

SZLOVÁK GÁBOR • TÓTH TIBOR • KŐRI GÁBOR

Adatbázis-kezelés, programozás dBASE IV-ben

2. kötet



**A dBASE IV adatbáziskezelő rendszer
programozása**

II. kötet

Függelék

LSI ATSZ – aPLUS Kft, Budapest, 1989

Szerzők:

Első kézirat: Kőri Gábor 2,6,E,F,G,I,J,K,M fejezetek
Molnár M. Attila Előszó, 1,2,A,B fejezetek
Szlovák Gábor 4,5,8,C,L fejezetek
Tóth Tibor 3,7,D,H fejezetek

Végső változat: Molnár M. Attila
Tóth Tibor

Szerkesztő: Molnár M. Attila

Lektorálta: Apor György

Copyright © aPLUS Kft.

ISBN: 963-592-973-0 I. kötet
963-592-974-9 II. kötet
963-592-975-7 összkötet

Megjelent az **LSI ATSZ** és az **Intersoft Kft.** közös kiadásában.

Felelős kiadó: Dr. Kovács Magda és Apor György

Témafelelős: Salgóné Sziklai Klára és Szűcs Ernő

A kézirat és szedés az **Intersoft Kft.** műhelyében készült.

Műszaki szerkesztő: Molnár M. Attila

A kiadványt az **InterSoft Kft.** menedzselte és finanszírozta.

Készült a Nógrád megyei Nyomdaipari Vállalat balassagyarmati üzemében.

Felelős vezető: Kelemen Gábor igazgató

Tartalomjegyzék

A) Néhány fogalom magyarázata	F-15
B) Jelölésrendszer, szimbólumok, konvenciók	F-19
C) Utasítások	F-23
C-1. Adat- és fájlkezelés	F-23
APPEND [BLANK]	F-23
APPEND FROM	F-23
APPEND FROM ARRAY	F-24
APPEND MEMO	F-24
AVERAGE	F-24
BROWSE	F-25
CALCULATE	F-26
CHANGE	F-27
CLOSE	F-27
COPY FILE	F-27
COPYINDEXES	F-28
COPY MEMO	F-28
COPYSTRUCTURE	F-28
COPYSTRUCTUREEXTENDED	F-29
COPY TAG	F-29
COPY TO	F-29
COPY TO ARRAY	F-30
COUNT	F-30
CREATE vagy MODIFY STRUCTURE	F-30
CREATE	F-31
CREATE VIEW ..FROM	F-32
DELETE	F-32
DELETETAG	F-32
EDIT	F-33
EXPORT	F-33
GO/GOTO	F-34
IMPORT	F-34
INDEX ON	F-35
INSERT	F-35
JOIN WITH	F-36
LOAD	F-36
PACK	F-36

RECALL	F-37
REINDEX	F-37
RENAME	F-37
REPLACE	F-37
ROLLBACK	F-38
SELECT	F-38
SKIP	F-38
SORT TO	F-39
SUM	F-39
TOTAL ON	F-40
UPDATE ON	F-41
USE	F-42
ZAP	F-42
C-2. Kereső utasítások	F-43
CONTINUE	F-43
FIND	F-43
LOCATE	F-43
SEEK	F-44
C-3. Adatmegjelenítés, adatbevitel	F-45
???	F-45
@ ...SAY ...GET	F-46
@ ...CLEAR	F-50
@ ...FILL	F-50
@ ...TO	F-51
LABEL FORM	F-51
LIST/DISPLAY	F-52
LIST/DISPLAY FILES	F-53
LIST/DISPLAY HISTORY	F-53
LIST/DISPLAY MEMORY	F-53
LIST/DISPLAY STATUS	F-54
LIST/DISPLAY STRUCTURE	F-54
LIST/DISPLAY USERS	F-54
TEXT	F-55
READ	F-55
REPORT FORM	F-55
TYPE	F-56
C-4. Értékadó utasítások	F-57
ACCEPT	F-57
INPUT	F-57
STORE	F-57
WAIT	F-58
C-5. Változók, szimbólumok kezelése	F-59

