
If Option1 = True Then
LstClave.Visible = True
End If
End Sub
```

```
Private Sub TexCveCalz_GotFocus()
```

```
If TexCveCalz <> "" Then
HighLight
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz & "", dbOpenSnapshot)
```

```
If Not MiTabla.EOF Then
TexPrecUni =
Format(CCur(PonCero(MiTabla.CostUnit)), "currency")
TexMarcaID = MiTabla.DllMarcas_ID
TexSexoID = MiTabla.DllSexo_ID
```

```

TexEstilosID = MiTabla.DIIEstilos_ID
TexMaterialID = MiTabla.DIIMaterial_ID
TexColoresID = MiTabla.DIICol_ID
TexLineasID = MiTabla.DIILineas_ID
TexModel = MiTabla.DIIModelo
TexDescrip = MiTabla.Descripcion
End If
End If
End Sub

Private Sub TexCveCalz_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    TexCveCalz = LstClave
    SendKeys "{tab}"
    LstClave.Visible = False
End If
End Sub

Private Sub TexEstilosID_Change()
BusquedaIdCombo TexEstilosID, CombEstilo
End Sub

Private Sub TexLineasID_Change()

BusquedaIdCombo TexLineasID, CombLinea

End Sub

Private Sub TexMarcaID_Change()
BusquedaIdCombo TexMarcaID, CombMarca
End Sub

Private Sub TexMaterialID_Change()
BusquedaIdCombo TexMaterialID, CombMater
End Sub

Private Sub TexModel_GotFocus()
LstClave.Visible = False
HighLight
End Sub

Private Sub TexModel_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{tab}"
End If
End Sub

Private Sub TexModel_LostFocus()
If Not IsNumeric(TexModel) And TexModel <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexModel = ""
    TexModel.SetFocus
End If
LstClave.Visible = False
End Sub

Private Sub TexColoresID_Change()
BusquedaIdCombo TexColoresID, CombColor
End Sub

Private Sub TexNoPar_Change()
Dim Mult As Variant
Dim Mult2 As Variant
Dim Mult3 As Variant

```

```

Dim Res As Variant
Dim Res2 As Variant
If Not IsNumeric(TexNoPar) And TexNoPar <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexNoPar = ""
    TexNoPar.SetFocus
Else
    Mult3 = PonCero(TexPrecUni) * PonCero(TexNoPar)
    LabelTotPag = FormatCurrency(Mult3)

    Mult = PonCero(TexPrecUni) * (Ganancia / 100)
    Res = PonCero(Mult) + PonCero(TexPrecUni)
    LabelPrecMen = FormatCurrency(Res)

    Mult2 = LabelPrecMen * (Descuento / 100)
    Res2 = LabelPrecMen - Mult2
    LabelPrecMay = FormatCurrency(Res2)

End If
End Sub

Private Sub TexNoPar_GotFocus()
LstClave.Visible = False
HighLight
End Sub

Private Sub TexNoPar_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{tab}"
End If
End Sub

Private Sub TexNoPar_LostFocus()
If Not IsNumeric(TexNoPar) And TexNoPar <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexNoPar = ""
    TexNoPar.SetFocus
End If
LstClave.Visible = False
End Sub

Private Sub TexPrecUni_Change()
Dim Mult As Variant
Dim Mult2 As Variant
Dim Mult3 As Variant
Dim Res As Variant
Dim Res2 As Variant
If Not IsNumeric(TexPrecUni) And TexPrecUni <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexPrecUni = ""
    TexPrecUni.SetFocus
Else
    Mult = PonCero(TexPrecUni) * (Ganancia / 100)
    Res = PonCero(Mult) + PonCero(TexPrecUni)
    LabelPrecMen = FormatCurrency(Res)

    Mult2 = LabelPrecMen * (Descuento / 100)
    Res2 = LabelPrecMen - Mult2
    LabelPrecMay = FormatCurrency(Res2)

    Mult3 = PonCero(TexPrecUni) * PonCero(TexNoPar)
    LabelTotPag = FormatCurrency(Mult3)
End If
End Sub

```



```
Private Sub TexPrecUni_GotFocus()
LstClave.Visible = False
HighLight
End Sub
```

```
Private Sub TexPrecUni_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End Sub
```

```
Private Sub TexPrecUni_LostFocus()
TexPrecUni = Format(CCur(PonCero(TexPrecUni)),
"currency")
End Sub
```

```
Private Sub TexSexoID_Change()
BusquedaIdCombo TexSexoID, CombSexo
End Sub
```

DETALLES VENTAS

```
Dim Devol As Integer
```

```
Private Sub BotAceptar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
Dim añadir As String
```

```
If Trim(TexCveCalz) = "" Or Trim(TexCostUni) = "" Or
Trim(TexNoPar) = "" Then
añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
Else
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
```

```
If ModoEdit = 1 Then
Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT DIIlaveProveedor,
DIIlMedidas, DIIlaveCalzado, DIIlNoPares, DIIlCostoUnit,
DIIlCostoPublico, DIIlImporte, DIIlDescripcion FROM
TbTmpDtllVentas WHERE DII_ID =" & TexDetalleID,
dbOpenDynaset) 'Costante para abrir en modo solo lectura y si
fuera dbOpenDynaset era para lectura/escritura'
```

```
MiRegistro.Edit
MiRegistro!DIIlaveCalzado = TexCveCalz
MiRegistro!DIIlaveProveedor = TexCvePro
MiRegistro!DIIlNoPares = TexNoPar
MiRegistro!DIIlCostoUnit = TexCostUni
MiRegistro!DIIlCostoPublico = TexCostPubl
MiRegistro!DIIlImporte = LabelImporte
MiRegistro!DIIlDescripcion = TexDescrip
MiRegistro!DIIlMedidas = CombMedidas
MiRegistro.Update
MiRegistro.Close
```

```
Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = " & TexCveCalz & """,
dbOpenDynaset) 'Costante para abrir en modo solo lectura y si
fuera dbOpenDynaset era para lectura/escritura'
MiRegistro.Edit
Devol = Devol - TexNoPar
MiRegistro!Existencias = MiRegistro!Existencias +
CLng(Devol)
```

```
MiRegistro.Update
```

```
Else
```

```
Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT ModoEdit,
DIIlaveProveedor, DIIlMedidas, DIIlaveCalzado,
DIIlNoPares, DIIlCostoUnit, DIIlCostoPublico, DIIlImporte,
DIIlDescripcion, DII_ID, DIIlFolFac FROM TbTmpDtllVentas
", dbOpenDynaset)
```

```
MiRegistro.AddNew
MiRegistro!DIIlaveCalzado = TexCveCalz
MiRegistro!DIIlaveProveedor = TexCvePro
MiRegistro!DIIlNoPares = TexNoPar
MiRegistro!DIIlCostoUnit = TexCostUni
MiRegistro!DIIlCostoPublico = TexCostPubl
MiRegistro!DII_ID = TexDetalleID
MiRegistro!DIIlFolFac = TexFolioFac
MiRegistro!DIIlImporte = LabelImporte
MiRegistro!DIIlDescripcion = TexDescrip
MiRegistro!DIIlMedidas = CombMedidas
MiRegistro!ModoEdit = 1
MiRegistro.Update
MiRegistro.Close
```

```
End If
Unload Me
End If
End Sub
```

```
Private Sub BotAceptar_GotFocus()
LstClave.Visible = False
LstClave2.Visible = False
End Sub
```

```
Private Sub BotCancelar_Click()
Unload Me
FoVentas.Show
End Sub
```

```
Private Sub CombMedidas_Click()
If CombMedidas <> "" Then
TexMedidasID =
CombMedidas.ItemData(CombMedidas.ListIndex)
End If
End Sub
```

```
Private Sub BotCancelar_GotFocus()
LstClave.Visible = False
LstClave2.Visible = False
End Sub
```

```
Private Sub Form_Activate()
TexCveCalz.SetFocus
```

```
If ModoEdit = 1 Then
TexDetalleID =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 0)
TexCveCalz =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 7)
TexNoPar =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 4)
```

```

TexCostUni =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 3)
TexCostPubl =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 5)
LabelImporte =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 6)
TexDescrip =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 2)
TexCvePro =
FoVentas.GDVentas.TextMatrix(FoVentas.GDVentas.Row, 1)
TexFolioFac = FoVentas.TextFolio
Else
TexFolioFac = FoVentas.TextFolio
TexDetalleID = IncrementadorID("TbDtllVentas")
End If
Devol = PonCero(TexNoPar)
End Sub

```

```

Private Sub TexDesc_Validate(Cancel As Boolean)
If Not IsNumeric(TexDesc) And TexDesc <> "" Then
MsgBox "Poner solo números", vbExclamation
TexDesc = ""
TexDesc.SetFocus
End If
End Sub

```

```

Private Sub Form_Load()
LlenarCombo LstClave2, "TbProveedores",
"CveProveedores", "Folio"
LlenarCombo LstClave, "TbInventario", "ClaveCalzado",
"Ventas_ID"
End Sub

```

```

Private Sub LstClave_DbClick()
TexCveCalz = LstClave
LstClave.Visible = False
TexCveCalz.SetFocus
End Sub

```

```

Private Sub LstClave_GotFocus()
LstClave2.Visible = False
End Sub

```

```

Private Sub LstClave2_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub TexCostPubl_LostFocus()
LabelImporte = TexNoPar * TexCostPubl
End Sub

```

```

Private Sub TexCveCalz_Change()

Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCveCalz.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True

End Sub

```

```

Private Sub TexCveCalz_DbClick()

```

```

LstClave.Visible = True

```

```

End Sub

```

```

Private Sub TexCveCalz_GotFocus()
LstClave.Visible = False
LstClave2.Visible = False
End Sub

```

```

Private Sub TexCveCalz_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexCveCalz = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End Sub

```

```

Private Sub TexCveCalz_LostFocus()

```

```

If TexCveCalz <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz & "'", dbOpenSnapshot)

```

```

If Not MiTabla.EOF Then 'Con este EOF quiere decir
que la clausula de arriba no se encontro un registro duplicado.'

```

```

TexCostUni =
Format(CCur(PonCero(MiTabla.CostUnit)), "currency")
TexCostPubl =
Format(CCur(PonCero(MiTabla.CostMen)), "currency")
TexDescrip = MiTabla.Descripcion
Else
MsgBox "La clave proporcionada no existe, trate de
nuevo", vbExclamation
TexCveCalz = ""
TexCveCalz.SetFocus

```

```

End If
End If

```

```

End Sub

```

```

Private Sub TexNoPar_GotFocus()
LstClave.Visible = False
LstClave2.Visible = False
End Sub

```

```

Private Sub TexNoPar_LostFocus()
LabelImporte = PonCero(TexNoPar) * PonCero(TexCostPubl)

```

```

Dim MiBD As Object
Dim MiTabla As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz & "'", dbOpenSnapshot)

```

```

If Not MiTabla.EOF Then

```

```

If CLng(MiTabla.Existencias) <
CLng(PonCero(TexNoPar)) Then
  MsgBox "Las existencias son insuficiente para la venta ",
vbExclamation
  TexNoPar = ""
  TexNoPar.SetFocus
End If
Else
End If

End Sub

```

```

Private Sub TexNoPar_Validate(Cancel As Boolean)
If Not IsNumeric(TexNoPar) And TexNoPar <> "" Then
  MsgBox "Poner solo números", vbExclamation
  TexNoPar = ""
  TexNoPar.SetFocus
End If

End Sub

```

```

Private Sub TexCvePro_Change()

Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexCvePro.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True

End Sub

```

```

Private Sub TexCvePro_DbIcClick()

LstClave2.Visible = True

End Sub

```

```

Private Sub TexCvePro_GotFocus()
LstClave.Visible = False
LstClave2.Visible = False
If TexCvePro <> "" Then
  TexCvePro = LstClave2

  Set MiBD = OpenDatabase(MiBDStr)
  Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE
CveProveedores = " & TexCvePro & "", dbOpenSnapshot)

  If MiTabla.EOF Then
    MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
    TexCvePro = ""
    TexCvePro.SetFocus
  End If
End If

End Sub

```

```

Private Sub TexCvePro_KeyPress(KeyAscii As Integer)

If KeyAscii = 13 Then
  TexCvePro = LstClave2
  SendKeys "{tab}"
  LstClave2.Visible = False
End If

End Sub

```

```

Private Sub LstClave2_DbIcClick()
TexCvePro = LstClave2
LstClave2.Visible = False
LstClave.Visible = False
End Sub

```

DEVOLUCION HISTORICO

```

Option Explicit
Dim Mibool As Boolean

```

```

Private Sub HabilitarBotones()
Dim MiReg As String

```

```

If DataDevolucion.Recordset.EditMode = 0 Then
If Not DataDevolucion.Recordset.EOF Then
  MiReg = TexFolDevol
  DataDevolucion.Recordset.MoveLast
End If

```

```

If DataDevolucion.Recordset.RecordCount <= 0 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotMenu.Enabled = True

```

```

ElseIf DataDevolucion.Recordset.RecordCount <= 1
Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotMenu.Enabled = True

```

```

Else
  DataDevolucion.Recordset.FindFirst "FolioDevolID ="
& PonCero(MiReg)
  If DataDevolucion.Recordset.AbsolutePosition = 0
Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True

```

```

  ElseIf DataDevolucion.Recordset.AbsolutePosition =
DataDevolucion.Recordset.RecordCount - 1 Then

```

```

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False

```

```

  Else
    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
  End If

```

```

  BotMenu.Enabled = True
End If

```

```

Else

```

```

  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False

```

```

BotMenu.Enabled = False

End If
BloqueaControles
End Sub

```

```

Private Sub BotAgregar_Click()

```

```

DataDevolucion.Recordset.AddNew
LabelFechDevol = Date
HabilitarBotones
OptionCambio.Value = True
TexFolDevol = IncrementadorID("TbDevolucion")
TexDesc1.BackColor = ColorGris
TexDesc2.BackColor = ColorGris
TexDesc3.BackColor = ColorGris
TexFolioVenta2.BackColor = ColorGris
TexNoPares1.BackColor = ColorGris
TexNoPares3.BackColor = ColorGris

```

```

If OptionCambio.Value = True Then

```

```

    TexFolioVenta1.Enabled = True
    TexCveCalz1.Enabled = True
    TexParCamb1.Enabled = True
    TexCveCalz2.Enabled = True
    TexNoPares2.Enabled = True
    TexCvePro.Enabled = True
    TexFolioVenta1.BackColor = vbWhite
    TexCveCalz1.BackColor = vbWhite
    TexParCamb1.BackColor = vbWhite
    TexCveCalz2.BackColor = vbWhite
    TexNoPares2.BackColor = vbWhite
    TexCvePro.BackColor = vbWhite

```

```

    TexFolioVenta3.Enabled = False
    TexCveCalz3.Enabled = False
    TexParCamb3.Enabled = False
    TexFolioVenta3.BackColor = ColorGris
    TexCveCalz3.BackColor = ColorGris
    TexParCamb3.BackColor = ColorGris

```

```

Else

```

```

    TexFolioVenta1.Enabled = False
    TexCveCalz1.Enabled = False
    TexParCamb1.Enabled = False
    TexCveCalz2.Enabled = False
    TexNoPares2.Enabled = False
    TexCvePro.Enabled = False
    TexFolioVenta1.BackColor = ColorGris
    TexCveCalz1.BackColor = ColorGris
    TexParCamb1.BackColor = ColorGris
    TexCveCalz2.BackColor = ColorGris
    TexNoPares2.BackColor = ColorGris
    TexCvePro.BackColor = ColorGris

```

```

    TexFolioVenta3.Enabled = True
    TexCveCalz3.Enabled = True
    TexParCamb3.Enabled = True
    TexFolioVenta3.BackColor = vbWhite
    TexCveCalz3.BackColor = vbWhite
    TexParCamb3.BackColor = vbWhite

```

```

End If
Check1.Value = 1

End Sub

```

```

Private Sub BotAnterior_Click()
DataDevolucion.Recordset.MovePrevious
HabilitarBotones
End Sub

```

```

Private Sub BotAnterior_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

```

```

Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
    DataDevolucion.Recordset.CancelUpdate
    HabilitarBotones
End If
End Sub

```

```

Private Sub BotCancelar_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
List3.Visible = False
End Sub

```

```

Private Sub BotGuardar_Click()
Dim añadir As String
Dim MiRegistro As Object
Dim MiRegistro12 As Object
Dim MiRegistro1 As Object
Dim MiBaseDatos As Object
Dim MiRegistro2 As Object

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)

```

```

If OptionCambio.Value = True Then

```

```

    If Trim(TexCveCalz2) = "" Or Trim(TexNoPares2) = "" Or
Trim(TexCvePro) = "" Then

```

```

        añadir = MsgBox("Si ha seleccionado Cambio debe llenar
los campos de Calzado a cambio. ", vbInformation)

```

```

    Else

```

```

        Set MiRegistro1 =
MiBaseDatos.OpenRecordset("SELECT FolioVenta,
CheckCancel, TipoCancel FROM TbFacturas WHERE
FolioVenta = " & TexFolioVenta1 & "", dbOpenDynaset)

```

```

        If Not MiRegistro1.EOF Then
            MiRegistro1.Edit
            MiRegistro1!CheckCancel = 1
            MiRegistro1!TipoCancel = "Cancelado por Cambio"

```

```

    MiRegistro1.Update
End If

Set      MiRegistro      =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = " & TexCveCalz1 &
"", dbOpenDynaset)

    MiRegistro.Edit
    MiRegistro!Existencias = MiRegistro!Existencias +
CLng(TexParCamb1)
    MiRegistro.Update
    MiRegistro.Close

Set      MiRegistro      =
MiBaseDatos.OpenRecordset("SELECT      DIINoPares,
DIIClaveCalzado FROM      TbDtIIVentas WHERE
DIIClaveCalzado ="" & TexCveCalz1 & "" AND DIIFolFac= ""
& TexFolioVenta1 & """, dbOpenDynaset)

    MiRegistro.Edit
    MiRegistro!DIINoPares = MiRegistro!DIINoPares -
CLng(TexParCamb1)
    MiRegistro.Update
    MiRegistro.Close

Set      MiRegistro      =
MiBaseDatos.OpenRecordset("SELECT DIIClaveProveedor,
DIIMedidas, DIIClaveCalzado, DIINoPares, DIICostoUnit,
DIICostoPublico, DIIMpporte, DIIDescripcion, DII_ID,
DIIFolFac FROM TbDtIIVentas ", dbOpenDynaset)

    MiRegistro.AddNew
    MiRegistro!DIIClaveCalzado = TexCveCalz2
    MiRegistro!DIIClaveProveedor = TexCvePro
    MiRegistro!DIINoPares = PonCero(TexNoPares2)
    MiRegistro!DIICostoUnit = PonCero(TexCostUni2)
    MiRegistro!DIICostoPublico = PonCero(TexCostPubl2)
    MiRegistro!DII_ID = IncrementadorID("TbDtIIVentas")
    MiRegistro!DIIFolFac = TexFolioVenta2
    MiRegistro!DIIMpporte = PonCero(LabelImporte2)
    MiRegistro!DIIDescripcion = TexDesc2
    MiRegistro.Update
    MiRegistro.Close

Set      MiRegistro      =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = " & TexCveCalz2 &
"", dbOpenDynaset)
    MiRegistro.Edit
    MiRegistro!Existencias = MiRegistro!Existencias -
CLng(PonCero(TexNoPares2))
    MiRegistro.Update
    MiRegistro.Close

End If

Else

Set      MiRegistro12      =
MiBaseDatos.OpenRecordset("SELECT      FolioVenta,
CheckCancel, TipoCancel FROM TbFacturas WHERE
FolioVenta= "" & TexFolioVenta3 & """, dbOpenDynaset)

If Not MiRegistro12.EOF Then
    MiRegistro12.Edit
    MiRegistro12!CheckCancel = 1
    MiRegistro12!TipoCancel = "Cancelado por
Devolución"
    MiRegistro12.Update

```

```

End If

Set MiRegistro2 = MiBaseDatos.OpenRecordset("SELECT
DIINoPares, DIIClaveCalzado FROM TbDtIIVentas WHERE
DIIClaveCalzado ="" & TexCveCalz3 & "" AND DIIFolFac= ""
& TexFolioVenta3 & """, dbOpenDynaset)

    MiRegistro2.Edit
    MiRegistro2!DIINoPares = MiRegistro2!DIINoPares -
CLng(TexParCamb3)
    MiRegistro2.Update
    MiRegistro2.Close

End If

DataDevolucion.Recordset.Update
DataDevolucion.Refresh
HabilitarBotones
End Sub

Private Sub BotGuardar_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
List3.Visible = False
End Sub

Private Sub BotMenu_Click()
FoMenuHist.Show
Unload FoDevolHist
End Sub

Private Sub BotModificar_Click()
DataDevolucion.Recordset.Edit
HabilitarBotones
TexDesc1.BackColor = ColorGris
TexDesc2.BackColor = ColorGris
TexDesc3.BackColor = ColorGris
TexFolioVenta2.BackColor = ColorGris
TexNoPares1.BackColor = ColorGris
TexNoPares3.BackColor = ColorGris

If OptionCambio.Value = True Then

    TexFolioVenta1.Enabled = True
    TexCveCalz1.Enabled = True
    TexParCamb1.Enabled = True
    TexCveCalz2.Enabled = True
    TexNoPares2.Enabled = True
    TexCvePro.Enabled = True
    TexFolioVenta1.BackColor = vbWhite
    TexCveCalz1.BackColor = vbWhite
    TexParCamb1.BackColor = vbWhite
    TexCveCalz2.BackColor = vbWhite
    TexNoPares2.BackColor = vbWhite
    TexCvePro.BackColor = vbWhite

    TexFolioVenta3.Enabled = False
    TexCveCalz3.Enabled = False
    TexParCamb3.Enabled = False
    TexFolioVenta3.BackColor = ColorGris
    TexCveCalz3.BackColor = ColorGris
    TexParCamb3.BackColor = ColorGris

Else

```

```

TexFolioVenta1.Enabled = False
TexCveCalz1.Enabled = False
TexParCamb1.Enabled = False
TexCveCalz2.Enabled = False
TexNoPares2.Enabled = False
TexCvePro.Enabled = False
TexFolioVenta1.BackColor = ColorGris
TexCveCalz1.BackColor = ColorGris
TexParCamb1.BackColor = ColorGris
TexCveCalz2.BackColor = ColorGris
TexNoPares2.BackColor = ColorGris
TexCvePro.BackColor = ColorGris

TexFolioVenta3.Enabled = True
TexCveCalz3.Enabled = True
TexParCamb3.Enabled = True
TexFolioVenta3.BackColor = vbWhite
TexCveCalz3.BackColor = vbWhite
TexParCamb3.BackColor = vbWhite

End If

End Sub

Private Sub BotPrimero_Click()
DataDevolucion.Recordset.MoveFirst
HabilitaBotones
End Sub

Private Sub BotPrimero_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

Private Sub BotSiguiente_Click()
If Not DataDevolucion.Recordset.EOF And
DataDevolucion.Recordset.RecordCount > 1 Then
DataDevolucion.Recordset.MoveNext
End If
HabilitaBotones
End Sub

Private Sub BotSiguiente_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

Private Sub BotUltimo_Click()
DataDevolucion.Recordset.MoveLast
HabilitaBotones
End Sub

Private Sub CombCondicion_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
End Sub

```

```

Private Sub BotUltimo_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

```

```

Private Sub Check1_Click()
If DataDevolucion.EditMode = 0 Then
If Check1.Value = 1 Then
OptionCambio.Value = True
Elseif Check2.Value = 1 Then
OptionDevolucion.Value = True
End If
End If
End Sub

```

```

Private Sub Check2_Click()
Check1_Click
End Sub

```

```

Private Sub Form_Activate()
DataDevolucion.Refresh
HabilitaBotones
End Sub

```

```

Private Sub Form_Load()
DataDevolucion.DatabaseName = MiBDStr
FoMenu.Hide
LlenarCombo LstClave2, "TbProveedores",
"CveProveedores", "Folio"
LlenarCombo List1, "TbInventario", "ClaveCalzado",
"Ventas_ID"

```

```

Dim MiBD As Object
Dim MiTab As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTab = MiBD.OpenRecordset("SELECT
Ventas_ID,NoVentas FROM TbVentas Where CheckCancel =
0 and CondHist = 1 ", dbOpenSnapshot)

```

```

Do Until MiTab.EOF
LstClave.AddItem IIf(IsNull(MiTab!NoVentas), "",
MiTab!NoVentas)
LstClave.ItemData(LstClave.NewIndex) =
MiTab!Ventas_ID

```

```

List3.AddItem IIf(IsNull(MiTab!NoVentas), "",
MiTab!NoVentas)
List3.ItemData(List3.NewIndex) = MiTab!Ventas_ID

```

```

MiTab.MoveNext
Loop
End Sub

```

```

Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub

```

```

Public Sub SeleccionarCampo()
    With Screen.ActiveForm
        If TypeOf .ActiveControl Is TextBox Then
            .ActiveControl.SelStart = 0
            .ActiveControl.SelLength = Len(.ActiveControl)
        End If
    End With
End Sub

```

```

Public Sub BloqueaControles()
    Dim i As Integer
    With Screen.ActiveForm
        For i = 0 To .Controls.Count - 1
            If TypeOf .Controls(i) Is TextBox Then
                If DataDevolucion.Recordset.EditMode = 0 Then
                    .Controls(i).Locked = True
                    .Controls(i).BackColor = &H80000004
                Else
                    .Controls(i).Locked = False
                    .Controls(i).BackColor = vbWhite
                End If
            End If
            If TypeOf .Controls(i) Is ComboBox Then
                If DataDevolucion.Recordset.EditMode = 0 Then
                    .Controls(i).Locked = True
                    .Controls(i).BackColor = &H80000004
                Else
                    .Controls(i).Locked = False
                    .Controls(i).BackColor = vbWhite
                End If
            End If
            If TypeOf .Controls(i) Is OptionButton Then
                If DataDevolucion.Recordset.EditMode = 0 Then
                    .Controls(i).Enabled = False
                Else
                    .Controls(i).Enabled = True
                End If
            End If
        Next
    End With
End Sub

```

```

Private Sub List1_DblClick()
    TexCveCalz2 = List1
    List1.Visible = False
    TexCveCalz2.SetFocus
End Sub

```

```

Private Sub List2_DblClick()
    TexCveCalz3 = List2
    List2.Visible = False
    TexCveCalz3.SetFocus
End Sub

```

```

Private Sub List3_DblClick()
    TexFolioVenta3 = List3
    List3.Visible = False
    TexFolioVenta3.SetFocus
End Sub

```

```

Private Sub LstClave_DblClick()
    TexFolioVenta1 = LstClave
    TexFolioVenta2 = LstClave
    LstClave.Visible = False
    TexFolioVenta1.SetFocus

```

```
End Sub
```

```

Private Sub LstClave2_DblClick()
    TexCvePro = LstClave2
    LstClave2.Visible = False
    List1.Visible = False
End Sub

```

```

Private Sub LstPares_DblClick()
    TexCveCalz1 = LstPares
    LstPares.Visible = False
    TexCveCalz1.SetFocus
End Sub

```

```

Private Sub TexCveCalz1_GotFocus()
    HighLight
    List1.Visible = False
    LstClave.Visible = False
    LstClave2.Visible = False
    List2.Visible = False
End Sub

```

```

Private Sub OptionCambio_Click()
    If DataDevolucion.EditMode <> 0 Then
        If OptionCambio.Value = True Then
            Check1.Value = 1
            Check2.Value = 0
        ElseIf OptionDevolucion.Value = True Then
            Check2.Value = 1
            Check1.Value = 0
        End If
    End If
End Sub

```

```

Private Sub OptionDevolucion_Click()
    OptionCambio_Click
End Sub

```

```

Private Sub TexCveCalz2_Change()
    If DataDevolucion.Recordset.EditMode <> 0 Then
        Dim Index As Integer
        On Error Resume Next
        Index = FindFirstMatch(List1, TexCveCalz2.Text, -1)
        List1.ListIndex = Index
        If Index >= 0 Then List1.Selected(Index) = True
    End If
End Sub

```

```

Private Sub TexCveCalz2_DblClick()
    List1.Visible = True
End Sub

```

```

Private Sub TexCveCalz2_GotFocus()
    List1.Visible = False
    LstClave2.Visible = False
    LstClave.Visible = False
    LstPares.Visible = False
End Sub

```

```

Private Sub TexCveCalz2_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then

```

```

TexCveCalz2 = LstClave
SendKeys "{tab}"
List1.Visible = False
End If
End Sub

```

```

Private Sub TexCveCalz2_LostFocus()
If TexCveCalz2 <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz2 & "", dbOpenSnapshot)

```

```

If Not MiTabla.EOF Then
TexCostUni2 = MiTabla.CostUnit
TexCostPubl2 = MiTabla.CostMen
TexDesc2 = MiTabla.Descripcion
Else
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCveCalz2 = ""
TexCveCalz2.SetFocus
End If
End If
End Sub

```

```

Private Sub TexCveCalz3_Change()

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(List2, TexCveCalz3.Text, -1)
List2.ListIndex = Index
If Index >= 0 Then List2.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCveCalz3_DbIclick()
If DataDevolucion.Recordset.EditMode <> 0 Then
List2.Visible = True
End If
End Sub

```

```

Private Sub TexCveCalz3_GotFocus()
LstClave.Visible = False
List1.Visible = False
LstPares.Visible = False
LstClave2.Visible = False
List3.Visible = False
End Sub

```

```

Private Sub TexCveCalz3_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
TexCveCalz3 = List2
SendKeys "{tab}"
List2.Visible = False
End If
End If
End Sub

```

```

Private Sub TexCveCalz3_LostFocus()
List3.Visible = False
Dim MiTab As Object
Dim MiBD As Object
Dim MiTabla As Object
If DataDevolucion.Recordset.EditMode <> 0 Then
If TexCveCalz3 <> "" Then
Set MiBD = OpenDatabase(MiBDStr)
Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DII_NoPares,DII_Descripcion FROM TbDtlVentas
Where DII_FolFac = " & TexFolioVenta3 & " and
DII_ClaveCalzado = " & TexCveCalz3 & "", dbOpenSnapshot)

```

