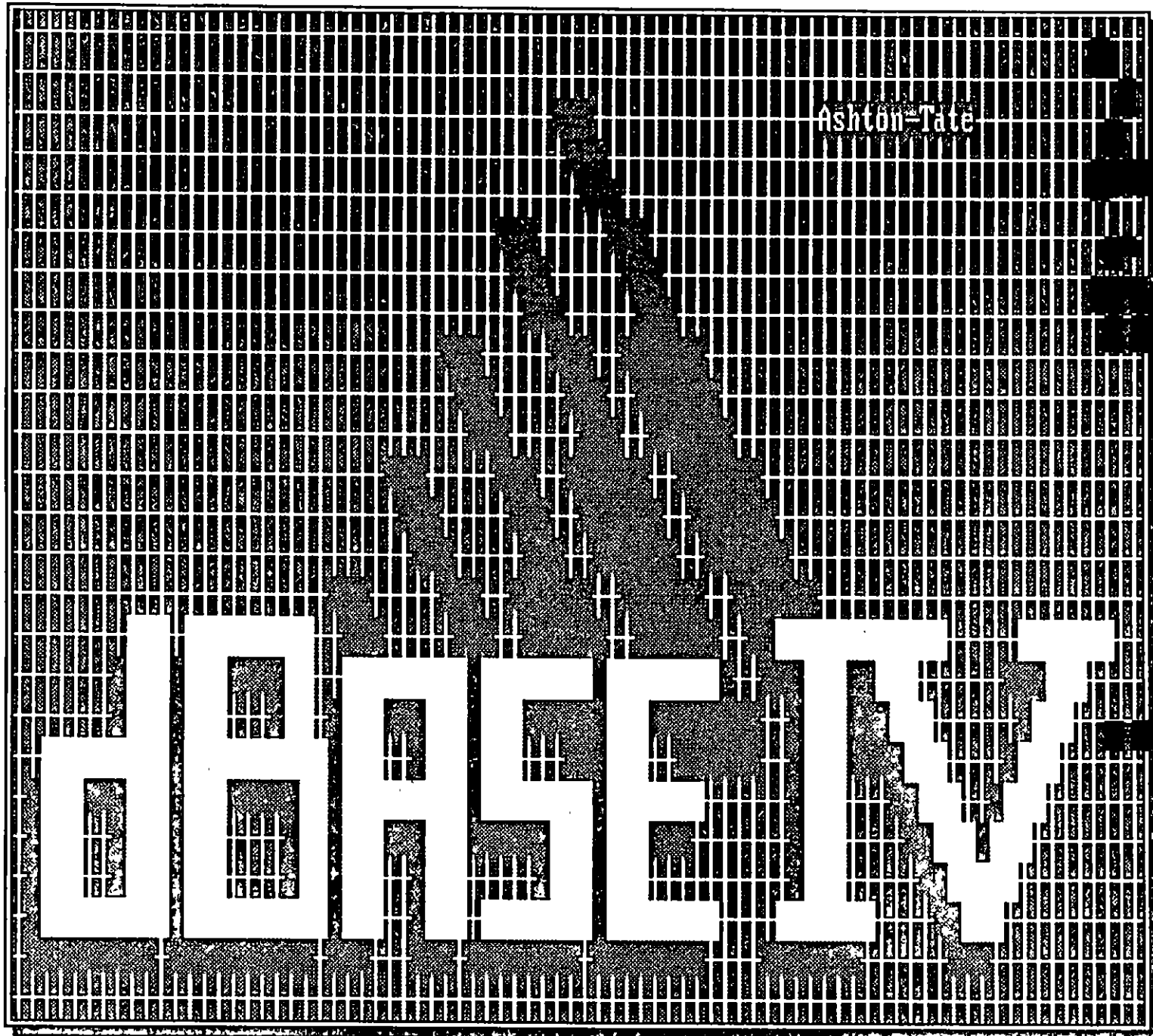


Ashton-Tate

PROFIT



dBASE IV PARA USUARIOS FINALES

OBJETIVO :

Al finalizar el curso, el participante estará familiarizado con la tecnología de Bases de Datos, a través de la utilización del Paquete Computacional dBASE IV.

Dominará los conceptos necesarios para el diseño, explotación y administración de Bases de Datos, a través del Control de Control. Comandos en línea y programación, propios del paquete.

A QUIEN VA DIRIGIDO :

El curso va dirigido a aquellas personas interesadas en la explotación eficiente de bancos de datos de volumen medio, a través de la generación inmediata de consultas y reportes.

Adicionalmente a personas que por su desarrollo profesional y/o superación personal, requieran incrementar su productividad por medio de la microcomputación, utilizando paquetes en el desarrollo de aplicaciones.

TEMARIO

I .- Introducción a dBASE IV.

- i Definiciones elementales.
- ii Configuración e instalación.
- iii Consideraciones en el arranque.

II .- Creación y Actualización de las Bases de Datos.

- i Diseño de la Base de Datos.
- ii Creación de la Base de Datos.
- iii Actualización de la Base de Datos.

III .- Métodos de Clasificación de la Información.

- i Ordenamientos.
- ii Indexación de archivos.

IV .- Recuperación de la información.

- i Posicionamiento Directo.
- ii Posicionamiento através de índices.
- iii Filtros.
- iv Conceptos del SQL.

V .- Emisión de Reportes y Etiquetas.

- i Creación de Reportes.
- ii Creación de Etiquetas.

VI .- Comandos SET y Funciones.

- i Comandos SET.
- ii Funciones Básicas.

VII .- Creación de Pantallas.

- i Generador de Pantallas.
- ii Archivos FORMAT.

VIII .- Introducción a la programación con dBASE IV.

- i Conceptos Básicos de Programación Estructurada.
- ii Creación y Ejecución de Programas en dBASE IV.
- iii Generador de Aplicaciones, una herramienta alterna.

Introducción a dBASE IV.

Los sistemas de administración de bases de datos (DBMS) han jugado un papel central en el éxito de la revolución de las PC'S. Han dado a los usuarios finales poder y control al solucionar sus problemas en el manejo y aplicación de la información.

Aunque base de datos es un término técnico, relacionado con una computadora, este también puede aplicarse a las formas en las cuales la información se cataloga, almacena y utiliza. Así, cualquier colección de datos relacionados y agrupados como un sólo ente, se le podrá denominar base de datos, teniendo una por ejemplo en un tarjetero; o en una agenda con nombres, telefonos y direcciones.

Cabe notar que lo que genera una base de datos es la forma en la cual está organizada la información y que el archivero o la agenda son sólo una ayuda y dBASE IV es una de tales ayudas.

De acuerdo a como está estructurada una base de datos se pueden clasificar como sigue :

- 1) JERARQUICAS También conocidas como DE ARBOL, dado que las unidades de datos se estructuran en niveles multiples que representan gráficamente un árbol invertido, así un registro está relacionado con otro(s) tipo(s) de registro(s) de menor jerarquía pero sólo con uno de mayor nivel.
- 2) DE RED A diferencia del modelo de árbol que no permite interconexión de ramas, este tipo de estructura lo permite y en forma multidireccional, es decir, cada tipo de registro puede tener más de un tipo de relación.
- 3) RELACIONALES Una estructura relacional se genera almacenando los datos como listas o relaciones interconectadas mediante ligas, por medio de las cuales se pueden simular los otros tipos de estructuras de base de datos.

Para crear, manipular y consultar una base de datos se requiere de programas especiales para realizar dichas tareas; a este software se le conoce como Sistemas Manejadores de Bases de Datos (DBMS data base management system), y existen para todos los tamaños de computadoras, macros, minis y micros. Un DBMS es capaz de organizar, procesar y presentar los datos seleccionados de diversas formas, de acuerdo a las aplicaciones y necesidades del usuario, es decir, proporciona una gran flexibilidad en el manejo de la información así como Una gran facilidad de acceso y actualización.

Pero no sólo se necesita software, sino también a una(s) persona(s) que se encarguen de diseñar y controlar la información. Sobre quien recae tal responsabilidad se le conoce como Administrador de Bases de Datos (DBA).

QUE ES dBASE IV ?

dBASE IV, es un Sistema Manejador de Bases de (DBMS). Que se puede definir como el conjunto de programas que permiten fácilmente, la administración de grandes volúmenes de información.

El tipo de operaciones que permite al usuario es el de crear, actualizar, ordenar, clasificar, seleccionar, emitir listados y reportes, elaboración de etiquetas para listas de correos, y un sin número de operaciones con la información que contenga el sistema que se desea controlar por medio de la computadora. Por ejemplo algunas de las aplicaciones más frecuentes en las que dBASE IV es utilizado son:

- Control de Inventarios
- Listas de Precios
- Directorios telefónicos
- Nóminas
- Contabilidad
- Control de Ordenes de Trabajo
- Control de Chequera
- Conciliación Bancaria

Y en general todos aquellos sistemas que manejen volúmenes de información considerables y que se desea llevar un control más fácil de esa información.

Antes de continuar con el estudio de este paquete, es necesario definir algunos conceptos básicos que serán de gran utilidad posteriormente.

DATOS .- Son la materia prima que al ser procesada dan como resultado la información. por ejemplo :

Son datos las horas trabajadas y el salario por hora, pero el importe a pagar por las horas trabajadas a la semana es la información.

Los datos son elementos que dan informes sobre el nombre, tipo e intervalo de valores. Ejemplo de esto es; el nombre, dirección, ciudad, estado, clave postal y el número del cliente.

CAMPO .- Un campo define la localización física de almacenamiento de una unidad de datos o información.

Un campo puede tener uno o más BYTES de longitud, siendo este la unidad más pequeña de datos o información que se puede definir y manejar, es el común denominador entre el USUARIO y la COMPUTADORA.

Un ejemplo de campos en una base de datos es:

<u>Campos</u>	<u>Datos</u>
NOMBRE :	DISTRIBUIDORA INTEGRAL DE COMPUTACION, SA. DE C.V.
DIRECCION :	MEDELLIN 184 DESPACHO 105
COLONIA :	ROMA
CIUDAD :	MEXICO
ESTADO :	DISTRITO FEDERAL
C.P. :	06700
TELEFONO :	584-88-22

REGISTRO .- Es un conjunto de campos sobre un sujeto o tema como es un proveedor, un cliente, un empleado, un vendedor, un producto, etc. A una colección de registros se le denomina ARCHIVO (o base de datos). 10,000 empleados activos en una organización implica 10,000 registros físicos en un archivo o base de datos. Existen dos tipos de archivos los que contienen registros principales con datos semipermanentes como en el caso de empleados y clientes y los archivos de transacciones que contienen datos eventuales como los pedidos, ordenes de trabajo, entradas y salidas al almacén, etc.

CONFIGURACION E INSTALACION.

dBASE IV es el sistema de base de datos más completo y poderoso jamás diseñado para una computadora personal, elevando el estándar a nuevos niveles de facilidad de uso, rapidez, automatización de aplicaciones, así como capacidad de conectividad y comunicación.

dBASE IV está diseñado para todos los niveles de usuarios. No es necesaria ninguna experiencia para los novatos, ya que sin necesidad de conocimientos en programación, es posible diseñar formas, reportes, etiquetas, inclusive automatización de aplicaciones completas. dBASE IV se encarga de generar automáticamente el código necesario para automatizar los desarrollos.

CARACTERISTICAS PRINCIPALES

OBTENCION DE AYUDA

En el caso de que necesite ayuda, el dBASE III PLUS le puede ofrecer una breve información sobre cuestiones que van desde el punto indicativo hasta la creación de una base de datos, información que está contenida en un archivo HELP del disco del sistema y que es accesible mientras se permanece en el dBASE III PLUS. Una serie de menús le ayudará a encontrar la información que necesite.

El dBASE III PLUS le ofrece ayuda aún sin que usted le pida. Si el dBASE III PLUS no entiende cómo intenta usted usar una orden le preguntará si necesita ayuda de los archivos HELP.

EL AYUDANTE

Todas las ayudas del ayudante que se refieren al cambio de archivos están contenidas en el menú Update. Para modificar la estructura de la base de datos, use la opción Database File del menú MODIFY. Desde el punto indicativo introduzca

ASSIST

El menú Set Up aparecerá resaltado. Como lo que queremos es ir a Update, pulse U para abrir el menú Update. Ver que el menú Update proporciona acceso a las mismas órdenes BROWSE, EDIT, DELETE, RECALL, PACK y REPLACE que se han usado a lo largo de este capítulo.

CENTRO DE CONTROL

dBASE IV ofrece un sin fin de beneficios para el usuario. Su facilidad de uso se debe principalmente a la introducción del Centro de Control. Este consta de seis paneles que proporcionan acceso, a través de menús, a las operaciones de creación de archivos, consultas a bases de datos, edición de formas, generación de reportes, diseño de etiquetas y generación de aplicaciones.

SQL

Es el estándar de la industria en consulta y manipulación de información. La combinación de comandos de SQL y dBASE proporciona mayor versatilidad en el desarrollo de aplicaciones de consulta en base de datos.

Una vez que el usuario especifique la tarea a ejecutar, así como las formas, reportes y archivos que desee utilizar, el generador de aplicaciones de dBASE IV los integra en una aplicación. El generador de aplicaciones se encarga de generar automáticamente todo el código de dBASE.

Compilador Automático...

Mayor rapidez y poder con el compilador integrado que acelera la velocidad de ejecución, haciéndolo 10 veces más rápido que dBASE III PLUS. Esto significa eficiencia en sus aplicaciones.

Query-by-Example (QBE)...

Esta característica hace de la recuperación y actualización de información, una tarea sumamente sencilla. El usuario principiante encontrará que extraer información de varios archivos simultáneamente, es una sola operación sumamente sencilla. El usuario avanzado lo encontrará asombroso.

Generador de Reportes...

El diseño de reportes columnares, formas pre-impresas o documentos de correo electrónico, se logra en un instante con la facilidad WYSIWYG (lo que ve es lo que obtiene) del generador de reportes.

Ambiente Multiusuario...

dBASE IV es la vanguardia en ambiente multiusuario. Sus aplicaciones individuales de dBASE III PLUS correrán como multi-usuario en dBASE IV, sin modificaciones. Cada dBASE LAN Pack contiene discos para hasta 5 usuarios simultáneos de dBASE IV y la documentación necesaria para usarlo en ambiente red.

Lenguaje Enriquecido...

dBASE IV incluye un set de 245 comandos nuevos, los cuales le permiten realizar operaciones tales como:

- Funciones definidas por el usuario.
- Arreglos bidimensionales.
- Funciones estadísticas, matemáticas y financieras.
- Campos calculados, tablas de entrada múltiple, validaciones.
- Manejo de ventanas y menus.
- Mantenimiento automático de hasta 47 índices simultáneos.

Mayor Poder...

Soporta hasta 10 bases de datos abiertas simultáneamente, 99 archivos abiertos en total, mil millones de registros por archivos, 255 campos por registro y hasta 255 caracteres por campo tipo carácter.

II.- CREACION Y ACTUALIZACION DE LAS BASES DE DATOS.

1.- Diseño de la Base de Datos.

Antes de realizar cualquier operación con dbase IV, es fundamental definir QUE es lo que deseamos hacer (análisis) y posteriormente COMO lo realizaremos (diseño). Dentro de éste último, está la etapa de diseño de los archivos que se van a manejar, así como las posibles relaciones que pueden existir entre ellos.

El primer paso para poder definir cual será la estructura de un archivo determinado, y más aún, cuantos archivos tendrán que definirse, es la definición de los datos que estarán involucrados en nuestra Base de Datos. Esto se logra analizando los requerimientos del Usuario, así como los documentos fuente y los resultados que desea obtener.

Una vez establecidos los datos que serán incluidos en los archivos, se deben agrupar de tal forma que los datos incluidos en cada grupo estén fuertemente relacionados entre si formando una unidad o una entidad independiente; ésta entidad será convertida en un archivo DBF, por lo cual se deben establecer los parámetros necesarios para poder ser definidos en dbase IV: tipo, tamaño, si es numérico, el número de decimales y si será con o sin punto flotante, y si será considerado campo clave para ser accedido.

Es importante decir que en caso de querer relacionar una entidad (o archivo) con otra, se deben definir los campos de iguales características para poder facilitar la implementación.

2.- Creación de la base de datos.

Para poder empezar a trabajar con dbase IV, debemos crear los archivos donde quedarán los datos. Esto se logra con la instrucción CREATE.

SINTAXIS

```
CREATE <filename>
```

Esta instrucción permitirá a través de una pantalla de edición, definir el nombre de los campos así como su longitud y tipo.

El nombre del archivo está limitado a 8 caracteres, comenzando con letra y solo puede incluir números, letras y el subguión. dbase IV le agregará la extensión .DBF, para identificarla como un archivo de base de datos.

Para realizar la creación de un archivo, basta escribir la instrucción y presionar la tecla RETURN.

Ejemplo:

```
.CREATE balumnos
```

Cuando ésta sea ejecutada, presentará una pantalla en forma tabular, con las siguientes columnas:

Layout Organize Append Go To Exit

Num	Field Name	Field Type	Width	Dec	Index
1		Character			N

Database | C:\dbaseiv\trabajo\balumnos | Field 1/1 |

La columna Num contiene la posición del campo en el registro. Esta es solo a nivel informativo ya que la secuencia no influye más que para aspectos visuales de revisión.

La columna Field Name contendrá el nombre del campo, éste podrá ser máximo de 10 caracteres aceptando solo letras, dígitos y el subguión.

La siguiente columna es muy importante, ya que define el tipo de valor que podrá almacenarse en el campo; los tipos disponibles en dbaseIV son:

Character (por omisión) permite escribir cualquier caracter ASCII y puede aceptar hasta 254 caracteres.

- Numeric** Este tipo almacena números fijos en un rango de 1 a 20 dígitos. Es conveniente usar este tipo cuando se manejan números exactos. Lo cual brindará mayor rapidez en las operaciones de suma y resta, éstos también son comúnmente llamados binary coded decimal.
- Float** En este tipo se almacenan valores de punto flotante también de 1 a 20 dígitos de longitud, los cuales son típicamente usados en aplicaciones científicas. Estos pueden ser usados para realizar operaciones en forma rápida, involucrando multiplicaciones y divisiones, sobre todo cuando se manejan valores muy altos o muy pequeños.
- Date** Este tipo define campos fecha, los cuales serán de longitud fija de 8, almacenados en formato norteamericano, pudiendolos visualizar en otro formato.
- Logical** Este tipo se define automáticamente de longitud unitaria y solo puede contener dos valores; verdadero (.T.) o falso (.F.), aunque existe la posibilidad de utilizar el formato .Y. (SI) o .N. (NO).
- Memo** Este campo tiene por longitud fija en la estructura de 10. Sin embargo esto no implica que sea el tamaño, más bien es la longitud máxima de la dirección dentro de un archivo que se crea automáticamente cuando se incluye cuando menos un campo de este tipo; este archivo anexo tendrá el mismo nombre de la base pero con la extensión .DBT. Este campo podrá contener el número máximo de caracteres que el procesador, utilizado para estos campos y definido en el config.db, permita.

La columna de Width (tamaño) establece de cuantos caracteres será el campo en cuestión.

La siguiente columna solo se usa en campos numéricos y define cuantos decimales serán considerados. Es importante señalar que estos dígitos junto con el punto están incluidos en el tamaño establecido en la columna de Width.

La última columna Index, permite definir a un campo como llave dentro de un archivo de índice (.ndx) o marcas (.mdx), el cual será generado al crear la base de datos.

Ejemplo:

Layout Organize Append Go To Exit

Num	Field Name	Field Type	Width	Dec	Index
1	BNUM_ALUMN	Numeric	4	0	Y
2	BAPELLIDOS	Character	25		Y
3	BNOMBRE	Character	15		N

Database | C:\dbaseiv\trabajo\balumnos | Field 1/1 |

3.- Actualización de la base de datos.

Apertura y cerrado de archivos de base de datos.

Cuando una base de datos se crea por primera vez, se generan los archivos dbf, dbt y mdx, en caso de haberlos definido a través de los campos de la misma. Al realizar ésta creación, los archivos quedarán ABIERTOS, es decir que están disponibles para almacenar, recuperar o actualizar datos.

Sin embargo, al abrir otro archivo o al salir de la sesión de dBASEIV, éstos quedarán CERRADOS, es decir no disponibles en la memoria para cualquier transacción sobre ellos. Por lo cual es importante saber como podremos abrirlos nuevamente y cerrarlos cuando así se requiera.

SINTAXIS

```
USE [<database filename>/? [IN <work area number>]]  
[[INDEX <.ndx or .mdx file list>] [ORDER <.ndx  
filename> / <.mdx tag> [OF <.mdx filename>]] [ALIAS  
<alias>][EXCLUSIVE] [NOUPDATE]
```

Esta instrucción permite abrir una base de datos en un área de memoria determinada, esto implica que estará, a partir del momento de aceptar la instrucción dBASEIV, disponible para cualquier actualización o consulta que se quiera efectuar sobre ella (dependiendo de cuales sean los atributos de Escritura y Lectura, tema que se tratará en otro curso).

Ejemplo:

```
USE balumnos
```

Lo cual pondrá a disposición la base balumnos que debió ser creada previamente.

Cuando se quiere cerrar una base de datos, es suficiente dar la instrucción USE sin nombre de archivo, para que la base abierta con los archivos de índice y de texto, sean cerrados. Sin embargo existe una instrucción alterna que es:

SINTAXIS

```
CLOSE DATABASES
```

Ejemplo:

```
.USE balumnos  
.USE  
.USE balumnos  
.CLOSE DATABASES
```

Alimentacion de datos.