CLEAR	F-59
DECLARE	F-60
PARAMETERS	F-60
PRIVATE	F-60
PUBLIC	F-61
RELEASE	F-61
RESTORE FROM	F-62
SAVE TO	F-62
C-6. Vezérlő szerkezetek	F-63
BEGIN TRANSACTION	F-63
CALL	F-63
CANCEL	F-63
DO	F-64
DO CASE	F-64
DO WHILE	F-65
FUNCTION	F-65
IF	F-66
ON ERROR	F-66
ON ESCAPE	F-67
ON KEY	F-68
ON PAD	F-69
ON PAGE	F-69
ONREADERROR	F-70
ON SELECTION PAD	F-70
ON SELECTION POPUP	F-71
PRINTJOB	F-71
RESUME	F-71
RETRY	F-72
RETURN	F-72
SCAN	F-73
SUSPEND	F-73
C-7. Ablak- és menükezelés	F-74
ACTIVATE MENU	F-74
ACTIVATE POPUP	F-74
ACTIVATE SCREEN	F-74
ACTIVATE WINDOW	F-75
DEACTIVATE MENU	F-75
DEACTIVATE POPUP	F-75
DEACTIVATE WINDOW	F-76
DEFINE BAR	F-77
DEFINE MENU	F-78
DEFINE PAD	F-78
DEFINE POPUP	F-79

DEFINE WINDOW	F-80
MOVE WINDOW	F-80
RESTORE WINDOW	F-80
SAVE WINDOW	F-81
SHOW MENU	F-81
SHOW POPUP	F-81
C-8. Tervező utasítások	F-82
CREATE/MODIFY APPLICATION	F-82
CREATE/MODIFY LABEL	F-82
CREATE/MODIFY QUERY/VIEW	F-82
CREATE/MODIFY REPORT	F-83
CREATE/MODIFY SCREEN	F-83
MODIFY COMMAND/FILE	F-84
C-9. Különleges utasítások	F-85
???	F-85
ASSIST	F-86
COMPILE	F-86
CONVERT	F-87
DEBUG	F-87
DEFINE BOX	F-88
DIR	F-88
EJECT	F-88
EJECT PAGE	F-89
ERASE	F-89
HELP	F-89
LOGOUT	F-89
NOTE	F-90
PLAY MACRO	F-90
PROTECT	F-90
RESET	F-91
RESTORE MACROS FROM	F-91
RUN	F-91
SAVE MACROS	F-91
UNLOCK	F-92
D) Set utasítások	F-93
SET	F-93
SET ALTERNATE	F-93
SET AUTOSAVE	F-94
SET BELL	F-94
SET BLOCKSIZE	F-94
SET BORDER	F-95

SET CARRY	F-95
SET CATALOG	F-95
SET CENTURY	F-96
SET CLOCK	F-97
SET COLOR	F-97
SET CONFIRM	F-98
SET CONSOLE	F-98
SET CURRENCY	F-99
SET CURRENCY LEFT/RIGHT	F-99
SET DATE	F-99
SET DEBUG	F-100
SET DECIMALS	F-100
SET DEFAULT	F-100
SET DELETED	F-100
SET DELIMITERS	F-101
SET DESIGN	F-101
SET DEVELOPMENT	F-101
SET DEVICE	F-101
SET DISPLAY	F-102
SET ECHO	F-102
SET ENCRYPTION	F-102
SET ESCAPE	F-102
SET EXACT	F-103
SET EXCLUSIVE	F-103
SET FIELDS	F-103
SET FILTER	F-104
SET FORMAT	F-104
SET FULLPATH	F-105
SET FUNCTION	F-105
SET HEADING	F-106
SET HELP	F-106
SET HISTORY	F-106
SET HOURS	F-106
SET INDEX	F-107
SET INSTRUCT	F-107
SET INTENSITY	F-107
SET LOCK	F-108
SET MARGIN	F-108
SET MARK	F-108
SET MEMOWIDTH	F-108
SET MESSAGE	F-108
SET NEAR	F-109
SET ODOMETER	F-109
SET ORDER	F-110