```

If Not MiTab.EOF Then
TexNoPares3 = MiTab.DII_NoPares
TexParCamb3 = MiTab.DII_NoPares
TexDesc3 = MiTab.DII_Descripcion
End If
End If
End Sub

```

```

Private Sub TexCvePro_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexCvePro.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCvePro_DbIclick()
LstClave2.Visible = True
End Sub

```

```

Private Sub TexCvePro_GotFocus()
List1.Visible = False
LstClave2.Visible = False
LstClave.Visible = False
LstPares.Visible = False

```

```

Dim MiBD As Object
Dim MiTabla As Object
If TexCvePro <> "" Then
TexCvePro = LstClave2

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE
CveProveedores = " & TexCvePro & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCvePro = ""
TexCvePro.SetFocus
End If
End If
End Sub

```

```

Private Sub TexCvePro_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexCvePro = LstClave2
SendKeys "{tab}"
LstClave2.Visible = False
End If

```


End Sub

```
Private Sub TexFechComp_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
End Sub
```

```
Private Sub TexFolioVenta1_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
    Dim Index As Integer
    On Error Resume Next
    Index = FindFirstMatch(LstClave, TexFolioVenta1.Text,
-1)
    LstClave.ListIndex = Index
    If Index >= 0 Then LstClave.Selected(Index) = True
End If
End Sub
```

```
Private Sub TexFolioVenta1_Db1Click()
If DataDevolucion.Recordset.EditMode <> 0 Then
    LstClave.Visible = True
End If
End Sub
```

```
Private Sub TexFolioVenta1_GotFocus()
HighLight
```

```
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
```

```
If DataDevolucion.Recordset.EditMode <> 0 Then
    If LstClave <> "" Then
        Dim MiBD As Object
        Dim MiTabla As Object

        Set MiBD = OpenDatabase(MiBDStr)
        Set MiTabla = MiBD.OpenRecordset("SELECT
TbVentas.* FROM TbVentas WHERE NoVentas = " &
TexFolioVenta1 & "", dbOpenSnapshot)
```

```
    If MiTabla.EOF Then
        MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
        TexFolioVenta1 = ""
        TexFolioVenta1.SetFocus
    Else
```

```
        TexFechComp = MiTabla.FechaVenta
```

```
        Dim MiTab As Object
        LstPares.Clear
```

```
        Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DIIClaveCalzado FROM TbDtl1Ventas Where
DIIIFolFac="" & TexFolioVenta1 & "", dbOpenSnapshot)
```

```
        Do Until MiTab.EOF
            LstPares.AddItem
            If(IsNull(MiTab!DIIIFolFac), "",
MiTab!DIIIFolFac) =
                LstPares.ItemData(LstPares.NewIndex) =
MiTab!DII_ID
                MiTab.MoveNext
        Loop
```

```
End If
```

```
End If
End If
End Sub
```

```
Private Sub TexFolioVenta1_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        TexFolioVenta1 = LstClave
        SendKeys "{tab}"
        LstClave.Visible = False
    End If
End If
End Sub
```

```
Private Sub TexCveCalz1_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
    Dim Index As Integer
    On Error Resume Next
    Index = FindFirstMatch(LstPares, TexCveCalz1.Text, -1)
    LstPares.ListIndex = Index
    If Index >= 0 Then LstPares.Selected(Index) = True
End If
End Sub
```

```
Private Sub TexCveCalz1_Db1Click()
If DataDevolucion.Recordset.EditMode <> 0 Then
    LstPares.Visible = True
End If
End Sub
```

```
Private Sub TexCveCalz1_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        TexCveCalz1 = LstPares
        SendKeys "{tab}"
        LstPares.Visible = False
    End If
End If
End Sub
```

```
Private Sub TexCveCalz1_LostFocus()
LstClave.Visible = False
    Dim MiTab As Object
    Dim MiBD As Object
    Dim MiTabla As Object
    If DataDevolucion.Recordset.EditMode <> 0 Then
        If TexCveCalz1 <> "" Then
            Set MiBD = OpenDatabase(MiBDStr)
            Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DIIIFolFac,DIIIFolFac FROM TbDtl1Ventas
Where DIIIFolFac="" & TexFolioVenta1 & "" and
DIIIFolFac="" & TexCveCalz1 & "", dbOpenSnapshot)
```

```
            If Not MiTab.EOF Then
                TexNoPares1 = MiTab.DIIIFolFac
                TexParCamb1 = MiTab.DIIIFolFac
                TexDesc1 = MiTab.DIIIFolFac
            End If
```

```
        End If
End If
End Sub
```

```
Private Sub TexFolioVenta3_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
```

```

Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(List3, TexFolioVenta3.Text, -1)
List3.ListIndex = Index
If Index >= 0 Then List3.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexFolioVenta3_DbClick()
If DataDevolucion.Recordset.EditMode <> 0 Then
List3.Visible = True
End If
End Sub

```

```

Private Sub TexFolioVenta3_GotFocus()
HighLight

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
If List3 <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbVentas.* FROM TbVentas WHERE NoVentas = " &
TexFolioVenta3 & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
TexFolioVenta3 = ""
TexFolioVenta3.SetFocus
Else
TexFechComp = MiTabla.FechaVenta

```

```

Dim MiTab As Object
List2.Clear

```

```

Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DIIClaveCalzado FROM TbDtlVentas Where
DIIFolFac="" & TexFolioVenta3 & "", dbOpenSnapshot)

```

```

Do Until MiTab.EOF
List2.AddItem IIf(IsNull(MiTab!DIIClaveCalzado),
"", MiTab!DIIClaveCalzado)
List2.ItemData(List2.NewIndex) = MiTab!DII_ID
MiTab.MoveNext
Loop

```

```

End If
End If
End If
End Sub

```

```

Private Sub TexFolioVenta3_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
TexFolioVenta3 = List3
SendKeys "{tab}"
List3.Visible = False
End If
End If
End Sub

```

```

Private Sub TexNoPares2_GotFocus()
List1.Visible = False
LstClave2.Visible = False

```

```

LstClave.Visible = False
LstPares.Visible = False
End Sub

```

```

Private Sub TexNoPares2_LostFocus()
LabelImporte2 = PonCero(TexNoPares2) *
PonCero(TexCostPubl2)

```

```

Dim MiBD As Object
Dim MiTabla As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz2 & "", dbOpenSnapshot)

```

```

If Not MiTabla.EOF Then
If CLng(MiTabla.Existencias) <
CLng(PonCero(TexNoPares2)) Then
MsgBox "Las existencias son insuficiente para la venta ",
vbExclamation
TexNoPares2 = ""
TexNoPares2.SetFocus
End If
Else
End If
End Sub

```

```

Private Sub TexNoPares2_Validate(Cancel As Boolean)
If Not IsNumeric(TexNoPares2) And TexNoPares2 <> ""
Then
MsgBox "Poner solo números", vbExclamation
TexNoPares2 = ""
TexNoPares2.SetFocus
End If
End Sub

```

```

Private Sub TexNoPares3_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub TexParCamb1_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
End Sub

```

```

Private Sub TexParCamb3_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
List3.Visible = False
End Sub

```

```

Private Sub Text1_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
End Sub

```

```

Private Sub Text3_GotFocus()
HighLight

```

DEVOLUCION

Option Explicit

Dim Mibool As Boolean

```

Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataDevolucion.Recordset.EditMode = 0 Then
  If Not DataDevolucion.Recordset.EOF Then
    MiReg = TexFolDevol
    DataDevolucion.Recordset.MoveLast
  End If

  If DataDevolucion.Recordset.RecordCount <= 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
    BotAgregar.Enabled = True
    BotGuardar.Enabled = False
    BotCancelar.Enabled = False
    BotEliminar.Enabled = False
    BotModificar.Enabled = False
    BotMenu.Enabled = True
    BotImprimir.Enabled = False
    BotBuscar.Enabled = False
  Elseif DataDevolucion.Recordset.RecordCount <= 1
  Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
    BotAgregar.Enabled = True
    BotGuardar.Enabled = False
    BotCancelar.Enabled = False
    BotEliminar.Enabled = True
    BotModificar.Enabled = True
    BotMenu.Enabled = True
    BotImprimir.Enabled = True
    BotBuscar.Enabled = True

  Else
    DataDevolucion.Recordset.FindFirst " foliodevolid ="
    & PonCero(MiReg)
    If DataDevolucion.Recordset.AbsolutePosition = 0
  Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True

  Elseif DataDevolucion.Recordset.AbsolutePosition =
  DataDevolucion.Recordset.RecordCount - 1 Then

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
  Else
    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
  End If

  BotAgregar.Enabled = True
  BotGuardar.Enabled = False

```

```

BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
End If

```

Else

```

BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True

```

```

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False

```

```

BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False

```

End If

```

BloqueaControles
End Sub

```

Private Sub BotAgregar_Click()

```

DataDevolucion.Recordset.AddNew
LabelFechDevol = Date
HabilitarBotones
OptionCambio.Value = True
TexFechComp.BackColor = ColorGris
TexFolDevol = IncrementadorID("TbDevolucion")
TexDesc1.BackColor = ColorGris
TexDesc2.BackColor = ColorGris
TexDesc3.BackColor = ColorGris
TexFolioVenta2.BackColor = ColorGris
TexNoPares1.BackColor = ColorGris
TexNoPares3.BackColor = ColorGris

```

If OptionCambio.Value = True Then

```

TexFolioVenta1.Enabled = True
TexCveCalz1.Enabled = True
TexParCamb1.Enabled = True
TexCveCalz2.Enabled = True
TexNoPares2.Enabled = True
TexCvePro.Enabled = True
TexFolioVenta1.BackColor = vbWhite
TexCveCalz1.BackColor = vbWhite
TexParCamb1.BackColor = vbWhite
TexCveCalz2.BackColor = vbWhite
TexNoPares2.BackColor = vbWhite
TexCvePro.BackColor = vbWhite

```

```

TexFolioVenta3.Enabled = False
TexCveCalz3.Enabled = False
TexParCamb3.Enabled = False
TexFolioVenta3.BackColor = ColorGris
TexCveCalz3.BackColor = ColorGris
TexParCamb3.BackColor = ColorGris

```

Else

```

TexFolioVenta1.Enabled = False
TexCveCalz1.Enabled = False

```

```

TexParCamb1.Enabled = False
TexCveCalz2.Enabled = False
TexNoPares2.Enabled = False
TexCvePro.Enabled = False
TexFolioVenta1.BackColor = ColorGris
TexCveCalz1.BackColor = ColorGris
TexParCamb1.BackColor = ColorGris
TexCveCalz2.BackColor = ColorGris
TexNoPares2.BackColor = ColorGris
TexCvePro.BackColor = ColorGris

```

```

TexFolioVenta3.Enabled = True
TexCveCalz3.Enabled = True
TexParCamb3.Enabled = True
TexFolioVenta3.BackColor = vbWhite
TexCveCalz3.BackColor = vbWhite
TexParCamb3.BackColor = vbWhite

```

```

End If
Check1.Value = 1

```

```

End Sub

```

```

Private Sub BotAnterior_Click()
DataDevolucion.Recordset.MovePrevious
HabilitaBotones
End Sub

```

```

Private Sub BotAnterior_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

```

```

Private Sub BotBuscar_Click()
FoBuscDevol.Show
End Sub

```

```

Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
DataDevolucion.Recordset.CancelUpdate
HabilitaBotones
BotAgregar.SetFocus
End If
End Sub

```

```

Private Sub BotCancelar_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
List3.Visible = False
End Sub

```

```

Private Sub BotEliminar_Click()
Dim Eliminar As String

```

```

Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")

```

```

If Eliminar = vbYes Then
With DataDevolucion.Recordset
.Delete
.Requery
If Not .EOF Then
.MoveLast
Else
BotEliminar.Enabled = False
.Requery
DataDevolucion.Refresh
End If
End With
End If
HabilitaBotones
End Sub

```

```

Private Sub BotGuardar_Click()
Dim añadir As String
Dim MiRegistro As Object
Dim MiRegistro12 As Object
Dim MiRegistro1 As Object
Dim MiBaseDatos As Object
Dim MiRegistro2 As Object

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)

```

```

If OptionCambio.Value = True Then

```

```

If Trim(TexCveCalz2) = "" Or Trim(TexNoPares2) = "" Or
Trim(TexCvePro) = "" Then

```

```

añadir = MsgBox("Si ha seleccionado Cambio debe llenar
los campos de Calzado a cambio.", vbInformation)

```

```

Else

```

```

Set MiRegistro1 =
MiBaseDatos.OpenRecordset("SELECT FolioVenta,
CheckCancel, TipoCancel FROM TbFacturas WHERE
FolioVenta = " & TexFolioVenta1 & """, dbOpenDynaset)

```

```

If Not MiRegistro1.EOF Then
MiRegistro1.Edit
MiRegistro1!CheckCancel = 1
MiRegistro1!TipoCancel = "Cancelado por Cambio"
MiRegistro1.Update
End If

```

```

Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = " & TexCveCalz1 &
"", dbOpenDynaset)

```

```

MiRegistro.Edit
MiRegistro!Existencias = MiRegistro!Existencias +
CLng(TexParCamb1)
MiRegistro.Update
MiRegistro.Close

```

```

Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT DIINoPares,
DIIClaveCalzado FROM TbDtIIVentas WHERE
DIIClaveCalzado = " & TexCveCalz1 & "" AND DIIFolFac = "
& TexFolioVenta1 & """, dbOpenDynaset)

```

```

MiRegistro.Edit

```

```

MiRegistro!DIINoPares = MiRegistro!DIINoPares -
CLng(TexParCamb1)
MiRegistro.Update
MiRegistro.Close

```

```

Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT DIIClaveProveedor,
DIIMedidas, DIIClaveCalzado, DIINoPares, DIICostoUnit,
DIICostoPublico, DIIMpporte, DIIDescripcion, DIID_ID,
DIIFolFac FROM TbDtIIVentas ", dbOpenDynaset)

```

```

MiRegistro.AddNew
MiRegistro!DIIClaveCalzado = TexCveCalz2
MiRegistro!DIIClaveProveedor = TexCvePro
MiRegistro!DIINoPares = PonCero(TexNoPares2)
MiRegistro!DIICostoUnit = PonCero(TexCostUni2)
MiRegistro!DIICostoPublico = PonCero(TexCostPubl2)
MiRegistro!DIID_ID = IncrementadorID("TbDtIIVentas")
MiRegistro!DIIFolFac = TexFolioVenta2
MiRegistro!DIIMpporte = PonCero(LabelImporte2)
MiRegistro!DIIDescripcion = TexDesc2
MiRegistro.Update
MiRegistro.Close

```

```

Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = "" & TexCveCalz2 &
"", dbOpenDynaset)
MiRegistro.Edit
MiRegistro!Existencias = MiRegistro!Existencias -
CLng(PonCero(TexNoPares2))
MiRegistro.Update
MiRegistro.Close

```

End If

Else

```

Set MiRegistro12 =
MiBaseDatos.OpenRecordset("SELECT FolioVenta,
CheckCancel, TipoCancel FROM TbFacturas WHERE
FolioVenta= "" & TexFolioVenta3 & "", dbOpenDynaset)

```

```

If Not MiRegistro12.EOF Then
MiRegistro12.Edit
MiRegistro12!CheckCancel = 1
MiRegistro12!TipoCancel = "Cancelado por
Devolución"
MiRegistro12.Update
End If

```

```

Set MiRegistro2 = MiBaseDatos.OpenRecordset("SELECT
DIINoPares, DIIClaveCalzado FROM TbDtIIVentas WHERE
DIIClaveCalzado="" & TexCveCalz3 & "" AND DIIFolFac=""
& TexFolioVenta3 & "", dbOpenDynaset)

```

```

MiRegistro2.Edit
MiRegistro2!DIINoPares = MiRegistro2!DIINoPares -
CLng(TexParCamb3)
MiRegistro2.Update
MiRegistro2.Close

```

End If

```

DataDevolucion.Recordset.Update
DataDevolucion.Refresh
HabilitarBotones
End Sub

```

Private Sub BotGuardar_GotFocus()

```

LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
List3.Visible = False
End Sub

```

```

Private Sub BotImprimir_Click()
FoOpRpDevol.Show vbModal
End Sub

```

```

Private Sub BotMenu_Click()
FoMenu.Show
Unload FoDevolucion
End Sub

```

```

Private Sub BotModificar_Click()
DataDevolucion.Recordset.Edit
HabilitarBotones
TexDesc1.BackColor = ColorGris
TexDesc2.BackColor = ColorGris
TexDesc3.BackColor = ColorGris
TexFolioVenta2.BackColor = ColorGris
TexNoPares1.BackColor = ColorGris
TexNoPares3.BackColor = ColorGris
TexFechComp.BackColor = ColorGris

```

If OptionCambio.Value = True Then

```

TexFolioVenta1.Enabled = True
TexCveCalz1.Enabled = True
TexParCamb1.Enabled = True
TexCveCalz2.Enabled = True
TexNoPares2.Enabled = True
TexCvePro.Enabled = True
TexFolioVenta1.BackColor = vbWhite
TexCveCalz1.BackColor = vbWhite
TexParCamb1.BackColor = vbWhite
TexCveCalz2.BackColor = vbWhite
TexNoPares2.BackColor = vbWhite
TexCvePro.BackColor = vbWhite

```

```

TexFolioVenta3.Enabled = False
TexCveCalz3.Enabled = False
TexParCamb3.Enabled = False
TexFolioVenta3.BackColor = ColorGris
TexCveCalz3.BackColor = ColorGris
TexParCamb3.BackColor = ColorGris

```

Else

```

TexFolioVenta1.Enabled = False
TexCveCalz1.Enabled = False
TexParCamb1.Enabled = False
TexCveCalz2.Enabled = False
TexNoPares2.Enabled = False
TexCvePro.Enabled = False
TexFolioVenta1.BackColor = ColorGris
TexCveCalz1.BackColor = ColorGris
TexParCamb1.BackColor = ColorGris
TexCveCalz2.BackColor = ColorGris
TexNoPares2.BackColor = ColorGris
TexCvePro.BackColor = ColorGris

```

```

TexFolioVenta3.Enabled = True
TexCveCalz3.Enabled = True

```

```

TexParCamb3.Enabled = True
TexFolioVenta3.BackColor = vbWhite
TexCveCalz3.BackColor = vbWhite
TexParCamb3.BackColor = vbWhite

```

```
End If
```

```
End Sub
```

```

Private Sub BotPrimero_Click()
DataDevolucion.Recordset.MoveFirst
HabilitaBotones
End Sub

```

```

Private Sub BotPrimero_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataDevolucion.Recordset.EOF And
DataDevolucion.Recordset.RecordCount > 1 Then
DataDevolucion.Recordset.MoveNext
End If
HabilitaBotones
End Sub

```

```

Private Sub BotSiguiente_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

```

```

Private Sub BotUltimo_Click()
DataDevolucion.Recordset.MoveLast
HabilitaBotones
End Sub

```

```

Private Sub CombCondicion_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
End Sub

```

```

Private Sub BotUltimo_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List3.Visible = False
List2.Visible = False
End Sub

```

```

Private Sub Check1_Click()
If DataDevolucion.EditMode = 0 Then
If Check1.Value = 1 Then

```

```

OptionCambio.Value = True
ElseIf Check2.Value = 1 Then
OptionDevolucion.Value = True
End If
End If
If OptionCambio.Value = True Then

```

```

TexFolioVenta1.Enabled = True
TexCveCalz1.Enabled = True
TexParCamb1.Enabled = True
TexCveCalz2.Enabled = True
TexNoPares2.Enabled = True
TexCvePro.Enabled = True
TexFolioVenta1.BackColor = vbWhite
TexCveCalz1.BackColor = vbWhite
TexParCamb1.BackColor = vbWhite
TexCveCalz2.BackColor = vbWhite
TexNoPares2.BackColor = vbWhite
TexCvePro.BackColor = vbWhite

```

```

TexFolioVenta3.Enabled = False
TexCveCalz3.Enabled = False
TexParCamb3.Enabled = False
TexFolioVenta3.BackColor = ColorGris
TexCveCalz3.BackColor = ColorGris
TexParCamb3.BackColor = ColorGris
TexFolioVenta3 = ""
TexCveCalz3 = ""
TexParCamb3 = ""
TexNoPares3 = ""
TexDesc3 = ""

```

```
Else
```

```

TexFolioVenta1.Enabled = False
TexCveCalz1.Enabled = False
TexParCamb1.Enabled = False
TexCveCalz2.Enabled = False
TexNoPares2.Enabled = False
TexCvePro.Enabled = False
TexFolioVenta1.BackColor = ColorGris
TexCveCalz1.BackColor = ColorGris
TexParCamb1.BackColor = ColorGris
TexCveCalz2.BackColor = ColorGris
TexNoPares2.BackColor = ColorGris
TexCvePro.BackColor = ColorGris
TexFolioVenta1 = ""
TexCveCalz1 = ""
TexParCamb1 = ""
TexCveCalz2 = ""
TexNoPares2 = ""
TexCvePro = ""
TexNoPares1 = ""
TexDesc1 = ""
TexFolioVenta2 = ""
TexDesc2 = ""

```

```

TexFolioVenta3.Enabled = True
TexCveCalz3.Enabled = True
TexParCamb3.Enabled = True
TexFolioVenta3.BackColor = vbWhite
TexCveCalz3.BackColor = vbWhite
TexParCamb3.BackColor = vbWhite

```

```
End If
```

```
End Sub
```

```
Private Sub Check2_Click()
Check1_Click
End Sub
```

```
Private Sub Form_Activate()
HabilitaBotones
End Sub
```

```
Private Sub Form_Load()
DataDevolucion.DatabaseName = MiBDStr
FoMenu.Hide
LlenarCombo LstClave2, "TbProveedores",
"CveProveedores", "Folio"
LlenarCombo List1, "TbInventario", "ClaveCalzado",
"Ventas_ID"
```

```
Dim MiBD As Object
Dim MiTab As Object
```

```
Set MiBD = OpenDatabase(MiBDStr)
Set MiTab = MiBD.OpenRecordset("SELECT
Ventas_ID,NoVentas FROM TbVentas Where CheckCancel =
0 and CondHist = 0 ", dbOpenSnapshot)
```

```
Do Until MiTab.EOF
LstClave.AddItem IIf(IsNull(MiTab!NoVentas), "",
MiTab!NoVentas)
LstClave.ItemData(LstClave.NewIndex) =
MiTab!Ventas_ID
```

```
List3.AddItem IIf(IsNull(MiTab!NoVentas), "",
MiTab!NoVentas)
List3.ItemData(List3.NewIndex) = MiTab!Ventas_ID
```

```
MiTab.MoveNext
Loop
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub
```

```
Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataDevolucion.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataDevolucion.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
```

```
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is OptionButton Then
If DataDevolucion.Recordset.EditMode = 0 Then
.Controls(i).Enabled = False
Else
.Controls(i).Enabled = True
End If
End If
Next
End With
```

```
End Sub
```

```
Private Sub List1_Db1Click()
TexCveCalz2 = List1
List1.Visible = False
TexCveCalz2.SetFocus
End Sub
```

```
Private Sub List2_Db1Click()
TexCveCalz3 = List2
List2.Visible = False
TexCveCalz3.SetFocus
End Sub
```

```
Private Sub List3_Db1Click()
TexFolioVenta3 = List3
List3.Visible = False
TexFolioVenta3.SetFocus
End Sub
```

```
Private Sub LstClave_Db1Click()
TexFolioVenta1 = LstClave
TexFolioVenta2 = LstClave
LstClave.Visible = False
TexFolioVenta1.SetFocus
End Sub
```

```
Private Sub LstClave2_Db1Click()
TexCvePro = LstClave2
LstClave2.Visible = False
List1.Visible = False
End Sub
```

```
Private Sub LstPares_Db1Click()
TexCveCalz1 = LstPares
LstPares.Visible = False
TexCveCalz1.SetFocus
End Sub
```

```
Private Sub TexCveCalz1_GotFocus()
HighLight
List1.Visible = False
LstClave.Visible = False
LstClave2.Visible = False
List2.Visible = False
End Sub
```

```

Private Sub OptionCambio_Click()
If DataDevolucion.EditMode <> 0 Then
  If OptionCambio.Value = True Then
    Check1.Value = 1
    Check2.Value = 0
  ElseIf OptionDevolucion.Value = True Then
    Check2.Value = 1
    Check1.Value = 0
  End If
End If
End Sub

Private Sub OptionDevolucion_Click()
OptionCambio_Click
End Sub

Private Sub TexCveCalz2_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(List1, TexCveCalz2.Text, -1)
List1.ListIndex = Index
If Index >= 0 Then List1.Selected(Index) = True
End Sub

Private Sub TexCveCalz2_DbClick()
List1.Visible = True
End Sub

Private Sub TexCveCalz2_GotFocus()
List1.Visible = False
LstClave2.Visible = False
LstClave.Visible = False
LstPares.Visible = False
End Sub

Private Sub TexCveCalz2_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
  TexCveCalz2 = LstClave
  SendKeys "{tab}"
  List1.Visible = False
End If
End Sub

Private Sub TexCveCalz2_LostFocus()
If TexCveCalz2 <> "" Then
  Dim MiBD As Object
  Dim MiTabla As Object

  Set MiBD = OpenDatabase(MiBDStr)
  Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
" & TexCveCalz2 & """, dbOpenSnapshot)

  If Not MiTabla.EOF Then
    TexCostUni2 = MiTabla.CostUnit
    TexCostPubl2 = MiTabla.CostMen
    TexDesc2 = MiTabla.Descripcion
  Else
    MsgBox "La clave proporcionada no existe,trate de
nuevo", vbExclamation
    TexCveCalz2 = ""
    TexCveCalz2.SetFocus
  End If
End If

```

```

End If
End Sub

```

```

Private Sub TexCveCalz3_Change()

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
  Dim Index As Integer
  On Error Resume Next
  Index = FindFirstMatch(List2, TexCveCalz3.Text, -1)
  List2.ListIndex = Index
  If Index >= 0 Then List2.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCveCalz3_DbClick()

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
  List2.Visible = True
End If
End Sub

```

```

Private Sub TexCveCalz3_GotFocus()

```

```

LstClave.Visible = False
List1.Visible = False
LstPares.Visible = False
LstClave2.Visible = False
List3.Visible = False
End Sub

```

```

Private Sub TexCveCalz3_KeyPress(KeyAscii As Integer)

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    TexCveCalz3 = List2
    SendKeys "{tab}"
    List2.Visible = False
  End If
End If
End Sub

```

```

Private Sub TexCveCalz3_LostFocus()

```

```

List3.Visible = False
Dim MiTab As Object
Dim MiBD As Object
Dim MiTabla As Object
If DataDevolucion.Recordset.EditMode <> 0 Then
  If TexCveCalz3 <> "" Then
    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DII_NoPares,DII_Descripcion FROM TbDtlVentas
Where DII_FolFac ="" & TexFolioVenta3 & "" and
DII_ClaveCalzado ="" & TexCveCalz3 & """, dbOpenSnapshot)

    If Not MiTab.EOF Then
      TexNoPares3 = MiTab.DII_NoPares
      TexParCamb3 = MiTab.DII_NoPares
      TexDesc3 = MiTab.DII_Descripcion
    End If
  End If
End If
End Sub

```

```

Private Sub TexCvePro_Change()

```

```

Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexCvePro.Text, -1)

```



```

LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True
End Sub

```

```

Private Sub TexCvePro_DblClick()
LstClave2.Visible = True
End Sub

```

```

Private Sub TexCvePro_GotFocus()

```

```

List1.Visible = False
LstClave2.Visible = False
LstClave.Visible = False
LstPares.Visible = False

```

```

Dim MiBD As Object
Dim MiTabla As Object
If TexCvePro <> "" Then
TexCvePro = LstClave2

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbProveedores.* FROM TbProveedores WHERE
CveProveedores = " & TexCvePro & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexCvePro = ""
TexCvePro.SetFocus
End If
End If
End Sub

```

```

Private Sub TexCvePro_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
TexCvePro = LstClave2
SendKeys "{tab}"
LstClave2.Visible = False
End If
End Sub

```

```

Private Sub TexFechComp_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
End Sub

```

```

Private Sub TexFolioVenta1_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexFolioVenta1.Text,
-1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexFolioVenta1_DblClick()
If DataDevolucion.Recordset.EditMode <> 0 Then
LstClave.Visible = True
End If
End Sub

```

```

Private Sub TexFolioVenta1_GotFocus()
HighLight

```

```

LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
If LstClave <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbVentas.* FROM TbVentas WHERE NoVentas = "" &
TexFolioVenta1 & "", dbOpenSnapshot)

```

```

If MiTabla.EOF Then
MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
TexFolioVenta1 = ""
TexFolioVenta1.SetFocus
Else
TexFechComp = MiTabla.FechaVenta

```

```

Dim MiTab As Object
LstPares.Clear

```

```

Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DIIClaveCalzado FROM TbDIIVentas Where
DIIFolFac = "" & TexFolioVenta1 & "", dbOpenSnapshot)

```

```

Do Until MiTab.EOF
LstPares.AddItem
IIf(IsNull(MiTab!DIIClaveCalzado), "",
MiTab!DIIClaveCalzado)
LstPares.ItemData(LstPares.NewIndex) =
MiTab!DII_ID
MiTab.MoveNext
Loop
End If
End If
End Sub

```

```

Private Sub TexFolioVenta1_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
TexFolioVenta1 = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End If
End Sub

```

```

Private Sub TexCveCalz1_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstPares, TexCveCalz1.Text, -1)
LstPares.ListIndex = Index
If Index >= 0 Then LstPares.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCveCalz1_DblClick()

```

```

If DataDevolucion.Recordset.EditMode <> 0 Then
  LstPares.Visible = True
End If
End Sub

```

```

Private Sub TexCveCalz1_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
  TexCveCalz1 = LstPares
  SendKeys "{tab}"
  LstPares.Visible = False
End If
End If
End Sub

```

```

Private Sub TexCveCalz1_LostFocus()
LstClave.Visible = False
Dim MiTab As Object
Dim MiBD As Object
Dim MiTabla As Object
If DataDevolucion.Recordset.EditMode <> 0 Then
  If TexCveCalz1 <> "" Then
    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DIINoPares,DIIDescripcion FROM TbDtlVentas
Where DIIFolFac = "" & TexFolioVenta1 & "" and
DIIClaveCalzado = "" & TexCveCalz1 & """, dbOpenSnapshot)