Una vez definido el archivo de Base de Datos, es importante saber como almacenar los datos correspondientes, para posteriormente poder procesar y así obtener la información requerida.

APPEND.- Este comando le permite adicionar un nuevo registro al final del archivo de B.D. activo.

SINTAXIS APPEND [BLANK]

La forma más simple es dando la instrucción APPEND solamente.

Al ejecutar la instrucción dBASE IV presentará en la pantalla los campos en blanco para ser llenados por el usuario .

Una vez terminado el último campo, grabará este registro y presentará nuevamente la pantalla mencionada de el registro siguiente.

Cabe hacer notar, que el número de registros lo asigna automáticamente dBASE IV, y siempre sigue un orden cronológico.

# REG.	bnum_alum	bapellidos	bnombre	ESTRUCTURA
1	1	PEREZ R.	JUAN	
	////////////////// E O F //////////////////////			

APPEND # REG.	bnum_alum	bapellidos	bnombre
1	1	PEREZ R.	JUAN
2	5	SALAS Z.	MARIA
	////////////////// E O F //////////////////////		

Para terminar presionar la tecla [RETURN] en el primer campo sin haber digitado algo previamente, o bien [ctrl] [End] o [ctrl] [W]

OPCIONES : BLANK

Esta opción permite adicionar un registro en blanco al final del archivo, pero sin mandar la pantalla de captura. Esta instrucción es utilizada conjuntamente con REPLACE que veremos posteriormente.

Otra opción para poder adicionar registros en la Base de Datos, es con la instrucción INSERT la cual permite insertar un registro, antes o después de donde se encuentre actualmente posicionado el apuntador del archivo.

SINTAXIS

```
INSERT [ BEFORE ] [ BLANK ]
```

La opción [BLANK] como en el caso del APPEND , inserta un registro en blanco para posteriormente ser editado en una pantalla en la que se muestran todos los campos.

El registro se adiciona después de donde esta el apuntador, a menos que se use la opción [BEFORE] que lo colocaría en una posición antes.

Cabe hacer notar que el orden en que fueron dados de alta se alteran con éste comando, renumerándose todos los registros posteriores.

Modificación de los datos.

Para poder realizar esta función importante, dBASEIV cuenta con una variedad de instrucciones para realizarla.

La instrucción EDIT es la que nos permite modificar todos o solamente algunos de los campos de un registro determinado.

SINTAXIS

```
EDIT [NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU] [NOEDIT]
[NODELETE] [NOCLEAR] [<record number>] [FIELDS<field list>]
[<scope>] [FOR<condition>] [WHILE<condition>]
```


DIC, S.A. de C.V. CURSO : dBASE IV

El formato más simple es EDIT el cual muestra en la pantalla todos los campos del registro en el cual se encuentra activo en el archivo de Base de Datos, al terminar de editarlo, pasa automáticamente al siguiente hasta que llega al fin del archivo.

Record 3

bnum_alum

bapellidos

bnombre

Cuando el número de campos es tal que no se puede desplegar en la pantalla, al mismo tiempo, dBASEIV presenta tantas pantallas como sea necesario para editar cada registro.

En muchos casos se presenta la situación de querer cambiar solo algunos campos, esto es no presentar en pantalla más que los que se requiera. Esto lo podemos lograr a través de la opción FIELDS, en la cual listamos los nombres de los campos que deseamos editar.

Por ejemplo la instrucción EDIT FIELDS bnum_alum, presentará en la pantalla solo el campo bnum_alum, para ser editado, en cada uno de los registros.

A las siguientes opciones les dedicaremos un poco más de explicación, ya que los encontraremos similarmente en la mayoría de los comandos de dBASEIV.

Lo primero que trataremos es el [<scope>] que en algunas interpretaciones en español le llaman <ámbito> o <comodin>.

Esta opción puede ser alguna de las siguientes:

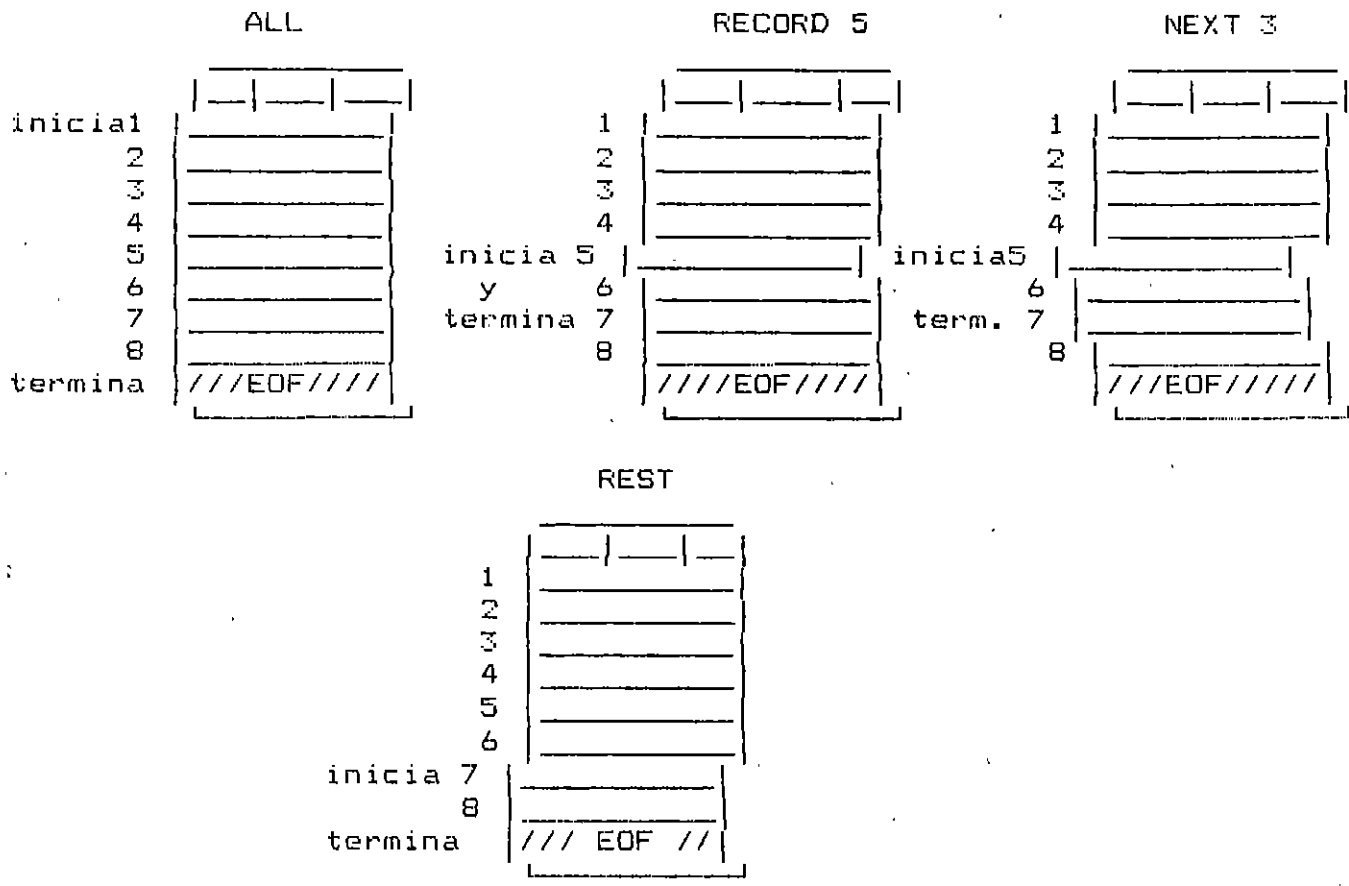
ALL

NEXT <n>

RECORD <n>

REST

ALL le indica a dBASEIV, que el comando actuará sobre todos los registros del archivo, comenzando siempre con el primero. RECORD <n> le indica que solo afectará al registro cuya posición es A, dejando posicionado el apuntador en este registro. NEXT <n> le indica que el comando se efectuará sobre los siguientes n registros, incluyendo el registro donde está posicionado el cursor y REST le indica que afectará a todos los registros a partir de donde se encuentra el cursor.



Ejemplos:

- .EDIT ALL** Qué sería idéntico a
- .EDIT** ya que en esta instrucción la opción ALL está dada por omisión
- .EDIT RECORD 5** En este caso solo editará el registro 5 y al terminar regresará el prompt o menú en caso de estar en la opción ASSIST
- .EDIT NEXT 3** En este caso editará el registro donde está posicionado más los dos restantes.
- .EDIT REST** Y por último editará a partir de donde está posicionado el apuntador hasta el último registro dejando el apuntador en la etiqueta de fin de archivo (EOF).

Existe la opción <record number>, la cual nos permite editar a partir de un registro determinado.

Ejemplo:

.EDIT 3

	1			
	2			
inicia la edición	3			
	4			
	5			
	6			
	7			
	8			
		/// EOF //		

Cabe hacer notar que esta instrucción es diferente a:

.EDIT RECORD 3

ya que esta última solo editará el registro 3, mientras que la primera, editará cualquiera pero comenzando a partir del registro 3.

Las opciones limitativas son, las siguientes:

- NOINIT** Permite usar la definición de la tabla de comandos mostrados en el menú superior de la pantalla, de la última edición hecha en la misma sesión de dBASEIV.
- NOFOLLOW** Esta opción se verá posteriormente, cuando se trate el tema de índices.
- NOAPPEND** Deshabilita la opción que tiene EDIT para poder dar de alta registros.
- NOMENU** No permite acceder al menú de opciones, que se despliega en la parte superior de la pantalla.
- NOEDIT** Muestra los registros sin permitir modificarlos.
- NODELETED** Desahabilita la posibilidad de dar de baja registros, con la opción de CTRL-U.
- NOCLEAR** Permanece la pantalla después de terminar el comando.

Para explicar las siguientes opciones, tendremos antes que definir las condiciones y como se construyen.

Comenzamos definiendo a una condición por un conjunto de operandos y operadores, ligando siempre a dos operandos por un operador.

< operando > < operador > < operando >

Los operandos pueden ser constantes, nombres de campos o nombres de variables.

Los operadores utilizados para condiciones son los lógicos, los cuales son :

= igual que	> = mayor o igual que
> mayor que	< = menor o igual que
< menor que	< > diferente que

\$ contenido en (para)

Ejemplos:

23>4 en éste caso la condición es cierta o verdadera (T) en dBASE una condición puede tener dos valores T verdadero o F falso .

bnum_alum = 35 solo será cierto cuando el campo bnum_alum valga 35

variable <= otra variable será cierto si la variable es menor o igual a otra variable.

Existen unos operadores que se les denomina conectores o ligadores lógicos, los cuales permiten combinar un conjunto de condiciones, estos son:

.AND.	Y LOGICO
.OR.	O LOGICO
.NOT.	NEGACION

Estos operadores estan regidos por las tablas de verdad como se muestra a continuación :

# RECOR	bnum_alum	bapellidos	bnombres
1	34	LOPEZ C.	MANUEL
2	65	HERNANDEZ J	RAQUEL
3	23	TORRES F.	ANA
4	54	LOPEZ H.	JOSE
5	3	KRAUZ Z.	DAVID
6	12	LOPEZ S.	SOFIA
////////////////// EOF ////////////////////			

En este caso sólo serán editados los registros 1,4 y 6 .

Es importante señalar que esta opción está limitada por el <scope> definido, vgr.

```
EDIT RECORD 2 for bapellidos = "LOPEZ"
```

No editará ningún registro, ya que el campo bapellido del registro 2 no contiene en su inicio a "LOPEZ", sin embargo

```
EDIT REST FOR bapellidos = "LOPEZ"
```

Editará el registro 4 y 6, ya que en la anterior instrucción el apuntador quedó posicionado en el registro 2.

La siguiente opción WHILE <condition >, la trataremos posteriormente, ya que se requiere una clarificación (es recomendable) previa para su mejor aprovechamiento.

La instrucción CHANGE funciona en forma similar a EDIT .

Otra instrucción que permite cambiar el contenido de un registro es: REPLACE, con la diferencia que ésta no presenta una pantalla para la edición, sino que directamente debemos definir cuál va a ser el nuevo contenido de cada campo que deseemos modificar.

SINTAXIS

```
REPLACE <field> WITH <exp.> [ADDITIVE] [,<field> WITH
<exp.> [ADDITIVE]][<scope>] [FOR <condition>] [WHILE
<condition>]
```

AND		resulta	OR		resulta	NOT	resulta
F	F	F	F	F	F	F	V
F	V	F	F	V	V	V	F
V	F	F	V	F	V		
V	V	V	V	V	V		

Siguiendo la estructura de los operandos y operadores, podemos ejemplificar lo siguiente:

- 23>4 AND 8>5 El resultado nos da T verdadero ya que 23>4 es T y 8>5 es T.
- 23>4 AND 8<5 En este caso, el resultado es falso por ser la 2a. condición falsa.
- 23>4 OR 8>5 En este ejemplo el resultado es T .
- 23>4 OR 8<5 En este caso el resultado también es T debido a que la primera condición se cumple.
- NOT 23>4 Aquí el resultado es F ya que la condición original es T que negada da el resultado mencionado.

La opción FOR <condition> repite el comando para todos aquellos registros que cumplan la condición.

ejemplo.

```
EDIT FOR bapellidos = "LOPEZ"
```

Esta instrucción leerá cada uno de los registros del archivo de la base de datos presentando para ser editados sólo aquellos cuyo campo bapellidos comience con la palabra "LOPEZ", cabe hacer notar que las comillas delimitan una constante de tipo caracter, por lo que en este caso el campo bapellido debe ser de tipo caracter, de lo contrario el sistema enviará el error DATA TYPE MISMATCH.

Esta instrucción afecta al registro corriente de la base de datos, a menos que se defina una opción de scope, FOR o WHILE que la modifique.

Ejemplo:

```
REPLACE bapellidos WITH "LOPEZ"
```

En este caso, el registro donde se encuentre posicionado el apuntador de la base abierta será el que reemplace su campo bapellidos por "LOPEZ". Sin embargo:

```
REPLACE ALL bapellidos WITH "LOPEZ"
```

reemplazará "LOPEZ" en el campo bapellidos para todos los registros de la base.

La cláusula ADDITIVE le permite construir un campo memo desde varias cadenas de caracteres. Esta opción es ignorada si no se tiene un campo memo.

En el caso de la expresión, pueden sólo utilizarse campos de igual tipo, con los operadores respectivos. Ejemplo:

```
REPLACE ALL bcosto WITH bcosto*1.15
```

La instrucción BROWSE nos permite modificar, borrar y adicionar registros a un archivo de base de datos, a través de una pantalla en la cual se puede visualizar más de un registro al mismo tiempo.

SINTAXIS

```
BROWSE [NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU] [NOEDIT]
[NODELETE] [NOCLEAR] [COMPRESS] [FORMAT] [LOCK <expN>]
[WIDTH <expN>] [FREEZE <field name>] [WINDOW <window
name>]
[FIELDS <field name 1> [/R] [/<column width>] /
<calculated field name 1> = <expression 1> [,<field
name 2> [/R] [/<column width>] / <calculated field
name 2> = <expression 2>]...
```

El formato más simple de esta instrucción es BROWSE. Al ejecutarla, dBASE IV mostrará en pantalla completa los registros de la base de datos apuntando al registro corriente.

En la parte superior aparecerá el menú de opciones, se presentarán los registros en forma tabular de tal forma que puedan verse a la vez más de un registro. Debido a que el ancho de la pantalla puede resultar pequeña para desplegar todos los campos de los registros, dBASE IV presentará los campos que solamente puedan ser mostrados completamente en la pantalla; sin embargo, se podrá acceder a ellos moviéndose con la tecla RETURN de tal forma que se desplace automáticamente la pantalla en forma horizontal.

Desde aquí, puede modificarse cualquier campo, sólo hay que posicionarse en él y podrá ser modificado.

En caso de querer presentar todo un registro para ser editado (igual que con el comando EDIT), presionar la tecla F2 y el registro en el que estamos posicionados será editado; si está presente algún archivo de formato (.fmt), éste será utilizado para la presentación del registro en la pantalla. Para regresar al modo BROWSE, presionar nuevamente la tecla F2.

Para adicionar nuevos registros, basta con posicionarse en el último registro y bajar una vez más el apuntador con la tecla [flecha hacia abajo] y DBASE IV presentará una leyenda preguntando si se quiere adicionar nuevos registros.

OPCIONES

NOINIT permite usar el menú de opciones seleccionado en la última utilización del BROWSE o EDIT, es decir, le indica a BROWSE no inicializar la tabla, seleccionada mediante el menú de opciones, del último BROWSE.

NOFOLLOW es aplicable para archivos indexados que se verán posteriormente.

NOAPPEND deshabilita la posibilidad de adicionar nuevos registros desde este comando.

NOMENU evita el acceso al menú de opciones.

NOEDIT previene cualquier cambio a los datos presentados en la pantalla, sin que implique evitar la adición o el borrado de ellos.

NODELETE previene el borrado de registros con CTRL-U.

NOCLEAR conserva la imagen de la pantalla después de haber terminado la instrucción.

COMPRESS reduce ligeramente el formato de la tabla para permitir que dos líneas más sean desplegadas en la pantalla, de tal forma que una pantalla de 25 líneas que presenta normalmente 17 registros pueda presentar 19.

FORMAT permite utilizar las opciones del comando @...GET que se encuentran especificadas en el archivo de formato (.fmt) activo.

LOCK especifica el número de campos contiguos a la izquierda de la pantalla que permanecerán fijos al moverse dentro de la pantalla del BROWSE. Se pueden fijar el movimiento hacia la izquierda o derecha presionando la F3 o F4 respectivamente. LOCK 0 no fijará ningún campo.

WIDTH define el tamaño de la columna en la cual se desplegarán los campos dentro del BROWSE, en caso de que no se especifique, se tomará el tamaño definido por la estructura de la base. Esta opción no se aplicará para los campos memo, ni los lógicos, y en el caso de que un numérico sea más grande que el tamaño especificado, éste no se mostrará.

FREEZE permite editar una sola columna, mostrando todas las demás sin poderlas modificar. Para deshabilitar esta opción se define el FREEZE sin ningún campo.

WINDOW activa una ventana previamente definida, en la cual será presentada la tabla del BROWSE, quedando la otra parte de la pantalla intacta, de tal forma que al término de la instrucción, ésta desaparecerá, siempre y cuando no se haya dado la opción de NOCLEAR.

FIELDS permite definir cuales y en que orden aparecerán los registros en la tabla del BROWSE. También cuales serán solamente de LECTURA con la opción /R. Se pueden tener campos calculados, los cuales solo podrán leerse; éstos se definen con una asignación (iva = bcosto * 0.15). Aquí también, se puede establecer el ancho con el que será desplegado cada campo, teniendo de 4 a 100 posiciones para campos de tipo caracter y de 8 a 100 para campos de tipo fecha y numéricos.

Ejemplo:

```
USE balumnos
GO TOP
DEFINE WINDOW ventana FROM 2,10 TO 9,60
BROWSE NOAPPEND NOEDIT NODELETE NOCLEAR
NOMENU COMPRESS WINDOW ventana
```

4.- BORRADO DE DATOS

En DBASE IV, antes de borrar un registro es necesario marcarlo para posteriormente borrarlo o bien recuperarlo (quitarle la marca).

Existen dos formas de marcar para borrado los registros, una es a través de los comandos APPEND, EDIT, CHANGE, INSERT y BROWSE, en los cuales sólo hay que presionar las teclas CTRL-U estando posicionado el registro que se quiere borrar. Para quitar esta marca, presionar nuevamente CTRL-U.

La otra forma es a través del comando DELETE, el cual marca el o los registros que se deseen borrar.

SINTAXIS

```
DELETE [<scope>] [FOR <condition>] [WHILE <condition>]
```

A menos que se defina un <scope> o las opciones WHILE o FOR, el comando actuará sólo sobre el registro en que se encuentra el apuntador.

Ejemplo:

```
.USE balumnos
.DELETED
```

En este caso como al abrir la base de datos, se posiciona siempre en el primer registro, éste será el que será marcado por la instrucción DELETE:

# Record	bnum_alumn	bapellidos	bnombres
* 1			
2			
3			
4			
////////////////// EOF ////////////////////			

DBASE muestra un asterisco para indicar los registros que están marcados para borrar, y la palabra DEL cuando se usan los comandos de pantalla completa mencionados anteriormente (APPEND, etc.).

Sin embargo, aunque un registro esté marcado, seguirá apareciendo cada vez que se accese, por lo que la finalidad de esta marca es simular un borrado lógico, de tal forma que en caso de quererse recuperar, se logre quitando la marca de borrado con el comando RECALL o, en el caso de los comandos de pantalla completa, presionando nuevamente CTRL-U.

Para que sean ignorados todos los registros marcados, dando la apariencia de que ya no existen, es necesario habilitar el siguiente comando:

```
SET DELETED OFF/ON
```

Cuando este comando está activo (ON), DBASE ignora los registros marcados para borrado, por lo cual dará la impresión de que ya no existen; si se quieren recuperar, es necesario que esta opción esté apagada (OFF) y se podrá realizar con el siguiente comando:

```
RECALL [<scope>] [FOR <condition>] [WHILE <condition>]
```

Esta instrucción actuará sólo sobre el registro corriente, a menos que se indique lo contrario a través de las opciones <scope> FOR y WHILE.