SET PATH	F-110
SET PAUSE	F-111
SET POINT	F-111
SET PRECISION	F-111
SET PRINTER	F-112
SET PROCEDURE	F-113
SET REFRESH	F-113
SET RELATION	F-113
SET REPROCESS	F-114
SET SAFETY	F-114
SET SCOREBOARD	F-115
SET SEPARATOR	F-115
SET SKIP	F-115
SET SPACE	F-116
SET SQL	F-116
SET STATUS	F-116
SET STEP	F-116
SET TALK	F-117
SET TITLE	F-117
SET TRAP	F-117
SET TYPEAHEAD	F-117
SET UNIQUE	F-118
SET VIEW	F-118
SET WINDOW	F-118

E) Függvények F-119

E-1. Egyszerű matematikai függvények F-119

ABS	F-119
CEILING	F-119
FLOOR	F-119
FIXED	F-119
FLOAT	F-119
INT	F-120
MAX	F-120
MIN	F-120
MOD	F-120
RAND	F-120
ROUND	F-121
SIGN	F-121
VAL	F-121

E-2. Bonyolultabb matematikai függvények F-122

ACOS	F-122
ASIN	F-122

ATAN	F-122
ATN2	F-122
COS	F-123
DTOR	F-123
EXP	F-123
LOG	F-123
LOG10	F-123
PI	F-124
RTOD	F-124
SIN	F-124
SQRT	F-124
TAN	F-124
E-3. Az adatkezelés függvényei	F-125
ALIAS	F-125
BOF	F-125
COMPLETED	F-125
DBF	F-125
DELETED	F-126
EOF	F-126
FIELD	F-126
FILE	F-126
FOUND	F-127
ISMARKED	F-127
KEY	F-127
LOOKUP	F-128
LUPDATE	F-128
MDX	F-128
NDX	F-129
ORDER	F-129
RECCOUNT	F-129
RECNO	F-129
RECSIZE	F-130
ROLLBACK	F-130
SEEK	F-130
SELECT	F-130
TAG	F-131
E-4. Sztringfüggvények	F-131
ASC	F-131
AT	F-131
CHR	F-132
ISALPHA	F-132
ISLOWER	F-132
ISUPPER	F-132

LEFT	F-132
LEN	F-133
LOWER	F-133
LTRIM	F-133
MEMLINES	F-133
MLINE	F-134
REPLICATE	F-134
RIGHT	F-134
RTRIM	F-135
SPACE	F-135
STR	F-135
STUFF	F-136
SUBSTR	F-136
TRANSFORM	F-136
UPPER	F-137
E-5. Dátumfüggvények	F-137
CDOW	F-137
CMONTH	F-137
CTOD	F-137
DATE	F-137
DAY	F-138
DMY	F-138
DOW	F-138
DTOC	F-138
DTOS	F-139
MDY	F-139
MONTH	F-139
TIME	F-139
YEAR	F-140
E-6. Képernyős adatbevitel függvényei	F-141
BAR	F-141
INKEY	F-141
LASTKEY	F-144
MENU	F-144
PAD	F-144
POPUP	F-144
PROMPT	F-145
READKEY	F-145
VARREAD	F-146
E-7. Környezeti függvények	F-146
COL	F-146
DISKSPACE	F-146

FKLABEL	F-146
FKMAX	F-146
GETENV	F-147
ISCOLOR	F-147
MEMORY	F-147
OS	F-147
PCOL	F-147
PRINTSTATUS	F-148
PROW	F-148
ROW	F-148
SET	F-148
TYPE	F-149
VERSION	F-149

E-8. Hálózati üzemmód függvényei **F-150**

ACCESS	F-150
CHANGE	F-150
FLOCK	F-150
LKSYS	F-150
NETWORK	F-151
RLOCK/LOCK	F-151
USER	F-151

E-9. Különleges függvények **F-152**

&	F-152
CALL	F-152
DIFFERENCE	F-153
ERROR	F-153
FV	F-153
IIF	F-153
LIKE	F-154
LINENO	F-154
MESSAGE	F-154
PAYMENT	F-155
PROGRAM	F-155
PV	F-155
SOUNDEX	F-156