    If Not MiTab.EOF Then
      TexNoPares1 = MiTab.DIINoPares
      TexParCamb1 = MiTab.DIINoPares
      TexDesc1 = MiTab.DIIDescripcion
    End If
  End If
End If
End Sub

```

```

Private Sub TexFolioVenta3_Change()
If DataDevolucion.Recordset.EditMode <> 0 Then
  Dim Index As Integer
  On Error Resume Next
  Index = FindFirstMatch(List3, TexFolioVenta3.Text, -1)
  List3.ListIndex = Index
  If Index >= 0 Then List3.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexFolioVenta3_DbClick()
If DataDevolucion.Recordset.EditMode <> 0 Then
  List3.Visible = True
End If
End Sub

```

```

Private Sub TexFolioVenta3_GotFocus()
HighLight
If DataDevolucion.Recordset.EditMode <> 0 Then
  If List3 <> "" Then
    Dim MiBD As Object
    Dim MiTabla As Object

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT
TbVentas.* FROM TbVentas WHERE NoVentas = "" &
TexFolioVenta3 & """, dbOpenSnapshot)

```

```

If MiTabla.EOF Then
  MsgBox "La clave proporcionada no existe,trate de
nuevo ", vbExclamation
  TexFolioVenta3 = ""
  TexFolioVenta3.SetFocus
Else
  TexFechComp = MiTabla.FechaVenta

  Dim MiTab As Object
  List2.Clear

  Set MiTab = MiBD.OpenRecordset("SELECT
DII_ID,DIIClaveCalzado FROM TbDtlVentas Where
DIIFolFac = "" & TexFolioVenta3 & """, dbOpenSnapshot)

  Do Until MiTab.EOF
    List2.AddItem IIf(IsNull(MiTab!DIIClaveCalzado),
"", MiTab!DIIClaveCalzado)
    List2.ItemData(List2.NewIndex) = MiTab!DII_ID
    MiTab.MoveNext
  Loop
End If
End If
End Sub

```

```

Private Sub TexFolioVenta3_KeyPress(KeyAscii As Integer)
If DataDevolucion.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
  TexFolioVenta3 = List3
  SendKeys "{tab}"
  List3.Visible = False
End If
End If
End Sub

```

```

Private Sub TexNoPares2_GotFocus()
List1.Visible = False
LstClave2.Visible = False
LstClave.Visible = False
LstPares.Visible = False
End Sub

```

```

Private Sub TexNoPares2_LostFocus()
LabelImporte2 = PonCero(TexNoPares2) *
PonCero(TexCostPubl2)

```

```

Dim MiBD As Object
Dim MiTabla As Object

```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT
TbInventario.* FROM TbInventario WHERE ClaveCalzado =
"" & TexCveCalz2 & """, dbOpenSnapshot)

```

```

If Not MiTabla.EOF Then
  If CLng(MiTabla.Existencias) <
CLng(PonCero(TexNoPares2)) Then
    MsgBox "Las existencias son insuficiente para la venta ",
vbExclamation
    TexNoPares2 = ""
    TexNoPares2.SetFocus
  End If
Else
End If
End Sub

```

```
Private Sub TexNoPares2_Validate(Cancel As Boolean)
If Not IsNumeric(TexNoPares2) And TexNoPares2 <> ""
Then
MsgBox "Poner solo números", vbExclamation
TexNoPares2 = ""
TexNoPares2.SetFocus
End If
End Sub
```

```
Private Sub TexNoPares3_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub TexParCamb1_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
End Sub
```

```
Private Sub TexParCamb3_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
List1.Visible = False
LstClave2.Visible = False
List2.Visible = False
List3.Visible = False
End Sub
```

```
Private Sub Text1_GotFocus()
LstClave.Visible = False
LstPares.Visible = False
End Sub
```

```
Private Sub Text3_GotFocus()
HighLight
End Sub
```

EMPLEADOS

```
Option Explicit
Dim MiEmp As String
```

```
Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataEmp.Recordset.EditMode = 0 Then
If Not DataEmp.Recordset.EOF Then
MiReg = TexCveEmp
DataEmp.Recordset.MoveLast
End If
If DataEmp.Recordset.RecordCount <= 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = True

```

```
BotImprimir.Enabled = False
BotBuscar.Enabled = False
ElseIf DataEmp.Recordset.RecordCount <= 1 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
```

Else

```
DataEmp.Recordset.FindFirst " CveEmpleado =" &
MiReg & ""
```

```
If DataEmp.Recordset.AbsolutePosition = 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = True
BotUltimo.Enabled = True
```

```
ElseIf DataEmp.Recordset.AbsolutePosition =
DataEmp.Recordset.RecordCount - 1 Then
```

```
BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
```

Else

```
BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = True
BotUltimo.Enabled = True
```

End If

```
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
```

End If

Else

```
BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True
```

```
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
```

```
BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
```

```
End If
BloqueaControles
End Sub
```

```
Private Sub BotAgregar_Click()
DataEmp.Recordset.AddNew
LabelIngreso = Date
TexCveEmp.SetFocus
HabilitarBotones
TexPais = "México"
TexEmpID = IncrementadorID("TbEmp")
End Sub
```

```
Private Sub BotAnterior_Click()
DataEmp.Recordset.MovePrevious
HabilitarBotones
End Sub
```

```
Private Sub BotBuscar_Click()
FoBuscEmp.Show vbModal
End Sub
```

```
Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
DataEmp.Recordset.CancelUpdate
HabilitarBotones
BotAgregar.SetFocus
End If
```

```
MiEmp = ""
End Sub
```

```
Private Sub BotEliminar_Click()
Dim Eliminar As String
Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
If Eliminar = vbYes Then
With DataEmp.Recordset
.Delete
.Requery
If Not .EOF Then
.MoveLast
Else
BotEliminar.Enabled = False
.Requery
DataEmp.Refresh
End If
End With
End If
HabilitarBotones
End Sub
```

```
Private Sub BotGuardar_Click()
Dim añadir As String
If TexCveEmp = "" Or TexNomEmp = "" Or TexTel = ""
Then
añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
TexCveEmp.SetFocus
Else
DataEmp.Recordset.Update
DataEmp.Refresh
MiEmp = ""
HabilitarBotones
End If
```

```
End Sub
```

```
Private Sub BotImprimir_Click()
FoOpRpEmp.Show
End Sub
```

```
Private Sub BotMenu_Click()
FoMenu.Show
Unload FoEmp
End Sub
```

```
Private Sub BotModificar_Click()
DataEmp.Recordset.Edit
MiEmp = TexCveEmp
HabilitarBotones
End Sub
```

```
Private Sub BotPrimero_Click()
DataEmp.Recordset.MoveFirst
HabilitarBotones
End Sub
```

```
Private Sub BotSiguiente_Click()
If Not DataEmp.Recordset.EOF And
DataEmp.Recordset.RecordCount > 1 Then
DataEmp.Recordset.MoveNext
End If
HabilitarBotones
End Sub
```

```
Private Sub BotUltimo_Click()
DataEmp.Recordset.MoveLast
HabilitarBotones
End Sub
```

```
Private Sub CombEstados_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub Form_Activate()
HabilitarBotones
BotAgregar.SetFocus
End Sub
```

```
Private Sub Form_Load()
FoMenu.Hide
DataEmp.DatabaseName = MiBDStr
LlenarCombo CombEstados, "TbEstados", "Estados",
"Estados_ID"
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
```

```

.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub

```

```

Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataEmp.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataEmp.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
Next
End With
End Sub

```

```

Private Sub TexCalle_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCalle_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCiud_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCiud_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCol_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCol_KeyPress(KeyAscii As Integer)

```

```

If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCP_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCP_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCveEmp_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCveEmp_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCveEmp_LostFocus()
TexCveEmp = DameFormato(TexCveEmp, TexCveEmp.Tag)
End Sub

```

```

Private Sub TexEstadosID_Change()
BusquedaIdCombo TexEstadosID, CombEstados
End Sub
Private Sub CombEstados_Click()
DesplegarEstados CombEstados, TexEstadosID
End Sub

```

```

Private Sub TexFechNaci_GotFocus()
HighLight
End Sub

```

```

Private Sub TexFechNaci_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexNomEmp_GotFocus()
HighLight
End Sub

```

```

Private Sub TexNomEmp_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then

```

```

    SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexPais_GotFocus()
HighLight
End Sub

```

```

Private Sub TexPais_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexPuesto_GotFocus()
HighLight
End Sub

```

```

Private Sub TexPuesto_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexRefer1_GotFocus()
HighLight
End Sub

```

```

Private Sub TexRefer1_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexRefer2_GotFocus()
HighLight
End Sub

```

```

Private Sub TexRefer2_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexSueldo_GotFocus()
HighLight
End Sub

```

```

Private Sub TexSueldo_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End Sub

```

```

End If
End Sub

```

```

Private Sub TexSueldo_LostFocus()
If Not IsNumeric(TexSueldo) And TexSueldo <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexSueldo = ""
    TexSueldo.SetFocus

```

```

End If
End Sub

```

```

Private Sub TexCveEmp_Validate(Cancel As Boolean)
If DataEmp.Recordset.EditMode <> 0 Then

```

```

    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

```

```

    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TbEmp.* From TbEmp WHERE CveEmpleado ='" &
TexCveEmp & "'", dbOpenSnapshot)

```

```

    If Not MiRegistro.EOF And MiEmp <> TexCveEmp Then
        MsgBox "Clave Duplicada", vbExclamation
        Cancel = True
        TexCveEmp = ""
    End If
End If
End Sub

```

```

Private Sub Text1_GotFocus()
HighLight
End Sub

```

```

Private Sub Text1_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TextTel_GotFocus()
HighLight
End Sub

```

```

Private Sub TextTel_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TextTel1_GotFocus()
HighLight
End Sub

```

```

Private Sub TextTel1_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End Sub

```

```
End If
End If
End Sub
```

```
Private Sub TexTel2_GotFocus()
HighLight
End Sub
```

```
Private Sub TexTel2_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexTelEmer_GotFocus()
HighLight
End Sub
```

```
Private Sub TexTelEmer_KeyPress(KeyAscii As Integer)
If DataEmp.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

FACTURAS

```
Option Explicit
Dim MiBol As Boolean
```

```
Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataFacturas.Recordset.EditMode = 0 Then
    If Not DataFacturas.Recordset.EOF Then
        MiReg = TexFolio
        DataFacturas.Recordset.MoveLast
    End If

    If DataFacturas.Recordset.RecordCount <= 0 Then
        BotPrimero.Enabled = False
        BotAnterior.Enabled = False
        BotSiguiente.Enabled = False
        BotUltimo.Enabled = False
        BotAgregar.Enabled = True
        BotGuardar.Enabled = False
        BotCancelar.Enabled = False
        BotEliminar.Enabled = False
        BotModificar.Enabled = False
        BotMenu.Enabled = True
        BotImprimir.Enabled = False
        BotBuscar.Enabled = False
    ElseIf DataFacturas.Recordset.RecordCount <= 1 Then
        BotPrimero.Enabled = False
        BotAnterior.Enabled = False
        BotSiguiente.Enabled = False
        BotUltimo.Enabled = False
        BotAgregar.Enabled = True
        BotGuardar.Enabled = False
        BotCancelar.Enabled = False
        BotEliminar.Enabled = True
        BotModificar.Enabled = True
    End If
End Sub
```

```
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
```

```
Else
```

```
DataFacturas.Recordset.FindFirst " Folio =" & MiReg
& ""
```

```
If DataFacturas.Recordset.AbsolutePosition = 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
End If
```

```
ElseIf DataFacturas.Recordset.AbsolutePosition =
DataFacturas.Recordset.RecordCount - 1 Then
```

```
    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
```

```
Else
```

```
    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
```

```
End If
```

```
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
```

```
End If
```

```
Else
```

```
BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True
```

```
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
```

```
BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
```

```
End If
```

```
BloqueaControles
ActualizarGrid
Cancelacion
End Sub
```

```
Private Sub BotAgregar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
```

```
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TbVentas.NoVentas, TbVentas.Ventas_ID,
```

```
TbFacturas.FolioVenta,      TbFacturas.CheckCancel,
TbVentas.CheckFact,      TbVentas.CheckCancel FROM
TbVentas left JOIN TbFacturas ON TbVentas.NoVentas =
TbFacturas.FolioVenta WHERE TbVentas.CondHist = 0 and
(TbVentas.CheckFact=True AND
TbVentas.CheckCancel=False) and
isnull(TbFacturas.CheckCancel) or
TbFacturas.CheckCancel=true", dbOpenSnapshot)
```

```
ListVenta.Clear
```

```
Do Until MiRegistro.EOF
  ListVenta.AddItem Iif(IsNull(MiRegistro!NoVentas), "",
MiRegistro!NoVentas)
  ListVenta.ItemData(ListVenta.NewIndex) =
MiRegistro!Ventas_ID
  MiRegistro.MoveNext
Loop
```

```
BorrarTab "TbTmpDtlFac"
```

```
DataFacturas.Refresh
DataFacturas.Recordset.AddNew
TexDesc = 0
HabilitarBotones
Cancelacion
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(Folio) as Folio1 From TbFacturas", dbOpenSnapshot)
'Costante para abrir en modo solo lectura y si fuera
dbOpendynaset era para lectura/escritura'
```

```
If Not IsNull(MiRegistro!Folio1) Then
  TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
  TexFolio.BackColor = ColorGris
  TexNomCli.BackColor = ColorGris
  TexCol.BackColor = ColorGris
  TexCalle.BackColor = ColorGris
  TexCP.BackColor = ColorGris
  TexRFC.BackColor = ColorGris
  TexTelNeg.BackColor = ColorGris
  TexTelPart.BackColor = ColorGris
  TexEstados.BackColor = ColorGris
  TexCiud.BackColor = ColorGris
  TexPais.BackColor = ColorGris
  Text1.BackColor = ColorGris
  GDFac.BackColor = ColorGris
```

```
Else
```

```
  TexFolio = "0001"
  TexFolio.BackColor = ColorGris
  TexNomCli.BackColor = ColorGris
  TexCol.BackColor = ColorGris
  TexCalle.BackColor = ColorGris
  TexCP.BackColor = ColorGris
  TexRFC.BackColor = ColorGris
  TexTelNeg.BackColor = ColorGris
  TexTelPart.BackColor = ColorGris
  TexEstados.BackColor = ColorGris
  TexCiud.BackColor = ColorGris
  TexPais.BackColor = ColorGris
  Text1.BackColor = ColorGris
  GDFac.BackColor = ColorGris
  TexFolVenta = ""
```

```
End If
```

```
TexDesc.BackColor = ColorGris
TexDesc.Locked = True
TexSubTotDesc.Locked = True
TexIVA.Locked = True
LabelFecha = Date
TexPais = "México"
TexNomEmp.BackColor = ColorGris
TexCveEmp.SetFocus
```

```
End Sub
```

```
Private Sub BotAnterior_Click()
DataFacturas.Recordset.MovePrevious
HabilitarBotones
Cancelacion
End Sub
```

```
Private Sub BotBuscar_Click()
FoBuscFac.Show vbModal
End Sub
```

```
Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
  DataFacturas.Recordset.CancelUpdate
  PasarDatos "TbDtlFac", "TbTmpDtlFac", "DII_Fac_Fol
=" & TexFolio & ""
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
NoVentas,Ventas_ID From TbVentas WHERE CheckFact =
true ", dbOpenSnapshot)
```

```
ListVenta.Clear
Do Until MiRegistro.EOF
  ListVenta.AddItem Iif(IsNull(MiRegistro!NoVentas), "",
MiRegistro!NoVentas)
  ListVenta.ItemData(ListVenta.NewIndex) =
MiRegistro!Ventas_ID
  MiRegistro.MoveNext
Loop
  HabilitarBotones
  BotAgrupar.SetFocus
End If
End Sub
```

```
Private Sub BotCancelar_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
  LstClave.Visible = False
  LstClave2.Visible = False
End If
End Sub
```

```
Private Sub BotEliminar_Click()
Dim Eliminar As String
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
  Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
  If Eliminar = vbYes Then
    If CheckCancel.Value = 1 Then
      LabelFecha.Enabled = False
      TexFolio.Enabled = False
      TexCveEmp.Enabled = False
      TexNoPedido.Enabled = False
      TexFolVenta.Enabled = False
      GDFac.Enabled = False
      LabelSubTotal.Enabled = False
      TexIVA.Enabled = False
      LabelTotal.Enabled = False
      BotModificar.Enabled = False
```



```

BotEliminar.Enabled = False
BotImprimir.Enabled = True
BotBuscar.Enabled = True
Else
Label21 = "¡Cancelada por Error!"
LabelFecha.Enabled = False
TexFolio.Enabled = False
TexCveEmp.Enabled = False
TexNoPedido.Enabled = False
TexFolVenta.Enabled = False
GDFac.Enabled = False
LabelSubTotal.Enabled = False
TexIVA.Enabled = False
LabelTotal.Enabled = False
BotModificar.Enabled = False
BotEliminar.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
End If
Else
LabelFecha.Enabled = True
TexFolio.Enabled = True
TexCveEmp.Enabled = True
TexNoPedido.Enabled = True
TexFolVenta.Enabled = True
GDFac.Enabled = True
LabelSubTotal.Enabled = True
TexIVA.Enabled = True
LabelTotal.Enabled = True
BotModificar.Enabled = True
BotEliminar.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
End If
HabilitarBotones
BotModificar.Enabled = False
End Sub

Private Sub BotGuardar_Click()
Dim añadir As String
Dim BuscarDatos
Dim Mult As Variant
If TexCveCli = "" Or Trim(TexCveEmp) = "" Or
Trim(TexNomCli) = "" Then
añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
Else
BorrarTab "TbDtllFac", "Dtl_Fac_Fol =" & TexFolio & ""
PasarDatos "TbTmpDtllFac", "TbDtllFac"
DataFacturas.Recordset.Update
BuscarDatos = TexCveCli
DataFacturas.Refresh
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
NoVentas,Ventas_ID From TbVentas WHERE CheckFact =
true", dbOpenSnapshot)
ListVenta.Clear
Do Until MiRegistro.EOF
ListVenta.AddItem IIf(IsNull(MiRegistro.NoVentas), "",
MiRegistro.NoVentas)
ListVenta.ItemData(ListVenta.NewIndex) =
MiRegistro.Ventas_ID
MiRegistro.MoveNext
Loop
DataFacturas.Recordset.FindFirst " CveClientes =" &
BuscarDatos & ""

```

```

HabilitarBotones
End If
End Sub

```

```

Private Sub BotGuardar_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
LstClave.Visible = False
LstClave2.Visible = False
End If
End Sub

```

```

Private Sub BotImprimir_Click()
FoOpRpFac.Show
End Sub

```

```

Private Sub BotMenu_Click()
FoMenu.Show
Unload FoFacturas
End Sub

```

```

Private Sub BotModificar_Click()
DataFacturas.Recordset.Edit
HabilitarBotones
TexFolio.BackColor = ColorGris
TexNomCli.BackColor = ColorGris
TexCol.BackColor = ColorGris
TexCalle.BackColor = ColorGris
TexCP.BackColor = ColorGris
TexRFC.BackColor = ColorGris
TexTelNeg.BackColor = ColorGris
TexTelPart.BackColor = ColorGris
TexEstados.BackColor = ColorGris
TexCiud.BackColor = ColorGris
TexPais.BackColor = ColorGris
Textl.BackColor = ColorGris
GDFac.BackColor = ColorGris
TexNomEmp.BackColor = ColorGris
TexFolVenta.Enabled = True
TexDesc.BackColor = ColorGris
TexDesc.Locked = True
TexSubTotDesc.Locked = True
TexIVA.Locked = True
End Sub

```

```

Private Sub BotPrimero_Click()
DataFacturas.Recordset.MoveFirst
HabilitarBotones
Cancelacion
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataFacturas.Recordset.EOF And
DataFacturas.Recordset.RecordCount > 1 Then
DataFacturas.Recordset.MoveNext
End If
HabilitarBotones
Cancelacion
End Sub

```

```

Private Sub BotUltimo_Click()
DataFacturas.Recordset.MoveLast
HabilitarBotones
Cancelacion

```

```

End Sub

Private Sub Form_Activate()
HabilitaBotones
TexFolio.BackColor = &H8000000F
TexNomCli.BackColor = &H8000000F
TexCol.BackColor = &H8000000F
TexCalle.BackColor = &H8000000F
TexCP.BackColor = &H8000000F
TexRFC.BackColor = &H8000000F
TexTelNeg.BackColor = &H8000000F
TexTelPart.BackColor = &H8000000F
TexEstados.BackColor = &H8000000F
TexCiud.BackColor = &H8000000F
TexPais.BackColor = &H8000000F
Text1.BackColor = &H8000000F
TexNomEmp.BackColor = ColorGris
TexFolVenta.BackColor = ColorGris

MiBol = True
End Sub

Private Sub Form_Load()
FoMenu.Hide
DataFacturas.DatabaseName = MiBDStr
LlenarCombo LstClave, "TbEmp", "CveEmpleado",
"Empleado_ID"
LlenarCombo LstClave2, "TbClientes", "CveClientes",
"Clientes_ID"
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
NoVentas,Ventas_ID From TbVentas WHERE CheckFact =
true and CondHist = 0 ", dbOpenSnapshot)

Do Until MiRegistro.EOF
ListVenta.AddItem IIf(IsNull(MiRegistro.NoVentas), "",
MiRegistro.NoVentas)
ListVenta.ItemData(ListVenta.NewIndex) =
MiRegistro.Ventas_ID
MiRegistro.MoveNext
Loop

MiBol = False
End Sub

Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub

Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub

Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataFacturas.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataFacturas.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is MSHFlexGrid Then
If DataFacturas.Recordset.EditMode = 0 Then
.Controls(i).BackColor = &H80000004
Else
.Controls(i).BackColor = vbWhite
End If
End If
Next
End With
End Sub

Private Sub GDFac_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
LstClave.Visible = False
LstClave2.Visible = False
HighLight
End If
End Sub

Private Sub LabelTotal_Change()
If MiBol = True Then
Text1 = ImporteLetra(LabelTotal)
End If
End Sub

Private Sub LstClave_Db1Click()
TexCveEmp = LstClave
LstClave.Visible = False
TexCveEmp.SetFocus
End Sub

Private Sub LstClave2_Db1Click()
TexCveCli = LstClave2
LstClave2.Visible = False
TexCveCli.SetFocus
End Sub

Private Sub TexCveCli_Change()
If DataFacturas.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexCveCli.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True

```

```
End If
End Sub
```

```
Private Sub TexCveCli_DbIclic()
If DataFacturas.Recordset.EditMode <> 0 Then
  LstClave2.Visible = True
End If
End Sub
```

```
Private Sub TexCveCli_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
  LstClave.Visible = False
  HighLight
  If TexCveCli <> "" Then
    Dim MiBD As Object
    Dim MiTabla As Object

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTabla = MiBD.OpenRecordset("SELECT Estados,
TbClientes.* FROM TbClientes INNER JOIN TbEstados ON
TbClientes.Estado = TbEstados.Estados_ID WHERE
CveClientes = " & TexCveCli & "", dbOpenSnapshot)
```

```
  If Not MiTabla.EOF Then
    TexNomCli = MiTabla.NombCliente
    TexCol = MiTabla.Colonia
    TexCalle = MiTabla.Calle
    TexCP = MiTabla.CP
    TexRFC = MiTabla.RFC
    TexTelNeg = MiTabla.TeleNeg
    TexTelPart = MiTabla.TelePart
    TexEstados = MiTabla.Estados
    TexCiud = MiTabla.Ciudad
    TexPais = MiTabla.Pais
  End If
End If
```

```
End If
End Sub
```

```
Private Sub TexCveCli_KeyPress(KeyAscii As Integer)
If DataFacturas.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    TexCveCli = LstClave2
    SendKeys "{tab}"
    LstClave.Visible = False
  End If
End If
End Sub
```

```
Private Sub TexCveEmp_GotFocus()
LstClave2.Visible = False
HighLight
If DataFacturas.Recordset.EditMode <> 0 Then
  If TexCveEmp <> "" Then
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set ~ MiRegistro =
MiBaseDatos.OpenRecordset("SELECT TbEmp.* From
Tbemp WHERE CveEmpleado = " & TexCveEmp & "",
dbOpenSnapshot)
```

```
  If MiRegistro.EOF Then
```

```
    MsgBox "La clave proporcionada no existe, favor de
verificar", vbExclamation
    TexCveEmp = ""
    TexCveEmp.SetFocus
  Else
    TexNomEmp = MiRegistro.Nombre
  End If
End If
End If
End Sub
```

```
Private Sub TexFolVenta_Change()
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(ListVenta, TexFolVenta.Text, -1)
ListVenta.ListIndex = Index
If Index >= 0 Then ListVenta.Selected(Index) = True
End Sub
```

```
Private Sub TexFolVenta_DbIclic()
If DataFacturas.EditMode <> 0 Then
  ListVenta.Visible = True
End If
End Sub
```

```
Private Sub TexFolVenta_GotFocus()
HighLight
End Sub
```

```
Private Sub TexFolVenta_KeyPress(KeyAscii As Integer)
TexFolVenta = ListVenta
SendKeys "{tab}"
End Sub
```

```
Private Sub TexFolVenta_LostFocus()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
Dim MiRegistro2 As Object
Dim añadir As String
Dim MiBD As Object
Dim MiTabla As Object

  BorrarTab "TbTmpdtlIFac"
  TexFolVenta = ListVenta
  Set MiBD = OpenDatabase(MiBDStr)
  Set MiTabla = MiBD.OpenRecordset("SELECT
TbTmpDtllVentas.* FROM TbTmpDtllVentas WHERE
DIIFolFac = " & TexFolVenta & "", dbOpenSnapshot)
  ListVenta.Visible = False
```

```
  Set MiBaseDatos = OpenDatabase(MiBDStr)
```

```
  Set ~ MiRegistro =
MiBaseDatos.OpenRecordset("SELECT DII_NoVentas,
DII_Cantidad, DII_Importe, DII_Descrip, DII_CostPub,
DII_ClaveCalzado, DII_ID, DII_Fac_Fol FROM
TbTmpdtlIFac ", dbOpenDynaset)
  Set ~ MiRegistro2 =
MiBaseDatos.OpenRecordset("SELECT TbDtllVentas.*
FROM TbDtllVentas WHERE DIIFolFac = " & TexFolVenta
& "", dbOpenDynaset)
  Do Until MiRegistro2.EOF
```

```
  MiRegistro.AddNew
```

```

        MiRegistro!DII_Cantidad =
MiRegistro2!DII_NoPares
        MiRegistro!DII_Importe = MiRegistro2!DII_Importe
        MiRegistro!DII_ClaveCalzado =
MiRegistro2!DII_ClaveCalzado
        MiRegistro!DII_Descrip =
MiRegistro2!DII_Descripcion
        MiRegistro!DII_CostPub =
MiRegistro2!DII_CostoPublico
        MiRegistro!DII_ID = MiRegistro2!DII_ID
        MiRegistro!DII_Fac_Fol = TexFolio
        MiRegistro!DII_NoVentas = TexFolVenta
        MiRegistro.Update
        MiRegistro2.MoveNext

```

```

        Loop
    ActualizarGrid
End Sub

```

```

Private Sub TexNoPedido_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
    LstClave.Visible = False
    LstClave2.Visible = False
    HighLight
End If
End Sub

```

```

Private Sub TexNoPedido_KeyPress(KeyAscii As Integer)
If DataFacturas.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexNoPedido_LostFocus()
If Not IsNumeric(TexNoPedido) And TexNoPedido <> ""
Then
    MsgBox "Poner solo números", vbExclamation
    TexNoPedido = ""
    TexNoPedido.SetFocus
End If
End Sub

```

```

Public Sub ActualizarGrid()
Dim i As Long
Dim MiSubtotal As Currency
i = 1
If DataFacturas.EditMode = 0 Then
    BorrarTab "TbTmpDtllFac"
    PasarDatos "TbDtllFac", "TbTmpDtllFac", "DII_Fac_Fol ="
    & TexFolio & ""
End If
With GDFac
    .Clear
    .Cols = 7
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

```

```

        Set MiBaseDatos = OpenDatabase(MiBDStr)
        Set MiRegistro = MiBaseDatos.OpenRecordset("select
DII_NoVentas, DII_ID, DII_Cantidad, DII_Importe,
DII_Descrip, DII_CostPub, DII_ClaveCalzado FROM
TbTmpdtllFac WHERE DII_Fac_Fol =" & TexFolio & "",
dbOpenSnapshot)

```

```

If MiRegistro.EOF Then
    GDFac.Clear
    GDFac.Rows = 2
End If
Do Until MiRegistro.EOF
    .Rows = MiRegistro.RecordCount + 2
    .TextMatrix(i, 0) = MiRegistro!DII_ID
    .TextMatrix(i, 2) = MiRegistro!DII_Cantidad
    .TextMatrix(i, 5) = MiRegistro!DII_ClaveCalzado
    .TextMatrix(i, 1) = MiRegistro!DII_Descrip
    .TextMatrix(i, 3) = Format(MiRegistro!DII_CostPub,
"currency")
    .TextMatrix(i, 4) = Format(MiRegistro!DII_Importe,
"currency")
    .TextMatrix(i, 6) = MiRegistro!DII_NoVentas
    MiSubtotal = MiSubtotal +
PonCero(MiRegistro!DII_Importe)
    i = i + 1
    MiRegistro.MoveNext
Loop

```

```

Dim MiRegistro2 As Object
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro2 = MiBaseDatos.OpenRecordset("SELECT
TbVentas.* From TbVentas WHERE NoVentas =" &
TexFolVenta & "", dbOpenSnapshot)

```

```

If Not MiRegistro2.EOF Then
    TexDesc = MiRegistro2.Descuento
End If
TexIVA.BackColor = ColorGris
TexSubTotDesc.BackColor = ColorGris
LabelSubTotal = Format(MiSubtotal, "currency")
TexSubTotDesc = Format((MiSubtotal -
(PonCero(TexDesc)), "currency")
TexIVA = Format(PonCero(TexSubTotDesc) * (IVA /
100), "currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")

```