Existe una función que permite, desde el indicador punto, saber si el registro se encuentra borrado:

```
DELETED ()
```

la cual será verdadera si el registro está marcado, y falsa si no lo está.

Para borrar físicamente del archivo los registros marcados, es necesario que la opción DELETED esté OFF y se realiza con el siguiente comando:

```
PACK
```

Es importante señalar que una vez dada esta instrucción ya no será posible recuperar algún registro.

Por otra parte, existe el comando ZAP el cual borrará todos los registros de la base de datos, estando o no marcados para borrado. Esta opción manda un mensaje para confirmar dicha acción.

Ejemplos:

```
.USE alumnos
.DELETE ALL FOR brum_alum = 0
.LIST
.SET DELETED ON
.LIST
.RECALL ALL
.SET DELETED OFF
.RECALL ALL
.LIST
.PACK
.LIST
.ZAP
```

III.- Métodos de clasificación de la Información.

1.- Ordenamientos.

La clasificación de datos, es un proceso muy común en cualquier sistema de información, por lo cual dBaseIV cuenta con dos formas de llevar a cabo esta función.

La primera que trataremos se realiza con la instrucción SORT.

SINTAXIS

```
SORT TO <filename> ON <field1> [/A] [/C] [/D] [,<field2>
[/A] [/C] [/D]...] [ASCENDING]/[DESCENDING] [<scope>]
[FOR<condition>] [WHILE<condition>]
```

La instrucción SORT crea una nueva base de datos, en la cual los registros de la base de datos activa, son posicionados en orden ascendente o descendente dependiendo de como se haya especificado.

Los campos definidos en esta instrucción, no pueden ser de tipo memo o lógicos. Se pueden ordenar sobre 10 campos como máximo y el archivo definido en la sintaxis no puede ser el mismo que está activo.

Es importante señalar que los registros en la nueva base de datos, no necesariamente deberán tener el mismo número de registro que en la base original, ya que al ser copiados en el orden que les corresponden, éste puede variar al que tuvieron inicialmente cuando fueron dados de alta.

La opción /C causa que la instrucción no diferencie entre mayúsculas y minúsculas, las opciones /A y /D, ascendente y descendente respectivamente; se aplican para cada campo, si ésta no se especifica se asume por omisión /A. Si se quiere combinar las opciones, se tienen que indicar sólo por una diagonal; vgr. /DC.

```
.USE alumnos  
.SORT TO ordenado ON bnum_alumn
```

AREA 1		DISCO	
alumnos.dbf		ordenado.dbf	
Rec	bnum_alumn	Rec	bnum_alumn
1	35	1	12
2	56	2	15
3	12	3	35
4	45	4	45
5	15	5	56

Como puede apreciarse en la figura anterior, los registros fueron copiados a una nueva base, la cual queda "CERRADA", desde la base en uso. De tal manera que si quieren verse los registros clasificados, tendrá que "ABRIRSE" la base ordenada, en este caso "ordenado".

2.- Indexados

El procedimiento de clasificación a través del SORT, resulta poco práctico si tenemos un ambiente en el cual los las bases de datos se actualizan constantemente. Por ejemplo, supongamos que ya se clasificó la base original, quedando la clasificada en la base ORDENADO. Si utilizamos ahora la base ORDENADO, los datos ya estarán clasificados; sin embargo, al agregar un nuevo registro, éste quedará posiblemente fuera de orden, lo que implicará nuevamente realizar el ordenamiento.

```
.USE ordenado  
.APPEND BLANK  
.REPLACE bnum_alumn WITH 13
```

ordenado.dbf

Rec	bnum_alumn	
1	12	
2	15	
3	35	
4	45	
5	56	
6	13	

registro nuevo

Para solucionar este problema, tenemos la alternativa de los archivos indexados, los cuales tendrán la extensión .ndx.

Estos archivos, son archivos que contienen dos campos: uno es el campo llave por el cual se desea tener clasificada la base; y el otro es el número del registro que guarda dicho registro en la base de datos. Cabe hacer mención que, éste no está a disposición del operador, es un archivo que le sirve a dBASEIV para acceder la base de datos, cuando está activo.

La forma de crear este archivo es por medio del siguiente comando:

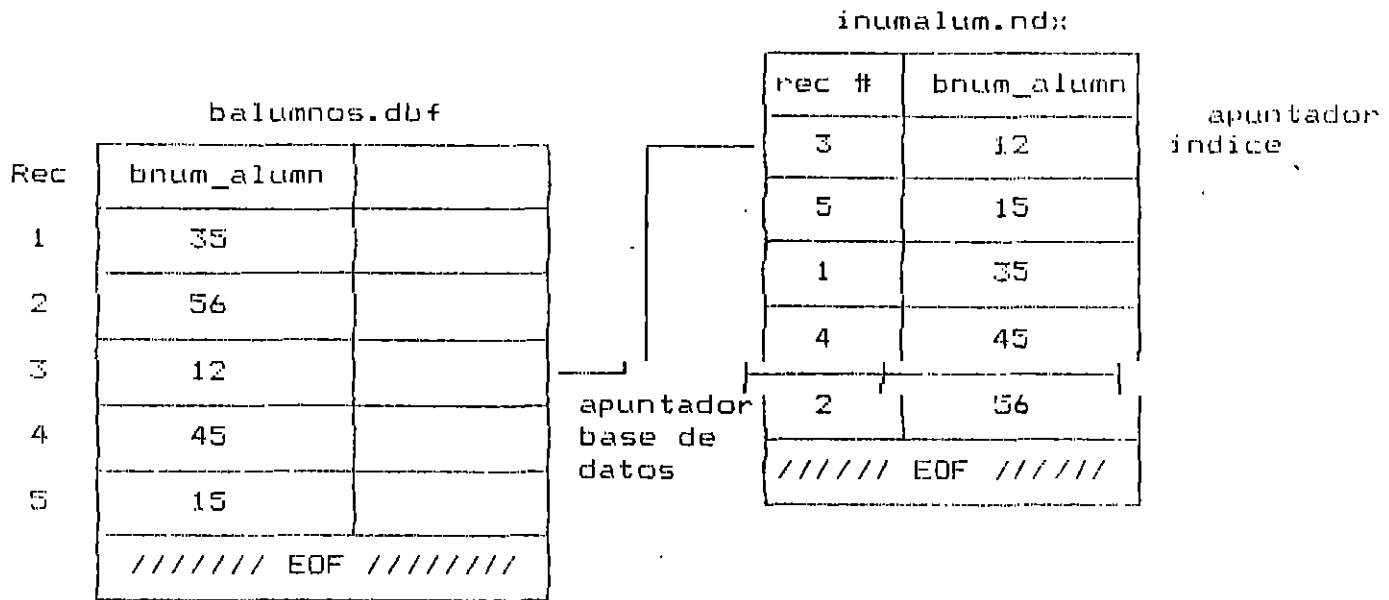
SINTAXIS

```
INDEX ON <key expression> TO <.ndx filename>/ TAG
<tagname> [OF <.mdx filename>] [UNIQUE] [DECENDING]
```

Donde <key expression> representa el campo, que puede estar incluido en una expresión, que será tomado como llave para la clasificación.

Ejemplo:

```
.INDEX ON bnum_alumn TO inumalum
```



Otra forma de crear éstos, es desde la definición de la base. Cuando se crea una base de datos con la instrucción CREATE, uno de los campos que se solicita es INDEX. Esto quiere decir que si ponemos la opción "Y" en el campo, se generará un archivo de INDICES con ese campo como llave. Sin embargo, existe la posibilidad de poner la misma opción en varios campos, esto no implicará que se creará un archivo de índices con varias llaves, sino que se generarán varios archivos de índices, uno para cada campo marcado; éstos archivos se guardarán en uno global de índices, el cual tendrá la extensión .ndx, y a cada índice contenido en éste, se le denominará en la sintaxis del comando como un archivo de TAG.

RECUPERACION DE LA INFORMACION

dBASE permite recuperar la información a través de diferentes mecanismos como son:

- POSICIONAMIENTO DIRECTO
- POSICIONAMIENTO A TRAVES DE INDICES
- FILTROS

El posicionamiento directo es a través de la instrucción

```
. GO <# registro>  
      ó  
. <# registro>
```

También es posible mover el apuntador de la base de datos con las instrucciones :

```
. SKIP <N>
```

donde <N> es el número de registros que se desea saltar, ya sea hacia adelante con un número positivo o hacia atrás con un negativo. En caso de no especificar cuantos registros se desea saltar por omisión brinca un registro hacia adelante.

Las instrucciones . GO TOP
. GO BOTTOM

Posicionan al apuntador al principio y al final de la base respectivamente.

El posicionamiento a través de índices se realiza con archivos indexados, (ver capítulo iii), utilizando para ello la asignación directa o indirecta de variables.

ASIGNACION INDIRECTA DE VARIABLES

ACCEPT [<prompt>] TO <memvar>

INPUT [<prompt>] TO <memvar>

WAIT [<prompt>] [TO <memvar>]

@ <row>, <col> GET <variable> [PICTURE <template>]
[RANGE <lower bound>, <upper bound>]

@ <row>, <col> SAY <expression> [PICTURE <template>]

@ <row>, <col> TO <row2>, <col2> [DOUBLE]

@ <row>, <col> CLEAR [TO <row2>, <col2>]

READ

SAVE TO <archivo.mem> [ALL LIKE <comodin>] [EXCEPT <comodin>]

RESTORE FROM <archivo.mem> [ADDITIVE]

RELEASE [ALL LIKE <comodin>] [EXCEPT <comodin>]

REPLACE

Otra instrucción que permite cambiar el contenido de un registro es: REPLACE, con la diferencia que ésta no presenta una pantalla para la edición, sino que directamente debemos definir cuál va a ser el nuevo contenido de cada campo que deseemos modificar.

SINTAXIS

```
REPLACE <field> WITH <exp.> [ADDITIVE] [, <field> WITH
<exp.> [ADDITIVE]][<scope>] [FOR <condition>] [WHILE
<condition>]
```

Esta instrucción afecta al registro corriente de la base de datos, a menos que se defina una opción de scope, FOR o WHILE que la modifique.

Ejemplo:

```
REPLACE bapellidos WITH "LOPEZ"
```

En este caso, el registro donde se encuentre posicionado el apuntador de la base abierta será el que reemplace su campo bapellidos por "LOPEZ". Sin embargo:

```
REPLACE ALL bapellidos WITH "LOPEZ"
```

reemplazará "LOPEZ" en el campo bapellidos para todos los registros de la base.

La cláusula ADDITIVE le permite construir un campo memo desde varias cadenas de caracteres. Esta opción es ignorada si no se tiene un campo memo.

En el caso de la expresión, pueden sólo utilizarse campos de igual tipo, con los operadores respectivos. Ejemplo:

```
REPLACE ALL bcosto WITH bcosto*1.15
```

BROWSE

La instrucción BROWSE nos permite modificar, borrar y adicionar registros a un archivo de base de datos, a través de una pantalla en la cual se puede visualizar más de un registro al mismo tiempo.

SINTAXIS

```
BROWSE [NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU] [NOEDIT]
[NODELETE] [NOCLEAR] [COMPRESS] [FORMAT] [LOCK <expN>]
[WIDTH <expN>] [FREEZE <field name>] [WINDOW <window
name>]
[FIELDS <field name 1> [/R] [/<column width>] /
<calculated field name 1> = <expression 1> [, <field
name 2> [/R] [/<column width>] / <calculated field
name 2> = <expression 2>]...]
```

El formato más simple de esta instrucción es BROWSE. Al ejecutarla, DBASE IV mostrará en pantalla completa los registros de la base de datos apuntando al registro corriente.

En la parte superior aparecerá el menú de opciones, se presentarán los registros en forma tabular de tal forma que puedan verse a la vez más de un registro. Debido a que el ancho de la pantalla puede resultar pequeña para desplegar todos los campos de los registros, DBASE IV presentará los campos que solamente puedan ser mostrados completamente en la pantalla; sin embargo, se podrá acceder a ellos moviéndose con la tecla RETURN de tal forma que se desplace automáticamente la pantalla en forma horizontal.

Desde aquí, puede modificarse cualquier campo, sólo hay que posicionarse en él y podrá ser modificado.

En caso de querer presentar todo un registro para ser editado (igual que con el comando EDIT), presionar la tecla F2 y el registro en el que estamos posicionados será editado; si está presente algún archivo de formato (.fmt), éste será utilizado para la presentación del registro en la pantalla. Para regresar al modo BROWSE, presionar nuevamente la tecla F2.

Para adicionar nuevos registros, basta con posicionarse en el último registro y bajar una vez más el apuntador con la tecla [flecha hacia abajo] y DBASE IV presentará una leyenda preguntando si se quiere adicionar nuevos registros.

OPCIONES

NOINIT permite usar el menú de opciones seleccionado en la última utilización del BROWSE o EDIT, es decir, le indica a BROWSE no inicializar la tabla, seleccionada mediante el menú de opciones, del último BROWSE.

NOFOLLOW es aplicable para archivos indexados que se verán posteriormente.

NOAPPEND deshabilita la posibilidad de adicionar nuevos registros desde este comando.

NOMENU evita el acceso al menú de opciones.

NOEDIT previene cualquier cambio a los datos presentados en la pantalla, sin que implique evitar la adición o el borrado de ellos.

NODELETE previene el borrado de registros con CTRL-U.

NOCLEAR conserva la imagen de la pantalla después de haber terminado la instrucción.

COMPRESS reduce ligeramente el formato de la tabla para permitir que dos líneas más sean desplegadas en la pantalla, de tal forma que una pantalla de 25 líneas que presenta normalmente 17 registros pueda presentar 19.

FORMAT permite utilizar las opciones del comando @...GET que se encuentran especificadas en el archivo de formato (.fmt) activo.

LOCK especifica el número de campos contiguos a la izquierda de la pantalla que permanecerán fijos al moverse dentro de la pantalla del BROWSE. Se pueden fijar el movimiento hacia la izquierda o derecha presionando la tecla F3 o F4 respectivamente. LOCK 0 no fijará ningún campo.

WIDTH define el tamaño de la columna en la cual se desplegarán los campos dentro del BROWSE, en caso de que no se especifique, se tomará el tamaño definido por la estructura de la base. Esta opción no se aplicará para los campos memo, ni los lógicos, y en el caso de que un número sea más grande que el tamaño especificado, éste no se mostrará.

FREEZE permite editar una sola columna, mostrando todas las demás sin poderlas modificar. Para deshabilitar esta opción se define el FREEZE sin ningún campo.

WINDOW activa una ventana previamente definida, en la cual será presentada la tabla del BROWSE, quedando la otra parte de la pantalla intacta, de tal forma que al término de la instrucción, ésta desaparecerá, siempre y cuando no se haya dado la opción de NOCLEAR.

FIELDS permite definir cuales y en que orden aparecerán los registros en la tabla del BROWSE. También cuales serán solamente de LECTURA con la opción /R. Se pueden tener campos calculados, los cuales solo podrán leerse; éstos se definen con una asignación (iva = bcosto * 0.15). Aquí también, se puede establecer el ancho con el que será desplegado cada campo, teniendo de 4 a 100 posiciones para campos de tipo caracter y de 8 a 100 para campos de tipo fecha y numéricos.

Ejemplo:

```
USE balumnos
GO TOP
DEFINE WINDOW ventana FROM 2,10 TO 9,60
BROWSE NOAPPEND NOEDIT NODELETE NOCLEAR
NOMENU COMPRESS WINDOW ventana
```

BORRADO DE DATOS

En DBASE IV, antes de borrar un registro es necesario marcarlo para posteriormente borrarlo o bien recuperarlo (quitarle la marca).

Existen dos formas de marcar para borrado los registros, una es a través de los comandos APPEND, EDIT, CHANGE, INSERT y BROWSE, en los cuales sólo hay que presionar las teclas CTRL-U estando posicionado el registro que se quiere borrar. Para quitar esta marca, presionar nuevamente CTRL-U.

La otra forma es a través del comando DELETE, el cual marca el o los registros que se deseen borrar.

SINTAXIS

DELETE [<scope>] [FOR <condition>] [WHILE <condition>]

A menos que se defina un <scope> o las opciones WHILE o FOR, el comando actuará sólo sobre el registro en que se encuentra el apuntador.

Ejemplo:

```
.USE balumnos
.DELE
```

En este caso como al abrir la base de datos, se posiciona siempre en el primer registro, éste será el que será marcado por la instrucción DELETE:

# Record	bnum_alumn	bapellidos	bnombres
▶ * 1			
2			
3			
4			
////////////////// EOF ////////////////////			

DBASE muestra un asterisco para indicar los registros que están marcados para borrar, y la palabra DEL cuando se usan los comandos de pantalla completa mencionados anteriormente (APPEND, etc.).

Sin embargo, aunque un registro esté marcado, seguirá apareciendo cada vez que se accese, por lo que la finalidad de esta marca es simular un borrado lógico, de tal forma que en caso de quererse recuperar, se logre quitando la marca de borrado con el comando RECALL o, en el caso de los comandos de pantalla completa, presionando nuevamente CTRL-U.

Para que sean ignorados todos los registros marcados, dando la apariencia de que ya no existen, es necesario habilitar el siguiente comando:

```
SET DELETED OFF/ON
```

Cuando este comando está activo (ON), DBASE ignora los registros marcados para borrado, por lo cual dará la impresión de que ya no existen; si se quieren recuperar, es necesario que esta opción esté apagada (OFF) y se podrá realizar con el siguiente comando:

```
RECALL [<scope>] [FOR <condition>] [WHILE <condition>]
```

Esta instrucción actuará sólo sobre el registro corriente, a menos que se indique lo contrario a través de las opciones <scope> FOR y WHILE.

Existe una función que permite, desde el indicador punto, saber si el registro se encuentra borrado:

```
DELETED ()
```

la cual será verdadera si el registro está marcado, y falsa si no lo está.

Para borrar físicamente del archivo los registros marcados, es necesario que la opción DELETED esté OFF y se realiza con el siguiente comando:

```
PACK
```

Es importante señalar que una vez dada esta instrucción ya no será posible recuperar algún registro.

Por otra parte, existe el comando ZAP el cual borrará todos los registros de la base de datos, estando o no marcados para borrado. Esta opción manda un mensaje para confirmar dicha acción.

Ejemplos:

```
.USE balumnos
.DELETE ALL FOR bnum_alumn = 0
.LIST
.SET DELETED ON
.LIST
.RECALL ALL
.LIST
.SET DELETED OFF
.RECALL ALL
.LIST
.PACK
.LIST
.ZAP
.LIST
```


EMISION DE REPORTES Y ETIQUETAS

Una de las funciones más importantes en el uso de las bases de datos, es el poder obtener la información que tenemos almacenada, de una forma que tenga ma's presentación que la utilizada por comandos como el LIST o DISPLAY, que nos muestran la información sin tener la opción de poner encabezados, pies de página, numeración en las hojas, etc. Y para lograr la presentación en nuestros reportes se tiene la opción de tener nuestra información en forma de reporte o de etiquetas.

La sintaxis del comando REPORT es la siguiente :

```
REPORT FORM <nombre del reporte .FRM> / ? [<comodin>]
      [FOR/WHILE <expresion>] [NOEJECT] [PLAIN]
      [HEADING <cadena de caracteres>] [TO PRINT]
      [TO FILE <nombre del archivo .TXT>] [SUMMARY]
```

Para la creación o modificación de reportes, es necesario dar la instrucción CREATE REPORT <reporte> o MODIFY REPORT <reporte>, y nos muestra un menú con siete opciones:

```
Layout  Fields  Bands  Words  Go To  Print  Exit  3:37:24
[.....▼1.....▼2.....▼3.....▼4.....▼5.....▼6.....▼7.....▼]
Page    Header  Band-----
Report  Intro   Band-----
Detail                                     Band-----
Report  Summary Band-----
Page    Footer  Band-----
```

```
Report  ||C:baseEPAGEND      ||Band 1/5      ||File:Agenda  ||      CapsIns
        Add field:F5   Select:F6     Move:F7      Copy:F8     Size:Shift-F7
```


COMANDOS SET

Para los comandos SET tiene que seleccionar ON u OFF, el default de los comandos es el escrito en mayúsculas.

SET ALTERNATE TO [<filename>] [ADDITIVE]

Ahora soporta la opción ADDITIVE que permite extender archivos de texto. La opción ADDITIVE agrga salidas en el fin de un archivo ALTERNATE en vez de crear un archivo nuevo.

SET AUTOSAVE on/OFF

Controla el contenido de los buffers de memoria que son escritos en disco, SET AUTOSAVE ON provoca que los buffers de memoria sean escritos automaticamente en disco. SET AUTOSAVE OFF requiere que las bases de datos y sus archivos relacionados sean cerrados antes de concluir; el contenido de los buffers es escrito en disco.

SET BELL TO [<frequency>,<duration>]

Permite modificar el tono y duración del sonido.

SET BLOCKSIZE TO <expN>

Determina el tamaño de bloque de creación de archivos index y archivos memo. <expN> es el rango de 1 a 32, el espacio actual del bloque es 512.

SET BORDER TO [SINGLE/DOUBLE/PANEL/NONE/<border definition string>]

Permite hacer cambios en el tipo de caracter definido en bordos de ventanas, cajas, menús y líneas.