F) Rendszerváltozók **F-157**

_aligment	F-157
_box	F-157
_indent	F-158
_lmargin	F-158
_padvance	F-158

_pageno	F-159
_pbpage	F-159
_pcolno	F-159
_pcopies	F-160
_pdriver	F-160
_pecode	F-160
_peject	F-161
_pepage	F-161
_pform	F-161
_plength	F-162
_plineno	F-162
_ploffset	F-162
_ppitch	F-163
_pquality	F-163
_pscode	F-163
_pspacing	F-164
_pwait	F-164
_rmargin	F-164
_tabs	F-165
_wrap	F-165

G) SQL utasítások és függvények F-167

G-1. SQL utasítások	F-167
ALTER TABLE	F-167
CLOSE	F-167
CREATEDATABASE	F-168
CREATEINDEX	F-168
CREATESYNONYM	F-169
CREATE TABLE	F-170
CREATE VIEW	F-172
DBCHECK	F-172
DBDEFINE	F-172
DECLARECURSOR	F-173
DELETE	F-174
DROPDATABASE	F-174
DROPINDEX	F-175
DROPSYNONYM	F-175
DROPTABLE	F-175
DROP VIEW	F-175
FETCH	F-175
GRANT	F-176
INSERT	F-178
LOAD DATA	F-179

OPEN	F-179
REVOKE	F-180
ROLLBACK	F-180
RUNSTATS	F-181
SELECT	F-181
SELECT - SELECT cikkely	F-182
SELECT - INTO cikkely	F-183
SELECT - FROM cikkely	F-183
SELECT - WHERE cikkely	F-184
SELECT - GROUP BY cikkely	F-185
SELECT - UNION cikkely	F-186
SELECT - ORDER BY cikkely	F-186
SELECT - FOR UPDATE OF cikkely	F-187
SELECT - SAVE TO TEMP cikkely	F-187
SHOWDATABASE	F-188
STARTDATABASE	F-188
STOPDATABASE	F-188
UNLOADDATA	F-188
UPDATE	F-189
G-2. SQL függvények	F-190
AVG	F-190
COUNT	F-190
MAX	F-191
MIN	F-191
SUM	F-191
H) Hibaüzenetek	F-193
D) Mozgás és szerkesztés a képernyőn	F-225
J) Fájl típusok és -kiterjesztések	F-227
K) Az adatfájlok belső szerkezete	F-231
K-1. A fejrész szerkezete	F-231
K-2. A mezőleíró tömb	F-232
K-3. Memo-mezők és memofájlok	F-233
L) A dBASE IV konfigurálása	F-235
L-1. Konfiguráló utasítások	F-235
L-2. Funkciós billentyűk	F-240

L-3. Környezeti jellemzők, szolgáltatások beállítása	F-241
M) A dBASE IV technikai jellemzői	F-247
Tárgymutató	F-250

A) Néhány fogalom magyarázata

Alkalmazás, alkalmazási program : A számítástechnikai eszközök (hardver és szofver) jellemzője, hogy sokkal több célra használhatók, mint általában más technikai eszközök. Éppen ezért az ilyen eszközök, mint például a dBASE IV önmagában nem működnek, nem használhatóak úgy, mint például egy automata mosógép vagy egy autó, amelyről a köznap ember is tudja mire való. Ezeket az általános célú eszközöket meg kell tanítani arra, hogy konkrét feladatokat oldjanak meg. Ez nem más mint az alkalmazásuk egy konkrét célra. Azt a szerkezetet, azaz rendezett utasításhalmazt alkalmazási programnak nevezzük, amely egy ilyen célra az általános eszközünket használhatóvá teszi. Így például a dBASE IV egy alkalmazása egy vállalat bérszámfejtése a segítségével, és maga a dBASE IV utasításokból álló programrendszer az alkalmazási program.