```

.ColWidth(0) = 0
.ColWidth(5) = 0
.TextMatrix(0, 2) = "Pares"
.ColWidth(2) = 600
.TextMatrix(0, 1) = "Descripción"
.ColWidth(1) = 4000
.TextMatrix(0, 3) = "Precio Publico"
.ColAlignment(3) = 7
.ColWidth(3) = 1200
.TextMatrix(0, 4) = "Importe"
.ColWidth(4) = 1000
.TextMatrix(0, 6) = "No Ventas"
.ColWidth(6) = 0

```

```

End With
End Sub

```

```

Private Sub TexCveEmp_Change()
If DataFacturas.Recordset.EditMode <> 0 Then
    Dim Index As Integer
    On Error Resume Next
    Index = FindFirstMatch(LstClave, TexCveEmp.Text, -1)
    LstClave.ListIndex = Index
    If Index >= 0 Then LstClave.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCveEmp_DblClick()
If DataFacturas.Recordset.EditMode <> 0 Then

```

```

LstClave.Visible = True
End If
End Sub

```

```

Private Sub TexCveEmp_KeyPress(KeyAscii As Integer)
If DataFacturas.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        TexCveEmp = LstClave
        SendKeys "{tab}"
        LstClave.Visible = False
    End If
End If
End Sub

```

```

Public Sub Cancelacion()
If CheckCancel = 1 Then
    LabelFecha.Enabled = False
    TexFolio.Enabled = False
    TexCveEmp.Enabled = False
    TexNoPedido.Enabled = False
    TexFolVenta.Enabled = False
    GDFac.Enabled = False
    LabelSubTotal.Enabled = False
    TexIVA.Enabled = False
    LabelTotal.Enabled = False
    BotModificar.Enabled = False
    BotEliminar.Enabled = False
    BotImprimir.Enabled = False
    BotBuscar.Enabled = False
Else
    LabelFecha.Enabled = True
    TexFolio.Enabled = True
    TexCveEmp.Enabled = True
    TexNoPedido.Enabled = True
    TexFolVenta.Enabled = True
    GDFac.Enabled = True
    LabelSubTotal.Enabled = True
    TexIVA.Enabled = True
    LabelTotal.Enabled = True
End If
End Sub

```

```

Public Sub PasarDatos(Tb1 As String, Tb2 As String, Optional
WH As String)
Dim MiBD As Object
Dim MiTab1 As Object, MiTab2 As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTab1 = MiBD.OpenRecordset("SELECT " & Tb1 &
"* FROM " & Tb1 & IIf(WH = "", "", " WHERE " & WH),
dbOpenSnapshot)
Set MiTab2 = MiBD.OpenRecordset("SELECT " & Tb2 &
"* FROM " & Tb2, dbOpenDynaset)

Do Until MiTab1.EOF
    MiTab2.AddNew
    MiTab2!DII_Cantidad = MiTab1!DII_Cantidad
    MiTab2!DII_PrecUnit = MiTab1!DII_PrecUnit
    MiTab2!DII_Importe = MiTab1!DII_Importe
    MiTab2!DII_Fac_Fol = MiTab1!DII_Fac_Fol
    MiTab2!DII_ID = MiTab1!DII_ID
    MiTab2!DII_Descrip = MiTab1!DII_Descrip
    MiTab2!DII_NoVentas = MiTab1!DII_NoVentas
    MiTab2!Ventas_ID = MiTab1!Ventas_ID
    MiTab2!DII_ClaveCalzado = MiTab1!DII_ClaveCalzado
    MiTab2!DII_CostPub = MiTab1!DII_CostPub
    MiTab2.Update

```

```

MiTab1.MoveNext
Loop
End Sub

```

FACTURAS HISTORICOS

```

Option Explicit
Dim MiBol As Boolean

```

```

Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataFacturas.Recordset.EditMode = 0 Then
    If Not DataFacturas.Recordset.EOF Then
        MiReg = TexFolio
        DataFacturas.Recordset.MoveLast
    End If

    If DataFacturas.Recordset.RecordCount <= 0 Then
        BotPrimero.Enabled = False
        BotAnterior.Enabled = False
        BotSiguiente.Enabled = False
        BotUltimo.Enabled = False
        BotGuardar.Enabled = True
    ElseIf DataFacturas.Recordset.RecordCount <= 1 Then
        BotPrimero.Enabled = False
        BotAnterior.Enabled = False
        BotSiguiente.Enabled = False
        BotUltimo.Enabled = False
        BotGuardar.Enabled = True
        BotMenu.Enabled = True
    Else
        DataFacturas.Recordset.FindFirst " Folio =" & MiReg
        & ""
        If DataFacturas.Recordset.AbsolutePosition = 0 Then
            BotPrimero.Enabled = False
            BotAnterior.Enabled = False
            BotSiguiente.Enabled = True
            BotUltimo.Enabled = True
        ElseIf DataFacturas.Recordset.AbsolutePosition =
DataFacturas.Recordset.RecordCount - 1 Then

            BotPrimero.Enabled = True
            BotAnterior.Enabled = True
            BotSiguiente.Enabled = False
            BotUltimo.Enabled = False
        Else
            BotPrimero.Enabled = True
            BotAnterior.Enabled = True
            BotSiguiente.Enabled = True
            BotUltimo.Enabled = True
        End If

        BotGuardar.Enabled = True
        BotMenu.Enabled = True

    End If
Else
    BotGuardar.Enabled = True

    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False

```

```
BotMenu.Enabled = False
```

```
End If
BloqueaControles
ActualizarGrid
Cancelacion
Text2.Locked = False
Text2.BackColor = vbWhite
Text3.Locked = False
Text3.BackColor = vbWhite
End Sub
```

```
Private Sub BotAgregar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
DataFacturas.Recordset.AddNew
TexDesc = 0
HabilitarBotones
Cancelacion
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(Folio) as Folio1 From TbFacturas", dbOpenSnapshot)
If Not IsNull(MiRegistro!Folio1) Then
    TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
    TexFolio.BackColor = ColorGris
    TexNomCli.BackColor = ColorGris
    TexCol.BackColor = ColorGris
    TexCalle.BackColor = ColorGris
    TexCP.BackColor = ColorGris
    TexRFC.BackColor = ColorGris
    TexTelNeg.BackColor = ColorGris
    TexTelPart.BackColor = ColorGris
    TexEstados.BackColor = ColorGris
    TexCiud.BackColor = ColorGris
    TexPais.BackColor = ColorGris
    Text1.BackColor = ColorGris
    GDFac.BackColor = ColorGris
Else
    TexFolio = "0001"
    TexFolio.BackColor = ColorGris
    TexNomCli.BackColor = ColorGris
    TexCol.BackColor = ColorGris
    TexCalle.BackColor = ColorGris
    TexCP.BackColor = ColorGris
    TexRFC.BackColor = ColorGris
    TexTelNeg.BackColor = ColorGris
    TexTelPart.BackColor = ColorGris
    TexEstados.BackColor = ColorGris
    TexCiud.BackColor = ColorGris
    TexPais.BackColor = ColorGris
    Text1.BackColor = ColorGris
    GDFac.BackColor = ColorGris
    TexFolVenta = ""
End If
TexDesc.BackColor = ColorGris
TexDesc.Locked = True
TexSubTotDesc.Locked = True
TexIVA.Locked = True
LabelFecha = Date
TexPais = "México"
TexNomEmp.BackColor = ColorGris
End Sub
```

```
Private Sub BotAnterior_Click()
DataFacturas.Recordset.MovePrevious
HabilitarBotones
Cancelacion
```

```
End Sub
```

```
Private Sub BotBuscar_Click()
FoBuscFac.Show vbModal
End Sub
```

```
Private Sub BotCancelar_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
    LstClave.Visible = False
    LstClave2.Visible = False
End If
End Sub
```

```
Private Sub BotEliminar_Click()
Dim Eliminar As String
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
    Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
    If Eliminar = vbYes Then
        CheckCancel.Value = 1
        Label21 = "¡Factura Cancelada!"
        LabelFecha.Enabled = False
        TexFolio.Enabled = False
        TexCveEmp.Enabled = False
        TexNoPedido.Enabled = False
        TexFolVenta.Enabled = False
        GDFac.Enabled = False
        LabelSubTotal.Enabled = False
        TexIVA.Enabled = False
        LabelTotal.Enabled = False
    Else
        Label21 = ""
        LabelFecha.Enabled = True
        TexFolio.Enabled = True
        TexCveEmp.Enabled = True
        TexNoPedido.Enabled = True
        TexFolVenta.Enabled = True
        GDFac.Enabled = True
        LabelSubTotal.Enabled = True
        TexIVA.Enabled = True
        LabelTotal.Enabled = True
    End If
HabilitarBotones
End Sub
```

```
Private Sub BotGuardar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
Dim Confirm As String
```

```
Confirm = MsgBox("Se dispone a pasar toda la información
contenida en la forma de Facturas hacia Historico Facturas, y
no se podrá regresar.¿Desea continuar?", vbYesNo +
vbCritical, "ADVERTENCIA")
If Confirm = vbYes Then
```

```
    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
CondHist FROM TbFacturas WHERE (((Fecha) Between # &
Format(Text2, "mm/dd/yy") & "# And #" & Format(Text3,
"mm/dd/yy") & "#))", dbOpenDynaset)
```

```
Do Until MiRegistro.EOF
    MiRegistro.Edit
```

```

MiRegistro!CondHist = 1
MiRegistro.Update
MiRegistro.MoveNext
Loop
End If
End Sub

```

```

Private Sub BotGuardar_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
LstClave.Visible = False
LstClave2.Visible = False
End If
End Sub

```

```

Private Sub BotImprimir_Click()
FoOpRpFac.Show
End Sub

```

```

Private Sub BotMenu_Click()
FoMenuHist.Show
Unload FoFacturasHist
End Sub

```

```

Private Sub BotModificar_Click()
DataFacturas.Recordset.Edit
HabilitaBotones
TexFolio.BackColor = ColorGris
TexNomCli.BackColor = ColorGris
TexCol.BackColor = ColorGris
TexCalle.BackColor = ColorGris
TexCP.BackColor = ColorGris
TexRFC.BackColor = ColorGris
TexTelNeg.BackColor = ColorGris
TexTelPart.BackColor = ColorGris
TexEstados.BackColor = ColorGris
TexCiud.BackColor = ColorGris
TexPais.BackColor = ColorGris
Text1.BackColor = ColorGris
GDFac.BackColor = ColorGris
TexNomEmp.BackColor = ColorGris
TexFolVenta.Enabled = True
TexDesc.BackColor = ColorGris
TexDesc.Locked = True
TexSubTotDesc.Locked = True
TexIVA.Locked = True
End Sub

```

```

Private Sub BotPrimero_Click()
DataFacturas.Recordset.MoveFirst
HabilitaBotones
Cancelacion
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataFacturas.Recordset.EOF And
DataFacturas.Recordset.RecordCount > 1 Then
DataFacturas.Recordset.MoveNext
End If
HabilitaBotones
Cancelacion
End Sub

```

```

Private Sub BotUltimo_Click()

```

```

DataFacturas.Recordset.MoveLast
HabilitaBotones
Cancelacion
End Sub

```

```

Private Sub BtFiltrar_Click()
If IsDate(Text2) And IsDate(Text3) Then
DataFacturas.RecordSource = "SELECT TbFacturas.*
FROM TbFacturas WHERE ((TbFacturas.Fecha) Between #"
& Format(Text2, "mm/dd/yy") & "# And #" & Format(Text3,
"mm/dd/yy") & "#)"
DataFacturas.Refresh
HabilitaBotones
ElseIf Not IsDate(Text2) Then
MsgBox "Fecha incorrecta, favor de verificar",
vbInformation
Text2 = ""
ElseIf Not IsDate(Text3) Then
MsgBox "Fecha incorrecta, favor de verificar",
vbInformation
Text3 = ""
End If
End Sub

```

```

Private Sub Form_Activate()
HabilitaBotones

```

```

Text2 = "01" & Format(Date, "-mm-yyyy")
Text3 = DiaUltimoDelMes(Date) & Format(Date, "-mm-
yyyy")
Text2.Locked = False
Text2.BackColor = vbWhite
Text3.Locked = False
Text3.BackColor = vbWhite

```

```

TexFolio.BackColor = &H8000000F
TexNomCli.BackColor = &H8000000F
TexCol.BackColor = &H8000000F
TexCalle.BackColor = &H8000000F
TexCP.BackColor = &H8000000F
TexRFC.BackColor = &H8000000F
TexTelNeg.BackColor = &H8000000F
TexTelPart.BackColor = &H8000000F
TexEstados.BackColor = &H8000000F
TexCiud.BackColor = &H8000000F
TexPais.BackColor = &H8000000F
Text1.BackColor = &H8000000F
TexNomEmp.BackColor = ColorGris
TexFolVenta.BackColor = ColorGris
MiBol = True
End Sub

```

```

Private Sub Form_Load()
FoMenu.Hide
DataFacturas.DatabaseName = MiBDStr
LlenarCombo LstClave, "TbEmp", "CveEmpleado",
"Empleado_ID"
LlenarCombo LstClave2, "TbClientes", "CveClientes",
"Clientes_ID"
Dim MiBaseDatos As Object
Dim MiRegistro As Object

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
NoVentas,Ventas_ID From TbVentas WHERE CheckFact =
true ", dbOpenSnapshot)

```

```

Do Until MiRegistro.EOF

```

```
ListVenta.AddItem IIf(IsNull(MiRegistro!NoVentas), "",
MiRegistro!NoVentas)
ListVenta.ItemData(ListVenta.NewIndex) =
MiRegistro!Ventas_ID
MiRegistro.MoveNext
Loop
```

```
MiBol = False
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub
```

```
Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataFacturas.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataFacturas.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is MSFlexGrid Then
If DataFacturas.Recordset.EditMode = 0 Then
.Controls(i).BackColor = &H80000004
Else
.Controls(i).BackColor = vbWhite
End If
End If
Next
End With
End Sub
```

```
Private Sub GDFac_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
LstClave.Visible = False
LstClave2.Visible = False
HighLight
End If
End Sub
```

```
Private Sub LabelTotal_Change()
If MiBol = True Then
Text1 = ImporteLetra(LabelTotal)
End If
```

```
End Sub
```

```
Private Sub LstClave_DbClick()
TexCveEmp = LstClave
LstClave.Visible = False
TexCveEmp.SetFocus
End Sub
```

```
Private Sub LstClave2_DbClick()
TexCveCli = LstClave2
LstClave2.Visible = False
TexCveCli.SetFocus
End Sub
```

```
Private Sub TexCveCli_Change()
If DataFacturas.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave2, TexCveCli.Text, -1)
LstClave2.ListIndex = Index
If Index >= 0 Then LstClave2.Selected(Index) = True
End If
End Sub
```

```
Private Sub TexCveCli_DbClick()
If DataFacturas.Recordset.EditMode <> 0 Then
LstClave2.Visible = True
End If
End Sub
```

```
Private Sub TexCveCli_GotFocus()
If DataFacturas.Recordset.EditMode <> 0 Then
LstClave.Visible = False
HighLight
If TexCveCli <> "" Then
Dim MiBD As Object
Dim MiTabla As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTabla = MiBD.OpenRecordset("SELECT Estados,
TbClientes.* FROM TbClientes INNER JOIN TbEstados ON
TbClientes.Estado = TbEstados.Estados_ID WHERE
CveClientes = " & TexCveCli & "", dbOpenSnapshot)
```

```
If Not MiTabla.EOF Then
TexNomCli = MiTabla.NombCliente
TexCol = MiTabla.Colonia
TexCalle = MiTabla.Calle
TexCP = MiTabla.CP
TexRFC = MiTabla.RFC
TexTelNeg = MiTabla.TeleNeg
TexTelPart = MiTabla.TelePart
TexEstados = MiTabla.Estados
TexCiud = MiTabla.Ciudad
TexPais = MiTabla.Pais
End If
End If
```

```
End If
```


End Sub

Private Sub TexCveCli_KeyPress(KeyAscii As Integer)

If DataFacturas.Recordset.EditMode <> 0 Then

If KeyAscii = 13 Then

 TexCveCli = LstClave2

 SendKeys "{tab}"

 LstClave.Visible = False

End If

End If

End Sub

Private Sub TexCveEmp_GotFocus()

LstClave2.Visible = False

HighLight

If DataFacturas.Recordset.EditMode <> 0 Then

If TexCveEmp <> "" Then

 Dim MiBaseDatos As Object

 Dim MiRegistro As Object

 Set MiBaseDatos = OpenDatabase(MiBDStr)

 Set MiRegistro =

 MiBaseDatos.OpenRecordset("SELECT TbEmp.* From
Tbemp WHERE CveEmpleado ='" & TexCveEmp & """,
dbOpenSnapshot)

 If MiRegistro.EOF Then

 MsgBox "La clave proporcionada no existe, favor de
verificar", vbExclamation

 TexCveEmp = ""

 TexCveEmp.SetFocus

 Else

 TexNomEmp = MiRegistro.Nombre

 End If

End If

End If

End Sub

Private Sub TexFolVenta_Change()

Dim Index As Integer

On Error Resume Next

Index = FindFirstMatch(ListVenta, TexFolVenta.Text, -1)

ListVenta.ListIndex = Index

If Index >= 0 Then ListVenta.Selected(Index) = True

End Sub

Private Sub TexFolVenta_Db1Click()

ListVenta.Visible = True

End Sub

Private Sub TexFolVenta_GotFocus()

HighLight

End Sub

Private Sub TexFolVenta_KeyPress(KeyAscii As Integer)

TexFolVenta = ListVenta

SendKeys "{tab}"

End Sub

Private Sub TexFolVenta_LostFocus()

Dim MiBaseDatos As Object

Dim MiRegistro As Object

Dim MiRegistro2 As Object

Dim añadir As String

Dim MiBD As Object

Dim MiTabla As Object

TexFolVenta = ListVenta

Set MiBD = OpenDatabase(MiBDStr)

Set MiTabla = MiBD.OpenRecordset("SELECT
TbTmpDtIIVentas.* FROM TbTmpDtIIVentas WHERE
DtIFolFac = "" & TexFolVenta & """, dbOpenSnapshot)

ListVenta.Visible = False

Set MiBaseDatos = OpenDatabase(MiBDStr)

Set MiRegistro =

MiBaseDatos.OpenRecordset("SELECT DtII_NoVentas,
DtII_Cantidad, DtII_Importe, DtII_Descrip, DtII_CostPub,
DtII_ClaveCalzado, DtII_ID, DtII_Fac_Fol FROM
TbTmptdtIIFac ", dbOpenDynaset)

Set MiRegistro2 =

MiBaseDatos.OpenRecordset("SELECT TbTmpDtIIVentas.*
FROM TbTmpDtIIVentas WHERE DtIFolFac = "" &
TexFolVenta & """, dbOpenDynaset)

Do Until MiRegistro2.EOF

 MiRegistro.AddNew

 MiRegistro!DtII_Cantidad

 MiRegistro2!DtII_NoPares

 MiRegistro!DtII_Importe = MiRegistro2!DtII_Importe

 MiRegistro!DtII_ClaveCalzado

 MiRegistro2!DtII_ClaveCalzado

 MiRegistro!DtII_Descrip

 MiRegistro2!DtII_Descrip

 MiRegistro!DtII_CostPub

 MiRegistro2!DtII_CostoPublico

 MiRegistro!DtII_ID = MiRegistro2!DtII_ID

 MiRegistro!DtII_Fac_Fol = TexFolio

 MiRegistro!DtII_NoVentas = TexFolVenta

 MiRegistro.Update

 MiRegistro2.MoveNext

Loop

ActualizarGrid

End Sub

Private Sub TexNoPedido_GotFocus()

If DataFacturas.Recordset.EditMode <> 0 Then

 LstClave.Visible = False

 LstClave2.Visible = False

 HighLight

End If

End Sub

Private Sub TexNoPedido_KeyPress(KeyAscii As Integer)

If DataFacturas.Recordset.EditMode <> 0 Then

 If KeyAscii = 13 Then

 SendKeys "{tab}"

 End If

End If

End Sub

Private Sub TexNoPedido_LostFocus()

If Not IsNumeric(TexNoPedido) And TexNoPedido <> ""
Then

 MsgBox "Poner solo números", vbExclamation

 TexNoPedido = ""

 TexNoPedido.SetFocus

```
End If
End Sub
```

```
Public Sub ActualizarGrid()
```

```
Dim i As Long
Dim MiSubtotal As Currency
```

```
i = 1
```

```
With GDFac
```

```
.Clear
```

```
.Cols = 7
```

```
Dim MiBaseDatos As Object
```

```
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("select
DII_NoVentas, DII_ID, DII_Cantidad, DII_Importe,
DII_Descrip, DII_CostPub, DII_ClaveCalzado FROM
TbdIIFac WHERE DII_Fac_Fol ='" & TexFolio & "',
dbOpenSnapshot)
```

```
If MiRegistro.EOF Then
```

```
GDFac.Clear
```

```
GDFac.Rows = 2
```

```
End If
```

```
Do Until MiRegistro.EOF
```

```
.Rows = MiRegistro.RecordCount + 2
```

```
.TextMatrix(i, 0) = MiRegistro!DII_ID
```

```
.TextMatrix(i, 2) = MiRegistro!DII_Cantidad
```

```
.TextMatrix(i, 5) = MiRegistro!DII_ClaveCalzado
```

```
.TextMatrix(i, 1) = MiRegistro!DII_Descrip
```

```
.TextMatrix(i, 3) = Format(MiRegistro!DII_CostPub,
```

```
"currency")
```

```
.TextMatrix(i, 4) = Format(MiRegistro!DII_Importe,
```

```
"currency")
```

```
.TextMatrix(i, 6) = MiRegistro!DII_NoVentas
```

```
MiSubtotal = MiSubtotal +
```

```
PonCero(MiRegistro!DII_Importe)
```

```
i = i + 1
```

```
MiRegistro.MoveNext
```

```
Loop
```

```
Dim MiRegistro2 As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
```

```
Set MiRegistro2 = MiBaseDatos.OpenRecordset("SELECT
TbVentas.* From TbVentas WHERE NoVentas ='" &
TexFolVenta & "', dbOpenSnapshot)
```

```
If Not MiRegistro2.EOF Then
```

```
TexDesc = MiRegistro2.Descuento
```

```
End If
```

```
TexIVA.BackColor = ColorGris
```

```
TexSubTotDesc.BackColor = ColorGris
```

```
LabelSubTotal = Format(MiSubtotal, "currency")
```

```
TexSubTotDesc = Format((MiSubtotal) -
```

```
(PonCero(TexDesc)), "currency")
```

```
TexIVA = Format(PonCero(TexSubTotDesc) * (IVA /
100), "currency")
```

```
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
```

```
.ColWidth(0) = 0
```

```
.ColWidth(5) = 0
```

```
.TextMatrix(0, 2) = "Pares"
```

```
.ColWidth(2) = 600
```

```
.TextMatrix(0, 1) = "Descripción"
```

```
.ColWidth(1) = 4000
```

```
.TextMatrix(0, 3) = "Precio Publico"
```

```
.ColAlignment(3) = 7
```

```
.ColWidth(3) = 1200
```

```
.TextMatrix(0, 4) = "Importe"
```

```
.ColWidth(4) = 1000
```

```
.TextMatrix(0, 6) = "No Ventas"
```

```
.ColWidth(6) = 0
```

```
End With
```

```
End Sub
```

```
Private Sub TexCveEmp_Change()
```

```
If DataFacturas.Recordset.EditMode <> 0 Then
```

```
Dim Index As Integer
```

```
On Error Resume Next
```

```
Index = FindFirstMatch(LstClave, TexCveEmp.Text, -1)
```

```
LstClave.ListIndex = Index
```

```
If Index >= 0 Then LstClave.Selected(Index) = True
```

```
End If
```

```
End Sub
```

```
Private Sub TexCveEmp_DbClick()
```

```
If DataFacturas.Recordset.EditMode <> 0 Then
```

```
LstClave.Visible = True
```

```
End If
```

```
End Sub
```

```
Private Sub TexCveEmp_KeyPress(KeyAscii As Integer)
```

```
If DataFacturas.Recordset.EditMode <> 0 Then
```

```
If KeyAscii = 13 Then
```

```
TexCveEmp = LstClave
```

```
SendKeys "{tab}"
```

```
LstClave.Visible = False
```

```
End If
```

```
End If
```

```
End Sub
```

```
Public Sub Cancelacion()
```

```
If CheckCancel = 1 Then
```

```
Label21 = "¡Factura Cancelada!"
```

```
LabelFecha.Enabled = False
```

```
TexFolio.Enabled = False
```

```
TexCveEmp.Enabled = False
```

```
TexNoPedido.Enabled = False
```

```
TexFolVenta.Enabled = False
```

```
GDFac.Enabled = False
```

```
LabelSubTotal.Enabled = False
```

```
TexIVA.Enabled = False
```

```
LabelTotal.Enabled = False
```

```
Else
```

```
Label21 = ""
```

```
LabelFecha.Enabled = True
```

```
TexFolio.Enabled = True
```

```
TexCveEmp.Enabled = True
```

```
TexNoPedido.Enabled = True
```

```
TexFolVenta.Enabled = True
```

```
GDFac.Enabled = True
```

```
LabelSubTotal.Enabled = True
```

```
TexIVA.Enabled = True
```

```
LabelTotal.Enabled = True
```

```
End If
```

```
End Sub
```

INVENTARIO

```
Option Explicit
```

```
Dim MiCalzado As String
```

```

Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataInventario.Recordset.EditMode = 0 Then
If Not DataInventario.Recordset.EOF Then
MiReg = TexCveCalz
DataInventario.Recordset.MoveLast
End If

If DataInventario.Recordset.RecordCount <= 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = False
BotModificar.Enabled = False
BtClave.Enabled = False
BotMenu.Enabled = True
BotImprimir.Enabled = False
BotBuscar.Enabled = False
ElseIf DataInventario.Recordset.RecordCount <= 1 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BtClave.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True

Else

DataInventario.Recordset.FindFirst " ClaveCalzado =""
& MiReg & ""
If DataInventario.Recordset.AbsolutePosition = 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = True
BotUltimo.Enabled = True

ElseIf DataInventario.Recordset.AbsolutePosition =
DataInventario.Recordset.RecordCount - 1 Then

BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
Else
BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = True
BotUltimo.Enabled = True
End If

BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BtClave.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True

```

```

BotBuscar.Enabled = True
End If

```

```
Else
```

```

BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True
BtClave.Enabled = True

```

```

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False

```

```

BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False

```

```
End If
```

```
BloqueaControles
```

```
End Sub
```

```

Private Sub BotAgregar_Click()
DataInventario.Recordset.AddNew
LabelFechIngre = Date
HabilitarBotones
TexDescrip.BackColor = &H8000000F
TexDescrip.Locked = True
TexCveCalz.BackColor = &H8000000F
TexCveCalz.Locked = True
TexInvenID = IncrementadorID("TbVentas")

```

```

CombMarca.Enabled = True
CombSexo.Enabled = True
CombEstilo.Enabled = True
CombLinea.Enabled = True
CombMater.Enabled = True
CombColor.Enabled = True
TexModel.Enabled = True
CombMarca.SetFocus
End Sub

```

```

Private Sub BotAnterior_Click()
DataInventario.Recordset.MovePrevious
HabilitarBotones
Stocks2
Stocks
End Sub

```

```

Private Sub BotBuscar_Click()
FoBuscInv.Show vbModal
End Sub

```

```

Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
DataInventario.Recordset.CancelUpdate
HabilitarBotones
End If
MiCalzado = ""
End Sub

```

```

Private Sub BotEliminar_Click()
Dim Eliminar As String
Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
If Eliminar = vbYes Then
With DataInventario.Recordset
.Delete
.Requery
If Not .EOF Then
.MoveLast
Else
BotEliminar.Enabled = False
.Requery
DataInventario.Refresh
End If
End With
End If
HabilitarBotones
End Sub

```

```

Private Sub BotGuardar_Click()
Dim CLAVE As String
Dim añadir As String

If Trim(TexCostUni) = "" Or Trim(TexStkMin) = "" Or
Trim(TexStkMax) = "" Then
añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
Else
If TexCveCalz = "" Then
CLAVE = MsgBox("Para poder guardar, debe generar la
Clave de Calzado", vbInformation)
BtClave.SetFocus
Else
DataInventario.Recordset.Update
DataInventario.Refresh
HabilitarBotones
MiCalzado = ""
BotAgregar.SetFocus
End If
End If
End Sub

```

```

Private Sub BotImprimir_Click()
FoOpRpInv.Show
End Sub

```

```

Private Sub BotMenu_Click()
FoMenu.Show
Unload FoInvent
End Sub

```

```

Private Sub BotModificar_Click()
DataInventario.Recordset.Edit
HabilitarBotones
MiCalzado = TexCveCalz
TexCveCalz.BackColor = &H8000000F
TexCveCalz.Locked = True
TexDescrip.BackColor = &H8000000F
TexDescrip.Locked = True

```

```

CombMarca.Enabled = False
CombSexo.Enabled = False
CombEstilo.Enabled = False
CombLinea.Enabled = False
CombMater.Enabled = False

```

```

CombColor.Enabled = False
TexModel.Enabled = False

CombMarca.BackColor = ColorGris
CombSexo.BackColor = ColorGris
CombEstilo.BackColor = ColorGris
CombLinea.BackColor = ColorGris
CombMater.BackColor = ColorGris
CombColor.BackColor = ColorGris
TexModel.BackColor = ColorGris
End Sub

```

```

Private Sub BotPrimero_Click()
DataInventario.Recordset.MoveFirst
HabilitarBotones
Stocks2
Stocks
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataInventario.Recordset.EOF And
DataInventario.Recordset.RecordCount > 1 Then
DataInventario.Recordset.MoveNext
End If
HabilitarBotones
Stocks2
Stocks
End Sub

```

```

Private Sub BotUltimo_Click()
DataInventario.Recordset.MoveLast
HabilitarBotones
Stocks2
Stocks
End Sub

```

```

Private Sub BtClave_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub CombColor_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub CombEstilo_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub CombLinea_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then

```

```

    SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub CombMarca_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub CombMater_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub CombSexo_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Form_Activate()
HabilitaBotones
BotAgrega.SetFocus

```