SET CARRY TO [<field list> [ADDITIVE]]

Selecciona los campos en donde van a ser copiados los datos usando APPEND, INSERT y BROWSE.

SET CLOCK on/OFF

Determina si el sistema de reloj va ha ser desplegado en la pantalla. El sistemas de reloj siempre es deplegado cuando usamos BROWSE, el Control Central, etiquetas, programas, Queries o reportes.

SET CLOCK TO [<row>,<column>]

Indica en que parte va a ser desplegado el sistema de reloj cuando usemos SET CLOCK ON.

SET COLOR OF HIGHLIGHT/MESSAGES/TITLES/BOX/INFORMATION/NORMAL/FIELDS TO [<color attribute>]

Permite seleccionar colores usados en Texto, Cajas y Encabezados.

SET CURRENCY LEFT/right

Determina si el simbolo de la moneda va a aparecer en el lado izquierdo o derecho de la cantidad.

SET CURRENCY TO [<expC>]

Cambia el simbolo usado para la moneda. El caracter por default es el simbolo dolar (\$) excepto si es una palabra arriba de 9 caracteres.

SET DATE [TO] AMERICAN/ansi/british/french/german/italian/japan/usa/mdy/dmy/ymd

Ahora soporta gran variedad de formatos de fecha.

CREACION DE PANTALLAS

Usted puede desplegar en la pantalla informaci3n de una base de datos con el BROWSE o con la pantalla de EDIT. El BROWSE despliega una pantalla estandar en la que cada renglon es un registro. La pantalla de EDIT es de mucha ayuda ya que es muy flexible para desplegar o modificar informaci3n.

Para dise1ar un formato de pantalla, usted puede ordenar los campos en la pantalla, adherir caja y lneas, omitir campos o adherir campos calculados.

Cada formato de pantalla es almacenado en tres diferentes archivos que son :

El archivo .SCR

Contiene el formato original del dise1o de la pantalla.

El archivo .FMT

Contiene el c3digo de dBASE IV, mientras el archivo .FM0 es la versi3n compilada del archivo .FMT.

Cuando usted grabe un archivo .SCR en otro drive u otro directorio tendra los archivos correspondientes .FMT o .FM0 del archivo .SCR.

MENU LAYOUT

```
+-----+
| Quick layout |
+-----+
| > Box        |
| > Line       |
+-----+
| Use different database file or view |
| Edit description of form            |
| Save this form                      |
+-----+
```

Para abrir el menu Layout u otro menu que se encuentran en la parte superior de la pantalla, es presionando la tecla <Alt> y la primer letra del menu que desee abrir en este caso es Layout.

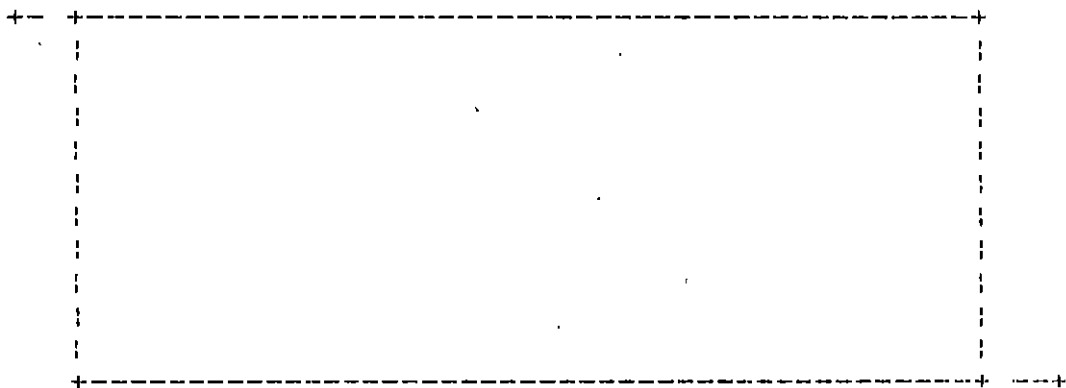
QUICK LAYOUT

Esta opción es usada para cargar todos los campos de la base de datos en el formato de pantalla.

BOX

Esta opción sirve para diseñar cajas y líneas. Primero se selecciona el tipo de borde de la línea que pueden ser línea simple, línea doble o un caracter en especial; Una vez seleccionado el tipo de borde de línea, presione RETURN en el punto en donde quiere que empiece su caja, despues posicione el cursor en la ubicación en que quiere que termine su cuadro como se muestra en la siguiente figura.

Primer RETURN



Segundo RETURN

Creando una caja con línea simple

LINE

Para crear una línea seleccione el tipo de línea que puede ser línea simple, línea doble o un caracter en especial, ya que seleccione el tipo de línea presione RETURN en el punto inicial donde quiere que comience la línea y otro RETURN en donde desee que termine su línea como se muestra en la siguiente figura.

Creando linea simple

Usando diferente Archivo de Base de Datos

Cuando se crea un formato de pantalla, es usualmente relacionada con una base de datos, con la opción :

Use Different Database file

Usted puede usar una fuente diferente de datos. Si usted tiene archivos de bases de datos con estructuras similares y desea modificar un formato de pantalla con una diferente base de datos aplicada, usted puede cargar el formato y substituir una base de datos diferentes, asi puede modificarlo y salvarlo con un nuevo nombre.

Save This Form

Esta opción es para salvar la pantalla que esta usted creando o modificando, puede ser con el mismo nombre o un nuevo nombre.

MENU DE CAMPOS

Las opciones de este menú nos sirven para adherir, borrar, modificar campos e insertar variables de memoria en el formato;

Add Field

Usamos esta opción para adherir campos en la pantalla (formato); Primero posicionamos el cursor en donde desea ubicar el campo. Cuando selecciona esta opción le muestra dos columnas; La primer columna contiene los nombre de los campos de la base de datos en uso. La segunda columna muestra los nombres de los campos usted puede seleccionar la opción especial <CREATE> para crear un nuevo campo Tambien puede agregar un nuevo campo en la posición del cursor presionando <F5> .

La información del menú de esta opción :

<Field Name, Type, Length y Decimals>

Vienen dentro de la base de datos; Antes de finalizar la modificación de la información presione <Ctrl>-<End> el campo es ubicado en la posición del cursor.

Adding a New Field

Para crear un nuevo campo seleccionamos la opción <CREATE> .

Specifying Fields Templates

La opción template permite desplegar los datos con el tipo de caracteres que se desee. Por ejemplo .

Si usted desea imprimir sus números telefónicos con una mascarilla como : (999) 999-99-99 para eso use este formato, entonces los paréntesis, espacio en blanco y guiones aparecerán en la pantalla, solo números son aceptados como datos.

Specifying Picture Function

Esta operación indica los atributos que usted puede asignar a un campo en particular. Una lista de los atributos aparecerán en la pantalla y usted podrá seleccionar una o más funciones, para finalizar y salvar su selección presione <Ctrl>-<End>.

INTRODUCCION A LA PROGRAMACION EN dBASE IV

Una de las características más notables de dBASE IV, es la capacidad de poder desarrollar programas en un lenguaje de alto nivel y de tipo estructurado, denominado DBASE, el cual nos permite desarrollar lo que se conoce como Aplicaciones o Sistemas de Información basados en poderosos comandos y utilizando además las diferentes herramientas que nos provee el paquete, como son :

Creación de Pantallas, Menus, Reportes, Etiquetas, Consultas, etc. de tal manera que combinandolas con los comandos que ya conocemos de tipo Punto y los comandos propios de programación nos da como resultado los Sistemas como pueden ser:

- Control de Agenda o Directorio Telefónico
- Control de Inventarios
- Control de Cursos
- etc.

Los elementos que debe tener un programa son :

1) INICIALIZACION DE PARAMETROS

```
SET TALK OFF
SET BELL ON
SET CONFIRM ON
etc.
```

2) INICIALIZACION DE BASES DE DATOS O VISTAS

```
SELECT 1
USE INVENTARIO
SELECT 2
USE CATALOGO INDEX LLAVE
etc.
```

```
o
SET VIEW TO INVENTARIO
```

3) INICIALIZACION DE VARIABLES

```
STORE 0          TO contador, ren, col
STORE SPACE(20) TO nombre
STORE .F.        TO bandera
STORE DATE()    TO fecha
```

4) PROGRAMA

En esta parte se tiene que utilizar comandos como

```
DO WHILE <condicion>    %% Repite mientras se cumpla
                        %% la condicion
    <comandos>
ENDDO                    %% Finaliza el ciclo
```

```
IF <Condicion>    %% Si se cumple la condición
                  %% realiza estos comandos
    <comandos>

ELSE              %% si no se cumple la condición
                  %% realiza estos otros comandos
    <comandos>

ENDIF            %% finaliza la pregunta

DO CASE          %% Selecciona una opción de un menú

    CASE opcion = 1
* Opcion de altas
        APPEND
    CASE opcion = 2
* Opcion de bajas
        PACK
    CASE opcion = 3
* Opcion de cambios
        EDIT
    CASE opcion = 4
* Opcion de consultas
        BROWSE
    CASE opcion = 5
* Opcion de reportes
        REPORT FORM <reporte> TO PRINT
    OTHERWISE      %% En caso que opcion no sea de 1 a 5
        <comandos>
ENDDO            %% Finaliza la selección de las opciones
```

Un ejemplo completo de una aplicación para el control de una agenda, es el listado siguiente, que fué generado con el centro de control de dBASE IV, con la opción de APPLICATIONS

```

*****
* Program.....: C:AGENDA
* Author.....: COINTO BARRERA
* Date.....: 2-24-89
* Notice.....: SISTEMAS DE INFORMACION DISTRIBUIDA, S.C.
* dBASE Ver....: 1.0
* Generated by.: APGEN version 1.0
* Description..: DATOS PERSONALIZADOS

* Notes.....:
*****

```

```

SET CONSOLE OFF
IF TYPE("gn_apgen") = "U" && We were not called from another APGEN program
  CLEAR ALL
  CLEAR WINDOW
  CLOSE ALL
  gn_apgen = 1
ELSE
  gn_apgen = gn_apgen + 1
  PRIVATE gc_bell, gc_carry, gc_clock, gc_century, gc_confirm, gc_deli,;
           gc_escape, gc_instruc, gc_safety, gc_status, gc_score, gc_talk
ENDIF

```

```

*-- Window for pause message box (ON ERROR)
DEFINE WINDOW Pause FROM 15,00 TO 19,79 DOUBLE
ON ERROR DO PAUSE WITH [Error occurred on line ]+LTRIM(STR(LINE())) +[ of pro
ON KEY LABEL F1 DO quickhlp

```

```

*-- Store initial SETs to variables
gc_bell   =SET("BELL")
gc_carry  =SET("CARRY")
gc_clock  =SET("CLOCK")
gc_century=SET("CENTURY")
gc_confirm=SET("CONFIRM")
gc_deli   =SET("DELIMITERS")
gc_escape =SET("ESCAPE")
gc_instruc=SET("INSTRUCT")
gc_safety =SET("SAFETY")
gc_status =SET("STATUS")
gc_score  =SET("SCOREBOARD")
gc_talk   =SET("TALK")

```

```

SET CLOCK OFF
SET COLOR TO
CLEAR
SET CONSOLE ON

```

```

*-- Sets for application
SET BELL OFF
SET CARRY OFF
SET CENTURY OFF
SET CONFIRM OFF

```

```

IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
SET CONFIRM ON
@ 3,1 SAY "Enter the key expression to search for:"
IF ln_type=3
  @ 4,1 GET ld_date PICT "@D"
ELSE
  IF ln_type=2
    @ 4,1 GET ln_num PICT "#####"
  ELSE
    @ 4,1 GET lc_lni
  ENDIF
ENDIF
READ
SET CONFIRM OFF
IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
  lc_temp=IIF(ln_type=1, "TRIM(lc_lni)", IIF(ln_type=2, "ln_num", "ld_da
  SEEK &lc_temp.
ENDIF
ENDIF
RELEASE WINDOWS Posit2
CASE lc_option='4'
  *-- Goto
  ln_type=1
  DEFINE POPUP Posit2 FROM 8,30
  DEFINE BAR 1 OF Posit2 PROMPT " GOTO:" SKIP
  DEFINE BAR 2 OF Posit2 PROMPT REPLICATE(CHR(196),10) SKIP
  DEFINE BAR 3 OF Posit2 PROMPT " TOP" MESSAGE "GOTO Top of File"
  DEFINE BAR 4 OF Posit2 PROMPT " BOTTOM" MESSAGE "GOTO Bottom of File"
  DEFINE BAR 5 OF Posit2 PROMPT " Record # " MESSAGE "GOTO A Specific Reco
ON SELECTION POPUP Posit2 DO get_sele
ACTIVATE POPUP posit2
ln_type = gn_send
IF LASTKEY() <> 27
  IF ln_type=5
    DEFINE WINDOW Posit2 FROM 8,26 TO 13,50 DOUBLE
    ACTIVATE WINDOW Posit2
    ln_num=0
    @ 3,1 SAY "Max. Record # = "+LTRIM(STR(RECCOUNT()))
    @ 1,1 SAY "Record to GOTO" GET ln_num PICT "#####" RANGE 1,RECCOU
    READ
    IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
      GOTO ln_num
    ENDIF
    RELEASE WINDOWS Posit2
  ELSE
    lc_temp=IIF(ln_type=3, "TOP", "BOTTOM")
    GOTO &lc_temp.
  ENDIF
ENDIF
CASE lc_option='5'
  *-- Locate
  DEFINE WINDOW Posit2 FROM 8,16 TO 14,66 DOUBLE
  ACTIVATE WINDOW Posit2

```

```

lc_option='0'      && main option ie. Seek, Goto and Locate
*-- Scope ie. ALL, REST, NEXT <n>
STORE SPACE(10) TO lc_scp
*-- 1 = Character SEEK, 2 = For clause, 3 = While clause
STORE SPACE(40) TO lc_ln1, lc_ln2, lc_ln3
lc_temp=""
@ 0,00 SAY "Index order: "+IIF(" "=ORDER(),"Database is in natural order",ORDER
@ 1,00 SAY "Listed below are the first 16 fields."
lc_temp=REPLICATE(CHR(196),19)
@ 2,0 SAY CHR(218)+lc_temp+CHR(194)+lc_temp+CHR(194)+lc_temp+CHR(194)+lc_temp
ln_num=240
DO WHILE ln_num < 560
  lc_temp=FIELD( (ln_num-240)/20 +1)
  @ (ln_num/80),MOD(ln_num,80) SAY CHR(179)+;
lc_temp+SPACE(11-LEN(lc_temp))+;
SUBSTR("= Char = Date = Logic = Num = Float = Memo ",;
AT(TYPE(lc_temp),"CDLNFMU")*8-7,8)
  ln_num=ln_num+20
ENDDO
ln_num=1

DEFINE POPUP Posit1 FROM 8,30
DEFINE BAR 1 OF Posit1 PROMPT " Position by " SKIP
DEFINE BAR 2 OF Posit1 PROMPT REPLICATE(CHR(196),15) SKIP
DEFINE BAR 3 OF Posit1 PROMPT " SEEK Record" MESSAGE "Search on index key" SKI
DEFINE BAR 4 OF Posit1 PROMPT " GOTO Record" MESSAGE "Position to specific rec
DEFINE BAR 5 OF Posit1 PROMPT " LOCATE Record " MESSAGE "Locate record for con
DEFINE BAR 6 OF Posit1 PROMPT " Return" MESSAGE "Return without positioning"
ON SELECTION POPUP Posit1 DO get_sele

SET CONFIRM ON
DO WHILE lc_option='0'
  ACTIVATE POPUP Posit1
  lc_option = ltrim(str(gn_send)) && for popup
  IF LASTKEY() = 27 .OR. lc_option="6"
    GOTO ln_rec
  EXIT
ENDIF
DO CASE
CASE lc_option='3'
  *-- Seek
  IF LEN(NDX(1))=0 .AND. LEN(MDX(1))=0
    DO Pause WITH "Can't use this option - No index files are open."
  LOOP
ENDIF
ln_type=1
lc_ln1=SPACE(40)
DEFINE WINDOW Posit2 FROM 8,19 TO 15,62 DOUBLE
ACTIVATE WINDOW Posit2
@ 1,1 SAY "Enter the type of expression:" GET ln_type PICT "#" RANGE 1,3
@ 2,1 SAY "(1=character, 2=numeric and 3=date.)"
READ

```

```

PROCEDURE Pause
PARAMETER lc_msg
*-- Parameters : lc_msg = message line
IF TYPE("lc_message")="U"
    gn_error=ERROR()
ENDIF
lc_msg = lc_msg
lc_option='0'
ACTIVATE WINDOW Pause
IF gn_error > 0
    IF TYPE("lc_message")="U"
        @ 0,1 SAY [An error has occurred !! - Error message: ]+MESSAGE()
    ELSE
        @ 0,1 SAY [Error # ]+lc_message
    ENDIF
ENDIF
ENDIF
@ 1,1 SAY lc_msg
WAIT " Press any key to continue..."
DEACTIVATE WINDOW Pause
RETURN

```

```

PROCEDURE quickhlp
*-- If you want to include help for a quickapp uncomment the lines below and
*-- put your help @ say's into the case statements
*ACTIVATE WINDOW desktop
*CLEAR
DO CASE
CASE BAR() = 1
CASE BAR() = 2
CASE BAR() = 3
CASE BAR() = 4
CASE BAR() = 5
CASE BAR() = 6
CASE BAR() = 7
CASE BAR() = 8
ENDCASE
*WAIT
*DEACTIVATE WINDOW desktop
RETURN

```

```

PROCEDURE Position
IF LEN(DBF()) = 0
    DO Pause WITH "Database not in use. "
    RETURN
ENDIF
SET SPACE ON
SET DELIMITERS OFF
ln_type=0          && sublevel selection
ln_rkey=READKEY() && test for ESC or Return
ln_rec=RECNO()    && DBF record number
ln_num=0          && for input of a number
ld_date=DATE()   && for input of a date

```

```

CASE gn_barv = 6
  *-- Run label form agenda
  SET MESSAGE TO 'Pick an option to locate a record or <ESC> for default'
  ACTIVATE WINDOW work
  gn_recno = RECNO()
  DO position
  DEACTIVATE WINDOW work
  STORE 0 TO gn_send, gn_pkey
  lc_toprnt = IIF(gn_recno <> recno(), 'REST ', '')
  ACTIVATE POPUP prntchk
  DO CASE
    CASE gn_send = 4
      lc_toprnt = 'TO PRINT'
    CASE gn_send = 5
      lc_toprnt = 'TO PRINT SAMPLE'
  ENDCASE
  IF .NOT. gn_send = 6
    SET MESSAGE TO 'Printing labels'
    ACTIVATE WINDOW desktop
    SET ESCAPE ON
    ON ERROR DO prnrtry
    LABEL FORM AGENDA &lc_toprnt.
    IF gn_pkey <> 27
      WAIT
    ENDIF
    SET ESCAPE ON
    DEACTIVATE WINDOW desktop
  ENDIF
  GOTO gn_recno
  ON ERROR DO PAUSE WITH [Error occurred on line ]+LTRIM(STR(LINE())) +[ 0
CASE gn_barv = 7
  *-- Reindex agenda
  ACTIVATE WINDOW desktop
  @ 3,0 SAY "Reindexing database AGENDA..."
  @ 4,0
  SET TALK ON
  REINDEX
  GO TOP
  ?
  WAIT
  SET TALK OFF
  DEACTIVATE WINDOW desktop
CASE gn_barv = 8
  DEACTIVATE POPUP
ENDCASE
SET MESSAGE TO
IF gc_status = "OFF"
  SET STATUS ON
ENDIF
SET FORMAT TO
RESTORE SCREEN FROM quick
RETURN

```

```

DO CASE
CASE gn_barv = 1
  *-- Add information
  SET MESSAGE TO 'Appending records to file AGENDA'
  APPEND
CASE gn_barv = 2
  *-- Change information
  SET MESSAGE TO 'Editing file AGENDA'
  EDIT
CASE gn_barv = 3
  *-- Browse information
  SET MESSAGE TO 'Browsing file AGENDA'
  BROWSE FORMAT NOEDIT NOAPPEND NOFOLLOW
CASE gn_barv = 4
  *-- Remove information (Pack file agenda)
  ACTIVATE WINDOW desktop
  @ 2,0 SAY "Packing database AGENDA to REMOVE records marked for deletion"
  @ 3,0
  SET TALK ON
  PACK
  GO TOP
  ?
  WAIT
  SET TALK OFF
  DEACTIVATE WINDOW desktop
CASE gn_barv = 5
  *-- Run report form agenda
  SET MESSAGE TO 'Pick an option to locate a record or <ESC> for default'
  ACTIVATE WINDOW work
  gn_recno = RECNO()
  DO position
  DEACTIVATE WINDOW work
  lc_toprnt = IIF(gn_recno <> recno(), 'REST ', '')
  STORE 0 TO gn_send, gn_pkey
  ACTIVATE POPUP prntchk
  IF gn_send = 4
    lc_toprnt = 'TO PRINT'
    ON ERROR DO prnrtry
  ENDIF
  IF .NOT. gn_send = 6
    SET MESSAGE TO 'Printing report AGENDA'
    ACTIVATE WINDOW desktop
    SET ESCAPE ON
    REPORT FORM AGENDA &lc_toprnt.
    IF gn_pkey <> 27
      WAIT
    ENDIF
    SET ESCAPE ON
    DEACTIVATE WINDOW desktop
  ENDIF
  GOTO gn_recno
  ON ERROR DO PAUSE WITH [Error occurred on line ]+LTRIM(STR(LINE())) +[ o

```


* EOP: C:AGENDA.PRG

```

*****
* Procedures... : C:AGENDA.Prc
* Author..... : COINTO BARRERA
* Date..... : 2-24-89
* Notice..... : SISTEMAS DE INFORMACION DISTRIBUIDA, S.C.
* dBASE Ver... : 1.0
* Generated by. : APGEN version 1.0
* Description.. : DATOS PERSONALIZADOS

```

* Notes..... :

```

*****

```

```

*-- Here is a sample procedure file to show the power of procdures.
*-- This example - Menubox displays a menu heading box with a centered heading

```

PROCEDURE MenuBox

PARAMETER lc_m_name

*-- Parameter lc_m_name - is the title variable for the menu

SET CLOCK OFF

@ 1,0 FILL TO 2,79 COLOR n/n

DO CASE

CASE gc_brdr = "0"

@ 1,0 CLEAR TO 3,79

CASE gc_brdr = "1"

@ 1,0 TO 3,79

CASE gc_brdr = "2"

lc_color = IIF(ISCOLOR(), "gr+/bg", "W+/N")

@ 1,0 TO 3,79 DOUBLE COLOR &lc_color.