ASCII szövegfájl : Egy általánosan használt, szabványos szerkezetű fájl, amelyben az információk az ASCII kódtáblázat szerint kódolt formában vannak. , kódtáblázatnak megfelelő karakterekből állnak. A fájlszerkezet és a kódtáblázat világszerte elfogadott szabvány, ezért a szoftverek (és hardverek) szinte mindegyike ezt kezelni, értelmezni képes, és általában ugyanazokat a karaktereket is jeleníti meg.

adatfájl eleje : (BOF, Beginning Of File) Az adatfájlban történő mozgás eredményeként az első rekord elé léphetünk. Ilyenkor a rekordmutató az 1-es sorszámmra mutat (akárcsak üres, vagy egyetlen rekordot tartalmazó adatfájlnál).

adatfájl vége : (EOF, End Of File) Az adatfájlban történő mozgás esetenként az utolsó rekordot is elhagyja. Ilyenkor a rekordmutató az utolsó rekordsorszámánál eggyel nagyobbat mutat. Az ún üres, vagy fantomrekord az aktuális.

argumentum : Egy függvény független változója (változói), jelen esetben az(ok) a paraméter(ek), amely(ek)et meg kell adnuk, és amely(ek)től a függvény kimenő (visszatérési) értéke függ.

bájt : A fájlhoz hasonlóan a számítástechnikában használatos *byte* szó fonetikus írása. A bájt az adat egyik legkisebb egysége (1 bájt = 8 bit), általában egy karakternek illetve az adat egy jelének felel meg. (pontosabban lásd számítástechnikai alapismereteket tárgyaló könyvekben). Az adatok terjedelmének mértékegysége is egyben. Többszöröse: 1 kilobájt = 1024 bájt, 1 Megabájt = 1024 kilobájt.

dBASE kulcsszó : lásd *kulcsszó*

Debugger : (Szabadon: "poloskátlanító") Nyomkövető, hibakeresést elősegítő eszköz (program), amely egy felhasználói program működése során lépésenként, vagy a program megadott pontjain a változók, mezők értékeiről, egyéb állapotjellem-

zokról szolgáltat információkat, úgy, hogy lehetővé teszi a működő program feltételhez kötött vagy kézi vezérlésű megszakítását.

decimális : Tízes számrendszerbeli. Számítástechnikában gyakran használatos a 16-os ún hexadecimális számrendszer is. Így az ASCII karakter-kódtáblázatban is a kódszámok általában mindkét számrendszerbeli alakban szerepelnek.

elérési út : (Path) Az az út, amelyen keresztül a kívánt fájlt el lehet érni. A teljes elérési út a meghajtó azonosítójával kezdődik, majd utána a DOS gyökérkönyvtártól lefelé haladva, egymástól a "\" jellel elválasztva kell megadni azoknak az egymásbaágyazott alkönyvtáraknak a nevét, amelyek közül az utolsó után (amiben maga a fájl található) a fájlnev szerepel. Pl: C:\PAPA\FIU\UNOKA\ez_kell.dbf.

eljáráshívás : Egy programból egy másik program, vagy programrész elindítása. Az elindított eljárás után a vezérlés a hívást követő utasításra kerül, hacsak az eljárásban más intézkedés nem történik.

FAT : (File Allocation Table) Az operációs rendszer adminisztrációs táblázata, amely a fájlok elhelyezkedési, tárolási adatait tartalmazza (Kezdőcím, stb)

forrásprogram : A programozó által az adott programnyelv szintaktikájának figyelembevételével megírt program, amely általában szövegszerkesztővel készül, de a számítógép számára nem végrehajtható. Ennek ún Compiler-rel történő lefordítása révén jön létre a számítógép számára értelmezhető ún. tárgyprogram. Ez utóbbinak kiterjesztése a dBASE IV-ben általában ".o"-ra végződik.

főprogram : Az a program(rész), eljárás, amely egy programrendszerben minden programot, programrészt közvetve, vagy közvetlenül vezérel, és így a programok hierarchiájában a csúcson áll. Rendszerint egy programrendszer indításakor ez töltődik be a memóriába.

háttértároló : Minden olyan adattárolásra szolgáló számítástechnikai adathordozó, amely nem a számítógép memóriája (RAM). Ilyenek: a merevlemez, hejlekonylemez, streamer, stb. Jellemzőjük, hogy a számítógép kikapcsolásakor is megőrzik az adatokat. Hozzá kell tenni, hogy ma már léteznek olyan memóriaegységek is, amelyek ugyanerre képesek.