```

Dim MiBaseDatos As Object
Dim MiRegistro As Object

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TxExist From TbSeg WHERE Contrasena ='" & Contrasena &
"'", dbOpenDynaset)

```

```

If MiRegistro!TxExist = False Then
    TxExist.Enabled = False
Else
    TxExist.Enabled = True
End If
Stocks2
End Sub

```

```

Private Sub Form_Load()
FoMenu.Hide
LlenarCombo CombColor, "TbColores", "Colores", "Co_ID"
LlenarCombo CombEstilo, "TbEstilos", "Estilos", "Estilos_ID"
LlenarCombo CombLinea, "TbLineas", "Lineas", "Lineas_ID"
LlenarCombo CombMarca, "TbMarcas", "Marcas",
"Marcas_ID"
LlenarCombo CombMater, "TbMaterial", "Material",
"Material_ID"
LlenarCombo CombSexo, "TbSexo", "Sexo", "Sexo_ID"
DataInventario.DatabaseName = MiBDStr
End Sub

```

```

Private Sub TexGotFocus(Index As Integer)

```

```

SeleccionarCampo
End Sub

```

```

Public Sub SeleccionarCampo()
With Screen.ActiveForm
    If TypeOf .ActiveControl Is TextBox Then
        .ActiveControl.SelStart = 0
        .ActiveControl.SelLength = Len(.ActiveControl)
    End If
End With
End Sub

```

```

Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
    For i = 0 To .Controls.Count - 1
        If TypeOf .Controls(i) Is TextBox Then
            If DataInventario.Recordset.EditMode = 0 Then
                .Controls(i).Locked = True
                .Controls(i).BackColor = &H80000004
            Else
                .Controls(i).Locked = False
                .Controls(i).BackColor = vbWhite
            End If
        End If
        If TypeOf .Controls(i) Is ComboBox Then
            If DataInventario.Recordset.EditMode = 0 Then
                .Controls(i).Locked = True
                .Controls(i).BackColor = &H80000004
            Else
                .Controls(i).Locked = False
                .Controls(i).BackColor = vbWhite
            End If
        End If
    Next
End With

```

```

End Sub

```

```

Private Sub TexCostMay_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCostMay_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexCostMay_LostFocus()
TexCostMay = PonCero(TexCostMay)
End Sub

```

```

Private Sub TexCostMay_Validate(Cancel As Boolean)
If Not IsNumeric(TexCostMay) And TexCostMay <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexCostMay = ""
    TexCostMay.SetFocus
End If
End Sub

```

```

Private Sub TexCostMen_GotFocus()
HighLight

```

End Sub

```
Private Sub TexCostMen_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexCostMen_LostFocus()
TexCostMen = PonCero(TexCostMen)
End Sub
```

```
Private Sub TexCostMen_Validate(Cancel As Boolean)
If Not IsNumeric(TexCostMen) And TexCostMen <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexCostMen = ""
    TexCostMen.SetFocus
End If
End Sub
```

```
Private Sub TexCostUni_GotFocus()
HighLight
End Sub
```

```
Private Sub TexCostUni_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexCostUni_Validate(Cancel As Boolean)
If Not IsNumeric(TexCostUni) And TexCostUni <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexCostUni = ""
    TexCostUni.SetFocus
End If
End Sub
```

```
Private Sub TexExist_GotFocus()
HighLight
End Sub
```

```
Private Sub TexExist_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexExist_Validate(Cancel As Boolean)
Stocks2
End Sub
```

```
Private Sub TexModel_GotFocus()
HighLight
End Sub
```

```
Private Sub TexModel_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexStkMax_GotFocus()
HighLight
End Sub
```

```
Private Sub TexStkMax_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexStkMax_Validate(Cancel As Boolean)
If Not IsNumeric(TexStkMax) And TexStkMax <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexStkMax = ""
    TexStkMax.SetFocus
End If
Stocks2
End Sub
```

```
Private Sub TexStkMin_GotFocus()
HighLight
End Sub
```

```
Private Sub TexStkMin_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub TexStkMin_Validate(Cancel As Boolean)
If Not IsNumeric(TexStkMin) And TexStkMin <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexStkMin = ""
    TexStkMin.SetFocus
End If
Stocks2
End Sub
```

```
Private Sub BtClave_Click()
Dim añadir As String
```

```
If CombMarca = "" Or CombSexo = "" Or CombEstilo = "" Or
CombLinea = "" Or CombMater = "" Or CombColor = "" Or
Trim(TexModel) = "" Then
    añadir = MsgBox("Para poder generar la clave, debe llenar
los campos", vbInformation)
Else
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object
```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbMarcas WHERE Marcas_ID =" &
CombMarca.ItemData(CombMarca.ListIndex),
dbOpenDynaset)

```

```

TexCveCalz = MiRegistro!Iniciales
TexDescrip = CombMarca
MiRegistro.Close

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbSexo WHERE Sexo_ID =" &
CombSexo.ItemData(CombSexo.ListIndex), dbOpenDynaset)
TexCveCalz = TexCveCalz + MiRegistro!Iniciales
TexDescrip = TexDescrip & ", " & CombSexo
MiRegistro.Close

```

```

TexDescrip = TexDescrip & ", " & CombEstilo

```

```

TexCveCalz = TexCveCalz & TexModel

```

```

TexDescrip = TexDescrip & ", " & CombMater

```

```

TexDescrip = TexDescrip & ", " & CombColor

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Lineas From TbLineas WHERE Lineas_ID =" &
CombLinea.ItemData(CombLinea.ListIndex),
dbOpenDynaset)
TexCveCalz = TexCveCalz + CStr(MiRegistro!Lineas)
TexDescrip = TexDescrip & ", " & CombLinea
MiRegistro.Close

```

```

TexCveCalz = UCase(TexCveCalz)
End If
End Sub

```

```

Private Sub CombColor_Click()
If CombColor <> "" Then
    TexColoresID =
CombColor.ItemData(CombColor.ListIndex)
End If
End Sub

```

```

Private Sub CombEstilo_Click()
If CombEstilo <> "" Then
    TexEstilosID =
CombEstilo.ItemData(CombEstilo.ListIndex)
End If
End Sub

```

```

Private Sub CombLinea_Click()
If CombLinea <> "" Then
    TexLineasID =
CombLinea.ItemData(CombLinea.ListIndex)
End If
End Sub

```

```

Private Sub CombMarca_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
If Trim(CombMarca) <> "" Then
    TexMarcaID =
CombMarca.ItemData(CombMarca.ListIndex)

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbMarcas WHERE Marcas_ID =" &
CombMarca.ItemData(CombMarca.ListIndex),
dbOpenDynaset)
End If
End Sub

```

```

Private Sub CombMater_Click()
If CombMater <> "" Then
    TexMaterialID =
CombMater.ItemData(CombMater.ListIndex)
End If
End Sub

```

```

Private Sub CombSexo_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
If Trim(CombSexo) <> "" Then
    TexSexoID = CombSexo.ItemData(CombSexo.ListIndex)
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbSexo WHERE Sexo_ID =" &
CombSexo.ItemData(CombSexo.ListIndex), dbOpenDynaset)
End If
End Sub

```

```

Private Sub TexEstilosID_Change()
BusquedaIdCombo TexEstilosID, CombEstilo
End Sub

```

```

Private Sub TexLineasID_Change()
BusquedaIdCombo TexLineasID, CombLinea
End Sub

```

```

Private Sub TexMarcaID_Change()
BusquedaIdCombo TexMarcaID, CombMarca
End Sub

```

```

Private Sub TexMaterialID_Change()
BusquedaIdCombo TexMaterialID, CombMater
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
FoInvent.Show
End Sub

```

```

Private Sub TexModel_LostFocus()
If Not IsNumeric(TexModel) And TexModel <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexModel = ""
    TexModel.SetFocus
End If
End Sub

```

```

Private Sub TexColoresID_Change()
BusquedaIdCombo TexColoresID, CombColor
End Sub

```

```

Private Sub TexSexoID_Change()

```

```

BusquedaldCombo TexSexoID, CombSexo
End Sub

```

```

Public Sub Stocks()
  If TexCveCalz <> "" Then
    If CInt(PonCero(TexStkMin)) < CInt(PonCero(TexExist))
    And CInt(PonCero(TexStkMax)) > CInt(PonCero(TexExist))
    Then
      TexStkMin.ForeColor = &H0& 'Color negro
      TexExist.ForeColor = &H0& 'Color negro
      TexStkMax.ForeColor = &H0& 'Color negro
      Elseif CInt(PonCero(TexStkMin)) <
CInt(PonCero(TexExist)) And CInt(PonCero(TexExist)) >
CInt(PonCero(TexStkMax)) Then
        MsgBox "La existencia actual es alta, se recomienda
ya no surtir.", vbExclamation
        TexStkMin.ForeColor = &H0& 'Color negro
        TexExist.ForeColor = &HC0& 'Color Rojo
        TexStkMax.ForeColor = &HC0& 'Color Rojo
        Elseif CInt(PonCero(TexStkMin)) >
CInt(PonCero(TexExist)) < CInt(PonCero(TexStkMax)) Then
          MsgBox "La existencia actual es baja, se recomienda
surtir.", vbExclamation
          TexStkMin.ForeColor = &HC0& 'Color Rojo
          TexExist.ForeColor = &HC0& 'Color Rojo
          TexStkMax.ForeColor = &H0& 'Color negro
          Elseif CInt(PonCero(TexStkMin)) >
CInt(PonCero(TexExist)) > CInt(PonCero(TexStkMax)) Then
            TexStkMin.ForeColor = &HC0& 'Color Rojo
            TexExist.ForeColor = &HC0& 'Color Rojo
            TexStkMax.ForeColor = &HC0& 'Color Rojo
            End If
          End If
        End Sub

```

```

Public Sub Stocks2()
  If TexCveCalz <> "" Then
    If CInt(PonCero(TexStkMin)) < CInt(PonCero(TexExist))
    And CInt(PonCero(TexStkMax)) > CInt(PonCero(TexExist))
    Then
      TexStkMin.ForeColor = &H0& 'Color negro
      TexExist.ForeColor = &H0& 'Color negro
      TexStkMax.ForeColor = &H0& 'Color negro
      Elseif CInt(PonCero(TexStkMin)) <
CInt(PonCero(TexExist)) And CInt(PonCero(TexExist)) >
CInt(PonCero(TexStkMax)) Then
        TexStkMin.ForeColor = &H0& 'Color negro
        TexExist.ForeColor = &HC0& 'Color Rojo
        TexStkMax.ForeColor = &HC0& 'Color Rojo
        Elseif CInt(PonCero(TexStkMin)) >
CInt(PonCero(TexExist)) < CInt(PonCero(TexStkMax)) Then
          TexStkMin.ForeColor = &HC0& 'Color Rojo
          TexExist.ForeColor = &HC0& 'Color Rojo
          TexStkMax.ForeColor = &H0& 'Color negro
          Elseif CInt(PonCero(TexStkMin)) >
CInt(PonCero(TexExist)) > CInt(PonCero(TexStkMax)) Then
            TexStkMin.ForeColor = &HC0& 'Color Rojo
            TexExist.ForeColor = &HC0& 'Color Rojo
            TexStkMax.ForeColor = &HC0& 'Color Rojo
            End If
          End If
        End Sub

```

```

Private Sub TexUbica_GotFocus()
  HighLight
End Sub

```

```

Private Sub TexUbica_KeyPress(KeyAscii As Integer)
  If DataInventario.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

MENU

```

Option Explicit
Dim MiCalzado As String

```

```

Private Sub HabilitarBotones()
  Dim MiReg As String
  BloqueaControles
  If DataInventario.Recordset.EditMode = 0 Then
    If Not DataInventario.Recordset.EOF Then
      MiReg = TexCveCalz
      DataInventario.Recordset.MoveLast
    End If

```

```

  If DataInventario.Recordset.RecordCount <= 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
    BotAgregar.Enabled = True
    BotGuardar.Enabled = False
    BotCancelar.Enabled = False
    BotEliminar.Enabled = False
    BotModificar.Enabled = False
    BtClave.Enabled = False
    BotMenu.Enabled = True
    BotImprimir.Enabled = False
    BotBuscar.Enabled = False
  Elseif DataInventario.Recordset.RecordCount <= 1 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
    BotAgregar.Enabled = True
    BotGuardar.Enabled = False
    BotCancelar.Enabled = False
    BtClave.Enabled = False
    BotEliminar.Enabled = True
    BotModificar.Enabled = True
    BotMenu.Enabled = True
    BotImprimir.Enabled = True
    BotBuscar.Enabled = True

```

```
Else
```

```

  DataInventario.Recordset.FindFirst " ClaveCalzado = "
  & MiReg & ""
  If DataInventario.Recordset.AbsolutePosition = 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True

```

```
Elseif DataInventario.Recordset.AbsolutePosition =
DataInventario.Recordset.RecordCount - 1 Then
```

```

  BotPrimero.Enabled = True
  BotAnterior.Enabled = True
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False

```



```

Else
  BotPrimero.Enabled = True
  BotAnterior.Enabled = True
  BotSiguiente.Enabled = True
  BotUltimo.Enabled = True
End If

```

```

BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BtClave.Enabled = False
BotEliminar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
End If

```

```
Else
```

```

BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True
BtClave.Enabled = True

```

```

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False

```

```

BotEliminar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False

```

```

End If
BloqueaControles
End Sub

```

```

Private Sub BotAgregar_Click()
  DataInventario.Recordset.AddNew
  LabelFechIngre = Date
  HabilitarBotones
  TexDescrip.BackColor = &H8000000F
  TexDescrip.Locked = True
  TexCveCalz.BackColor = &H8000000F
  TexCveCalz.Locked = True
  TexInvenID = IncrementadorID("TbVentas")

```

```

CombMarca.Enabled = True
CombSexo.Enabled = True
CombEstilo.Enabled = True
CombLinea.Enabled = True
CombMater.Enabled = True
CombColor.Enabled = True
TexModel.Enabled = True
CombMarca.SetFocus
End Sub

```

```

Private Sub BotAnterior_Click()
  DataInventario.Recordset.MovePrevious
  HabilitarBotones
  Stocks2
  Stocks
End Sub

```

```

Private Sub BotBuscar_Click()
  FoBusInv.Show vbModal
End Sub

```

```

Private Sub BotCancelar_Click()
  Dim CANCELAR As String
  CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
  If CANCELAR = vbYes Then
    DataInventario.Recordset.CancelUpdate
    HabilitarBotones
  End If
  MiCalzado = ""
End Sub

```

```

Private Sub BotEliminar_Click()
  Dim Eliminar As String
  Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
  If Eliminar = vbYes Then
    With DataInventario.Recordset
      .Delete
      .Requery
    End With
    If Not .EOF Then
      .MoveLast
    Else
      BotEliminar.Enabled = False
      .Requery
      DataInventario.Refresh
    End If
  End With
  End If
  HabilitarBotones
End Sub

```

```

Private Sub BotGuardar_Click()
  Dim CLAVE As String
  Dim añadir As String

```

```

If Trim(TexCostUni) = "" Or Trim(TexStkMin) = "" Or
Trim(TexStkMax) = "" Then
  añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
Else
  If TexCveCalz = "" Then
    CLAVE = MsgBox("Para poder guardar, debe generar la
Clave de Calzado", vbInformation)
    BtClave.SetFocus
  Else
    DataInventario.Recordset.Update
    DataInventario.Refresh
    HabilitarBotones
    MiCalzado = ""
    BotAgregar.SetFocus
  End If
End If
End Sub

```

```

Private Sub BotImprimir_Click()
  FoOpRpInv.Show
End Sub

```

```

Private Sub BotMenu_Click()
  FoMenu.Show
Unload FoInvent

```

End Sub

```
Private Sub BotModificar_Click()
DataInventario.Recordset.Edit
HabilitarBotones
MiCalzado = TexCveCalz
TexCveCalz.BackColor = &H8000000F
TexCveCalz.Locked = True
TexDescrip.BackColor = &H8000000F
TexDescrip.Locked = True
```

```
CombMarca.Enabled = False
CombSexo.Enabled = False
CombEstilo.Enabled = False
CombLinea.Enabled = False
CombMater.Enabled = False
CombColor.Enabled = False
TexModel.Enabled = False
```

```
CombMarca.BackColor = ColorGris
CombSexo.BackColor = ColorGris
CombEstilo.BackColor = ColorGris
CombLinea.BackColor = ColorGris
CombMater.BackColor = ColorGris
CombColor.BackColor = ColorGris
TexModel.BackColor = ColorGris
End Sub
```

```
Private Sub BotPrimero_Click()
DataInventario.Recordset.MoveFirst
HabilitarBotones
Stocks2
Stocks
End Sub
```

```
Private Sub BotSiguiente_Click()
If Not DataInventario.Recordset.EOF And
DataInventario.Recordset.RecordCount > 1 Then
DataInventario.Recordset.MoveNext
End If
HabilitarBotones
Stocks2
Stocks
End Sub
```

```
Private Sub BotUltimo_Click()
DataInventario.Recordset.MoveLast
HabilitarBotones
Stocks2
Stocks
End Sub
```

```
Private Sub BtClave_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub CombColor_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
```

End If
End If
End Sub

```
Private Sub CombEstilo_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub CombLinea_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub CombMarca_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub CombMater_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub CombSexo_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub Form_Activate()
HabilitarBotones
BotAgregar.SetFocus
```

```
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TxExist From TbSeg WHERE Contraseña =" & Contraseña &
""", dbOpenDynaset)
```

```
If MiRegistro!TxExist = False Then
TexExist.Enabled = False
Else
TexExist.Enabled = True
End If
Stocks2
End Sub
```

```

Private Sub Form_Load()
FoMenu.Hide
LlenarCombo CombColor, "TbColores", "Colores", "Col_ID"
LlenarCombo CombEstilo, "TbEstilos", "Estilos", "Estilos_ID"
LlenarCombo CombLinea, "TbLineas", "Lineas", "Lineas_ID"
LlenarCombo CombMarca, "TbMarcas", "Marcas",
"Marcas_ID"
LlenarCombo CombMater, "TbMaterial", "Material",
"Material_ID"
LlenarCombo CombSexo, "TbSexo", "Sexo", "Sexo_ID"
DataInventario.DatabaseName = MiBDStr
End Sub

```

```

Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub

```

```

Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub

```

```

Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataInventario.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataInventario.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H80000004
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
Next
End With
End Sub

```

```

Private Sub TexCostMay_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCostMay_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCostMay_LostFocus()
TexCostMay = PonCero(TexCostMay)
End Sub

```

```

Private Sub TexCostMay_Validate(Cancel As Boolean)
If Not IsNumeric(TexCostMay) And TexCostMay <> "" Then
MsgBox "Poner solo números", vbExclamation
TexCostMay = ""
TexCostMay.SetFocus
End If
End Sub

```

```

Private Sub TexCostMen_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCostMen_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCostMen_LostFocus()
TexCostMen = PonCero(TexCostMen)
End Sub

```

```

Private Sub TexCostMen_Validate(Cancel As Boolean)
If Not IsNumeric(TexCostMen) And TexCostMen <> "" Then
MsgBox "Poner solo números", vbExclamation
TexCostMen = ""
TexCostMen.SetFocus
End If
End Sub

```

```

Private Sub TexCostUni_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCostUni_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCostUni_Validate(Cancel As Boolean)
If Not IsNumeric(TexCostUni) And TexCostUni <> "" Then
MsgBox "Poner solo números", vbExclamation
TexCostUni = ""
TexCostUni.SetFocus
End If
End Sub

```

```

Private Sub TexExist_GotFocus()
HighLight
End Sub

```

```
Private Sub TexExist_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexExist_Validate(Cancel As Boolean)
Stocks2
End Sub
```

```
Private Sub TexModel_GotFocus()
HighLight
End Sub
```

```
Private Sub TexModel_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexStkMax_GotFocus()
HighLight
End Sub
```

```
Private Sub TexStkMax_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexStkMax_Validate(Cancel As Boolean)
If Not IsNumeric(TexStkMax) And TexStkMax <> "" Then
  MsgBox "Poner solo números", vbExclamation
  TexStkMax = ""
  TexStkMax.SetFocus
End If
Stocks2
End Sub
```

```
Private Sub TexStkMin_GotFocus()
HighLight
End Sub
```

```
Private Sub TexStkMin_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub TexStkMin_Validate(Cancel As Boolean)
If Not IsNumeric(TexStkMin) And TexStkMin <> "" Then
  MsgBox "Poner solo números", vbExclamation
```

```
TexStkMin = ""
TexStkMin.SetFocus
End If
Stocks2
End Sub
```

```
Private Sub BtClave_Click()
Dim añadir As String
```

```
If CombMarca = "" Or CombSexo = "" Or CombEstilo = "" Or
CombLinea = "" Or CombMater = "" Or CombColor = "" Or
Trim(TexModel) = "" Then
  añadir = MsgBox("Para poder generar la clave, debe llenar
los campos", vbInformation)
Else
```

```
  Dim MiBaseDatos As Object
  Dim MiRegistro As Object
```

```
  Set MiBaseDatos = OpenDatabase(MiBDStr)
  Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbMarcas WHERE Marcas_ID =" &
CombMarca.ItemData(CombMarca.ListIndex),
dbOpenDynaset)
```

```
  TexCveCalz = MiRegistro!Iniciales
  TexDescrip = CombMarca
  MiRegistro.Close
```

```
  Set MiBaseDatos = OpenDatabase(MiBDStr)
  Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Iniciales From TbSexo WHERE Sexo_ID =" &
CombSexo.ItemData(CombSexo.ListIndex), dbOpenDynaset)
  TexCveCalz = TexCveCalz + MiRegistro!Iniciales
  TexDescrip = TexDescrip & ", " & CombSexo
  MiRegistro.Close
```

```
  TexDescrip = TexDescrip & ", " & CombEstilo
```

```
  TexCveCalz = TexCveCalz & TexModel
```

```
  TexDescrip = TexDescrip & ", " & CombMater
```

```
  TexDescrip = TexDescrip & ", " & CombColor
```

```
  Set MiBaseDatos = OpenDatabase(MiBDStr)
  Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
Lineas From TbLineas WHERE Lineas_ID =" &
CombLinea.ItemData(CombLinea.ListIndex),
dbOpenDynaset)
```

```
  TexCveCalz = TexCveCalz + CStr(MiRegistro!Lineas)
  TexDescrip = TexDescrip & ", " & CombLinea
  MiRegistro.Close
```

```
  TexCveCalz = UCase(TexCveCalz)
End If
End Sub
```

```
Private Sub CombColor_Click()
If CombColor <> "" Then
  TexColoresID =
CombColor.ItemData(CombColor.ListIndex)
End If
End Sub
```

```
Private Sub CombEstilo_Click()
If CombEstilo <> "" Then
```

```

    TexEstilosID
    CombEstilo.ItemData(CombEstilo.ListIndex)
    End If
    End Sub

Private Sub CombLinea_Click()
    If CombLinea <> "" Then
        TexLineasID
        CombLinea.ItemData(CombLinea.ListIndex)
    End If
    End Sub

Private Sub CombMarca_Click()
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object
    If Trim(CombMarca) <> "" Then
        TexMarcaID
        CombMarca.ItemData(CombMarca.ListIndex)
        Set MiBaseDatos = OpenDatabase(MiBDStr)
        Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
    Iniciales From TbMarcas WHERE Marcas_ID =" &
    CombMarca.ItemData(CombMarca.ListIndex),
    dbOpenDynaset)
    End If
    End Sub

Private Sub CombMater_Click()
    If CombMater <> "" Then
        TexMaterialID
        CombMater.ItemData(CombMater.ListIndex)
    End If
    End Sub

Private Sub CombSexo_Click()
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object
    If Trim(CombSexo) <> "" Then
        TexSexoID = CombSexo.ItemData(CombSexo.ListIndex)
        Set MiBaseDatos = OpenDatabase(MiBDStr)
        Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
    Iniciales From TbSexo WHERE Sexo_ID =" &
    CombSexo.ItemData(CombSexo.ListIndex), dbOpenDynaset)
    End If
    End Sub

Private Sub TexEstilosID_Change()
    BusquedaIdCombo TexEstilosID, CombEstilo
    End Sub

Private Sub TexLineasID_Change()
    BusquedaIdCombo TexLineasID, CombLinea
    End Sub

Private Sub TexMarcaID_Change()
    BusquedaIdCombo TexMarcaID, CombMarca
    End Sub

Private Sub TexMaterialID_Change()
    BusquedaIdCombo TexMaterialID, CombMater
    End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
    FolInvent.Show
    End Sub

```

```

Private Sub TexModel_LostFocus()
    If Not IsNumeric(TexModel) And TexModel <> "" Then
        MsgBox "Poner solo números", vbExclamation
        TexModel = ""
        TexModel.SetFocus
    End If
    End Sub

```

```

Private Sub TexColoresID_Change()
    BusquedaIdCombo TexColoresID, CombColor
    End Sub

```

```

Private Sub TexSexoID_Change()
    BusquedaIdCombo TexSexoID, CombSexo
    End Sub

```

```

Public Sub Stocks()
    If TexCveCalz <> "" Then
        If CInt(PonCero(TexStkMin)) < CInt(PonCero(TexExist))
        And CInt(PonCero(TexStkMax)) > CInt(PonCero(TexExist))
        Then
            TexStkMin.ForeColor = &H0& 'Color negro
            TexExist.ForeColor = &H0& 'Color negro
            TexStkMax.ForeColor = &H0& 'Color negro
        ElseIf CInt(PonCero(TexStkMin)) <
        CInt(PonCero(TexExist)) And CInt(PonCero(TexExist)) >
        CInt(PonCero(TexStkMax)) Then
            MsgBox "La existencia actual es alta, se recomienda
            ya no surtir.", vbExclamation
            TexStkMin.ForeColor = &H0& 'Color negro
            TexExist.ForeColor = &HC0& 'Color Rojo
            TexStkMax.ForeColor = &HC0& 'Color Rojo
        ElseIf CInt(PonCero(TexStkMin)) >
        CInt(PonCero(TexExist)) < CInt(PonCero(TexStkMax)) Then
            MsgBox "La existencia actual es baja, se recomienda
            surtir.", vbExclamation
            TexStkMin.ForeColor = &HC0& 'Color Rojo
            TexExist.ForeColor = &HC0& 'Color Rojo
            TexStkMax.ForeColor = &H0& 'Color negro
        ElseIf CInt(PonCero(TexStkMin)) >
        CInt(PonCero(TexExist)) > CInt(PonCero(TexStkMax)) Then
            TexStkMin.ForeColor = &HC0& 'Color Rojo
            TexExist.ForeColor = &HC0& 'Color Rojo
            TexStkMax.ForeColor = &HC0& 'Color Rojo
        End If
    End If
    End Sub

```

```

Public Sub Stocks2()
    If TexCveCalz <> "" Then
        If CInt(PonCero(TexStkMin)) < CInt(PonCero(TexExist))
        And CInt(PonCero(TexStkMax)) > CInt(PonCero(TexExist))
        Then
            TexStkMin.ForeColor = &H0& 'Color negro
            TexExist.ForeColor = &H0& 'Color negro
            TexStkMax.ForeColor = &H0& 'Color negro
        ElseIf CInt(PonCero(TexStkMin)) <
        CInt(PonCero(TexExist)) And CInt(PonCero(TexExist)) >
        CInt(PonCero(TexStkMax)) Then
            TexStkMin.ForeColor = &H0& 'Color negro
            TexExist.ForeColor = &HC0& 'Color Rojo

```

```

    TexStkMax.ForeColor = &HC0& 'Color Rojo
  ElseIf      CInt(PonCero(TexStkMin)) >
CInt(PonCero(TexExist)) < CInt(PonCero(TexStkMax)) Then
    TexStkMin.ForeColor = &HC0& 'Color Rojo
    TexExist.ForeColor = &HC0& 'Color Rojo
    TexStkMax.ForeColor = &H0& 'Color negro
  ElseIf      CInt(PonCero(TexStkMin)) >
CInt(PonCero(TexExist)) > CInt(PonCero(TexStkMax)) Then
    TexStkMin.ForeColor = &HC0& 'Color Rojo
    TexExist.ForeColor = &HC0& 'Color Rojo
    TexStkMax.ForeColor = &HC0& 'Color Rojo
  End If
End If
End Sub

```

```

Private Sub TexUbica_GotFocus()
HighLight
End Sub

```

```

Private Sub TexUbica_KeyPress(KeyAscii As Integer)
If DataInventario.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub

```

MENU HISTORICO

```

Private Sub BotAcept_Click()
If OptionHistVent.Value = True Then
  FoVentasHist.Show
Else
  If OptionHistComp.Value = True Then
    FoComprasHist.Show
  Else
    If OptionHistFac.Value = True Then
      FoFacturasHist.Show
    Else
      If OptionHistDevol.Value = True Then
        FoDevolHist.Show
      End If
    End If
  End If
End If
Unload Me
End Sub

```

```

Private Sub BotMenu_Click()
FoMenu.Show
Unload Me
End Sub

```

PROVEEDORES

```

Option Explicit
Dim MiFolio As String
Dim MisProv As String ~

```

```

Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If DataProvee.Recordset.EditMode = 0 Then

```

```

If Not DataProvee.Recordset.EOF Then
  MiReg = TexCvePro
  DataProvee.Recordset.MoveLast
End If

```

```

If DataProvee.Recordset.RecordCount <= 0 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotAgregar.Enabled = True
  BotGuardar.Enabled = False
  BotCancelar.Enabled = False
  BotEliminar.Enabled = False
  BotImprimir.Enabled = False
  BotBuscar.Enabled = False
  BotModificar.Enabled = False
  BotMenu.Enabled = True

```

```

ElseIf DataProvee.Recordset.RecordCount <= 1 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotAgregar.Enabled = True
  BotGuardar.Enabled = False
  BotCancelar.Enabled = False
  BotEliminar.Enabled = True
  BotImprimir.Enabled = True
  BotBuscar.Enabled = True
  BotModificar.Enabled = True
  BotMenu.Enabled = True

```

Else

```

  DataProvee.Recordset.FindFirst " CveProveedores ="
  & MiReg & ""

```

```

  If DataProvee.Recordset.AbsolutePosition = 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True

```