ENDCASE

SET CLOCK TO 2,68

@ 2,1 SAY SUBSTR(CDOW(DATE()), 1, 3) + '. ' + DTOC(DATE()) + ' '

@ 2,41 - (LEN(lc_m_name)/2) SAY lc_m_name

lc_color = IIF(ISCOLOR(), "w+/b", "W+/N")

@ 2,1 FILL TO 2,78 COLOR &lc_color.

RETURN

PROCEDURE get_sele

*-- Get the user selection & store BAR into variable

gn_send = BAR() && Variable for print testing

DEACTIVATE POPUP

RETURN

PROCEDURE Action

PARAMETERS bar

*-- Get the user selection & store BAR into variable

gn_barv = bar

SET MESSAGE TO

IF LTRIM(STR(gn_barv)) \$ "123"

*-- Set format file AGENDA for edit/append/browse

SET FORMAT TO AGENDA

ENDIF

SID, S.C.

CURSO : dBASE IV

```
DEFINE BAR 5 OF prntchk PROMPT " Label Sample " MESSAGE "Printer LPT1: with  
DEFINE BAR 6 OF prntchk PROMPT " Return" MESSAGE "Return to Main Menu"  
ON SELECTION POPUP prntchk DO get_sele
```

```
*-- Window to cover work surface during edit, append, etc.
```

```
DEFINE WINDOW work FROM 0,0 TO 21,79 NONE
```

```
*-- Window for area below menu heading & for running reports/labels in
```

```
DEFINE WINDOW desktop FROM 4,0 TO 21,79 NONE
```

```
DEFINE WINDOW printemp FROM 10,25 TO 15,56
```

```
*-- Display heading centered on the screen.
```

```
DO menubox WITH lc_heading
```

```
*-- Show the menu so we don't get a flash if the user hits arrow keys or ESC
```

```
SHOW POPUP quick
```

```
SAVE SCREEN TO quick
```

```
*-- Display Quickapp menu centered on the screen.
```

```
DO WHILE gn_barv <> 8 && Prevent user from exiting with arrow keys or ESC
```

```
ACTIVATE POPUP quick
```

```
ENDDO
```

```
* Restore SET environment the best we can
```

```
SET BELL &gc_bell.
```

```
SET CARRY &gc_carry.
```

```
SET CLOCK TO
```

```
SET CLOCK &gc_clock.
```

```
SET CENTURY &gc_century.
```

```
SET CONFIRM &gc_confirm.
```

```
SET DELIMITERS &gc_deli.
```

```
SET ESCAPE &gc_escape.
```

```
SET INSTRUCT &gc_instruc.
```

```
SET STATUS &gc_status.
```

```
SET SAFETY &gc_safety.
```

```
SET SCORE &gc_score.
```

```
SET TALK &gc_talk.
```

```
SET FORMAT TO
```

```
IF gn_apgen = 1 && We were not called from another APGEN program
```

```
CLEAR WINDOW
```

```
CLEAR POPUP
```

```
CLEAR ALL
```

```
CLOSE ALL
```

```
ELSE
```

```
RELEASE WINDOWS work, desktop
```

```
RELEASE SCREEN quick
```

```
RELEASE POPUP quick
```

```
gn_apgen = gn_apgen - 1
```

```
ENDIF
```

```
ON ERROR
```

```
ON KEY LABEL F1
```

```
RETURN
```

SID, S. C.

CURSO : dBASE IV

```
SET DELIMITERS TO ""
SET DELIMITER OFF
SET ESCAPE ON
SET INSTRUCT OFF
SET SAFETY ON
SET SCOREBOARD OFF
SET STATUS OFF
SET TALK OFF
```

*-- Set global variables

```
gn_barv = 0          && Initialize bar value variable
gn_error = 0        && Variable to store error() number
gn_send = 0         && Return variable from popup
gc_brdr = "2"       && Border style for menu box - See Procedure
lc_heading = "AGENDA PERSONAL" && Menu heading string
ll_color = ISCOLOR()
```

CLEAR

```
SET ESCAPE ON
SET STATUS ON
```

*-- Set colors

```
IF ll_color
  SET COLOR OF NORMAL TO w+/b
  SET COLOR OF MESSAGES TO w+/n
  SET COLOR OF TITLES TO w/b
  SET COLOR OF HIGHLIGHT TO gr+/bg
  SET COLOR OF BOX TO gr+/bg
  SET COLOR OF INFORMATION TO b/w
  SET COLOR OF FIELDS TO n/bg
```

ENDIF

USE AGENDA

SET ORDER TO TAG BAPELLIDOS

*-- Define the main popup menu for Quickapp

SET BORDER TO DOUBLE

DEFINE POPUP quick FROM 7,27

```
DEFINE BAR 1 OF quick PROMPT " ALTAS           " MESSAGE "Add records to databa
DEFINE BAR 2 OF quick PROMPT " CAMBIOS         " MESSAGE "Edit records in da
DEFINE BAR 3 OF quick PROMPT " CONSULTAS       " MESSAGE "Browse database AG
DEFINE BAR 4 OF quick PROMPT " BAJAS A REG. MARCADOS " MESSAGE "Purge deleted
DEFINE BAR 5 OF quick PROMPT " REPORTE         " MESSAGE "Run report form AGENDA"
DEFINE BAR 6 OF quick PROMPT " ETIQUETAS       " MESSAGE "Run label form AGENDA"
DEFINE BAR 7 OF quick PROMPT " ORDENAMIENTO    " MESSAGE "Reindex database AGE
DEFINE BAR 8 OF quick PROMPT " SALIDA         " MESSAGE "Exit program to dBAS
```

ON SELECTION POPUP quick DO Action WITH BAR()

*-- Define the popup menu for print redirection

DEFINE POPUP prntchk FROM 10,55

```
DEFINE BAR 1 OF prntchk PROMPT " Send to..." SKIP
DEFINE BAR 2 OF prntchk PROMPT REPLICATE(CHR(196),14). SKIP
DEFINE BAR 3 OF prntchk PROMPT " Screen " MESSAGE "Screen only"
DEFINE BAR 4 OF prntchk PROMPT " Printer " MESSAGE "Printer LPT1:"
```

```

@ 1,19 SAY "ie. ALL, NEXT <n>, and REST"
@ 1,01 SAY "Scope:" GET lc_scp
@ 2,01 SAY "For:  " GET lc_ln2
@ 3,01 SAY "While:" GET lc_ln3
READ
IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
  lc_temp=TRIM(lc_scp)
  lc_temp=lc_temp + IIF(LEN(TRIM(lc_ln2)) > 0, " FOR "+TRIM(lc_ln2), "")
  lc_temp=lc_temp + IIF(LEN(TRIM(lc_ln3)) > 0, " WHILE "+TRIM(lc_ln3), "")
  IF LEN(lc_temp) > 0
    LOCATE &lc_temp.
  ELSE
    DO Pause WITH "All fields were blank."
  ENDIF
ENDIF
RELEASE WINDOW Posit2
ENDCASE
IF EOF()
  DO Pause WITH "Record not found."
  GOTO ln_rec
ENDIF
IF READKEY()=12 .OR. READKEY()= 268 .OR. LASTKEY()=27  && Esc was hit
  lc_option='0'
ENDIF
ENDDO
SET DELIMITERS &gc_deli.
SET CONFIRM OFF
RETURN

```

```

PROC prntrtry
PRIVATE lc_escape
lc_escape = SET("ESCAPE")
IF .NOT. PRINTSTATUS()
  IF lc_escape = "ON"
    SET ESCAPE OFF
  ENDIF
  gn_pkey = 0
  ACTIVATE WINDOW printemp
  @ 1,0 SAY "Please ready your printer or"
  @ 2,0 SAY "      press ESC to cancel"
  DO WHILE ( .NOT. PRINTSTATUS()) .AND. gn_pkey <> 27
    gn_pkey = INKEY()
  ENDDO
  DEACTIVATE WINDOW printemp
  SET ESCAPE &lc_escape
  IF gn_pkey <> 27
    RETRY.
  ENDIF
ENDIF
RETURN
* EOF: C:AGENDA. PRG

```

CLASIFICACION DE LOS COMANDOS.

Durante la utilización normal de dBASE IV, se utilizan en forma combinada algunos comandos para llevar a cabo una tarea en particular. Algunos comandos de dBASE IV son modelos de construcciones estructuradas que la mayoría de los lenguajes de alto nivel poseen.

CREACION DE ARCHIVOS: Los siguientes comandos crean archivos de bases de datos y archivos asociados.

CREATE	COPY	COPY FILE	INDEX	CREATE SCREEN
JOIN	CREATE LABEL	CREATE REPORT	SAVE	CREATE QUERY
SORT	TOTAL	CREATE VIEW		

ADICION DE DATOS: Los siguientes comandos adicionan nuevos registros de datos a las bases de datos.

APPEND	BROWSE	INSERT
--------	--------	--------

CORRECCION DE DATOS: Los siguientes comandos editan los registros de una base de datos para su posible corrección.

BROWSE	CHANGE	DELETE	EDIT
PACK	READ	RECALL	REPLACE
UPDATE	ZAP		

ASISTENCIA AL USUARIO: Los siguientes comandos brindan información en línea al usuario

ASSIST	DIR	DISPLAY MEMORY
DISPLAY STATUS	DISPLAY STRUCTURE	HELP
DISPLAY HISTORY	ACTIVATE MENU	ACTIVATE POPUP
ACTIVATE SCREEN	ACTIVATE WINDOW	SHOW MENU
SHOW POPUP		

DESPLIEGUE DE DATOS: Los siguientes comandos despliegan datos seleccionados de una base de datos no importando el estado de SET TALK ON/OFF.

@..SAY	?	AVERAGE	BROWSE
COUNT	DISPLAY	LIST	REPORT
SUM	TEXT		
@..TO			

POSICIONAMIENTO DEL APUNTADOR A LOS REGISTROS: Los siguientes comandos cambian la posición del apuntador a los registros según sea dirigido por los comandos.

CONTINUE	FIND	GOTO	LOCATE
SEEK	SKIP	COPY FILE	COPY STRUCTURE EXTENDED
COPY STRUCTURE	COPY INDEXES	JOIN	COPY MEMO
COPY TAG	COPY TO ARRAY	SCAN	

MANIPULACION DE BASES DE DATOS: Los siguientes comandos afectan enteramente a los archivos de las bases de datos.

APPEND FROM	CLOSE	COPY	CONVERT
ERASE	MODIFY STRUCTURE	QUIT	INDEX
REINDEX	RENAME	SELECT	SORT
USE			

MANIPULACION DE OTROS TIPOS DE ARCHIVOS: Los siguientes comandos afectan a archivos de tipos que no son bases de datos.

MODIFY COMMAND	MODIFY LABEL	MODIFY REPORT	RUN
BEGIN TRANSACTION	END TRANSACTION	ROLLBACK	

UTILIZACION DE VARIABLES DE MEMORIA: Los siguientes comandos manipulan variables de memoria.

ACCEPT	AVERAGE	CLEAR ALL	CLEAR GETS
CLEAR MEMORY	COUNT	DISPLAY MEMORY	INPUT
PRIVATE	PUBLIC	READ	RELEASE
RESTORE	SAVE	STORE	SUM
WAIT	CLEAR FIELDS	LIST MEMORY	CLEAR TYPEAHEAD

PROGRAMACION: Los siguientes comandos ayudan en el control y utilización de los archivos tipo programas.

ACCEPT	CANCEL	DO	DO CASE
DO WHILE	IF	INPUT	MODIFY COMMAND
NOTE/*	PARAMETERS	QUIT	RETURN
RUN	TEXT	WAIT	PROCEDURE
@..SAY	@..GET	@..DOUBLE	@..CLEAR
LOOP	EXIT	READ	RECALL
PLAY MACRO	PRINTJOB	RESUME	ON PAGE
ON SELECTION	ON READERROR	ON PAD	ON ERROR
FUNCTION	CALCULATE	COMPILE	CONTINUE
DEBUG	DECLARE		

COMANDOS DEL MEDIO AMBIENTE: Los siguientes comandos controlan el medio ambiente de dBASE IV y sus dispositivos perifericos.

EJECT	CLEAR	RUN
LIST HISTORY	LIST STATUS	DEFINE BAR
DEFINE BOX	DEFINE MENU	DEFINE PAD
DEFINE POPUP	DEFINE WINDOW	DEFINE LOGOUT
MOVE WINDOW	PROTECT	

COMANDOS DE RED :

PROTECT	UNLOCK
---------	--------

CONEXION CON OTROS PAQUETES : Los siguientes comandos sirven para intercomunicar a dBASE IV con otros paquetes y/o lenguajes.

EXPORT	IMPORT	CALL
LOAD		

CONTROL DE PARAMETROS: Los siguientes parametros pueden ser habilitados (ON) ó deshabilitados (OFF) o puestos en algun valor en especial (TO) dentro de dBASE IV. Todos comienzan con SET.

ALTERNATE	BELL	CARRY	CATALOG
CENTURY	COLOR	CONFIRM	CONSOLE
DEBUG	DECIMALS	DEFAULT	DELETED
DELIMITERS	DEVICE	DOHISTORY	ECHO
ESCAPE	EXACT	FIELDS	FILTER
FIXED	FORMAT	FUNCTION	HEADING
HELP	HISTORY	INDEX	INTENSITY
MARGIN	MEMOWIDTH	MENU	MESSAGE
ORDER	PATH	PRINT	PRINTER
PROCEDURE	RELATION	SAFETY	SCOREBOARD
STATUS	STEP	TALK	TITLE
TYPEAHEAD	UNIQUE	VIEW	
AUTOSAVE	BLOCKSIZE	BORDER	CLOCK
CURRENCY	DATE	DESIGN	DEVELOPMENT
DISPLAY	ENCRYPTION	FULLPATH	HOURS
INSTRUCT	LOCK	MARK	NEAR
ODOMETER	PAUSE	POINT	PRECISION
REFRESH	REPROCESS	SEPARATOR	SKIP
SPACE	SQL	TRAP	WINDOW

APENDICE I

TECLAS DE CONTROL AUXILIARES PARA LA OPERACION EN PANTALLA.

TECLA	ALTERNATIVA	FUNCION
↑	^E	Desplaza el cursor una linea o campo hacia arriba.
↓	^X	Desplaza el cursor hacia abajo - un campo o linea.
→	^S	Desplaza el cursor un espacio a la izquierda. En menus mueve el cursor un campo a la izquierda.
←	^D	Desplaza el cursor un espacio a la derecha, en menus, mueve el cursor un campo a la derecha.
^→	^B	Se translada al campo a la derecha en BROWSE. En MODIFY COMMAND, desplaza el cursor al final de la linea.
^←	^Z	Se translada al campo de la izquierda. En MODIFY REPORT, despliega la estructura del archivo en scroll hacia abajo. En MODIFY COMMAND, mueve el cursor al inicio de la linea.
←←	(BS)	Borra el caracter a la izquierda del cursor (tecla backspace).
Del	^G	Elimina el caracter sobre el que esta posicionado el cursor.
End	^F	Desplaza el cursor una palabra hacia la derecha.
^End	^W	Termina guardando en disco todas las actualizaciones efectuadas al archivo de comandos y en la utilización de APPEND y BROWSE.
Esc	^Q	Termina sin almacenar los cambios efectuados al archivo de comandos en APPEND y BROWSE no salva los cambios efectuados al último registro visitado.

TECLA	ALTERNATIVA	FUNCION
Home	^A	Desplaza el cursor una palabra a la izquierda.
^Home		Habilita o inhibe las opciones - en los comandos de manejo de pantalla.
Ins	^V	Habilita o inhibe la opción de - inserción de caracteres. Si esta habilitada inserta caracteres a continuación del cursor, si esta deshabilitada reescribe sobre -- los caracteres en esa línea.
^KW		En MODIFY COMMAND, escribe el archivo entero sobre otro archivo.
^KR		En MODIFY COMMAND, lee un archivo del disco sobre el archivo en que se está trabajando.
^N		Inserta una nueva línea o definición de campo.
PgUp	^R	Se desplaza hacia el registro o pantalla anterior, o a los 17 registros anteriores (ventana anterior) si estamos en BROWSE.
^PgUp	^W	Termina y salva en operaciones - de pantalla, tales como: edición de campos memo.
PgDn	^C	Se desplaza al siguiente registro o a la siguiente pantalla - de despliegue, o a la siguiente ventana de 17 registros en el - comando BROWSE.
^PgDn		Se introduce y permite la edición de campos memo.

TECLA	ALTERNATIVA	FUNCION
<RETURN>		Desplaza el cursor al siguiente campo o línea. En APPEND termina y salva si esta en el primer carácter de un registro en blanco. En EDIT, termina y salva si esta en el último campo del último registro. En MODIFY COMMAND, inserta una línea si INSERT esta habilitado. En menus, selecciona una opción.
^T		Elimina una palabra a la derecha de la posición del cursor.
^U		Marca un registro para ser eliminado en EDIT y BROWSE. Elimina una definición de campo en MODIFY REPORT ó MODIFY STRUCTURE.
^Y		Elimina hasta el final del campo o toda una línea si estamos en MODIFY COMMAND.

Caracteres de control cuando no estamos en operaciones de facilidad en pantalla.

TECLA	ALTERNATIVA	FUNCION
<--	^H	Elimina caracteres al tiempo que se regresa el cursor.
<RETURN>	^M	Tiene el mismo efecto que la tecla de <RETURN>.
^P		Habilita o inhibe la utilización de la impresora, al ponerla en paralelo con la computadora.
^S		Detiene y continua con el despliegue de la pantalla o con la acción de un comando.
^X		Elimina una línea de comandos.

APENDICE II

TERMINOLOGIA UTILIZADA.

<alias>	Nombre alternativo de archivo.
<condición>	Expresión lógica.
<cadena>	Cadena de caracteres.
<dec>	Número de lugares decimales.
<delimitador>	Comillas sencillas o dobles, o parentesis cuadrados para datos.
<exp>	Campos, Variables de memoria, funciones de cadenas de caracteres, y combinaciones de estas.
<lista exp>	Lista de expresiones separadas por comas.
<Cexp>	Expresión de caracteres.
<Dexp>	Expresión fecha.
<Nexp>	Expresión numérica.
<campo>	El nombre de un campo.
<lista de campos>	Una lista de nombres de campos separadas por comas.
<archivo>	El nombre de un archivo.
<archivo index>	El nombre de un archivo indexado.
<llave>	La(s) porción(es) de un archivo tipo base de datos, utilizadas para crear un archivo indexado.
<var mem>	El nombre de una variable de memoria.
<lista var mem>	Variables de memoria separadas por comas.
<n>	Un número.
<procedure>	Un archivo de comandos, similar a un subprograma.
<comodin>	ALL, NEXT n, y RECORD n.

<skeleton>

Sustituye * o ? por parte de un nombre.

<.mdx file>

Archivo múltiple de index

APENDICE III
FUNCIONES DE dBASE IV

&(Cvar mem)	Macro substitución.
ABS (Nexp)	Valor Absoluto
ACCESS()	Accesa al nivel que el usuario a definido con el comando PROTECT
ACOS (Nexp)	Arco coseno
ALIAS ([<Nexp>])	El alias de una area de trabajo
ASC()	Conversión de caracter a codigo ASCII.
ASIN(<Nexp>)	Arco seno
AT(<Cexp 1>, <Cexp 2>)	Busqueda de subcadena.
ATAN(<Nexp>)	Arco tangente
ATAN(<Nexp1>, <Nexp2>)	Arco tangente del seno <Nexp1> y el coseno <Nexp2>
BAR()	Número del último comando seleccionado desde el pop-up menú
BOF()	Verifica si es comienzo de archivo.
CALL (<Cexp>[, <exp1>/<var1> [, <exp2>/<var2>]]...)	Ejecuta un programa .BIN en memoria
CDOW(<Dexp>)	Día de la semana (por ejemplo Lunes).
CEILING (<Nexp>)	El límite del entero mayor o igual al valor especificado
CHANGE()	Determina cual registro ha sido modificado por un usuario en una red local

CHR()	Conversión de código ASCII a caracter.
CMONTH(<Dexp>)	Mes (por ejemplo Enero).
COL()	Se posiciona en esa columna sobre la pantalla.
COMPLETED()	Transacción completa o no
COS(<Nexp>)	El coseno de un ángulo en radianes
CTOD(<Cexp>)	Conversión caracter a fecha.
DATE()	Toma la fecha del sistema.
DAY(<Dexp>)	Día del mes (por ejemplo 15).
DBF([<alias>])	Nombre de la Base de Datos en uso.
DELETED()	Registro eliminado verifica que sea este su estado.
DIFERENCE(<Cexp1>,<Cexp2>)	Diferencia entre dos expresiones tipo caracter
DISKSPACE()	Espacio libre (bytes) en el disco
DMY(<Dexp>)	Formato de conversión DD/MES/AA
DOW(<Dexp>)	Día de la semana (por ejemplo, 3).
DTOC(<Dexp>)	Conversión fecha a caracter.
DTOR(<Nexp>)	Conversión de grados a radianes
DTOS(<Dexp>)	Conversión de fecha para ordenamientos, YYYYMMDD
EOF()	Verifica si es fin de archivo.
ERROR()	Número para condición ON ERROR
EXP(<Nexp>)	Exponencial base (e).