```

  ElseIf DataProvee.Recordset.AbsolutePosition =
DataProvee.Recordset.RecordCount - 1 Then

```

```

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False

```

Else

```

  BotPrimero.Enabled = True
  BotAnterior.Enabled = True
  BotSiguiente.Enabled = True
  BotUltimo.Enabled = True

```

End If

```

  BotAgregar.Enabled = True
  BotGuardar.Enabled = False
  BotCancelar.Enabled = False
  BotEliminar.Enabled = True
  BotImprimir.Enabled = True
  BotBuscar.Enabled = True
  BotModificar.Enabled = True
  BotMenu.Enabled = True

```

End If

Else

```

  BotAgregar.Enabled = False

```

```
BotGuardar.Enabled = True
BotCancelar.Enabled = True
```

```
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
```

```
BotEliminar.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
```

```
End If
BloqueaControles
End Sub
```

```
Private Sub BotAnterior_Click()
DataProvee.Recordset.MovePrevious
HabilitarBotones
End Sub
```

```
Private Sub BotAnterior_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub
```

```
Private Sub BotBuscar_Click()
FoBuseProv.Show vbModal
End Sub
```

```
Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar los cambios?",
vbYesNo + vbQuestion, "CANCELAR")
If CANCELAR = vbYes Then
DataProvee.Recordset.CancelUpdate
HabilitarBotones
BotAgrega.SetFocus
End If
MiFolio = ""
MisProv = ""
End Sub
```

```
Private Sub BotEliminar_Click()
Dim Eliminar As String
Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
If Eliminar = vbYes Then
With DataProvee.Recordset
.Delete
.Requery
If Not .EOF Then ~
.MoveLast
Else
BotEliminar.Enabled = False
.Requery
DataProvee.Refresh
End If
End Sub
```

```
End With
End If
HabilitarBotones
End Sub
```

```
Private Sub BotGuardar_Click()
Dim añadir As String
If CombPag = Trim(TexNomPro) = "" Or Trim(TexRaSo) =
"" Or Trim(TexCalle) = "" Or CombEstados = "" Then
```

```
Else
DataProvee.Recordset.Update
DataProvee.Refresh
HabilitarBotones
MiFolio = ""
MisProv = ""
BotAgrega.SetFocus
End If
End Sub
```

```
Private Sub BotAgrega_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
DataProvee.Recordset.AddNew
LabelFecha = Date
HabilitarBotones
TexCvePro.BackColor = &H8000000F
TexCvePro.Locked = True
TexPais = "México"
TexRaSo.SetFocus
```

```
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(Folio) as Folio1 From TbProveedores", dbOpenSnapshot)
```

```
If Not IsNull(MiRegistro!Folio1) Then
TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
Else
TexFolio = "0001"
End If
```

```
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
MiRegistro.Close
```

```
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(CveProveedores) as Clave From TbProveedores",
dbOpenSnapshot)
```

```
If Not IsNull(MiRegistro!CLAVE) Then
TexCvePro = Format((MiRegistro!CLAVE) + 1, "0000")
Else
TexCvePro = "0001"
End If
End Sub
```

```
Private Sub BotImprimir_Click()
FoOpRpProv.Show
End Sub
```

```
Private Sub BotMenu_Click()
FoMenu.Show
Unload FoProv
End Sub
```

```
Private Sub BotModificar_Click()
If CombPag = "Contado" Then
  CombBanco.BackColor = ColorGris
  CombCred.BackColor = ColorGris
  CombBanco.Locked = True
  CombCred.Locked = True
```

```
Else
  CombBanco.Locked = False
  CombCred.Locked = False
  CombBanco.BackColor = vbWhite
  CombCred.BackColor = vbWhite
End If
```

```
DataProvee.Recordset.Edit
HabilitaBotones
MiFolio = TexFolio
MisProv = TexCvePro
TexRaSo.SetFocus
TexCvePro.BackColor = &H8000000F
TexCvePro.Locked = True
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
End Sub
```

```
Private Sub BotPrimero_Click()
DataProvee.Recordset.MoveFirst
HabilitaBotones
End Sub
```

```
Private Sub BotSiguiente_Click()
If Not DataProvee.Recordset.EOF And
DataProvee.Recordset.RecordCount > 1 Then
  DataProvee.Recordset.MoveNext
End If
HabilitaBotones
End Sub
```

```
Private Sub BotUltimo_Click()
DataProvee.Recordset.MoveLast
HabilitaBotones
End Sub
```

```
Private Sub CombBanco_Click()
If CombBanco <> "" Then
  TexBancoID =
  CombBanco.ItemData(CombBanco.ListIndex)
End If
End Sub
```

```
Private Sub CombBanco_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub CombCred_Click()
If CombCred <> "" Then
  TexCredID = CombCred.ItemData(CombCred.ListIndex)
End If
End Sub
```

```
Private Sub CombCred_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub CombEstados_Click()
DesplegarEstados CombEstados, TexEstadosID
End Sub
```

```
Private Sub CombEstados_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub CombPag_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub
```

```
Private Sub Form_Activate()
HabilitaBotones
BotAgregar.SetFocus
End Sub
```

```
Private Sub CombPag_Click()
If CombPag <> "" Then
  TexPagID = CombPag.ItemData(CombPag.ListIndex)
End If
If CombPag = "Contado" Then
```

```
  CombCred.Locked = True
  CombCred.Enabled = False
  CombCred.ListIndex = -1
  CombCred.BackColor = ColorGris
```

```
  TexNoCuenta.Locked = True
  TexNoCuenta.Enabled = False
  TexNoCuenta.BackColor = &H8000000F
  TexNoCuenta = ""
```

```
  CombBanco.Locked = True
  CombBanco.Enabled = False
  CombBanco.ListIndex = -1
  CombBanco.BackColor = ColorGris
```

```
Else
  CombCred.Locked = False
  CombCred.Enabled = True
  CombCred.BackColor = &H80000009
```

```
  CombBanco.Locked = False
  CombBanco.Enabled = True
  CombBanco.BackColor = &H80000009
```



```

TexNoCuenta.Locked = False
TexNoCuenta.Enabled = True

TexNoCuenta.BackColor = &H80000009
End If
End Sub

Private Sub Form_Load()
LlenarCombo CombPag, "TbFormaPag", "FormaPag",
"FormaPag_ID"
LlenarCombo CombCred, "TbFormaCred", "FormaCred",
"FormaCred_ID"
LlenarCombo CombEstados, "TbEstados", "Estados",
"Estados_ID"
LlenarCombo CombBanco, "TbBancos", "Bancos",
"Bancos_ID"
DataProvee.DatabaseName = MiBDStr
End Sub
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub

Public Sub SeleccionarCampo()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.SelStart = 0
.ActiveControl.SelLength = Len(.ActiveControl)
End If
End With
End Sub

Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
For i = 0 To .Controls.Count - 1
If TypeOf .Controls(i) Is TextBox Then
If DataProvee.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H8000000F
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
If DataProvee.Recordset.EditMode = 0 Then
.Controls(i).Locked = True
.Controls(i).BackColor = &H8000000F
Else
.Controls(i).Locked = False
.Controls(i).BackColor = vbWhite
End If
End If
Next
End With
End Sub

Private Sub TexBancoID_Change()
BusquedaldCombo TexBancoID, CombBanco
End Sub

Private Sub TexCalle_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCalle_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCiud_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCiud_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCol_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCol_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCP_GotFocus()
HighLight
End Sub

```

```

Private Sub TexCP_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexCredID_Change()
BusquedaldCombo TexCredID, CombCred
End Sub

```

```

Private Sub TexCvePro_Validate(Cancel As Boolean)
If DataProvee.Recordset.EditMode <> 0 Then

```

```

Dim MiBaseDatos As Object
Dim MiRegistro As Object

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TbProveedores.* From TbProveedores WHERE
CveProveedores =" & TexCvePro & """, dbOpenSnapshot)

```

```

If Not MiRegistro.EOF And MisProv <> TexCvePro Then

```

```

MsgBox "Número Duplicado", vbExclamation
Cancel = True
End If

```

```

End If
End Sub

```

```

Private Sub TexDescPro_GotFocus()
HighLight
End Sub

```

```

Private Sub TexDescPro_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexDescPro_LostFocus()
If Not IsNumeric(TexDescPro) And TexDescPro <> "" Then
MsgBox "Poner solo números", vbExclamation
TexDescPro = ""
TexDescPro.SetFocus
End If
End Sub

```

```

Private Sub TexEstadosID_Change()
BusquedaIdCombo TexEstadosID, CombEstados
End Sub

```

```

Private Sub TexFax_GotFocus()
HighLight
End Sub

```

```

Private Sub TexFax_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexGananc_GotFocus()
HighLight
End Sub

```

```

Private Sub TexGananc_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexGananc_LostFocus()
If Not IsNumeric(TexGananc) And TexGananc <> "" Then
MsgBox "Poner solo números", vbExclamation
TexGananc = ""
TexGananc.SetFocus
End If

```

```

End Sub

```

```

Private Sub TexMail_GotFocus()
HighLight
End Sub

```

```

Private Sub TexMail_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexNoCuenta_GotFocus()
HighLight
End Sub

```

```

Private Sub TexNoCuenta_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexNomPro_GotFocus()
HighLight
End Sub

```

```

Private Sub TexNomPro_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexPagID_Change()
BusquedaIdCombo TexPagID, CombPag
End Sub

```

```

Private Sub TexPais_GotFocus()
HighLight
End Sub

```

```

Private Sub TexPais_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexRaSo_GotFocus()
HighLight
End Sub

```

```

Private Sub TexRaSo_KeyPress(KeyAscii As Integer)

```

```

If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub

```

```

Private Sub TexRaSo_LostFocus()
TexRaSo = DameFormato(TexRaSo, TexRaSo.Tag)
End Sub

```

```

Private Sub TexRFC_GotFocus()
HighLight
End Sub

```

```

Private Sub TexRFC_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub

```

```

Private Sub TexRFC_LostFocus()
TexRFC = DameFormato(TexRFC, TexRFC.Tag)
End Sub

```

```

Private Sub Text1_GotFocus()
HighLight
End Sub

```

```

Private Sub Text1_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub

```

```

Private Sub TexTelNeg_GotFocus()
HighLight
End Sub

```

```

Private Sub TexTelNeg_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If
End Sub

```

```

Private Sub TexTelPart_GotFocus()
HighLight
End Sub

```

```

Private Sub TexTelPart_KeyPress(KeyAscii As Integer)
If DataProvee.Recordset.EditMode <> 0 Then
  If KeyAscii = 13 Then
    SendKeys "{tab}"
  End If
End If

```

```

End If
End Sub

```

SEGURIDAD

```

Option Explicit
Dim MiFolio As String
Dim MisProv As String

```

```

Public Sub BloqueaControles()
  Dim i As Integer
  With Screen.ActiveForm
    For i = 0 To .Controls.Count - 1
      If TypeOf .Controls(i) Is TextBox Then
        If Data1.Recordset.EditMode = 0 Then
          .Controls(i).Locked = True
          .Controls(i).BackColor = &H8000000F
        Else
          .Controls(i).Locked = False
          .Controls(i).BackColor = vbWhite
        End If
      End If
      If TypeOf .Controls(i) Is ComboBox Then
        If Data1.Recordset.EditMode = 0 Then
          .Controls(i).Locked = True
          .Controls(i).BackColor = &H8000000F
        Else
          .Controls(i).Locked = False
          .Controls(i).BackColor = vbWhite
        End If
      End If
      If TypeOf .Controls(i) Is CheckBox Then
        If Data1.Recordset.EditMode = 0 Then
          .Controls(i).Enabled = False
        Else
          .Controls(i).Enabled = True
        End If
      End If
    Next
  End With
End Sub

```

```

Private Sub HabilitarBotones()
Dim MiReg As String
BloqueaControles
If Data1.Recordset.EditMode = 0 Then
  If Not Data1.Recordset.EOF Then
    MiReg = TxG(4)
    Data1.Recordset.MoveLast
  End If

```

```

If Data1.Recordset.RecordCount <= 0 Then
  BotPrimero.Enabled = False
  BotAnterior.Enabled = False
  BotSiguiente.Enabled = False
  BotUltimo.Enabled = False
  BotAgregar.Enabled = True
  BotGuardar.Enabled = False
  BotCancelar.Enabled = False
  BotEliminar.Enabled = False
  BotImprimir.Enabled = False
  BotBuscar.Enabled = False
  BotModificar.Enabled = False
  BotMenu.Enabled = True

```

```

ElseIf Data1.Recordset.RecordCount <= 1 Then

```

```

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True

Else

Data1.Recordset.FindFirst " ID =" & MiReg
If Data1.Recordset.AbsolutePosition = 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True

ElseIf    Data1.Recordset.AbsolutePosition =
Data1.Recordset.RecordCount - 1 Then

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
Else
    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
End If

BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
End If

Else

BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False

BotEliminar.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False

End If

End Sub

Private Sub BotAnterior_Click()
    Data1.Recordset.MovePrevious
    HabilitarBotones
End Sub

Private Sub BotBuscar_Click()
    FoBusSeg.Show vbModal
End Sub

Private Sub BotCancelar_Click()
    Dim CANCELAR As String
    FoSeg.Show
    CANCELAR = MsgBox("¿Desea cancelar los cambios?",
vbYesNo + vbQuestion, "CANCELAR")
    If CANCELAR = vbYes Then
        Data1.Recordset.CancelUpdate
        HabilitarBotones
        BotAgregar.SetFocus
    End If
    MiFolio = ""
    MisProv = ""
End Sub

Private Sub BotEliminar_Click()
    Dim Eliminar As String
    Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
    If Eliminar = vbYes Then
        With Data1.Recordset
            .Delete
            .Requery
        End With
        If Not .EOF Then
            .MoveLast
        Else
            BotEliminar.Enabled = False
            Data1.Refresh
        End If
    End With
End Sub

HabilitarBotones
End Sub

Private Sub BotGuardar_Click()
    If TxG(1) = "" Then
        MsgBox "Para poder guardar, debe llenar el Nombre",
vbInformation
        TxG(0).SetFocus
    ElseIf TxG(1) = "" Then
        MsgBox "Para poder guardar, debe llenar el Usuario",
vbInformation
        TxG(1).SetFocus
    ElseIf TxG(2) = "" Then
        MsgBox "Para poder guardar, debe llenar la Contraseña",
vbInformation
        TxG(2).SetFocus
    ElseIf TxG(3) = "" Then
        MsgBox "Para poder guardar, debe llenar Confirmar
Contraseña", vbInformation
        TxG(3).SetFocus
    ElseIf TxG(3) <> TxG(2) Then
        MsgBox "La contraseña no es la misma al momento de
confirmarla, favor de verificar", vbInformation
        TxG(3) = ""
        TxG(2) = ""
        TxG(2).SetFocus
    End If
End Sub

```

```

Else
  If Data1.EditMode = 2 Then
    TxG(4) = IncrementadorID("TbSeg")
  End If
  Data1.Recordset.Update
  Data1.Refresh
  HabilitarBotones
  BotAgrega.SetFocus
End If
End Sub

```

```

Private Sub BotAgrega_Click()
  Data1.Recordset.AddNew
  HabilitarBotones
End Sub

```

```

Private Sub BotImprimir_Click()
  FoOpRpProv.Show
End Sub

```

```

Private Sub BotMenu_Click()
  FoMenu.Show
  Unload Me
End Sub

```

```

Private Sub BotModificar_Click()

```

```

  Data1.Recordset.Edit
  HabilitarBotones
  If TxG(1) <> "admin" Then
    Check16.Value = 1
    Check16.Enabled = True
  Else
    Check16.Value = 0
    Check16.Enabled = False
  End If

```

```

End Sub

```

```

Private Sub BotPrimero_Click()
  Data1.Recordset.MoveFirst
  HabilitarBotones
End Sub

```

```

Private Sub BotSiguiente_Click()
  If Not Data1.Recordset.EOF And
  Data1.Recordset.RecordCount > 1 Then
    Data1.Recordset.MoveNext
  End If
  HabilitarBotones
End Sub

```

```

Private Sub BotUltimo_Click()
  Data1.Recordset.MoveLast
  HabilitarBotones
End Sub

```

```

Private Sub Check1_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If

```

```

End If
End Sub

```

```

Private Sub Check10_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check11_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check12_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check13_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check14_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check15_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check16_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
      SendKeys "{tab}"
    End If
  End If
End Sub

```

```

Private Sub Check17_KeyPress(KeyAscii As Integer)
  If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then

```

```

    SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub Check2_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check3_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check4_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check5_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check6_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check7_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check8_KeyPress(KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Check9_KeyPress(KeyAscii As Integer)

```

```

If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub Form_Activate()
HabilitaBotones
End Sub

```

```

Private Sub Form_Load()
Data1.DatabaseName = MiBDStr
If FoSeg.TxG(1) <> "admin" Then
    Check16.Value = 1
    Check16.Enabled = True
Else
    Check16.Value = 0
    Check16.Enabled = False
End If
End Sub

```

```

Private Sub TxG_GotFocus(Index As Integer)
HighLight
End Sub

```

```

Private Sub TxG_KeyPress(Index As Integer, KeyAscii As Integer)
If Data1.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

VENTAS

```

Option Explicit
Dim MiBan As Boolean

```

```

Private Sub HabilitaBotones()
Dim MiReg As String
BloqueaControles
If DataVentas.Recordset.EditMode = 0 Then
    If Not DataVentas.Recordset.EOF Then
        MiReg = TexFolio
        DataVentas.Recordset.MoveLast
    End If

```

```

If DataVentas.Recordset.RecordCount <= 0 Then
    BotPrimero.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False
    BotAgregar.Enabled = True
    BotGuardar.Enabled = False
    BotCancelar.Enabled = False
    BotEliminar.Enabled = False
    BotImprimir.Enabled = False
    BotBuscar.Enabled = False
    BotModificar.Enabled = False
    BotMenu.Enabled = True
ElseIf DataVentas.Recordset.RecordCount <= 1 Then

```

```

BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True

Else

DataVentas.Recordset.FindFirst " NoVentas =" &
MiReg & ""
If DataVentas.Recordset.AbsolutePosition = 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = True
BotUltimo.Enabled = True

ElseIf DataVentas.Recordset.AbsolutePosition =
DataVentas.Recordset.RecordCount - 1 Then

BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
Else
BotPrimero.Enabled = True
BotAnterior.Enabled = True
BotSiguiente.Enabled = True
BotUltimo.Enabled = True
End If

BotAgregar.Enabled = True
BotGuardar.Enabled = False
BotCancelar.Enabled = False
BotEliminar.Enabled = True
BotImprimir.Enabled = True
BotBuscar.Enabled = True
BotModificar.Enabled = True
BotMenu.Enabled = True
End If

Else

BotAgregar.Enabled = False
BotGuardar.Enabled = True
BotCancelar.Enabled = True
BotEliminar.Enabled = False
BotImprimir.Enabled = False
BotBuscar.Enabled = False
BotModificar.Enabled = False
BotMenu.Enabled = False
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False

End If
BloqueaControles
ActualizarGrid
Cancelacion

End Sub

```

```

Private Sub BotAgregar_Click()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
BorrarTab "TbTmpDtlVentas"

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
last(NoVentas) as Folio1 From TbVentas", dbOpenSnapshot)

```

```

DataVentas.Refresh
DataVentas.Recordset.AddNew

```

```

BotEliminar.Enabled = True
BotImprimir.Enabled = False
BotBuscar.Enabled = False
BotModificar.Enabled = False

```

```

HabilitarBotones
TexDesc = 0
Cancelacion
LabelFechVta = Date
TexFolVentID = IncrementadorID("TbVentas")
TexNomEmp.BackColor = ColorGris
TexSubTotDesc.BackColor = ColorGris
TexIVA.BackColor = ColorGris
CheckFacturar.Enabled = False
TexCveEmp.SetFocus

```

```

If Not IsNull(MiRegistro!Folio1) Then
TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
Else
TexFolio = "0001"
TexFolio.BackColor = &H8000000F
TexFolio.Locked = True
End If
End Sub

```

```

Private Sub BotAnterior_Click()
DataVentas.Recordset.MovePrevious
HabilitarBotones
End Sub

```

```

Private Sub BotBuscar_Click()
FoBuseVent.Show vbModal
End Sub

```

```

Private Sub BotCancelar_Click()
Dim CANCELAR As String
CANCELAR = MsgBox("¿Desea cancelar el registro?",
vbYesNo + vbInformation, "CANCELAR")
If CANCELAR = vbYes Then
DataVentas.Recordset.CancelUpdate
PasarDatos "TbDtlVentas", "TbTmpDtlVentas",
"DtlFolFac =" & TexFolio & ""
HabilitarBotones
BotAgregar.SetFocus
End If
LstClave.Visible = False
End Sub

```

```

Private Sub BotEliminar_Click()

```

```

Dim Eliminar As String
Dim MiBaseDatos As Object

```

```

Dim MiRegistro As Object
Dim MiBaseDatos1 As Object
Dim MiRegistro1 As Object

Eliminar = MsgBox("¿Desea eliminar el registro?",
vbYesNo + vbQuestion, "ELIMINAR")
If Eliminar = vbYes Then
    CheckCancel = 1
    Label3 = "¡Cancelada por Error!"
    LabelFechVta.Enabled = False
    TexCveEmp.Enabled = False
    CombPag.Enabled = False
    GDVentas.Enabled = False
    LabelSubTotal.Enabled = False
    TexDesc.Enabled = False
    LabelTotal.Enabled = False
    BotModificar.Enabled = True
    BotEliminar.Enabled = True
    BotImprimir.Enabled = True
    BotBuscar.Enabled = True
    TexFolio.Enabled = False
Else
    Label3 = ""
    LabelFechVta.Enabled = True
    TexCveEmp.Enabled = True
    CombPag.Enabled = True
    GDVentas.Enabled = True
    LabelSubTotal.Enabled = True
    TexDesc.Enabled = True
    LabelTotal.Enabled = True
    BotModificar.Enabled = True
    BotEliminar.Enabled = True
    BotImprimir.Enabled = True
    BotBuscar.Enabled = True
    TexFolio.Enabled = True
End If
HabilitarBotones
End Sub

Private Sub BotGuardar_Click()

Dim añadir As String
Dim BuscarDatos
Dim Mult As Variant

If DataVentas.Recordset.EditMode <> 0 Then

    If Trim(TexCveEmp) = "" Or CombPag = "" Then
        añadir = MsgBox("Para poder guardar, debe llenar los
campos", vbInformation)
    Else
        BuscarDatos = TexCveEmp
        Borrartab "TbDtllVentas", "DIIIFolFac =" & TexFolio &
""
        PasarDatos "TbTmpDtllVentas", "TbDtllVentas"
        DataVentas.Recordset.Update
        DataVentas.Refresh
        DataVentas.Recordset.FindFirst " CveEmpleado =" &
BuscarDatos & ""
        HabilitarBotones
    End If
End If

LstClave.Visible = False
End Sub

Private Sub BotImprimir_Click()

```

```

FoOpRpVent.Show
End Sub

```

```

Private Sub BotMenu_Click()
    FoMenu.Show
    Unload FoVentas
End Sub

```

```

Private Sub BotModificar_Click()
    DataVentas.Recordset.Edit
    HabilitarBotones
    TexFolio.Locked = True
    TexFolio.BackColor = ColorGris
    TexNomEmp.BackColor = ColorGris
    TexSubTotDesc.BackColor = ColorGris
    TexIVA.BackColor = ColorGris
End Sub

```

```

Private Sub BotPrimero_Click()
    DataVentas.Recordset.MoveFirst
    HabilitarBotones
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataVentas.Recordset.EOF And
DataVentas.Recordset.RecordCount > 1 Then
    DataVentas.Recordset.MoveNext
End If
    HabilitarBotones
End Sub

```

```

Private Sub BotUltimo_Click()
    DataVentas.Recordset.MoveLast
    HabilitarBotones
End Sub

```

```

Private Sub CheckIVA_Click()
Dim i As Long
Dim MiSubtotal As Currency
i = 1
With GDVentas
    .Clear
    .Cols = 8
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIIClaveProveedor, DIIIMedidas, DII_ID, DIIIClaveCalzado,
DIIInoPares, DIIICostoUnit, DIIImporte, DIIICostoPublico,
DIIIDescripcion FROM TbTmpDtllVentas WHERE DIIIFolFac
=" & TexFolio & """, dbOpenSnapshot)
    If MiRegistro.EOF Then
        GDVentas.Clear
        GDVentas.Rows = 2
    End If
    Do Until MiRegistro.EOF
        .Rows = MiRegistro.RecordCount + 2
        .TextMatrix(i, 0) = MiRegistro!DII_ID
        .TextMatrix(i, 7) = MiRegistro!DIIIClaveCalzado
        .TextMatrix(i, 1) = MiRegistro!DIIIClaveProveedor
        .TextMatrix(i, 4) = MiRegistro!DIIInoPares
        .TextMatrix(i, 2) = MiRegistro!DIIIDescripcion
        .TextMatrix(i, 3) = Format(MiRegistro!DIIICostoUnit,
"currency")
    End Sub

```



```

        .TextMatrix(i, 5) = Format(MiRegistro!DIICostoPublico,
"currency")
        .TextMatrix(i, 6) = Format(MiRegistro!DIIImporte,
"currency")
        MiSubtotal = MiSubtotal + MiRegistro!DIIImporte
        i = i + 1
        MiRegistro.MoveNext
    Loop

    If CheckIVA = 1 Then
        LabelSubTotal = Format(MiSubtotal, "currency")
        TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
        TextIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")
        LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
        CheckFacturar.Enabled = False
        CheckFacturar = 1
    Else
        CheckFacturar = 0
        CheckFacturar.Enabled = False
        TextIVA = 0
        LabelSubTotal = Format(MiSubtotal, "currency")
        TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
        LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
    End If

    MiRegistro.Close
    .ColWidth(0) = 0
    .ColWidth(7) = 0
    .TextMatrix(0, 1) = "Clave Proveedor"
    .ColWidth(1) = 1200
    .TextMatrix(0, 4) = "Pares"
    .ColWidth(4) = 500
    .TextMatrix(0, 2) = "Descripción"
    .ColWidth(2) = 4000
    .TextMatrix(0, 3) = "Costo Unitario"
    .ColAlignment(3) = 7
    .ColWidth(3) = 1000
    .TextMatrix(0, 5) = "Costo Publico"
    .ColAlignment(5) = 7
    .ColWidth(5) = 1000
    .TextMatrix(0, 6) = "Importe"
    .ColAlignment(6) = 7
    .ColWidth(6) = 1000

End With
End Sub

```

```

Private Sub CheckIVA_Validate(Cancel As Boolean)

```

```

Dim i As Long

```

```

Dim MiSubtotal As Currency

```

```

i = 1

```

```

With GDVentas

```

```

    .Clear

```

```

    .Cols = 8

```

```

    Dim MiBaseDatos As Object

```

```

    Dim MiRegistro As Object

```

```

    Set MiBaseDatos = OpenDatabase(MiBDStr)

```

```

    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIClaveProveedor, DIIMedidas, DII_ID, DIIClaveCalzado,
DIINoPares, DIICostoUnit, DIIImporte, DIICostoPublico,
DIIDescripcion FROM TbTmpDtIIVentas WHERE DIIFolFac
= "" & TextFolio & """, dbOpenSnapshot)

```

```

    If MiRegistro.EOF Then

```

```

        GDVentas.Clear
        GDVentas.Rows = 2
    End If
    Do Until MiRegistro.EOF
        .Rows = MiRegistro.RecordCount + 2
        .TextMatrix(i, 0) = MiRegistro!DII_ID
        .TextMatrix(i, 7) = MiRegistro!DIIClaveCalzado
        .TextMatrix(i, 1) = MiRegistro!DIIClaveProveedor
        .TextMatrix(i, 4) = MiRegistro!DIINoPares
        .TextMatrix(i, 2) = MiRegistro!DIIDescripcion
        .TextMatrix(i, 3) = Format(MiRegistro!DIICostoUnit,
"currency")
        .TextMatrix(i, 5) = Format(MiRegistro!DIICostoPublico,
"currency")
        .TextMatrix(i, 6) = Format(MiRegistro!DIIImporte,
"currency")
        MiSubtotal = MiSubtotal + MiRegistro!DIIImporte
        i = i + 1
        MiRegistro.MoveNext
    Loop

    If CheckIVA = 1 Then
        LabelSubTotal = Format(MiSubtotal, "currency")
        TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
        TextIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")
        LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
        CheckFacturar.Enabled = False
        CheckFacturar = 1
    Else
        CheckFacturar = 0
        CheckFacturar.Enabled = False
        TextIVA = 0
        LabelSubTotal = Format(MiSubtotal, "currency")
        TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
        LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
    End If

    MiRegistro.Close
    .ColWidth(0) = 0
    .ColWidth(7) = 0
    .TextMatrix(0, 1) = "Clave Proveedor"
    .ColWidth(1) = 1200
    .TextMatrix(0, 4) = "Pares"
    .ColWidth(4) = 500
    .TextMatrix(0, 2) = "Descripción"
    .ColWidth(2) = 4000
    .TextMatrix(0, 3) = "Costo Unitario"
    .ColAlignment(3) = 7
    .ColWidth(3) = 1000
    .TextMatrix(0, 5) = "Costo Publico"
    .ColAlignment(5) = 7
    .ColWidth(5) = 1000
    .TextMatrix(0, 6) = "Importe"
    .ColAlignment(6) = 7
    .ColWidth(6) = 1000

```

```

End With

```

```

End Sub

```

```

Private Sub CombPag_Click()

```

```

    If CombPag <> "" Then

```

```

        TexPagoID = CombPag.ItemData(CombPag.ListIndex)

```

```

    End If

```

```

End Sub

```

```
Private Sub CombPag_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub CombPag_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub Form_Activate()
If MiBan = True Then
HabilitaBotones
    TexFolio.Locked = True
    BotAgregar.SetFocus
    TexNomEmp.BackColor = ColorGris
    TexFolio.BackColor = ColorGris
    MiBan = False
End If
HabilitaBotones
    TexNomEmp.BackColor = ColorGris
    TexFolio.BackColor = ColorGris
End Sub
```

```
Private Sub Form_Load()
FoMenu.Hide
DataVentas.DatabaseName = MiBDStr
LlenarCombo CombPag, "TbFormaPag", "FormaPag",
"FormaPag_ID"
LlenarCombo LstClave, "TbEmp", "CveEmpleado",
"Empleado_ID"
TexNomEmp.BackColor = ColorGris
TexFolio.BackColor = ColorGris
MiBan = True
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
With Screen.ActiveForm
    If TypeOf .ActiveControl Is TextBox Then
        .ActiveControl.SelStart = 0
        .ActiveControl.SelLength = Len(.ActiveControl)
    End If
End With
End Sub
```

```
Public Sub BloqueaControles()
Dim i As Integer
With Screen.ActiveForm
    For i = 0 To .Controls.Count - 1
        If TypeOf .Controls(i) Is TextBox Then
            If DataVentas.Recordset.EditMode = 0 Then
                .Controls(i).Locked = True
                .Controls(i).BackColor = &H80000004
            Else
                .Controls(i).Locked = False
                .Controls(i).BackColor = vbWhite
            End If
        End If
    Next
End With
```

```
End If
End If
If TypeOf .Controls(i) Is ComboBox Then
    If DataVentas.Recordset.EditMode = 0 Then
        .Controls(i).Locked = True
        .Controls(i).BackColor = &H80000004
    Else
        .Controls(i).Locked = False
        .Controls(i).BackColor = vbWhite
    End If
End If
If TypeOf .Controls(i) Is MSHFlexGrid Then
    If DataVentas.Recordset.EditMode = 0 Then
        .Controls(i).BackColor = &H80000004
    Else
        .Controls(i).BackColor = vbWhite
    End If
End If
If TypeOf .Controls(i) Is CheckBox Then
    If DataVentas.Recordset.EditMode = 0 Then
        .Controls(i).Enabled = False
    Else
        .Controls(i).Enabled = True
    End If
End If
Next
End With
End Sub
```

```
Private Sub GDVentas_DbClick()
If DataVentas.Recordset.EditMode <> 0 Then
    If GDVentas.TextMatrix(GDVentas.Row, 0) = "" Then
        FoDetalleVtas.ModoEdit = 2
    End If
    FoDetalleVtas.Show vbModal
End If
End Sub
```

```
Private Sub GDVentas_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub GDVentas_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub GDVentas_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
Dim MiBaseDatos As Object
Dim MiRegistro As Object
```

```
If KeyCode = 46 And GDVentas.TextMatrix(GDVentas.Row, 0) <> "" Then
```

```
    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("select TbTmpDtlVentas.* FROM TbTmpDtlVentas WHERE Dtl_ID =" & GDVentas.TextMatrix(GDVentas.Row, 0), dbOpenDynaset)
```

```
    If Not MiRegistro.EOF Then
```

```

MiRegistro.Delete
MiRegistro.Close

Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = '" &
GDVentas.TextMatrix(GDVentas.Row, 7) & "'",
dbOpenDynaset)
MiRegistro.Edit
MiRegistro!Existencias = MiRegistro!Existencias +
CLng(GDVentas.TextMatrix(GDVentas.Row, 3))
MiRegistro.Update
ActualizarGrid
End If
End If
End Sub

Public Sub ActualizarGrid()
Dim i As Long
Dim MiSubtotal As Currency
i = 1
If DataVentas.EditMode = 0 Then
BorrarTab "TbTmpDtlVentas"
PasarDatos "TbDtlVentas", "TbTmpDtlVentas",
"DtlFolFac="" & TexFolio & ""
End If
With GDVentas
.Clear
.Cols = 8
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DtlClaveProveedor, DtlMedidas, Dtl_ID, DtlClaveCalzado,
DtlNoPares, DtlCostoUnit, DtlImporte, DtlCostoPublico,
DtlDescripcion, DtlNoPares*DtlCostoPublico as dtlImporte1
FROM TbTmpDtlVentas WHERE DtlFolFac = "" & TexFolio
& """, dbOpenSnapshot)
If MiRegistro.EOF Then
GDVentas.Clear
GDVentas.Rows = 2
End If
Do Until MiRegistro.EOF
.Rows = MiRegistro.RecordCount + 2
.TextMatrix(i, 0) = MiRegistro!Dtl_ID
.TextMatrix(i, 7) = MiRegistro!DtlClaveCalzado
.TextMatrix(i, 1) = MiRegistro!DtlClaveProveedor
.TextMatrix(i, 4) = MiRegistro!DtlNoPares
.TextMatrix(i, 2) = MiRegistro!DtlDescripcion
.TextMatrix(i, 3) = Format(MiRegistro!DtlCostoUnit,
"currency")
.TextMatrix(i, 5) = Format(MiRegistro!DtlCostoPublico,
"currency")
.TextMatrix(i, 6) = Format(MiRegistro!DtlImporte1,
"currency")
MiSubtotal = MiSubtotal + MiRegistro!DtlImporte1
i = i + 1
MiRegistro.MoveNext
Loop

If CheckIVA = 1 Then
LabelSubTotal = Format(MiSubtotal, "currency")
TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
TexIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")

```

```

LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
CheckFacturar.Enabled = False
CheckFacturar = 1
Else
CheckFacturar = 0
CheckFacturar.Enabled = False
TexIVA = 0
LabelSubTotal = Format(MiSubtotal, "currency")
TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
End If

```

```

MiRegistro.Close
.ColWidth(0) = 0
.ColWidth(7) = 0
.TextMatrix(0, 1) = "Clave Proveedor"
.ColWidth(1) = 1200
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 2) = "Descripción"
.ColWidth(2) = 4000
.TextMatrix(0, 3) = "Costo Unitario"
.ColAlignment(3) = 7
.ColWidth(3) = 1000
.TextMatrix(0, 5) = "Costo Publico"
.ColAlignment(5) = 7
.ColWidth(5) = 1000
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000

```

```

End With
End Sub

```

```

Private Sub LstClave_Db1Click()
TexCveEmp = LstClave
LstClave.Visible = False
TexCveEmp.SetFocus
End Sub

```

```

Private Sub TexCveEmp_Change()
If DataVentas.Recordset.EditMode <> 0 Then
Dim Index As Integer
On Error Resume Next
Index = FindFirstMatch(LstClave, TexCveEmp.Text, -1)
LstClave.ListIndex = Index
If Index >= 0 Then LstClave.Selected(Index) = True
End If
End Sub

```

```

Private Sub TexCveEmp_Db1Click()
If DataVentas.Recordset.EditMode <> 0 Then
LstClave.Visible = True
End If
End Sub

```

```

Private Sub TexCveEmp_GotFocus()
LstClave.Visible = False
HighLight
If DataVentas.Recordset.EditMode <> 0 And TexCveEmp <>
"" Then

```

```

Dim MiBaseDatos As Object

```

```

Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TbEmp.* From Tbemp WHERE CveEmpleado ='" &
TexCveEmp & """, dbOpenSnapshot)

If MiRegistro.EOF Then
MsgBox "La clave proporcionada no existe, favor de
verificar", vbExclamation
TexCveEmp = ""
TexCveEmp.SetFocus
Else
TexNomEmp = MiRegistro.Nombre
End If
End If
End Sub

Private Sub TexCveEmp_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
TexCveEmp = LstClave
SendKeys "{tab}"
LstClave.Visible = False
End If
End If
End Sub

Private Sub TexDesc_Change()
Dim MiCur As Double
If MiBan = False Then
MiCur = PonCero(TexDesc)
Dim i As Long
Dim MiSubtotal As Currency
i = 1
With GDVentas
.Clear
.Cols = 8
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIIClaveProveedor, DIIIMedidas, DII_ID, DIIIClaveCalzado,
DIIINoPares, DIIICostoUnit, DIIImporte, DIIICostoPublico,
DIIIDescripcion FROM TbTmpDIIVentas WHERE DIIIFolFac
=" & TexFolio & """, dbOpenSnapshot)
If MiRegistro.EOF Then
GDVentas.Clear
GDVentas.Rows = 2
End If
Do Until MiRegistro.EOF
.Rows = MiRegistro.RecordCount + 2
.TextMatrix(i, 0) = MiRegistro!DII_ID
.TextMatrix(i, 7) = MiRegistro!DIIIClaveCalzado
.TextMatrix(i, 1) = MiRegistro!DIIIClaveProveedor
.TextMatrix(i, 4) = MiRegistro!DIIINoPares
.TextMatrix(i, 2) = MiRegistro!DIIIDescripcion
.TextMatrix(i, 3) = Format(MiRegistro!DIIICostoUnit,
"currency")
.TextMatrix(i, 5) = Format(MiRegistro!DIIICostoPublico,
"currency")
.TextMatrix(i, 6) = Format(MiRegistro!DIIImporte,
"currency")
MiSubtotal = MiSubtotal + MiRegistro!DIIImporte
i = i + 1
MiRegistro.MoveNext
Loop

```

```

If CheckIVA = 1 Then
LabelSubTotal = Format(MiSubtotal, "currency")
TexSubTotDesc = Format((MiSubtotal -
(PonCero(TexDesc)), "currency")
TexIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
CheckFacturar.Enabled = False
CheckFacturar = 1
Else
CheckFacturar = 0
CheckFacturar.Enabled = False
TexIVA = 0
LabelSubTotal = Format(MiSubtotal, "currency")
TexSubTotDesc = Format((MiSubtotal -
(PonCero(TexDesc)), "currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
End If

```

```

MiRegistro.Close
.ColWidth(0) = 0
.ColWidth(7) = 0
.TextMatrix(0, 1) = "Clave Proveedor"
.ColWidth(1) = 1200
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 2) = "Descripción"
.ColWidth(2) = 4000
.TextMatrix(0, 3) = "Costo Unitario"
.ColAlignment(3) = 7
.ColWidth(3) = 1000
.TextMatrix(0, 5) = "Costo Publico"
.ColAlignment(5) = 7
.ColWidth(5) = 1000
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000

```

End With

End If
End Sub

```

Private Sub TexDesc_GotFocus()
LstClave.Visible = False
HighLight
End Sub

```

```

Private Sub TexDesc_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub TexDesc_LostFocus()
TexDesc = PonCero(TexDesc)
End Sub

```

```

Private Sub TexDesc_Validate(Cancel As Boolean)
If Not IsNumeric(TexDesc) And TexDesc <> "" Then
MsgBox "Poner solo números", vbExclamation

```

```

TexDesc = ""
TexDesc.SetFocus
End If
End Sub

```

```

Private Sub TexPagoID_Change()
BusquedaIdCombo TexPagoID, CombPag
End Sub

```

```

Public Sub Cancelacion()

```

```

    If CheckCancel = 1 Then
Label3 = "¡Cancelada por Error!"
LabelFechVta.Enabled = False
TexCveEmp.Enabled = False
CombPag.Enabled = False
GDVentas.Enabled = False
LabelSubTotal.Enabled = False
TexDesc.Enabled = False
LabelTotal.Enabled = False
BotModificar.Enabled = False
BotEliminar.Enabled = False
BotImprimir.Enabled = True
BotBuscar.Enabled = True
TexFolio.Enabled = False
Else
Label3 = ""
LabelFechVta.Enabled = True
TexCveEmp.Enabled = True
CombPag.Enabled = True
GDVentas.Enabled = True
LabelSubTotal.Enabled = True
TexDesc.Enabled = True
LabelTotal.Enabled = True
TexFolio.Enabled = True
End If

```

```

End Sub

```

```

Public Sub PasarDatos(Tb1 As String, Tb2 As String, Optional
WH As String)
Dim MiBD As Object
Dim MiTab1 As Object, MiTab2 As Object, MiTab3 As
Object

```

```

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTab1 = MiBD.OpenRecordset("SELECT " & Tb1 &
".* FROM " & Tb1 & If(WH = "", "", " WHERE " & WH),
dbOpenSnapshot)
    Set MiTab2 = MiBD.OpenRecordset("SELECT " & Tb2 &
".* FROM " & Tb2, dbOpenDynaset)

```

```

Do Until MiTab1.EOF
MiTab2.AddNew
MiTab2!DIINoPares = MiTab1!DIINoPares
MiTab2!DIICostoUnit = MiTab1!DIICostoUnit
MiTab2!DIICostoPublico = MiTab1!DIICostoPublico
MiTab2!DIID_ID = MiTab1!DIID_ID
MiTab2!DIIFolFac = MiTab1!DIIFolFac
MiTab2!DIIMpporte = MiTab1!DIIMpporte
MiTab2!DIIDescripcion = MiTab1!DIIDescripcion
MiTab2!DIIMedidas = MiTab1!DIIMedidas
MiTab2!DIIClaveCalzado = MiTab1!DIIClaveCalzado =
MiTab2!DIIClaveProveedor
MiTab1!DIIClaveProveedor
MiTab2.Update
If Tb1 = "TbTmpDtllVentas" Then
    If MiTab1!ModoEdit = 1 Then

```

```

        Set MiTab3 = MiBD.OpenRecordset("SELECT
Existencias FROM TbInventario WHERE ClaveCalzado = "
& MiTab1!DIIClaveCalzado & "", dbOpenDynaset)
        MiTab3.Edit
        MiTab3!Existencias = MiTab3!Existencias -
MiTab1!DIINoPares
        MiTab3.Update
    End If
End If

MiTab1.MoveNext
Loop
End Sub

```

VENTAS HISTORICOS

```

Option Explicit
Dim MiBan As Boolean

```

```

Private Sub HabilitarBotones()

```

```

Dim MiReg As String
BloqueaControles
If DataVentas.Recordset.EditMode = 0 Then
If Not DataVentas.Recordset.EOF Then
    MiReg = TexFolio
    DataVentas.Recordset.MoveLast
End If

```

```

If DataVentas.Recordset.RecordCount <= 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotGuardar.Enabled = True
BotMenu.Enabled = True

```

```

ElseIf DataVentas.Recordset.RecordCount <= 1 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = False
BotUltimo.Enabled = False
BotGuardar.Enabled = True
BotMenu.Enabled = True

```

```

Else

```

```

    DataVentas.Recordset.FindFirst " NoVentas =" &
MiReg & ""

```

```

If DataVentas.Recordset.AbsolutePosition = 0 Then
BotPrimero.Enabled = False
BotAnterior.Enabled = False
BotSiguiente.Enabled = True
BotUltimo.Enabled = True

```

```

ElseIf DataVentas.Recordset.AbsolutePosition =
DataVentas.Recordset.RecordCount - 1 Then

```

```

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False

```

```

Else

```

```

    BotPrimero.Enabled = True
    BotAnterior.Enabled = True
    BotSiguiente.Enabled = True
    BotUltimo.Enabled = True
End If

```

```

BotGuardar.Enabled = True

```

```

        BotMenu.Enabled = True
    End If

Else

    BotGuardar.Enabled = True
    BotMenu.Enabled = False
    BotPrimerio.Enabled = False
    BotAnterior.Enabled = False
    BotSiguiente.Enabled = False
    BotUltimo.Enabled = False

End If
BloqueaControles
ActualizarGrid
Cancelar
Text1.Locked = False
Text1.BackColor = vbWhite
Text3.Locked = False
Text3.BackColor = vbWhite
End Sub

Private Sub BotAgregar_Click()
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
    last(NoVentas) as Folio1 From TbVentas", dbOpenSnapshot)

    DataVentas.Recordset.AddNew

    HabilitarBotones
    TexDesc = 0
    Cancelacion
    LabelFechVta = Date
    TexFolVentID = IncrementadorID("TbVentas")
    TexNomEmp.BackColor = ColorGris
    TexSubTotDesc.BackColor = ColorGris
    TexIVA.BackColor = ColorGris
    CheckFacturar.Enabled = False
    TexCveEmp.SetFocus

    If Not IsNull(MiRegistro!Folio1) Then
        TexFolio = Format((MiRegistro!Folio1) + 1, "0000")
        TexFolio.BackColor = &H8000000F
        TexFolio.Locked = True
    Else
        TexFolio = "0001"
        TexFolio.BackColor = &H8000000F
        TexFolio.Locked = True
    End If
End Sub

Private Sub BotAnterior_Click()
    DataVentas.Recordset.MovePrevious
    HabilitarBotones
    Cancelacion
End Sub

Private Sub BotBuscar_Click()
    FoBuscVent.Show vbModal
End Sub

Private Sub BotCancelar_Click()
    Dim CANCELAR As String

```

```

    CANCELAR = MsgBox("¿Desea cancelar el registro?",
    vbYesNo + vbInformation, "CANCELAR")
    If CANCELAR = vbYes Then
        DataVentas.Recordset.CancelUpdate
        HabilitarBotones
    End If
    LstClave.Visible = False
End Sub

```

```

Private Sub BotEliminar_Click()
    Dim Eliminar As String
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object
    Dim MiBaseDatos1 As Object
    Dim MiRegistro1 As Object

```

```

    Eliminar = MsgBox("¿Desea eliminar el registro?",
    vbYesNo + vbQuestion, "ELIMINAR")
    If Eliminar = vbYes Then
        CheckCancel = 1
        Label3 = "¡Venta Cancelada!"
        LabelFechVta.Enabled = False
        TexCveEmp.Enabled = False
        CombPag.Enabled = False
        GDVentas.Enabled = False
        LabelSubTotal.Enabled = False
        TexDesc.Enabled = False
        LabelTotal.Enabled = False
        TexFolio.Enabled = False
    Else
        Label3 = ""
        LabelFechVta.Enabled = True
        TexCveEmp.Enabled = True
        CombPag.Enabled = True
        GDVentas.Enabled = True
        LabelSubTotal.Enabled = True
        TexDesc.Enabled = True
        LabelTotal.Enabled = True
        TexFolio.Enabled = True
    End If
    HabilitarBotones
End Sub

Private Sub BotGuardar_Click()
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object
    Dim Confirm As String
    Dim MiRegistro2 As Object

    Confirm = MsgBox("Se dispone a pasar toda la información
    contenida en la forma de Ventas hacia Historico Ventas y de
    Devoluciones hacia Historicos Devoluciones, ya que se
    encuentran relacionadas, una vez guardado no se podrá
    regresar la información. ¿Desea continuar?", vbYesNo +
    vbCritical, "ADVERTENCIA")
    If Confirm = vbYes Then

        Set MiBaseDatos = OpenDatabase(MiBDStr)
        Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
        CondHist, NoVentas FROM TbVentas WHERE
        (((FechaVenta) Between #" & Format(Text1, "mm/dd/yy") &
        "# And #" & Format(Text3, "mm/dd/yy") & "#)",
        dbOpenDynaset)

        Do Until MiRegistro.EOF
            Set MiRegistro2 =
            MiBaseDatos.OpenRecordset("SELECT CondHist FROM

```

```

TbDevolucion WHERE FolioVta1 = "" &
MiRegistro!NoVentas & "" or FolioVta3 = "" &
MiRegistro!NoVentas & "", dbOpenDynaset)
If Not MiRegistro2.EOF Then
    MiRegistro2.Edit
    MiRegistro2!CondHist = 1
    MiRegistro2.Update
    MiRegistro2.MoveNext
End If
MiRegistro2.Close
MiRegistro.Edit
MiRegistro!CondHist = 1
MiRegistro.Update
MiRegistro.MoveNext
Loop

```

```

End If
End Sub

```

```

Private Sub BotImprimir_Click()
FoOpRpVent.Show
End Sub

```

```

Private Sub BotMenu_Click()
FoMenuHist.Show
Unload FoVentasHist
End Sub

```

```

Private Sub BotModificar_Click()
DataVentas.Recordset.Edit
HabilitaBotones
TexFolio.Locked = True
TexFolio.BackColor = ColorGris
TexNomEmp.BackColor = ColorGris
TexSubTotDesc.BackColor = ColorGris
TexIVA.BackColor = ColorGris
End Sub

```

```

Private Sub BotPrimero_Click()
DataVentas.Recordset.MoveFirst
HabilitaBotones
Cancelacion
End Sub

```

```

Private Sub BotSiguiente_Click()
If Not DataVentas.Recordset.EOF And
DataVentas.Recordset.RecordCount > 1 Then
    DataVentas.Recordset.MoveNext
End If
HabilitaBotones
Cancelacion
End Sub

```

```

Private Sub BotUltimo_Click()
DataVentas.Recordset.MoveLast
HabilitaBotones
Cancelacion
End Sub

```

```

Private Sub BtFiltrar_Click()
If IsDate(Text1) And IsDate(Text3) Then
    DataVentas.RecordSource = "SELECT TbVentas.* FROM
TbVentas WHERE (((FechaVenta) Between #" &

```

```

Format(Text1, "mm/dd/yy") & "# And #" & Format(Text3,
"mm/dd/yy") & "#))"
DataVentas.Refresh
HabilitaBotones
ElseIf Not IsDate(Text1) Then
    MsgBox "Fecha incorrecta, favor de verificar",
vbInformation
Text1 = ""
ElseIf Not IsDate(Text3) Then
    MsgBox "Fecha incorrecta, favor de verificar",
vbInformation
Text3 = ""
End If
End Sub

```

```

Private Sub CheckIVA_Click()

```

```

Dim i As Long
Dim MiSubtotal As Currency
i = 1
With GDVentas
    .Clear
    .Cols = 8
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

```

```

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIClaveProveedor, DIIMedidas, DII_ID, DIIClaveCalzado,
DIINoPares, DIICostoUnit, DIImporte, DIICostoPublico,
DIIDescripcion FROM TbDIIVentas WHERE DIIFolFac = ""
& TexFolio & "", dbOpenSnapshot)

```

```

If MiRegistro.EOF Then
    GDVentas.Clear
    GDVentas.Rows = 2
End If
Do Until MiRegistro.EOF
    .Rows = MiRegistro.RecordCount + 2
    .TextMatrix(i, 0) = MiRegistro!DII_ID
    .TextMatrix(i, 7) = MiRegistro!DIIClaveCalzado
    .TextMatrix(i, 1) = MiRegistro!DIIClaveProveedor
    .TextMatrix(i, 4) = MiRegistro!DIINoPares
    .TextMatrix(i, 2) = MiRegistro!DIIDescripcion
    .TextMatrix(i, 3) = Format(MiRegistro!DIICostoUnit,
"currency")
    .TextMatrix(i, 5) = Format(MiRegistro!DIICostoPublico,
"currency")
    .TextMatrix(i, 6) = Format(MiRegistro!DIImporte,
"currency")
    MiSubtotal = MiSubtotal + MiRegistro!DIImporte
    i = i + 1
    MiRegistro.MoveNext
Loop

```

```

If CheckIVA = 1 Then
    LabelSubTotal = Format(MiSubtotal, "currency")
    TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
    TexIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")
    LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
    CheckFacturar.Enabled = False
    CheckFacturar = 1
Else
    CheckFacturar = 0
    CheckFacturar.Enabled = False
    TexIVA = 0
    LabelSubTotal = Format(MiSubtotal, "currency")

```

```

TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
End If

MiRegistro.Close
.ColWidth(0) = 0
.ColWidth(7) = 0
.TextMatrix(0, 1) = "Clave Proveedor"
.ColWidth(1) = 1200
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 2) = "Descripción"
.ColWidth(2) = 4000
.TextMatrix(0, 3) = "Costo Unitario"
.ColAlignment(3) = 7
.ColWidth(3) = 1000
.TextMatrix(0, 5) = "Costo Publico"
.ColAlignment(5) = 7
.ColWidth(5) = 1000
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000

End With
End Sub

```

```

Private Sub CheckIVA_Validate(Cancel As Boolean)
Dim i As Long
Dim MiSubtotal As Currency
i = 1
With GDVentas
.Clear
.Cols = 8
Dim MiBaseDatos As Object
Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIClaveProveedor, DIIMedidas, DIID_ID, DIIClaveCalzado,
DIINoPares, DIICostoUnit, DIIMpporte, DIICostoPublico,
DIIDescripcion FROM TbDtlVentas WHERE DIIFolFac = ""
& TexFolio & """, dbOpenSnapshot)
If MiRegistro.EOF Then
GDVentas.Clear
GDVentas.Rows = 2
End If
Do Until MiRegistro.EOF
.Rows = MiRegistro.RecordCount + 2
.TextMatrix(i, 0) = MiRegistro!DIID_ID
.TextMatrix(i, 7) = MiRegistro!DIIClaveCalzado
.TextMatrix(i, 1) = MiRegistro!DIIClaveProveedor
.TextMatrix(i, 4) = MiRegistro!DIINoPares
.TextMatrix(i, 2) = MiRegistro!DIIDescripcion
.TextMatrix(i, 3) = Format(MiRegistro!DIICostoUnit,
"currency")
.TextMatrix(i, 5) = Format(MiRegistro!DIICostoPublico,
"currency")
.TextMatrix(i, 6) = Format(MiRegistro!DIIMpporte,
"currency")
MiSubtotal = MiSubtotal + MiRegistro!DIIMpporte
i = i + 1
MiRegistro.MoveNext
Loop

If CheckIVA = 1 Then
LabelSubTotal = Format(MiSubtotal, "currency")

```

```

TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
TexIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
CheckFacturar.Enabled = False
CheckFacturar = 1
Else
CheckFacturar = 0
CheckFacturar.Enabled = False
TexIVA = 0
LabelSubTotal = Format(MiSubtotal, "currency")
TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
End If

```

```

MiRegistro.Close
.ColWidth(0) = 0
.ColWidth(7) = 0
.TextMatrix(0, 1) = "Clave Proveedor"
.ColWidth(1) = 1200
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 2) = "Descripción"
.ColWidth(2) = 4000
.TextMatrix(0, 3) = "Costo Unitario"
.ColAlignment(3) = 7
.ColWidth(3) = 1000
.TextMatrix(0, 5) = "Costo Publico"
.ColAlignment(5) = 7
.ColWidth(5) = 1000
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000

```

```

End With
End Sub

```

```

Private Sub CombPag_Click()
If CombPag <> "" Then
TexPagoID = CombPag.ItemData(CombPag.ListIndex)
End If
End Sub

```

```

Private Sub CombPag_GotFocus()
LstClave.Visible = False
End Sub

```

```

Private Sub CombPag_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
If KeyAscii = 13 Then
SendKeys "{tab}"
End If
End If
End Sub

```

```

Private Sub Form_Activate()
If MiBan = True Then
HabilitaBotones

```

```

Text1 = "01" & Format(Date, "-mm-yyyy")
Text3 = DiaUltimoDelMes(Date) & Format(Date, "-mm-
yyyy")

```



```
Text1.Locked = False
Text1.BackColor = vbWhite
Text3.Locked = False
Text3.BackColor = vbWhite
```

```
    TexFolio.Locked = True
    TexNomEmp.BackColor = ColorGris
    TexFolio.BackColor = ColorGris
    MiBan = False
End If
    HabilitarBotones
    TexNomEmp.BackColor = ColorGris
    TexFolio.BackColor = ColorGris
End Sub
```

```
Private Sub Form_Load()
FoMenu.Hide
DataVentas.DatabaseName = MiBDStr
LlenarCombo CombPag, "TbFormaPag", "FormaPag",
"FormaPag_ID"
LlenarCombo LstClave, "TbEmp", "CveEmpleado",
"Empleado_ID"
TexNomEmp.BackColor = ColorGris
TexFolio.BackColor = ColorGris
MiBan = True
End Sub
```

```
Private Sub TexGotFocus(Index As Integer)
SeleccionarCampo
End Sub
```

```
Public Sub SeleccionarCampo()
    With Screen.ActiveForm
        If TypeOf .ActiveControl Is TextBox Then
            .ActiveControl.SelStart = 0
            .ActiveControl.SelLength = Len(.ActiveControl)
        End If
    End With
End Sub
```

```
Public Sub BloqueaControles()
    Dim i As Integer
    With Screen.ActiveForm
        For i = 0 To .Controls.Count - 1
            If TypeOf .Controls(i) Is TextBox Then
                If DataVentas.Recordset.EditMode = 0 Then
                    .Controls(i).Locked = True
                    .Controls(i).BackColor = &H80000004
                Else
                    .Controls(i).Locked = False
                    .Controls(i).BackColor = vbWhite
                End If
            End If
            If TypeOf .Controls(i) Is ComboBox Then
                If DataVentas.Recordset.EditMode = 0 Then
                    .Controls(i).Locked = True
                    .Controls(i).BackColor = &H80000004
                Else
                    .Controls(i).Locked = False
                    .Controls(i).BackColor = vbWhite
                End If
            End If
            If TypeOf .Controls(i) Is MSHFlexGrid Then
                If DataVentas.Recordset.EditMode = 0 Then
                    .Controls(i).BackColor = &H80000004
                Else
```

```
                .Controls(i).BackColor = vbWhite
            End If
        End If
        If TypeOf .Controls(i) Is CheckBox Then
            If DataVentas.Recordset.EditMode = 0 Then
                .Controls(i).Enabled = False
            Else
                .Controls(i).Enabled = True
            End If
        End If
    Next
End With
End Sub
```

```
Private Sub GDVentas_DbClick()
If DataVentas.Recordset.EditMode <> 0 Then
    If GDVentas.TextMatrix(GDVentas.Row, 0) = "" Then
        FoDetalleVtas.ModoEdit = 2
    End If
    FoDetalleVtas.Show vbModal
End If
End Sub
```

```
Private Sub GDVentas_GotFocus()
LstClave.Visible = False
End Sub
```

```
Private Sub GDVentas_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub
```

```
Private Sub GDVentas_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object
```

```
If KeyCode = 46 And GDVentas.TextMatrix(GDVentas.Row, 0) <> "" Then
```

```
    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("select
TbDtIlVentas.* FROM TbDtIlVentas WHERE DII_ID = " &
GDVentas.TextMatrix(GDVentas.Row, 0), dbOpenDynaset)
```

```
If Not MiRegistro.EOF Then
```

```
    MiRegistro.Delete
    MiRegistro.Close
```

```
    Set MiRegistro =
MiBaseDatos.OpenRecordset("SELECT Existencias FROM
TbInventario WHERE ClaveCalzado = " &
GDVentas.TextMatrix(GDVentas.Row, 7) & "",
dbOpenDynaset)
    MiRegistro.Edit
    MiRegistro!Existencias = MiRegistro!Existencias +
CLng(GDVentas.TextMatrix(GDVentas.Row, 3))
    MiRegistro.Update
    ActualizarGrid
End If
End If
```

End Sub

Public Sub ActualizarGrid()