FIELD(Nexp)	Nombre de campos en la Base de Datos
FILE("<nombrearchivo>")	Verifica si existe el archivo.
FIXED(<Nexp>)	Convierte la longitud, real de un número con punto flotante a código binario decimal.
FKLABEL(Nexp)	Nombre de teclas de funciones.
FKMAX(Nexp)	No. Máximo de teclas de funciones.
FLOAT(<Nexp>)	Convierte un código binario decimal a un número con punto flotante
FLOCK([alias])	Bloquea una base de datos
FLOOR(<Nexp>)	El límite del entero, menor o igual a el valor especificado.
FOUND()	Resultado de una búsqueda en una Base de Datos.
FV(<Cant. a pagar>, <tasa>, r <periodos>)	Valor futuro
GETENV	Variables ambientales del Sistema Operativo.
IIF(<condicion>, verdadero, falso)	Una expresión u otra.
INKEY()	Retorna el valor de la tecla que fue presionada durante la ejecución del programa.
INT(<Nexp>)	Toma la parte entera.
ISALPHA(<Cexp>)	Evaluación para letra.
ISCOLOR()	Evaluación para modo de color.
ISLOWER(<Cexp>)	Evaluación para minúsculas.
ISMARKEDI[<alias>]	Determina si una base de datos esta en transición o no.
ISUPPER(<Cexp>)	Evaluación para mayúsculas.
KEY([<.mdx file>], <Nexp> [alias])	Indica la llave del index

LASTKEY()	Valor en ASCII de la tecla para salir de un comando de full-screen.
LEFT(<Cexp>/<campo memo>, <Nexp>)	Selección de subcadena a partir del lado izquierdo.
LEN(<Cexp>)	Longitud de una cadena.
LIKE(<pat tern>, <Cexp>)	Compara cadenas usando wildcards
LINENO()	Regresa el número de línea a ejecutarse en un programa.
LESYS (Nexp)	Regresa el tiempo, fecha, y clave para un archivo bloqueado en una red: Nexp = 0, da la hora en que fué bloqueado Nexp = 1, da la fecha en que fué bloqueado Nexp = 2, da la clave del usuario que bloqueo el registro
LOCK([<lista Cexp>] [, <alias>])	Bloquea uno o más registro trabajando en una red; (similar a RLOCK())
LOG (<Nexp>)	Logaritmo natural base e.
LOG10 (<Nexp>)	Logaritmo natural base 10.
LOOKUP(<return field>, <look-for exp>, <look-in field>)	Bloquea un registro desde otra base de datos
LOWER(<Cexp>)	Conversión a minúsculas.
LTRIM(<Cexp>)	Eliminación de espacios iniciales en blanco.
LUPDATE()	Ultima actualización de la Base de Datos.
MAX (<Nexp1>/<Dexp1>, <Nexp2>/<Dexp2>)	Determina el mayor de dos valores.

NDX (<Nexp1>/[,alias>]	Informa el nombre del archivo de índices múltiples abierto
MDY (<Dexp>)	Convierte la fecha al formato Mes, DD,YY.
MEMLINES (<memo field name>)	
MEMORY ([O])	
MENU ()	Nombre del menú activo
MESSAGE ()	Secuencia del mensaje ON ERROR.
MIN (<Nexp1>/<Dexp1>, <Nexp2>/<Dexp2>)	Determina el menor de dos valores.
MLINE (<campo memo>,<Nexp>)	Especifica una línea de un campo memo
MOD (<Nexp1>,<Nexp2>)	Módulo.
MONTH (<Dexp>)	Mes del año (por ejemplo 4).
NDX (<Nexp> [,<alias>]	Nombres de los Archivos de Índices Abiertos.
NETWORK ()	Determina si dBASE IV está instalado y corriendo sobre una red o no.
ORDER ([<alias>])	Número del orden primario en los index o .mdx etiquetas
OS ()	Nombre del Sistema Operativo.
PAD ()	
PAYMENT (<principal>,<tasa>, <periodos>)	
PCOL ()	Se posiciona en una determinada columna de la impresora.
PI ()	
POPUP ()	

PRINTSTATUS()	
PROGRAM()	
PROMPT()	
PROW()	Se posiciona en un determinado renglon de la impresora.
PV(<Cant. a pagar>, <tasa>, <periodos>)	Valor presente
RAND([<Nexp>])	Valor aleatorio.
READKEY()	Determina el valor de la tecla oprimida, para dejar el modo de pantalla completa.
RECCOUNT([<alias>])	No. de registros en la Base de Datos.
RECNO([<alias>])	Me dice el número de registro en uso.
RECSIZE([<alias>])	Indica el tamaño del registro.
REPLICATE(<Cexp>, <Nexp>)	Repite la expresión de caracteres, el número de veces que se le indique.
RIGHT(<Cexp>/<campo memo>, <Nexp>)	Selección de subcadena a partir del lado derecho.
RLOCK([<lista Cexp>] [, <alias>])	Bloque uno o más registros en la base de datos en red
ROLLBACK()	
ROUND(<Nexp>, <dec>)	Redondea a un número de decimales.
ROW()	Posición actual de líneas de pantalla.
RTOD(<Nexp>)	Convierte radianes a grados
RTRIM(<Cexp>)	Eliminación de últimos espacios en blanco.

SEEK (<Cexp> [, <alias>])	Revisa si la llave del index se encuentra o no.
SELECT ()	Regresa el número mayor sin usar de las areas de trabajo.
SET (<Cexp>)	Informa si los parametros de SET están prendidos o apagados
SIGN (<Nexp>)	Signo matemático
SIN (<Nexp>)	función seno.
SOUNDEX (<Cexp>)	
SPACE (<Nexp>)	Genera espacios en blanco.
SQRT (<Nexp>)	Raiz cuadrada positiva.
STR (<Nexp>[, <longitud>][, <dec>])	Conversión numérica a caracter.
STUFF	Reemplazo de una parte de la cadena.
SUBSTR (<Cexp>, <inic>[, <longitud>])	Selección de Subcadena.
TAG ([<.mdx file>,]<Nexp>[, <alias>])	Retorna la etiqueta de un archivo .mdx
TAN (<Nexp>)	Tangente
TIME ()	Tiempo del sistema (hh:mm:ss).
TRANSFORM (<Nexp>, <formato>)	Tranforma la expresión numérica a caracter con un formato de salida PICTURE.
TRIM (<Cexp>)	Remueve espacios sobrantes a los extremos de la cadena.
TYPE (<Cexp>)	Valida la expresión.
UPPER (<Cexp>)	Conversión a mayusculas.
USER ()	
VAL (<Cexp>)	Conversión caracter a número.
VARREAD ()	Nombre del campo o variable de memoria que se esta editando

VERSION()

Versión de dBASE IV

YEAR(<Dexp>)

Año (por ejemplo, 1985).

COMANDOS DE dBASE IV.

? <exp>
 Despliega la expresión o su resultado en la siguiente línea.

??<exp>
 Despliega la expresión o su resultado sobre la misma línea.

??/? [<expresión1>] [PICTURE <Cexp>]
 [FUNCTION <lista de funciones>] [AT <Nexp>]
 [STYLE <número de font>]
 [, <expresión2> ...] [,]

??? <Cexp>
 Envía directamente a la impresora.

@ < renglon, columna > [SAY <exp>
 [PICTURE <formato>]
 [FUNCTION <lista de funciones>
 [GET <variable>
 [[OPEN] WINDOW <nombre de la ventana>]
 [PICTURE <formato>]
 [FUNCTION <lista de funciones>
 [RANGE [<menor>] [, <mayor>]
 [VALID <condición>]
 [ERROR <Cexp>]]
 [WHEN <Condición> [DEFAULT <expresión>]
 [MESSAGE <Cexp>]]
 [COLOR [<estandar>] [, <resaltado>]]

SAY despliega los datos del usuario formateados por el mismo sobre la pantalla o la impresora.

GET despliega los datos formateados por el usuario sobre la pantalla para su edición.

@ < renglon1, columna1 > [CLEAR] TO
 <RENGLON2, COLUMNA2>
 Limpia una porción de la pantalla en la ventana activa

@ < renglon1, columna1 > FILL TO < renglon2, columna2 >
 [COLOR <atributos>]
 Permite cambiar los colores en una parte específica.

@ < renglon1, columna1 > TO < renglon2, columna2 > [DOUBLE/PANEL
 / <definición del borde>] [COLOR <atributos>]

Dibuja una caja en la pantalla, con línea sencilla, doble u otro carácter especificado.

ACCEPT [**<indicador>**] TO **<var mem>**
 Almacena una cadena de caracteres dentro de una variable de memoria.

ACTIVATE MENU **<menu>** [PAD **<pad>**]
 Activa un menú bar existente y los despliega para su uso

ACTIVATE POPUP **<popup>**
 Activa un menú pop-up previamente definido para su uso

ACTIVATE SCREEN
 Recupera una pantalla completa

ACTIVATE WINDOW **<lista de las ventanas>/ALL**
 Activa y despliega una ventana previamente definida

APPEND [BLANK]
 Adición de registros al final de un archivo tipo base de datos.

APPEND FROM **<archivo>** [FOR **<exp>**] [TYPE] [**<tipo de archivo>**]
 Adiciona registros de otros archivos a los archivos tipo base de datos.

APPEND FROM ARRAY **<arreglo>** [FOR **<condición>**]
 Adiciona registros a la base de datos desde un arreglo

APPEND MEMO **<campo memo>** FROM **<archivo>** [OVERWRITE]
 Importa un archivo de textos a un campo memo

ASSIST
 Menú autoexplicativo para la generación y administración de bases de datos. (Centro de Control)

AVERAGE **<lista exp>** [**<comodin>**] [FOR/WHILE **<exp>**]
 [TO **<lista de variables>/TO ARRAY <arreglo>**]
 Obtiene el valor promedio de una expresión numérica.

BEGIN TRANSACTION [**<path name>**]
<comandos de transacciones>

END TRANSACTION
 Inicia un archivo de transacciones, con la alternativa de un ROLLBACK (regreso sin afectar) y realiza los cambios en los registros de la base de datos.

BROWSE

```
[NOINIT] [NOEDIT] [NODELETE] [NOCLEAR] [COMPRES] [FORMAT]
[LOCK <Nexp>] [FREEZE <campo>] [NOFOLLOW] [NOMENU]
[WIDTH <Nexp>] [NOAPPEND] [WINDOW <ventana>]
[FIELDS <campo1> [/R] [/<ancho>]
      /<campo calculado 1> = <expresión 1>
      <campo2> [/R] [/<ancho>]
      /<campo calculado 2> = <expresión 2>]....]
```

Posee un menú de asistencia, edita utilizando las facilidades de pantalla y ventanas; despliega hasta 17 registros por pantalla.

```
CALCULATE [comodin] <lista de opciones> [FOR/WHILE <condición>]
          [TO <lista de var.>/TO ARRAY <arreglo>]
```

Donde la lista de opciones puede ser una de las siguientes funciones:

```
AVG(<Nexp>), CNT(), MAX(<expresión>), MIN(<expresión>)
NPV(<tasa, flujo, inicial>), STD(<Nexp>), SUM(<Nexp>)
VAR(<Nexp>)
```

```
CALL <nombre de módulo> [WITH <Cexp> / <variable de memoria>]
```

Procesa un archivo (o módulo) binario que ha sido colocado en la memoria por medio del comando LOAD.

CANCEL

Aborta la ejecución de programas y regresa al indicador punto (.).

CHANGE

```
[NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU] [NOEDIT]
[NODELETE] [NOCLEAR] [ <núm. reg.>]
[FIELDS <lista de campos>] [FOR/WHILE <exp>]
```

Edita los campos y registros especificados de una Base de Datos.

CLEAR

Borra la pantalla.

CLEAR

```
[ALL/FIELDS/GETS/MEMORY/MENUS/POFUPS/TYPERHEAD/WINDOWS]
```

Limpia la pantalla, Cierra todos los archivos tipo base de datos, los archivos index, los archivos de formatos de pantalla, las relaciones. También libera todas las variables de memoria, y selecciona, el área de trabajo uno.

CLOSE

```
ALL/ALTERNATE/DATABASES/FORMAT/INDEX/PROCEDURE
```

Cierra los tipos de archivo especificados.

CONTINUE

Posiciona el apuntador al número de registro al siguiente registro que cumpla con la condición de búsqueda establecida previamente por el comando LOCATE.

COPY FILE <nombre archivo> TO <nombre archivo>
Duplica cualquier clase de archivo.

COPY STRUCTURE TO <nombre de archivo> [FIELDS <lista de campos>]
Duplica la estructura del archivo en uso a una nueva Base de Datos.

COPY TO <nombre archivo> [<comodin>][FIELDS <lista campos>]
[FOR/WHILE <exp>][TYPE][<tipo de archivo>]
Copy la base de datos utilizada en otra base de datos o a un archivo de texto, si son utilizadas las opciones SDF o DELIMITED.

COPY TO <archivo nuevo> STRUCTURE EXTENDED
Copia la definición de la estructura del archivo en uso a una nueva Base de Datos en donde la descripción de la estructura original es convertida en registros.

COUNT [<comodin>][FOR/WHILE <exp>][TO <var mem>]
Cuenta el número de registros que acaso cumplan con la condición impuesta por FOR ó WHILE el número de registros se especifica en la opción de comodin.

CREATE <nombre archivo .DBF>
Definición de la estructura de un archivo tipo base de datos incluyendo, la adición de este nuevo archivo al directorio del disco.

CREATE LABEL <nombre archivo .LBL> /?
Controlado a través de un menú; sirve para la generación de un archivo tipo LABEL para la generación de etiquetas en la impresora; el archivo contiene el formato de impresión de las etiquetas.

CREATE QUERY <nombre de archivo .QRY> /?
Controlado por un menú; crea una condición de filtro y la almacena en un archivo .QRY .

CREATE REPORT <nombre archivo .FRM> /?
Controlado por un menú; crea un archivo de formato de reportes.

CREATE SCREEN <nombre de archivo .SCR> /?
Controlado por un menú; crea un formato de pantalla de acuerdo con los requerimientos del cliente y lo almacena en un archivo .SCR También genera un archivo .FMT; tiene opciones para crear y para modificar la estructura de una base de datos.

CREATE VIEW <nombre de archivo .VUE> /?
 Controlado por un menú; crea una relación entre archivos y establece un ambiente de trabajo.

CREATE VIEW <nombre de archivo .VUE> FROM ENVIRONMENT
 Crea un archivo view (.VUE) en base a un ambiente de trabajo.

DELETE [(comodin)] [(FOR/WHILE <exp>)]
 Marca los registros seleccionados para ser eliminados posteriormente mediante el comando PACK ó ZAP.

DIR [(drive:)] [(directorio) \] [(skeleton)]
 Muestra los archivos especificados del drive especificados, si ninguna especificación de archivos existe, solo muestra los archivos con extensión .DBF.

DISPLAY [(comodin)] [(FIELDS <lista campos>)] [(FOR/WHILE <exp>)] [(OFF)]
 Despliega los registros y campos especificados de la base de datos que esta siendo utilizada.

DISPLAY HISTORY [(LAST <Nexp>)] [(TO PRINT)]
 Muestra los comandos almacenados en el HISTORY en orden inverso al cronológico. Muestra todas las líneas al menos que LAST <Nexp> sea especificado.

DISPLAY MEMORY [(TO PRINT)]
 Despliega las variables de memoria actuales.

DISPLAY STATUS [(TO PRINT)]
 Despliega información actual acerca de las bases de datos activas, archivos index, archivos alternos, y parametros del sistema.

DISPLAY STRUCTURE [(TO PRINT)]
 Despliega la estructura de la base de datos que esta siendo utilizada.

DO <nombre archivo .PRG>/<nombre procedure> [(WITH <lista param.>)]
 Provoca la ejecución de un programa o procedure opcionalmente con paso de parametros.

DO CASE. .CASE. .[(OTHERWISE)]. . .ENDCASE
 Permite la ejecución de solo un bloque de instrucciones (CASE) de entre varios, dadas unas ciertas condiciones de entrada. Opcionalmente, incluye la alternativa (OTHERWISE "si no es ningún caso anterior, entonces"). El DO CASE debera terminar con un ENDCASE.

DO WHILE. <comandos>. ENDDO

Permite la elaboración de bloques de comandos con estructura de anillo (Loops) en un programa. El comando LOOP brinca los comandos entre el mismo y la instrucción ENDDO de tal manera que no los ejecuta. DO WHILE debe terminar con un ENDDO.

EDIT [<comodin>] [FIELDS <lista de campos>] [WHILE/FOR <condición>]

Permite efectuar cambios al contenido de los campos de los archivos tipo base de datos.

EJECT

Envia un salto de página (FORM FEED) a la impresora.

ERASE <nombre archivo>

Elimina el archivo especificado del directorio.

EXIT

Permite el "escape" de un loop DO sin terminar la ejecución de un programa.

EXPORT TO <nombre de archivo> TYPE PFS

Crea archivos del PFS en base a los archivos del dBASE IV.

FIND <cadena caracteres>

Posiciona el apuntador al registro, en el primer registro, con una llave index, que sea igual a la cadena de caracteres especificada.

GO/GOTO BOTTOM/TOP/<Nexp>

Posiciona el apuntador de registros directamente en un registro específico.

HELP [<comando o duda buscada>]

Menú de ayuda, que explica los comandos de DBASE IV e información adicional.

IF. [ELSE]. ENDIF

Permite la ejecución condicional de comandos en un programa, opcionalmente con una alternativa (ELSE). Cada IF debe terminar con un ENDIF.

IMPORT FROM <nombre de archivo> TYPE PFS

Crea archivos del dBASE IV en base a archivos PFS.

INDEX ON <exp> TO <nombre archivo .NDX> [UNIQUE]

Esto provoca que la base de datos sea ordenada de acuerdo a una llave especificada, el resultado del ordenamiento se guarda en el archivo .NDX, además este ordenamiento provoca, que la forma de presentar el archivo en pantalla sea de acuerdo a este nuevo ordenamiento.

INPUT [<indicador>] TO <var mem>
 Permite la entrada de datos a una variable de memoria.

INSERT [BLANK][BEFORE]
 Inserta un registro en la posición especificada dentro de la base de datos.

JOIN WITH <alias> TO <archivo nuevo> FOR <condición>
 [FIELDS <lista campos>]
 Combina los registros y campos especificados de dos bases de datos; el resultado los deja en otra base de datos.

LABEL FORM <nombre archivo .LBL> [/? [SAMPLE] [<scope>]
 [FOR/WHILE <condición>][TO PRINT][TO FILE <nombre archivo>]
 Imprime etiquetas utilizando para esto un archivo que contiene un formato específico de impresión.

LIST [<comodin>][FOR/WHILE <condición>][FIELDS <lista campos>]
 [OFF] [TO PRINT]
 Lista registros y campos específicos de la base de datos.

LIST HISTORY [LAST <Nexp>] [TO PRINT]
 Muestra los comandos almacenados en HISTORY MODE en orden cronológico. Los muestra todos a menos que LAST (<Nexp>) sea especificado.

LIST MEMORY [TO PRINT]
 Muestra el nombre, el tipo y el tamaño de cada variable de memoria.

LIST STATUS
 Muestra la información relativa a las operaciones del dBASE. IV hasta el momento.

LIST STRUCTURE [TO PRINT]
 Muestra la información relativa a la base de datos en uso.

LOAD <nombre del archivo binario> [,<extensión>]
 Coloca un archivo binario en la memoria a modo de que pueda ser procesado por medio de un comando CALL.

LOCATE [<comodin>] FOR <condición>
 Posiciona el apuntador de registros en un registro tal que cumpla con la condición especificada por <condición>.

LOOP
 Salta todos los comandos entre esta instrucción y el ENDDO.

MODIFY COMMAND <nombre archivo>
 Procesador de palabra que permite la creación y edición de archivos ASCII de texto, incluyendo los archivos de programas (.PRG).

MODIFY LABEL <nombre archivo .LBL> /?
 Control a través de menú que permite la creación y edición, de archivos de formatos para la impresión de etiquetas.

MODIFY QUERY <nombre de archivo .QRY> /?
 Equivale a la opción correspondiente de CREATE, pero modifica en vez de crear.

MODIFY REPORT <nombre archivo .FRM> /?
 Control a través de menú para la creación y edición de archivos de formato de reportes.