Dim i As Long

Dim MiSubtotal As Currency

i = 1

With GDVentas

.Clear

.Cols = 8

Dim MiBaseDatos As Object

Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)

Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIClaveProveedor, DIIMedidas, DII_ID, DIIClaveCalzado,
DIINoPares, DIICostoUnit, DIIMpporte, DIICostoPublico,
DIIDescripcion FROM TbDIIVentas WHERE DIIFolFac = ""
& TexFolio & """, dbOpenSnapshot)

If MiRegistro.EOF Then

GDVentas.Clear

GDVentas.Rows = 2

End If

Do Until MiRegistro.EOF

.Rows = MiRegistro.RecordCount + 2

.TextMatrix(i, 0) = MiRegistro!DII_ID

.TextMatrix(i, 7) = MiRegistro!DIIClaveCalzado

.TextMatrix(i, 1) = MiRegistro!DIIClaveProveedor

.TextMatrix(i, 4) = MiRegistro!DIINoPares

.TextMatrix(i, 2) = MiRegistro!DIIDescripcion

.TextMatrix(i, 3) = Format(MiRegistro!DIICostoUnit,

"currency")

.TextMatrix(i, 5) = Format(MiRegistro!DIICostoPublico,

"currency")

.TextMatrix(i, 6) = Format(MiRegistro!DIIMpporte,

"currency")

MiSubtotal = MiSubtotal + MiRegistro!DIIMpporte

i = i + 1

MiRegistro.MoveNext

Loop

If CheckIVA = 1 Then

LabelSubTotal = Format(MiSubtotal, "currency")

TexSubTotDesc = Format((MiSubtotal) -

(PonCero(TexDesc)), "currency")

TexIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),

"currency")

LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +

CCur(PonCero(TexIVA)), "currency")

CheckFacturar.Enabled = False

CheckFacturar = 1

Else

CheckFacturar = 0

CheckFacturar.Enabled = False

TexIVA = 0

LabelSubTotal = Format(MiSubtotal, "currency")

TexSubTotDesc = Format((MiSubtotal) -

(PonCero(TexDesc)), "currency")

LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +

CCur(PonCero(TexIVA)), "currency")

End If

MiRegistro.Close

.ColWidth(0) = 0

.ColWidth(7) = 0

.TextMatrix(0, 1) = "Clave Proveedor"

.ColWidth(1) = 1200

.TextMatrix(0, 4) = "Pares"

.ColWidth(4) = 500

.TextMatrix(0, 2) = "Descripción"

.ColWidth(2) = 4000

.TextMatrix(0, 3) = "Costo Unitario"

.ColAlignment(3) = 7

.ColWidth(3) = 1000

.TextMatrix(0, 5) = "Costo Publico"

.ColAlignment(5) = 7

.ColWidth(5) = 1000

.TextMatrix(0, 6) = "Importe"

.ColAlignment(6) = 7

.ColWidth(6) = 1000

End With

End Sub

Private Sub LstClave_DblClick()

TexCveEmp = LstClave

LstClave.Visible = False

TexCveEmp.SetFocus

End Sub

Private Sub TexCveEmp_Change()

If DataVentas.Recordset.EditMode <> 0 Then

Dim Index As Integer

On Error Resume Next

Index = FindFirstMatch(LstClave, TexCveEmp.Text, -1)

LstClave.ListIndex = Index

If Index >= 0 Then LstClave.Selected(Index) = True

End If

End Sub

Private Sub TexCveEmp_DblClick()

If DataVentas.Recordset.EditMode <> 0 Then

LstClave.Visible = True

End If

End Sub

Private Sub TexCveEmp_GotFocus()

LstClave.Visible = False

HighLight

If DataVentas.Recordset.EditMode <> 0 And TexCveEmp <>

"" Then

Dim MiBaseDatos As Object

Dim MiRegistro As Object

Set MiBaseDatos = OpenDatabase(MiBDStr)

Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TbEmp.* From Ttemp WHERE CveEmpleado = "" &
TexCveEmp & """, dbOpenSnapshot)

If MiRegistro.EOF Then

MsgBox "La clave proporcionada no existe, favor de
verificar", vbExclamation

TexCveEmp = ""

TexCveEmp.SetFocus

Else

TexNomEmp = MiRegistro.Nombre

End If

End If

End Sub

Private Sub TexCveEmp_KeyPress(KeyAscii As Integer)

If DataVentas.Recordset.EditMode <> 0 Then

If KeyAscii = 13 Then

TexCveEmp = LstClave

```

    SendKeys "{tab}"
    LstClave.Visible = False
End If
End If
End Sub

```

```

Private Sub TexDesc_Change()
Dim MiCur As Double
If MiBan = False Then
    MiCur = PonCero(TexDesc)
    Dim i As Long
Dim MiSubtotal As Currency
i = 1
With GDVentas
    .Clear
    .Cols = 8
    Dim MiBaseDatos As Object
    Dim MiRegistro As Object

    Set MiBaseDatos = OpenDatabase(MiBDStr)
    Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
DIIClaveProveedor, DIIMedidas, DII_ID, DIIClaveCalzado,
DIINoPares, DIICostoUnit, DIIMporte, DIICostoPublico,
DIIDescripcion FROM TbDIIVentas WHERE DIIFolFac = ""
& TexFolio & """, dbOpenSnapshot)
    If MiRegistro.EOF Then
        GDVentas.Clear
        GDVentas.Rows = 2
    End If
    Do Until MiRegistro.EOF
    .Rows = MiRegistro.RecordCount + 2
    .TextMatrix(i, 0) = MiRegistro!DII_ID
    .TextMatrix(i, 7) = MiRegistro!DIIClaveCalzado
    .TextMatrix(i, 1) = MiRegistro!DIIClaveProveedor
    .TextMatrix(i, 4) = MiRegistro!DIINoPares
    .TextMatrix(i, 2) = MiRegistro!DIIDescripcion
    .TextMatrix(i, 3) = Format(MiRegistro!DIICostoUnit,
"currency")
    .TextMatrix(i, 5) = Format(MiRegistro!DIICostoPublico,
"currency")
    .TextMatrix(i, 6) = Format(MiRegistro!DIIMporte,
"currency")
    MiSubtotal = MiSubtotal + MiRegistro!DIIMporte
    i = i + 1
    MiRegistro.MoveNext
    Loop

    If CheckIVA = 1 Then
    LabelSubTotal = Format(MiSubtotal, "currency")
    TexSubTotDesc = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
    TexIVA = Format(PonCero(TexSubTotDesc) * (IVA / 100),
"currency")
    LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
    CheckFacturar.Enabled = False
    CheckFacturar = 1
    Else
    CheckFacturar = 0
    CheckFacturar.Enabled = False
    TexIVA = 0
    LabelSubTotal = Format(MiSubtotal, "currency")
    TexSubTotDesc ~ = Format((MiSubtotal) -
(PonCero(TexDesc)), "currency")
    LabelTotal = Format(CCur(PonCero(TexSubTotDesc)) +
CCur(PonCero(TexIVA)), "currency")
    End If

    MiRegistro.Close

```

```

.ColWidth(0) = 0
.ColWidth(7) = 0
.TextMatrix(0, 1) = "Clave Proveedor"
.ColWidth(1) = 1200
.TextMatrix(0, 4) = "Pares"
.ColWidth(4) = 500
.TextMatrix(0, 2) = "Descripción"
.ColWidth(2) = 4000
.TextMatrix(0, 3) = "Costo Unitario"
.ColAlignment(3) = 7
.ColWidth(3) = 1000
.TextMatrix(0, 5) = "Costo Publico"
.ColAlignment(5) = 7
.ColWidth(5) = 1000
.TextMatrix(0, 6) = "Importe"
.ColAlignment(6) = 7
.ColWidth(6) = 1000

```

End With

End If
End Sub

```

Private Sub TexDesc_GotFocus()
LstClave.Visible = False
HighLight
End Sub

```

```

Private Sub TexDesc_KeyPress(KeyAscii As Integer)
If DataVentas.Recordset.EditMode <> 0 Then
    If KeyAscii = 13 Then
        SendKeys "{tab}"
    End If
End If
End Sub

```

```

Private Sub TexDesc_LostFocus()
TexDesc = PonCero(TexDesc)
End Sub

```

```

Private Sub TexDesc_Validate(Cancel As Boolean)
If Not IsNumeric(TexDesc) And TexDesc <> "" Then
    MsgBox "Poner solo números", vbExclamation
    TexDesc = ""
    TexDesc.SetFocus
End If
End Sub

```

```

Private Sub TexPagoID_Change()
BusquedaldCombo TexPagoID, CombPag
End Sub

```

```

Public Sub Cancelacion()
If CheckCancel = 1 Then
    Label3 = "¡Venta Cancelada!"
    LabelFechVta.Enabled = False
    TexCveEmp.Enabled = False
    CombPag.Enabled = False
    GDVentas.Enabled = False
    LabelSubTotal.Enabled = False
    TexDesc.Enabled = False
    LabelTotal.Enabled = False
    TexFolio.Enabled = False
Else

```

```

Label3 = ""
LabelFechVta.Enabled = True
TexCveEmp.Enabled = True
CombPag.Enabled = True
GDVentas.Enabled = True
LabelSubTotal.Enabled = True
TexDesc.Enabled = True
LabelTotal.Enabled = True
TexFolio.Enabled = True
End If
End Sub

```

```

Global MiBDStr As String
Global Descuento As Currency
Global Ganancia As Currency
Global IVA As Currency
Global Contraseña As String
Global MiID As Long

```

```
Sub Main()
```

```

Dim MiBD As Object
Dim MiReg As Object
Dim MiTab As Object

```

```
MiBDStr = App.Path & "\BDZapateria.mdb"
```

```

Set MiBD = OpenDatabase(MiBDStr)
Set MiTab = MiBD.OpenRecordset("SELECT TbValDefault.*
FROM TbValDefault", dbOpenSnapshot)
Ganancia = MiTab!Valor
MiTab.MoveNext
Descuento = MiTab!Valor
MiTab.MoveNext
IVA = MiTab!Valor
MiTab.Close

```

```
FoContra.Show
```

```
End Sub
```

```

Public Sub LlenarCombo(MiComboBox As Object, MiTb As
String, MiCampo As String, Optional MiID As String)
Dim MiBD As Object
Dim MiTab As Object

```

```
Set MiBD = OpenDatabase(MiBDStr)
```

```
If Trim(MiID) <> "" Then
```

```

Set MiTab = MiBD.OpenRecordset("SELECT " & MiTb & ".*
FROM " & MiTb, dbOpenSnapshot)
Do Until MiTab.EOF
MiComboBox.AddItem IIf(IsNull(MiTab(MiCampo)), "",
MiTab(MiCampo))
MiComboBox.ItemData(MiComboBox.NewIndex) =
MiTab(MiID)
MiTab.MoveNext
Loop
Else

```

```

Set MiTab = MiBD.OpenRecordset("SELECT " & MiTb & ".*
FROM " & MiTb, dbOpenSnapshot)
Do Until MiTab.EOF
MiComboBox.AddItem IIf(IsNull(MiTab(MiCampo)), "",
MiTab(MiCampo))

```

```

MiTab.MoveNext
Loop
End If

```

```
End Sub
```

```
Public Sub BusquedaIdCombo(Textos As Variant, Combos As
ComboBox)
```

```
Dim i As Long
```

```
i = 0
```

```
If Textos <> "0" And Textos <> "" Then
```

```
Do Until Textos = Combos.ItemData(i)
```

```
i = i + 1
```

```
Loop
```

```
Combos.ListIndex = i
```

```
Else
```

```
Combos.ListIndex = -1
```

```
End If
```

```
End Sub
```

```
Function IncrementadorID(ValorID As String) As Long
```

```
Dim MiBD As Object
```

```
Dim MiTab As Object
```

```
Set MiBD = OpenDatabase(MiBDStr)
```

```
Set MiTab = MiBD.OpenRecordset("SELECT TbIDs.* FROM
TbIDs WHERE TbTablas = " & ValorID & "",
dbOpenDynaset)
```

```
IncrementadorID = MiTab!IDs + 1
```

```
MiTab.Edit
```

```
MiTab!IDs = IncrementadorID
```

```
MiTab.Update
```

```
End Function
```

```
Public Sub DesplegarEstados(CombEdos As ComboBox,
```

```
TexEdosID As TextBox)
```

```
If CombEdos <> "" Then
```

```
TexEdosID = CombEdos.ItemData(CombEdos.ListIndex)
```

```
End If
```

```
End Sub
```

```
Public Sub AvisoLlenado(CuadrosTex As TextBox, Datas As
Data, MiAlgo As Variant)
```

```
If CuadrosTex = "" Then
```

```
MsgBox "Para poder guardar, debe llenar el campo",
vbExclamation
```

```
Else
```

```
If CuadrosTex <> "" Then
```

```
Datas.Recordset.Update
```

```
Datas.Refresh
```

```
MiAlgo = ""
```

```
End If
```

```
End If
```

```
End Sub
```

```
Function FechaDevolucion(TextoFecha As TextBox, Boton As
CommandButton)
```

```
If (CDate(TextoFecha) + 15) < Date Then
```

```
MsgBox "La fecha de devolucion ha expedido, dar
cancelar.", vbExclamation
```

```
TextoFecha = ""
```

```
Boton.SetFocus
```

```
Else
```

```
If CDate(TextoFecha) > Date Then
```

```

MsgBox "Ha puesto una fecha equivocada, rectificar.",
vbExclamation
TextoFecha = ""
TextoFecha.SetFocus
End If
End If
End Function

```

```

Public Function ColorGris() As Variant
ColorGris = -2147483644
End Function

```

```

Public Sub BusquedaIdList(Textos As TextBox, Listas As
ListBox)
Dim i As Long
i = 0
If Textos <> "" And Textos <> "" Then
Do Until Textos = Listas.ItemData(i)
i = i + 1
Loop
Listas.ListIndex = i
Else
Listas.ListIndex = -1
End If
End Sub

```

```

Public Sub LlenarList(MiListBox As ListBox, MiTb As
String, MiCampo As String, MiID As String)
Dim MiBD As Object
Dim MiTab As Object

Set MiBD = OpenDatabase(MiBDStr)
Set MiTab = MiBD.OpenRecordset("SELECT " & MiTb & ".*"
FROM " & MiTb, dbOpenSnapshot)

```

```

Do Until MiTab.EOF
MiListBox.AddItem IIf(IsNull(MiTab(MiCampo)), "",
MiTab(MiCampo))
MiListBox.ItemData(MiListBox.NewIndex) = MiTab(MiID)
MiTab.MoveNext
Loop
End Sub

```

```

Public Function QuitaNull(miVar As Variant) As Variant
QuitaNull = IIf(IsNull(miVar), "", miVar)
End Function

```

```

Public Function PonCero(miVar As Variant) As Variant
PonCero = IIf(IsNull(miVar) Or Trim(miVar) = "", 0, miVar)
End Function

```

```

Public Function DameFormato(Valor As Variant, Formato As
String) As String
Select Case Formato
Case "MY"
DameFormato = UCase(Valor)
Case "NM"
DameFormato = Format(Valor, "currency")
End Select
End Function

```

```

Public Sub DameBorrar()
Dim MiBaseDatos As Object
Dim MiRegistro As Object
Set MiBaseDatos = OpenDatabase(MiBDStr)
Set MiRegistro = MiBaseDatos.OpenRecordset("SELECT
TbTmpdtlIFac.* FROM TbTmpdtlIFac", dbOpenDynaset)
'Costante para abrir en modo solo lectura y si fuera
dbOpenDynaset era para lectura/escritura'
Do Until MiRegistro.EOF
MiRegistro.Delete
MiRegistro.MoveNext
Loop
End Sub

```

```

Public Sub HighLight()
With Screen.ActiveForm
If TypeOf .ActiveControl Is TextBox Then
.ActiveControl.Select = 0
.ActiveControl.SelectLength = Len(.ActiveControl)
End If
End With
End Sub

```

```

Public Sub Respaldo()

```

```

Dim MiError As Variant

```

```

On Error GoTo ErrorMain

```

```

DoEvents
FileCopy "C:\Archivos de programa\Microsoft Visual
Studio\VB98\Programa de Tesis\BDZapateria.mdb",
"A:\BDZapateria.mdb"

```

```

Exit Sub

```

```

ErrorMain:

```

```

MiError = Err
If Err = 53 Or Err = 76 Then
MsgBox "No se pude copiar correctamente la Base de
Datos. Reintentar. "

```

```

End If
End Sub

```

```

Public Sub Pausar(Minutos)
Dim PauseTime, Start
PauseTime = Minutos * 60
Start = Timer
Do While Timer < Start + PauseTime
DoEvents
Loop
End Sub

```

```

Public Sub Recuperar()

```

```

Dim MiError As Variant

```

```

On Error GoTo ErrorMain

```

```

FileCopy "A:\BDZapateria.mdb", MiBDStr

```

```

Exit Sub

```

```

ErrorMain:

```

```

MiError = Err
If Err = 53 Or Err = 76 Then
    MsgBox "No se pude cargar correctamente la Base de
    Datos. Reintentar. "
End If
End Sub

```

```

Public Function PonCero2(miVar As Variant) As Variant
PonCero2 = IIf(Not IsNumeric(miVar), 0, miVar)
End Function

```

```

Public Sub MostrarMenuPrin(Optional MiIO As Integer)
If MiIO = 0 Then
    FoMenu.Show
    FoMenu.Enabled = True
    FoMenu.Height = 3030
ElseIf MiIO = 1 Then
    FoMenu.Enabled = False
    FoMenu.Height = 100
End If
End Sub

```

```

Public Function DiaUltimoDelMes(MiFecha As String)
Dim i As Integer
Dim MiFecha2 As String
If MiFecha <> "" And MiFecha <> " " Then
MiFecha = Format(MiFecha, "1/mm/yy")
MiFecha2 = Format(DateAdd("m", 1, MiFecha),
"1/mm/yyyy")
Do Until CDate(MiFecha) >= CDate(MiFecha2)
    MiFecha = MiFecha + CDate(1)
    DiaUltimoDelMes = DiaUltimoDelMes + 1
Loop
End If
End Function

```

```

Public Sub BorrarTab(Tb1 As String, Optional WH As String)
Dim MiBD As Object
Dim MiTab1 As Object

    Set MiBD = OpenDatabase(MiBDStr)
    Set MiTab1 = MiBD.OpenRecordset("SELECT " & Tb1 &
"* FROM " & Tb1 & IIf(WH = "", "", " WHERE " & WH),
dbOpenDynaset) 'Costante para abrir en modo solo lectura y si
fuera dbOpenDynaset era para lectura/escritura'

    Do Until MiTab1.EOF
        MiTab1.Delete
        MiTab1.MoveNext
    Loop
End Sub

```

ANEXOS



Fecha: 01/11/2002

PROVEEDORES

Pág. 1

Clave del proveedor : 0001
Razon social : DISTRIBUIDORA DE CALZADO SAN MATEO
Nombre del proveedor : Eric Huante Heredia
Calle : Hacienda Pasteje # 30
Colonia : Santa Elena
Telefono del negocio : 017221498530
C.P. : 5211
E-mail: e_huante@hotmail.com
Estado : México
Banco : Banamex

Fax :
RFC : 12345EHHH

Ciudad : Toluca
Cuenta : 54774578875676

Clave del proveedor : 0002
Razon social : ZAPATERIA LA HERRADURA
Nombre del proveedor : Carlos Hernandez Jimenez
Calle : Espinos # 350
Colonia : Prados Verdes
Telefono del negocio : (01728)2834516
C.P. : 38567
E-mail: juancho_65@hotmail.com
Estado : México
Banco :

Fax : (01728)2854917
RFC : 3456CHJ78

Ciudad : Toluca
Cuenta :

Clave del proveedor : 0003
Razon social : CALZADO INTERNACIONAL GOMEZ
Nombre del proveedor : Antonio Gomez Salazar
Calle : San Nicolas #45
Colonia : Allende
Telefono del negocio : 7895467
C.P. : 40056
E-mail: control34@hotmail.com
Estado : Guerrero

Fax : 7895467
RFC : 2454GMSA55

Ciudad : Acapulco



COMPRAS

No. de Factura

379

Pág. 1

Fecha de Compra 09/09/2002

Proveedor: Carlos Hernandez Jimenez

CLAVE	DESCRIPCION	MENUDEO	MAYOREO	PARES	IMPORTE
ALFMER36922	Alfa, Mujer, Zapatilla, Charol, Negro, 22	\$494.00	\$469.30	1	\$380.00
ALFMER36922.5	Alfa, Mujer, Zapatilla, Charol, Negro, 22.5	\$494.00	\$469.30	1	\$380.00
ALFMER36923	Alfa, Mujer, Zapatilla, Charol, Negro, 23	\$494.00	\$469.30	1	\$380.00
ALFMER36923.5	Alfa, Mujer, Zapatilla, Charol, Negro, 23.5	\$494.00	\$469.30	1	\$380.00
ALFMER36924	Alfa, Mujer, Zapatilla, Charol, Negro, 24	\$494.00	\$469.30	1	\$380.00
ALFMER36924.5	Alfa, Mujer, Zapatilla, Charol, Negro, 24.5	\$494.00	\$469.30	1	\$380.00
OZOHRE24525	Ozono, Hombre, Formal, Piel, Negro, 25	\$286.00	\$271.70	1	\$220.00
OZOHRE24525.5	Ozono, Hombre, Formal, Piel, Negro, 25.5	\$286.00	\$271.70	1	\$220.00
OZOHRE24526	Ozono, Hombre, Formal, Piel, Negro, 26	\$286.00	\$271.70	1	\$220.00
OZOHRE24526.5	Ozono, Hombre, Formal, Piel, Negro, 26.5	\$286.00	\$271.70	1	\$220.00
OZOHRE24527	Ozono, Hombre, Formal, Piel, Negro, 27	\$286.00	\$271.70	1	\$220.00
OZOHRE24527.5	Ozono, Hombre, Formal, Piel, Negro, 27.5	\$286.00	\$271.70	1	\$220.00
Cuatro mil ciento cuarenta pesos 00/100 M.N.				SUB-TOTAL	\$3,600.00
				I.V.A.	\$540.00
				TOTAL	\$4,140.00



Fecha: 01/11/2002

INVENTARIO

Pág. 1

CLAVE	DESCRIPCION	FECHA	UBICACION	STOCK MAXIMO	STOCK MINIMO	EXIST. REAL
OZOHRE24526	Ozono, Hombre, Formal, Piel, Negro, 26	01/11/2002	Pasillo # 10, Anaquel # 2	4	1	2
OZOHRE24526.5	Ozono, Hombre, Formal, Piel, Negro, 26.5	01/11/2002	Pasillo # 10, Anaquel # 2	4	1	2
OZOHRE24527	Ozono, Hombre, Formal, Piel, Negro, 27	01/11/2002	Pasillo # 10, Anaquel # 2	4	1	2
OZOHRE24527.5	Ozono, Hombre, Formal, Piel, Negro, 27.5	01/11/2002	Pasillo # 10, Anaquel # 2	4	1	2
FLEHRE34625.5	Flexi, Hombre, Casual, Piel, Cafe, 25.5	09/09/2002	Pasillo # 2 , Anaquel # 4	10	5	6
ALFMER36922	Alfa, Mujer, Zapatilla, Charol, Negro, 22	09/09/2002	Pasillo # 2 , Anaquel # 4	5	2	1
ALFMER36922.5	Alfa, Mujer, Zapatilla, Charol, Negro, 22.5	09/09/2002	Pasillo # 2 , Anaquel # 4	5	1	1
ALFMER36923	Alfa, Mujer, Zapatilla, Charol, Negro, 23	09/09/2002	Pasillo # 2 , Anaquel # 4	5	2	3
ALFMER36923.5	Alfa, Mujer, Zapatilla, Charol, Negro, 23.5	09/09/2002	Pasillo # 2 , Anaquel # 5	5	2	1
ALFMER36924	Alfa, Mujer, Zapatilla, Charol, Negro, 24	09/09/2002	Pasillo # 2 , Anaquel # 5	5	2	2
ALFMER36924.5	Alfa, Mujer, Zapatilla, Charol, Negro, 24.5	09/09/2002	Pasillo # 2 , Anaquel # 5	5	2	2
FLEHRE34626	Flexi, Hombre, Casual, Piel, Cafe, 26	09/09/2002	Pasillo # 2 , Anaquel # 5	8	3	4
OZOHRE24525	Ozono, Hombre, Formal, Piel, Negro, 25	01/11/2002	Pasillo # 10, Anaquel # 2	4	1	2
OZOHRE24525.5	Ozono, Hombre, Formal, Piel, Negro, 25.5	01/11/2002	Pasillo # 10, Anaquel # 2	4	1	2



VENTAS

No. de Venta

0001

Pág. 1

Forma de Pago: Crédito

Fecha de Venta

10/10/2002

Atendido por : Javier Garfias Heredia

CLAVE PROV.	DESCRIPCION	PRECIO	PARES	IMPORTE
0002	Alfa, Mujer, Zapatilla, Charol, Negro, 22	\$494.00	1	\$494.00
0002	Alfa, Mujer, Zapatilla, Charol, Negro, 23.5	\$494.00	2	\$988.00
SUB-TOTAL				\$1,482.00
DESCUENTO				\$0.00
SUBTOTAL				
DESCUENTO				\$1,482.00
IVA				\$222.30
TOTAL				\$1,704.30



FACTURAS

Folio

0003

Pág. 1

Fecha

01/11/2002

Cliente

Jennifer Lopez Ruiz
Chapultepec Sur
Jacarandas # 565
Morelia
Michoacán
R.F.C. JNLORU70659874 TEL. 3156987

Atendido por

Salvador Padilla Heredia

Número de Pedido

DESCRIPCION	PRECIO	PARES	IMPORTE
Flexi, Hombre, Casual, Piel, Cafe, 25.5		\$1.00	\$300.00
Ozono, Hombre, Formal, Piel, Negro, 27		\$1.00	\$286.00
Seiscientos sesenta y siete pesos 00/100 M.N.		SUB-TOTAL	\$586.00
		DESCUENTO	\$6.00
		SUBTOTAL	
		DESCUENTO	\$580.00
		I.V.A.	\$87.00
		TOTAL	\$667.00

Fecha: 01/11/2002



CLIENTES

Pág. 1

Clave del cliente : 0001
Nombre del cliente : Edgar Rafael Contreras Heredia
Calle : Tanganxoan # 887
Colonia : Félix Ireta
Telefono del negocio : 014433164633
Telefono particular : 014433164633
C.P. : 58070
RFC : EGHM827824

Ciudad : Morelia
Estado : Michoacán

Clave del cliente : 0002
Nombre del cliente : Sergio Andrei Cardenas Suarez
Calle : Av. Cuahutemoc # 853
Colonia : Izcalli
Telefono del negocio : 2119872
Telefono particular : 2134567
C.P. : 6009
RFC : 5745SEC65

Ciudad : Uruapan
Estado : Michoacán

Clave del cliente : 0003
Nombre del cliente : Jesus Aguilar Garcia
Calle : Tucuran # 678
Colonia : Valle de Mexico
Telefono del negocio : 4568900
Telefono particular : 4336723
C.P. : 52234
RFC : JEAG4576

Ciudad : San Juan Del Rio
Estado : Queretaro

Fecha: 01/11/2002



EMPLEADOS

Pág. 1

Clave del Empleado : JAV01
Nombre del Empleado : Javier Garfias Heredia
Calle : Arriaga Rivera # 58
Colonia : Col. Centro
Telefono particular : 014363632733
Telefono de Emergencia : 014363636150
C.P. : 58600
Puesto: Mostrador

Ciudad : Morelia
Estado : Michoacán
Sueldo : \$700.00

Clave del Empleado : MAG02
Nombre del Empleado : Magdalena Heredia Mendoza
Calle : Ventura Puente # 657
Colonia : Rio Grande
Telefono particular : 5007890
Telefono de Emergencia : 4890062
C.P. : 58070
Puesto: Administrador

Ciudad : Morelia
Estado : Michoacán
Sueldo : \$4,500.00

Clave del Empleado : CHAVA03
Nombre del Empleado : Salvador Padilla Heredia
Calle : Cuahutemoc
Colonia : Naranja # 14
Telefono particular : 3264876
Telefono de Emergencia : 3548792
C.P. : 948478
Puesto: Mostrador

Ciudad : Morelia
Estado : Michoacán
Sueldo : \$500.00



DEVOLUCIONES

Pág. 1

Fecha Devolución : 10/10/2002

Fecha Compra : 10/10/2002

CAMBIO

CALZADO A CAMBIAR

Folio Venta :

Clave Calzado :

Pares a Cambiar :

No. de Pares :

Descripción Calzado :

CALZADO A CAMBIO

Folio Venta

Clave Calzado :

Cantidad Pares :

Clave Proveedor

Descripción Calzado :

DEVOLUCION

Folio Venta : 0001

Clave Calzado : ALFMER36922

Pares a Devolver : 1

No. de Pares: 2

Descripción

Alfa, Mujer, Zapatilla, Charol, Negro, 22

Observaciones

Fecha: 01/11/2002



CATALOGOS

Pág. 1

MARCA	SEXO	ESTILO	MATERIAL	COLOR	LINEA
Nike	Hombre	Casual	Suela de	Azul	10
Adidas	Joven	Bota fox	Cuero	Verde	10.5
Fila	Niño	Zapatilla	Suela de	Blanco	11
Flexi	Bebito	Tenis	Baquelita	Morado	11.5
Tobi	Mujer	Futbol	Sintético	Negro	12
Alfa	Jovencita	Formal	Charol	Negro	12
Zandac	Niña	Formal	Piel	Cafe	12.5
Niko	Bebita	Basketbol	Binipiel	Vino	13
Adiora		Sandalia	Plastico	Plata	13.5
Lobo Solo		Bota Vaqu		Azul Marini	14
ioba				Rojo	14.5
Ozono					15
Ozonía					15.5
Reebok					16
Mediterrane					16.5
Caracy					17
DKNY					17.5
Hush Puppi					18
Alondra					18.5
K-Suiss					19
Leonardos					19.5
Brasil					20
Bubble Gun					20.5
					21
					21.5
					22
					22.5
					23
					23.5
					24
					24.5
					25
					25.5
					26