MODIFY SCREEN <nombre de archivo .SCR > /?
 Igual que la opción correspondiente de CREATE, pero modifica en vez de crear.

MODIFY STRUCTURE
 Altera la estructura de la base de datos en uso.

MODIFY VIEW <nombre de archivo .VUE> /?
 Igual que la opción correspondiente de CREATE, pero modifica en vez de crear.

NOTE/* <cadena de caracteres sin delimitadores>
 Inserta un comentario no ejecutable dentro del código de un archivo de programas.

ON ERROR/ESCAPE/KEY <comando del dBASE>
 Procesa el comando especificado cuando ocurre un error, cuando la tecla ESC es oprimida o cuando es precionada cualquier otra tecla.

PACK
 Elimina en forma permanente los registros previamente marcados, para ser eliminados.

PARAMETERS <lista parametros>
 Especifica variables de memoria, que son utilizadas para pasar información desde o hacia los programas ejecutados a través del comando DO. .WITH.

PRIVATE [ALL/LIFE/EXCEPT <skeleton>]]<lista de var mem>
 Impide la transferencia o consulta de valores de las variables que se incluyen en la lista a programas de nivel más alto.

PROCEDURE <nombre del procedimiento>
 Identifica el principio de cada rutina en un archivo de procedimientos.

PUBLIC <lista de variables de memoria>
 Marca las variables de memoria como globales.

QUIT
 Cierra todos los archivos y termina la ejecución de dBase regresando al sistema operativo.

READ [SAVE]
 Permite la entrada de datos a un campo o variable a través de una instrucción GET.

RECALL [[<comodin>][FOR/WHILE <condición>]
 Elimina las marcas de los registros que iban a ser eliminados.

REINDEX
 Reconstruye el archivo de índices (index), de la base de datos que esta siendo utilizada y tiene ya un archivo index.

RELEASE [/<lista de var mem>][ALL [LIKE/EXCEPT <skeleton>]]
 [MODULE <nombre de módulo>]
 Elimina las variables de memoria actuales o algún módulo cargado en memoria.

RENAME <nombre actual del archivo> TO <nuevo nombre del archivo>
 Asigna un nuevo nombre al archivo.

REPLACE [<comodin>] <campo1> WITH <exp1> [,<campo2 WITH <exp2>,...]
 [FOR/WHILE <condición>]
 Cambia el contenido de los campos de datos a valores especificados.

REPORT FORM <nombre archivo .FRM> /? [[<comodin>][FOR/WHILE
 <exp>][PLAIN] [HEADING <cadena de caracteres>][NOBJECT][TO
 PRINT] [TO FILE <nombre archivo>] [SUMMARY]
 Despliega un reporte de datos en forma tabular.

RESTORE FROM <nombre archivo .MEM> [ADDITIVE]
 Reestablece un conjunto de variables de memoria previamente salvadas sobre un archivo.

RESUME
 Hace que un programa suspendido reanude su ejecución.

RETRY
 Termina un archivo de comandos y procesa nuevamente un mismo comando en el programa que lo activo.

RETURN [TO MASTER]

Termina la ejecución de un programa. Esta es la última línea ejecutable.

RUN <comando>

Ejecuta un programa fuera del contexto de DBASE IV.

SAVE TO <nombre archivo .MEM> [ALL LIKE/EXCEPT <skeleton>]

Copia el valor de las variables de memoria en ese momento, y las guarda sobre un archivo de memoria.

SEEK <expresión>

Posiciona el apuntador del registro al primer registro, con una llave index que sea igual a la expresión especificada.

SELECT [n/alias]

Selecciona el switcheo entre las 10 áreas de trabajo.

SET

Incluye un menú de ayuda para establecer los parámetros de control de DBASE IV.

SET ALTERNATE TO [<nombre archivo>]

Crea un archivo donde es almacenada la salida.

SET ALTERNATE on/OFF

Habilita o inhibe la posibilidad de enviar la salida a este archivo.

SET ALTERNATE TO [<nombre de archivo>]

Crea un archivo para almacenar los datos generados.

SET BELL ON/off

Habilita/inhibe el sonido de la campana en la entrada de datos.

SET CARRY on/OFF

Habilita/inhibe la posibilidad de escribir el contenido del último registro en el registro adicionado mediante APPEND.

SET CATALOG ON/off

AGREGA/no agrega nombres de archivos al catálogo abierto.

SET CATALOG TO [<nombre de archivo .CAT> /?]

Crea un nuevo catálogo; abre otro existente; o cierra un catálogo abierto.

SET CENTURY on/OFF

Muestra/NO MUESTRA el siglo en la presentación de la fecha.

SET COLOR ON/OFF

Cambia de un monitor a color a otro monocromático, y reciprocamente.

SET COLOR TO <display estandar>[,display resaltado][, <bordes>]
Habilita atributos de la pantalla de despliegue.

SET CONFIRM on/OFF
Efectua o no el salto en forma automatica al siguiente campo de entrada de datos cuando este campo esta lleno.

SET CONSOLE ON/off
Habilita/inhibe el envio de informacion de salida a la pantalla.

SET DATE AMERICAN/ANSI/BRITISH/ITALIAN/FRENCH/GERMAN
Determina el formato para la fecha. Normalmente la fecha es establecida en formato AMERICANO MM/DD/AA.

SET DEBUG on/OFF
Envia o no la salida del ECHO a la impresora.

SET DECIMALS TO <n>
Establece el numero minimo de decimales desplegado en el resultado de ciertas operaciones y funciones efectuadas.

SET DEFAULT TO <drive>
Especifica el drive de default para la interaccion con discos.

SET DELETED on/OFF
Oculta/procesa los registros marcados para ser eliminados.

SET DELIMITERS on/OFF
Despliega los campos de las variables delimitadas en forma normal o en video inverso.

SET DELIMITERS TO [<cadena de caracteres>] [DEFAULT]
Especifica los delimitadores de campos en los despliegues de las variables para entrada y salida en las operaciones de pantalla.

SET DEVICE TO SCREEN/print
Envia el resultado de los comandos @.SAY a la PANTALLA/impresora.

SET DOHISTORY on/OFF
Determina si los archivos de comandos estan, o no, registrados en HISTORY .

SET ECHO on/OFF
Hace eco/NO HACE ECO de los comandos en la impresora o en la pantalla.

SET ESCAPE ON/off
 ABORTA/continua la ejecución de un archivo de comandos cuando la tecla escape es digitada.

SET EXACT on/OFF
 REQUIERE/no requiere que la comparación de cadenas de caracteres sea exacta.

SET FIELDS on/OFF
 Respeta/IGNORA la lista de campos especificados.

SET FIELDS TO [<lista de campos> ALL]
 Determina que campos están disponibles.

SET FILTER TO [FILE <nombre de archivo .QRY> /?][<condición>]
 Provoca que un archivo tipo base de datos de la apariencia de que sólo contiene registros que pasen la condición especificada.

SET FIXED on/OFF
 Fija/NO FIJA el número de decimales que serán desplegados.

SET FORMAT TO [<nombre de archivo .FMT> /?]
 Abre un archivo de formato para la captura de datos.

SET FUNCTION <tecla de número> TO <cadena de caracteres>
 Establece el valor de las teclas de función.

SET HEADING ON/off
 MUESTRA/no muestra los nombres de campos arriba de los campos en LIST o en DISPLAY.

SET HELP ON/off
 Determina esta opción si se le preguntara o no al usuario si necesita o no ayuda cuando cometa algún error.

SET HISTORY ON/off
 ACTIVA o desactiva la ayuda HISTORY.

SET HISTORY TO (Nexp)
 Especifica el número de comandos ejecutados que han de ser almacenados en HISTORY. El número preestablecido de comandos almacenados es 20.

SET INDEX TO <lista archivos .NDX> /?
 Abre los archivos index invocados.

SET INTENSITY ON/off
 HABILITA/inhíbe los efectos de video inverso para algunos comandos.

SET MARGIN TO <n>
 Establece el margen izquierdo de la impresora.

SET MEMOWIDTH TO <Nexp>
Ajusta el ancho de la columna de datos del campo memo. El valor preestablecido es 50.

SET MENUS ON/off
Determina si seran o no desplegados los menus para algunas operaciones de pantalla.

SET MESSAGE TO <secuencia de caracteres>
Muestra un mensaje definido por el usuario en la linea de la parte baja de la pantalla.

SET ORDER TO [<Nexp>]
Establece que cualquier archivo de indice abierto ha de funcionar como el de control sin cerrar ni volver a abrir los archivos.

SET PATH TO <lista de esquemas>
Establece un esquema para la busqueda de archivos.

SET PRINT on/OFF
Envia/NO ENVIA la salida a la impresora.

SET PRINTER TO <dispositivo DOS>
Redirecciona la salida de datos hacia el dispositivo especificado.

SET PROCEDURE TO <nombre procedure>
Abre el archivo de procedure invocado.

SET RELATION TO [<llave>/RECNO()/<Nexp> INTO <alias>]
Liga dos bases de datos de acuerdo a una expresion en la llave.

SET SAFETY ON/off
HABILITA/inhibe una proteccion contra escritura al archivo.

SET STATUS ON/off
MUESTRA/no muestra la barra de estado en el modo punto.

SET STEP on/OFF
Se detiene/NO SE DETIENE el programa en su ejecucion cada vez se ejecuta un comando.

SET TALK ON/off
ENVIA/no envia el resultado de la ejecucion del comando a la pantalla.

SET TITLE ON/off
PIDE/no pide el encabezado de un archivo de catalogo cuando un nuevo archivo es agregado a un catalogo.

SET TYPEAHEAD TO <Nexp>
 Especifica el tamaño de la memoria intermedia de escritura adelantada. El tamaño preestablecido es 20.

SET UNIQUE on/OFF
 El primero/TODOS LOS REGISTROS con llave idéntica aparecen en un archivo de índices (index).

SET VIEW TO <nombre de archivo .VUE> /?
 Abre un archivo de relación.

SKIP <n>
 Provoca que el apuntador al número de registro avance o retroceda en forma relativa a la posición que ocupa.

SORT TO <nombre archivo nuevo> ON <campo1> [/A]/[D]/[C]
 [, <campo2> [/A]/[D]/[C]...][<comodin>][FOR/WHILE <condición>]
 Crea una copia ordenada de la base de datos, arreglada de acuerdo a uno de los campos de datos.

STORE <exp> TO <var mem> [, <lista var mem>]
 Almacena una expresión dentro de una o más variables de memoria.

SUM [<comodin>][<lista expresiones> TO <lista var mem>]
 [FOR/WHILE <condición>]
 Calcula y despliega la suma de una expresión para los registros de la base de datos especificada por el comodín y por la condición.

SUSPEND
 Detiene el proceso del procedimiento o del archivo de comandos que está siendo ejecutado y presenta el punto de interacción. Continúa el proceso escribiendo RESUME.

TEXT. .ENDTEXT
 Despliega un bloque de texto desde un archivo de comandos. TEXT deberá ser terminado con ENDTEXT.

TOTAL TO <nombre archivo> ON <llave> {<comodin>]
 [FIELDS <lista de campos>]
 [FOR/WHILE <condición>]
 Crea un sumario de un archivo de base de datos preordenado, conteniendo totales numéricos.

TYPE <nombre de archivo> [TO PRINT]
 Muestra el contenido de un archivo de texto.

UPDATE [RANDOM] ON <campo llave>
 FROM <alias>
 REPLACE <campo1> WITH <exp1>
 [, <campo2> WITH <exp2> . . .] [RANDOM]
 Permite modificaciones BATCH a una base de datos.

USE [<nombre archivo .DBF> /?][INDEX <nombre archivo .NDX>]
[ALIAS <alias>]
Especifica el archivo tipo base de datos que sera utilizado para todas las operaciones hasta que se utilice nuevamente el comando USE.

WAIT [<indicador>][TO <var mem>]
Suspende la ejecución de un programa hasta que sea digitada cualquier tecla.

ZAP
Elimina todos los registros de la base de datos que esta siendo utilizada.

GLOSARIO DE TERMINOS Y SIMBOLOS.

Esta sección del manual cubre aquellas palabras y símbolos que tiene un significado especial cuando se utilizan para describir, los comandos y funciones de dBASE IV.

< > : Pico parentesis, indican que el elemento encerrado deberá ser digitado por el usuario. Los parentesis no forman parte del comando.

[] : Parentesis cuadrados, indican que el elemento encerrado, puede ser omitido, esto es; el elemento es opcional. Los parentesis cuadrados no forman parte del comando.

^ : Indica la tecla de control. Cuando este símbolo es utilizado, la tecla de control (etiquetada como Ctrl), deberá ser presionada y continuar la presión hasta que otra tecla (que también se indicara) se presione. Por ejemplo, ^U indica que mientras la tecla control se mantiene bajo presión se digita la tecla U.

/ : La diagonal indica que debemos elegir una opción de entre varias que están separadas por diagonales.

Abortar : Suspender la ejecución de un programa de computadora, o la operación de un comando de dBase IV.

Acceso Secuencial : Un método de acceso por medio del cual se busca a través de la base de datos registro por registro hasta que se localiza el registro deseado.

Alias : Un pseudónimo opcional (otro nombre, adicional), con que nos referimos a un campo o a un archivo, cuando estamos interactuando con este. Un alias puede ser hasta de 10 dígitos de longitud. Puede contener letras, números y underscore (_). Debe comenzar con letra y no puede tener espacios intermedios.

Ancho del Campo : El número de caracteres que caben en un campo.

Apuntador al Registro : Un número utilizado por Dbase III para identificar el registro de la base de datos sobre el que está posicionado.

Archivo : Siempre se refiere a un archivo en disco. Existen algunos comandos, que siempre van seguidos del nombre del archivo, para indicar que el comando afecta directamente a todo el archivo (por ejemplo, DELETE FILE archivo).

Archivo Activo : El archivo tipo base de datos con que se está trabajando en el área de trabajo seleccionada.

Archivo de Comandos : Un programa de computadora, escrito con comandos de dBase IV y almacenado como un archivo en disco.

Archivo de Memoria : Un archivo en disco utilizado para almacenar en forma permanente las variables de memoria.

se refiere al comando actual o a todos los registros de la base de datos, según sea el caso. Las frases FOR/WHILE habilitan el comodín de default a todos (ALL). El comodín puede ser:

RECORD n: un registro solamente.

NEXT n: n registros comenzando con el actual.

ALL: Todos los registros de la base de datos.

REST n: Todos los registros que restan a partir del actual.

Concatenación : La unión de dos o más cadenas de caracteres, para formar una nueva cadena que las incluye. En dBase IV la concatenación se efectúa a través del signo más (+). Por ejemplo, "2" + "2" se convierte en "22".

Condición : Lo define en forma específica a un comando que hacer. La condición misma es una comparación entre dos o más elementos. Por ejemplo en el comando DISPLAY FOR Nombre='Perez' la condición es Nombre='Perez'. El contenido del campo Nombre es comparado con la cadena de caracteres 'Perez' para cada registro en la base de datos. Solo aquellos registros para los cuales se cumple, la condición serán desplegados. Se pueden definir condiciones múltiples, mediante la utilización de los operadores lógicos .AND. y .OR.

Consola : El video monitor y el teclado.

Constante : Es un dato que no cambia durante una operación (o serie de operaciones). Las constantes pueden ser literales, o algún valor almacenado en una variable de memoria.

CRT : Abreviación de "Cathode Ray Tube". Utilizado como sinónimo de video monitor.

Ctrl : ver Tecla de Control.

Cursor : Una marca visible que nos indica la posición sobre la pantalla.

Ccadena : Una abreviación para cadena de caracteres.

DEMS : Abreviación de "Database Management System" o Sistema de Administración de Bases de Datos.

Default : Una elección preprogramada. Una acción o selección que toma dBase IV a menos que se asigne otra.

Delimitadores : Símbolos utilizados para que la computadora pueda identificar en forma separada a diferentes elementos (por ejemplo DISPLAY Nombre,Dirección,Ciudad los elementos están separados por comas). O bien para identificar el contenido de las cadenas de caracteres (por ejemplo, STORE "Hola" TO X), esto se logra con las comillas sencillas o dobles.

Directorio : El catálogo de archivos de un disco.

Archivo Index : Un archivo index o archivo de índices, es un archivo de propósito especial que relaciona una llave (que es un elemento de interés para el usuario (un campo o expresión de campos), a los números de registro de la base de datos. Las llaves index están en orden ascendente.

ASCII : Abreviación de "American Standard Code for Information Interchange". Código de siete bits que define 128 caracteres estándar, dentro de los que se incluyen: caracteres de control, letras, números, y símbolos especiales.

Atributo : Una característica descriptiva. En este manual el atributo se refiere específicamente a las características de despliegue del video monitor, tales como: intensidad, inverso o no y subrayado.

Cadena : ver Cadena de Caracteres

Cadena Nula : Una cadena de caracteres conteniendo solo el carácter de nulo (ASCII 0). Esta cadena tiene una longitud de cero. Todas las comparaciones contra esta cadena aparecerán como ciertas.

Cadena de Caracteres : Una secuencia de números, letras y símbolos. La secuencia Medellín 184, Despacho 105 Col. Roma es una cadena de caracteres. El contenido de un campo tipo carácter o de una variable de memoria de este tipo, puede normalmente ser substituida por cadenas de caracteres. Las cadenas de caracteres están normalmente encerradas entre comillas u otros delimitadores.

Campo : Contiene un elemento identificado de información en un archivo tipo base de datos.

Campo LLave : Un campo o expresión por el cual se basa dBase IV, para ordenar (SORT), indexar (INDEX), unir (JOIN), actualizar (UPDATE), o totalizar (TOTAL).

Campo Numérico : Contiene números que pueden ser utilizados directamente en cálculos aritméticos.

Carácter : Un símbolo posible de imprimir. Dentro de los caracteres se incluyen las letras, los números, y algunos símbolos especiales.

Columna : Un número el cual es utilizado para indicar la posición horizontal de inicio (en pantalla o en impresora), los valores de las columnas se incrementan de izquierda a derecha comenzando con el valor de cero. Sobre la impresora, el valor de la columna se suma al valor del margen izquierdo que se tenga (ver SET MARGIN).

Comando : Una instrucción de dBase IV a la computadora. Estos se describen en la sección de comandos.

Comodin : Una parte opcional de muchos de los comandos, que especifica el monto de la base de datos, para la cual se aplica el comando. Cada comando que tiene la opción de comodin tiene un valor de default que

DOS : Abreviación de "Disk Operating System" que es el sistema operativo que trabaja sobre las PC actuales.

Drive : Indica la identificación del controlador de discos (por ejemplo, A o B). Cuando se utiliza como parte de un nombre de archivo, el identificador del drive (controlador) deberá tener a continuación dos puntos (por ejemplo, A:).

Enter : Utilizada en dos contextos: (1) Digitar un comando o un dato como respuesta a un mensaje que así nos lo indique. (2) como un sinónimo de la tecla RETURN.

Entero (Integer) : Un número que no tiene parte fraccional.

Entrada (Input) : Datos proporcionados por el usuario a través del teclado.

Esc : ver Tecla Esc.

Estructura : La definición de los campos de la base de datos.

Exp : Una abreviación de expresión.

Expresión : Puede consistir de un campo, una variable de memoria, una función, una constante, o cualquier combinación válida de los elementos anteriores. La abreviación utilizada para expresión es, exp. Las letras C, D, L, o N son letras que precederan la abreviación, para indicar un tipo específico de expresión. Por ejemplo Hexp, se refiere a una expresión numérica.

Extensión : Un punto y hasta 3 caracteres que se adicionan al final de un nombre de archivo.

Fin de Archivo : Un caracter ASCII utilizado para indicar el final de un archivo que esta almacenado sobre un disco.

FOR <condición> : Una frase opcional en muchos comandos, la cual, indica que el comando se aplicara a cada registro de la base de datos que cumpla con la <condición> establecida.

Form Feed : Se refiere a (1) una acción que provoca que el papel en una impresora avance a la siguiente página. (2) un caracter de control ASCII.

Función : Una operación de proposito especial, que puede ser utilizada en forma independiente o conjuntamente con algunos comandos. Ver el apéndice de funciones de estas notas.

Impresión : Indica a la impresora como parte de un comando. TO PRINT es una parte opcional de algunos comandos, que habilita la impresora mientras el comando esta activo. La impresora tambien puede ser habilitada (ON) o inhibida (OFF) a través del comando, SET.

Indicador (Prompt) : Un símbolo, Una palabra, o una frase descriptiva, que se utiliza para indicar que la computadora esta lista para recibir datos o comandos como entrada.

Literal : Una valor explicito que no puede sufrir alteración. Por ejemplo, el verbo de un comando a continuación del indicador ".", es una literal.

Lista de Campos : Consiste de uno o mas nombres de campo, cuando, mas de un nombre de campo se utiliza, los nombres deben de estar separados por coma.

Lista de Expresiones : Consiste de una o mas expresiones. Si mas de una expresión esta presente, deberan las expresiones ser separadas por comas.

Lista Var Mem : Consiste de una o mas nombres de variables de memoria. Si se enuncia mas de un nombre de variable, los nombres de las variables, se separan por comas.

LLave : Ver Campo LLave.

Menú : Una lista de opciones.

Menú-driven : Un proceso que permite al usuario seleccionar un curso de acción para lograr un objetivo, basandonos en una trayectoria de opciones elegidas a traves de varios menus. Help y Assist son ejemplos de sistemas menu-driven.

Monitor : Sinónimo de Video.

n : Indica un número (por ejemplo, 2137). En la mayoría de los casos, una expresión numérica (por ejemplo, Costo * Porcentaje), puede ser utilizada de la misma forma que cualquier cifra numérica, por ejemplo 123.

Nombre de Archivo : El nombre de un archivo del disco, Un nombre de archivo puede consistir de hasta 8 caracteres, formado por letras, numeros, y underscores (_). Pero comenzando siempre con una letra, y sin tener espacios en blanco intermedios.

Nombre del Campo : El nombre asignado a un campo. Un nombre de campo, puede ser de hasta 10 caracteres, que pueden ser: letras números, y underscore (_), debera comenzar con una letra y no puede contener espacios en blanco.

Número de Registro : Un número único de identificación, asignado por dBase IV a cada registro de la base de datos.

Operador : Un símbolo que indica la forma en que seran relacionados dos elementos. Por ejemplo el signo mas (+), como operador aritmético significa adición.

Orden Ascendente : Ordenamiento en el que se comienza con el valor mas pequeño y se termina en el valor mas grande (por ejemplo, de A a Z o de 0 a 9). En dBase IV el orden ascendente es con respecto al valor ASCII.

Orden Descendente : En esta forma de ordenar se comienza con el valor mas grande y se termina con el valor mas pequeño (por ejemplo, Z a A o de 9 a 0). En dBase IV se toma en cuenta el valor inverso del orden ASCII.

Orden Lógico : El orden de los registros establecido en el archivo index. Por ejemplo si un campo de los registros de la base es de la base, sera alfabético y se establecerá con respecto a los nombres que contenga este campo en todos los registros de la base.

Parámetro : Se refiere a (1) características ajustables de procesamiento ajustables por dBase IV y que son controladas a través de los comandos SET. Por ejemplo el status de la impresora (ON, u OFF) son parametros de procesamiento. (2) PARAMETERS es tambien un comando que asigna nombres, a variables de memoria, campos de registros, y expresiones pasadas a los programas a través del comando DO.

Path : Se refiere a la ruta especificada de busqueda de archivos que no se encuentren en el directorio sobre el que estamos trabajando. Ver manual del Sistema Operativo.

Posiciones Decimales : El número de digitos a la derecha del punto decimal.

Procedure : Un programa de aplicación especial de dBase III PLUS. Puede contener hasta 32 programas, cada uno debe comenzar con la palabra PROCEDURE. Su función es reducir el número de pequeños programas, para formar un archivo mas grande que los contenga.

Programa : Una serie ordenada de instrucciones de software, diseñada para lograr que la computadora ejecute una secuencia de acciones.

Query : Se refiere a consultas a la base de datos a través del teclado.

Registro : Unidad básica de información direccionable de que se compone un archivo.

Registro Actual : El registro sobre el cual esta posicionado el apuntador, o el registro que se esta utilizando.

Registros Lógicos : Son los registros de un archivo index o similar, que consiste de una imagen parcial del contenido de los registros del archivo tipo base de datos.

Ren glón : Un número que es utilizado, para indicar la línea sobre el video o la página de impresión. Los renglones se numeran de arriba hacia abajo comenzando con el valor de cero.

Respaldo (Backup) : El proceso de copiar en forma total o parcial un disco, sobre otro disco, para asegurarnos de que en caso de pérdida o falla del primer disco, tenemos un segundo disco del cual podemos obtener la información, para nosotros valiosa.

Salida : La transferencia de información de la computadora hacia los dispositivos externos, como por ejemplo, la impresora y el video monitor.

Sintaxis : La estructura de los comandos.

Sistema de Base de Datos Relacional : Sistema de administración de bases de datos, donde la base de datos consiste de una o mas tablas rectangulares de renglones y columnas. Los registros de datos son ligados (referencias cruzadas) por el contenido de columnas que son comunes entre los diferentes archivos.

Skeleton : Una mascara utilizada para definir nombres de archivos o de variables de memoria, que tienen elementos comunes. El skeleton utiliza los caracteres ?, y * para generalizar partes de los nombres con elementos comunes. Por ejemplo DIR *.idx despliega el nombre de todos los archivos index. El asterisco se aplica, a cadenas de caracteres de diferentes longitudes. El caracter ?, se aplica para generalizar caracteres individuales. Por ejemplo ??SE.* despliega todos los nombres de archivos de cualquier tipo, pero con una longitud en el nombre de 4 caracteres y que tengan, la terminación SE.

Tecla de Control : Una tecla que asigna otra función a cada letra del teclado. La forma de utilizarla es como la tecla de SHIFT. Se digita y se mantiene presionada mientras se digita la letra requerida. En las notas se representa como "^" (por ejemplo, ^A significa Control A).

Tecla Esc : Tecla que genera el código ASCII 27. dBase IV utiliza la tecla Esc para abortar la ejecución de un comando o de un programa.

Template : Una mascara que ayuda en la definición y delimitación, de los datos de entrada. El Template 99/99/99 sirve como una ayuda en la captura de fechas dentro de variables tipo caracter. Los digitos 9 indican que solo admite datos numericos entre 0 y 9 y las diagonales, son parte del campo y me ayudan para un mejor despliegue visual.

Tipo de Campo : Indica la clase de datos que pueden ser almacenados en un campo. Los tipos de campo son: (C) caracter, (D) fecha (L) Logicos, (N) numéricos, y (M) memo.

TO : Una parte de muchos comandos de dBase IV, que indica el destino del comando. TO normalmente es seguido por un nombre de archivo o nombre de variable de memoria. Algunos comandos utilizan, TO como un conector natural, para especificar una característica de procesamiento (por ejemplo, SET COLOR TO).

Toggle : Se refiere a aquellas situaciones donde el mismo comando provoca que se efectue un cambio entre dos estados estables, posibles. El estado habilitado sera dependiente del estado en que se encuentre la maquina en el momento en el que el comando sea ejecutado. Por ejemplo *P habilita o inhibe la impresora, dependiendo del estado de la impresora cuando el comando se efectue.

Truncar : acción de eliminar digitos en numeros y caracteres en cadenas de caracteres. Por ejemplo 4 es el resultado de efectuar el truncamiento sobre 4.342. "cadena" es el resultado de efectuar el truncamiento sobre "cadena de caracteres".

Variable : Indica : (1) El nombre de un campo o variable de memoria, o (2) un elemento posible de cambiar.

Var Mem : Una abreviación para variable de memoria.

Variable de Memoria : elementos identificables que contienen datos importantes y no forman parte de la base de datos. Las variables de memoria, son normalmente elementos temporales. Pueden ser almacenadas en disco y restablecidas a la memoria a traves de los comandos SAVE TO y RESTORE FROM respectivamente utilizando para este fin los archivos de memoria (.MEM).

WHILE <condición> : Un elemento opcional en muchos comandos, los cuales indican que el comando sera repetido, siempre que el resultado de la condición sea verdadero (TRUE). Cuando WHILE <condición> se utiliza en un comando que se aplica a los registros de una base de datos (tal como LIST), cada registro es evaluado en secuencia, y el comando termina cuando se termine la condición del comodín o cuando la condición del WHILE evaluada sea falsa.

WITH : Utilizada en aquellos comandos donde el contenido del campo esta siendo substituido con algun nuevo valor. WITH es tambien utilizado para indicar que se estan pasando parametros a un programa a traves de utilizar el comando DO.

MENSAJES DE ERROR.

Este Apéndice contiene todos los mensajes de error en orden alfabético. El número de error asociado a cada mensaje, se muestra en paréntesis cuadrados al final del mensaje. Estos números no aparecen en la pantalla con el mensaje del error, pero pueden ser identificados mediante el uso del comando ON ERROR y usando la función ERROR() para obtener el número del error.

Los mensajes sin número de error son originados por el Centro de Control y no pueden ser identificados.

Se sugieren acciones correctivas cada que sea posible.

Replace query must have at least one WITH operator [370]
Se trató de utilizar un "query" de actualización (de reemplazo) y no hay declaración WITH.

A UNIQUE aggregate must be the only aggregate in a QUERY [324]
Un QUERY agregado puede tener solo un UNIQUE.

ACOS(): Out of range [293]
El valor de la expresión que se utilizó con la función de arcocoseno está fuera del rango de -1.0 a +1.0.

ALIAS expression not in range [232]
Se está utilizando una expresión de alias fuera del rango de 1 a 10.

ALIAS name already in use [24]
Se trató de utilizar USE para un archivo de base de datos que ya estaba abierto, tiene el mismo alias, o tiene un nombre o alias dentro del rango de default de A hasta J.

ALIAS not found [133]
Trató de usarse SELECT para seleccionar una área de base de datos fuera del rango de A hasta J o de 1 a 10, o se utilizó un alias indefinido.

All allowed slots have been filled
Se ha utilizado el máximo número de indexaciones permitido: 10 .ndx y 47 .mdx.

All database must be closed before using PROTECT [173]
Cerrar las bases de datos antes de intentar hacer uso de PROTECT.

ASIN(): Out of range [291]
El valor de la expresión utilizada con la función arcoseno está fuera del rango de -1.0 a +1.0.

ATAN(): Out of range [294]
El rango permitido de ángulos en radianes para la función arcotangente está entre $+\pi/2$ y $-\pi/2$.

MENSAJES DE ERROR

- Bad array dimension(s)** [230]
Este mensaje indica que se utilizó un valor ilegal (tal como cero) durante la declaración de un arreglo, o se declaró un arreglo que excede los 1,023 elementos.
- Bad EXPRESSION** [40]
La sintaxis de una expresión es ilegal en un comando DO WHILE, IF o CASE.
- Bad PROCEDURE name** [32]
Se está tratando de usar un "procedure" que dBASE IV no puede encontrar. Verificar por completo las especificaciones del archivo DOS de este archivo "procedure".
- BAR position must be a positive number** [167]
Se debe definir las barras de un menú o "popup" usando solamente números positivos.
- Beginning of file encountered** [38]
Se está intentando realizar una búsqueda hacia atrás en un archivo de base de datos en el que ya se estaba al principio de la base de datos.
- Branching must end before the first @ command** [308]
En un archivo de formato, la primera sección de inicialización puede contener cualquier comando, pero cualquier estructura de control, tal como un IF, debe terminarse con un ENENDIF antes de que la segunda sección, la cual contiene solamente comandos @, dé inicio. Verificar el archivo de formato y dar por terminado las ramificaciones del programa antes de empezar con la sección de comandos @.
- Calculate field requires an expression**
Se trató de crear un campo calculado sin una expresión. Presionar Ctrl-End in el submenú de validación, y así no habrá expresión ejecutada.
- Calculated fields in the VIEW must not be empty** [323]
Se borró una expresión de un campo calculado que estaba en un VIEW.
- Cannot add an empty or erroneous calculated field to the view**
Se intentó sumar un archivo calculado inválido para el VIEW.
- Cannot add a group band to this type of band**
Esto ocurre cuando se trata de sumar una banda de grupo a una banda que no es banda de cabeza de página, banda de reporte intro o banda de grupo intro.
- Cannot add fields to the view in an update query**
Se intentó sumar campos al VIEW siendo que hay un comando de actualización en el pothandel del esqueleto de un archivo.

- Cannot append in column order** [147]
Se intentó utilizar APPEND desde una hoja de cálculo MultiPlan (formato SYLK) cuyos renglones no están en orden ascendente. Verificar si las columnas de la hoja de cálculo están en orden antes de reutilizar este comando.
- Cannot clear menu in use** [176]
Un menú activo en pantalla no puede ser limpiado mediante los comandos CLEAR MENU o RELEASE MENU.
- Cannot clear popup in use** [177]
Un menú "pop-up" activo en pantalla no puede ser limpiado mediante los comandos CLEAR POPUP o RELEASE POPUP.
- Cannot close database when transaction is in process** [185]
Debe completarse una transacción y utilizar un comando END TRANSACTION, o ejecutar un ROLLBACK antes de que se pueda cerrar la base de datos involucrada en la transacción.
- Cannot close index files when transaction is in process** [187]
Debe completarse una transacción y utilizar un comando END TRANSACTION, o ejecutar un ROLLBACK antes de que se pueda cerrar los archivos indexados involucrados en la transacción.
- Cannot create a link in the pothandle**
El cursor estaba en el pothandle cuando se seleccionó Create link por posicionamiento (diseño query).
- Cannot create more than 20 calculated fields**
Se trató de sumar el campo calculado número 21 al esqueleto de campo calculado.
- Cannot create! SQL table exists with the same name** [374]
Se está tratando de crear un archivo .dbf, y existe una tabla SQL con el mismo nombre. Seleccionar un nombre de archivo diferente para la base de datos.
- Cannot delete SQL created TAG: <tagname>** [376]
Se puede borrar un SQL created TAG solo durante el modo SQL interactivo.
- Cannot enter SUM or AVERAGE in a character field**
No se puede utilizar SUM o AVERAGE en un campo de carácter en QBE.
- Cannot erase open file** [89]
Cerrar los archivos de base de datos o de indexación antes de intentar borrarlos.
- Cannot erase a read-only file** [336]
No puede borrarse un archivo que se ha accedido por read-

only. Ver el administrador del sistema para verificar el nivel de prioridad de archivos.

Cannot execute this command while transaction is in process [186]
Ver la descripción del comando BEGIN TRANSACTION para identificar la lista de comandos que no son permitidos durante esta transacción.

Cannot go to Browse/Edit if errors exist in the query design[344]
Este mensaje ocurre cuando se ha intentado utilizar Browse/Edit siendo que hay un error en el diseño del query.

Cannot have more than one aggregate followed by UNIQUE
Se tiene más de un agregado seguido de una cláusula UNIQUE.

Cannot have more than one aggregate operator in a column [380]
Se tiene más de un operador agregado en una columna.

Cannot have more than one GROUP BY in a column [339]
Se introdujo dos veces GROUP BY en una columna.

Cannot insert. Expression length at limit [139]
Solo se pueden unir dos archivos de base de datos diferentes. Se necesita nombrar otra base de datos mediante USE.

Cannot link a file skeleton to itself [345]
Se intentó ligar dos campos en el mismo esqueleto del archivo.

Cannot load more than eight files in query design
Puede utilizarse un máximo de ocho esqueletos de archivo en QBE.

Cannot modify SQL table [375]
No puede modificarse una tabla SQL en el modo DBASE IV; debe utilizarse un modo SQL interactivo.

Cannot move empty field
Este es un error del Centro de Control de la base de datos. Se obtiene este mensaje si se presiona la tecla F7 en un campo en blanco en la tabla en sort.

Cannot MOVE or COPY without a defined selection
Se intentó mover o copiar sin haber definido de antemano el rango seleccionado.

Cannot move the cursor beyond bottom of window
Se trató de mover el cursor fuera de la ventana de pop-up por vez primera. Si se intenta mover el cursor en la misma dirección otra vez, la ventana queda cerrada.

Cannot move the cursor beyond left edge of window
Se trató de mover el cursor fuera de la ventana de pop-up

por vez primera. Si se intenta mover el cursor en la misma direccióm otra vez, la ventana queda cerrada.

Cannot move the cursor beyond right edge of window

Se trató de mover el cursor fuera de la ventana de pop-up por vez primera. Si se intenta mover el cursor en la misma direccióm otra vez, la ventana queda cerrada.

Cannot move the cursor beyon top of window

Se trató de mover el cursor fuera de la ventana de pop-up por vez primera. Si se intenta mover el cursor en la misma direccióm otra vez, la ventana queda cerrada.

Cannot nest transactions

[198]

Las transacciones no pueden ser anidadas dentro de otras transacciones. Utilizar un comando END TRANSACTION antes de dar comienzo a una nueva transacción.

Cannot open <filename> resource file

[398]

Este archivo fuente específico (.res) no puede ser leído. Este es un error fatal y trae como resultado el término de DBASE IV. Puede tratar de localizarse este archivo .res en los disketes del programa original y copiarlo del diskette al directorio desde el cual se inició DBASE IV en el disco duro. Si no puede ser localizado, llamar al Centro de Apoyo de Software de Ashton-Tate para solicitar ayuda, ya que DBASE IV no funcionará sin este archivo.

Cannot overwrite SQL TAG: <tagname>

[377]

No se puede sobrescribir en un SQL created TAG excepto en el modo SQL interactivo.

Cannot place a link in a field that has an error

[346]

Se está tratando de ligar un campo que tiene un error.

Cannot recover the damaged index file

[364]

Los contenidos de un archivo indexado están dañados. Ejecutar un nuevo comando INDEX para crear un nuevo índice para el archivo de base de datos.

Cannot re-define menu in use

[174]

Se está intentando utilizar un comando DEFINE MENU siendo que no hay menu activo en memoria. Utilizar un comando DEACTIVATE MENU primero, y luego usar DEFINE MENU.

Cannot re-define popup in use

[175]

Se está intentando utilizar un comando DEFINE POPUP siendo que no hay un popup activo en memoria. Utilizar un comando DEACTIVATE POPUP primero, luego usar DEFINE POPUP.

Cannot release memvar in use: <memvar name>

[395]

No puede usarse RELEASE o CLEAR en una variable de memoria que está siendo usada en un @...GET...READ o como parámetro

- Column number be between 0 and either right margin or 255 [223]**
Se ha especificado un número de columna de impresión que está fuera de página. Reutilizar el comando usando un número de columna que esté dentro de los límites de ancho.
- Command not allowed in programs [306]**
Se ha usado un comando en un archivo de programa que no está permitido en los programas como tales (como RESUME).
- Command not allowed in SQL [307]**
Este comando de dBASE IV no está permitido en modo SQL.
- Command not allowed in a user defined function [384]**
Ver Capítulo de Fundamentos, para verificar la lista de comandos permitidos en funciones definidas por el usuario.
- Command not functional in dBASE IV [93]**
Se está usando un comando tal como SET MENU o SET DOHISTORY que es no funcional en dBASE IV, pero que ha sido retenido para compatibilidad anterior.
- Command not supported by RunTime [290]**
Ciertos comandos tales como MODIFY SRSTRUCTURE y CREATE SCREEN no son permitidos en archivos runtime. Se tiene un archivo de comandos que tiene alguno de estos.
- Command not valid in RunTime environment [348]**
El comando que se está tratando de usar no puede ser utilizado durante runtime.
- Command only valid in programs [260]**
Se está tratando de usar un comando que solamente es permitido en programas, en el punto prompt. Escribir un programa o un procedimiento si se quiere usar este comando.
- Command too long, press CTRL-HOME to edit in zoom window [402]**
Sólo puede editarse un comando de 254 caracteres sobre la línea de comando. Si se necesita una línea más larga, presionar Ctrl-Home para editar el comando en la ventana ampliada.
- Command will never be reached [369]**
Se tiene un comando en el programa que no puede ser ejecutado debido a que está localizado después de un comando RETURN o EXIT. Los comandos localizados después de la terminación del programa nunca pueden ser ejecutados.
- Compilation error [360]**
Un archivo de programa no se compilará debido a errores de sintaxis. Corregir los errores en el archivo de programa y recompilar.

de un PROCEDURE o FUNCTION.

Cannot select requested database [17]

Se ha seleccionado un número inválido de área de trabajo. Usar un número de 1 a 10 o una letra de A a J.

Cannot use a sort priority number higher than nine

Este es un error del Centro de Control. Se obtiene este mensaje cuando se ha introducido un número de prioridad de sort que ha sido mayor que nueve, debido a que no se puede utilizar un sort sobre más de nueve campos.

Cannot write to a read-only file [111]

Se está tratando de escribir un archivo read-only.

Cannot write to database due to incomplete transaction [201]

Ha ocurrido una transacción incompleta en la base de datos conforme se ha usado. Si no se empezó con la transacción, verificar si la base de datos está involucrada en una transacción con la función COMPLETED(). Si es falso, (.F), esperar hasta que la transacción sea completada. Si se ha empezado la transacción, utilizar ya sea END TRANSACTION para salvar cambios o un comando ROLLBACK para deshacer los cambios y limpiar la integridad de la bandera del archivo de base de datos.

Cannot write to transaction log file [188]

El archivo de transacción log requerido no está abierto o disponible. Utilizar un comando END TRANSACTION para abandonar la transacción que se ha intentado.

Catalog has not been established [122]

Un catálogo debe ser creado antes de ser utilizado.

CHANGE(), not enough memory [161]

Este error aparece cuando no hay memoria disponible para ejecutar una función CHANGE().

CHANGE(), record locked by another [160]

Este error aparece cuando se intenta utilizar la función CHANGE() para determinar si el registro ha sido cambiado y otro usuario tiene el registro protegido.

CHR(): Out of range [57]

El argumento proporcionado a la función CHR() es menor que 0 o mayor que 255. Este rango representa la extensión de caracteres ASCII de IBM.

Column is full (255 character maximum)

Este mensaje aparece cuando se selecciona Create link by pointing o Sort on this field para una columna que no tiene suficientes caracteres para acomodar la longitud de la variable ejemplo o de la dirección del sort.

Condition box cannot be extended any further

Se ha alcanzado el la region del fondo de la caja de condición.

Condition box has not been created yet

Se ha seleccionado Show condition box desde un menu Condition, siendo que no se ha creado la caja de condición.

Conflicting data types

En un diseño query, se ha introducido una expresión que tiene conflicting data types.

CONTINUE with LOCATE

[42]

No se puede utilizar CONTINUE sin ejecutar un comando LOCATE.

Control Code Exceeds 255

[401]

No se puede tener un código de control de impresión que exceda los 255 caracteres.