interaktív üzemmód : A felhasználó által adott parancs azonnal végrehajtásra kerül, majd újra kész a rendszer a parancsfogadásra. Általában a személyi számítógépes rendszerek interaktív módon működnek.

környezet : A környezet gyakorlatilag azoknak a jellemzők és szolgáltatásoknak a halmaza, amelyek egy program futása során rendelkezésre állnak. Másszóval a program infrastruktúrájának is nevezhető. A környezet több szintű, és a szintek egymásba ágyazódnak, egymást "héjként" veszik körül. Így a "legkülső" környezet maga a hardverkonfiguráció, amely rendelkezésre áll. Az ezen futó operációs rendszer a második, és a harmadik a dBASE IV beállítható jellemzői, szol-

A) Néhány fogalom magyarázata

gáltatásai által meghatározott környezet. Ebben a könyvben a környezet szó a legutóbbit takarja.

kiterjesztés : A fájltypusok megkülönböztetésére használt háromkarakteres utótag, amelyet a fájlnevtől ponttal kell elválasztani. Pl: *filemile.dbf*. A dBASE IV a kiterjesztéseket automatikusan kezeli, de több helyütt el lehet térni az ajánlott kiterjesztésektől, azonban ilyenkor mindig a teljes fájlnevet kell az utasításokban használni.

kulcsszó : Minden olyan szó, amely valamely dBASE utasításban előfordul, így nemcsak az utasítások neve, hanem azok cikkelyeinek kulcsszavai is.

lebegőpontos számábrázolás : Olyan számábrázolási forma, amelyben a szám értékét előjel, mantissa és karakterisztika formában tárolják, amely tulajdonképpen a szám értékes jegyeinek és 10 hatványkitevőjének (a szám tízes alapú logaritmus egész részének) tárolását jelenti. Ezzel az ábrázolható számtartomány lényegesen megnő. Pl: $978 = 9,78E + 2$, azaz $9,78 * 10^2$; $0,001234 = 1,234E-3$, azaz $1,234 * 10^{-3}$.

mentés, elmentés : A fájlokban vagy memóriában lévő információk tárolása más fájlokban későbbi használat illetve helyreállítás céljából. A tárolás rendszerint a merevlemezen, illetve az ott lévő fájlok tekintetében hajlékony lemezen történik.

munkaterület : A memóriában elkülönített terület, amely az adatfájlok aktuális adatainak tárolására szolgál. (Rekordpuffer) Tíz munkaterület jelölhető ki, azaz egyszerre 10 adatfájl nyitható meg. A munkaterületekre 1 - 10 terjedő egész számokkal, vagy az A - J betűkkel lehet hivatkozni.

puffer : Átmeneti tároló a memóriában. A rendszer kezeli, de nagysága a konfiguráláskor szabályozható.

paraméter : Számítástechnikában általában a azokat az értékeket, adatokat nevezik paraméternek, amelyek egy utasítás, program, programrész, függvény — egyszóval eljárás — működését befolyásolják. A paraméterek esetenként ezek működéséhez is szükségesek. A paraméterek lehetnek bemeneti és kimeneti paraméterek. A bemenőket az eljárások indításakor kell rendelkezésre bocsátani (megadni), a kimenő paraméterek az eljárás végrehajtásának eredményeit vagy azok során keletkezett információkat tartalmazzák, amelyekhez az eljárás végrehajtása után lehet hozzáférni. A paraméterek megadása általában az eljárás meghívásakor, indításakor az eljáráshívó utasítás mögött listászerűen történik. A paraméterlistában rendszerint a sorrendiség a döntő, a paraméterértékeket tartalmazó változók, mezők neve nem lényeges.

szintaktika : Egy programnyelv utasításainak formai szerkezetére vonatkozó szabályok. A számítógép, pontosabban az interpreter vagy a fordító csak a szabályoknak megfelelő formában leírt utasításokat képes értelmezni. Egyébként hibát jelez (Syntax error).

tulajdonos-tag viszony : (owner-member) A hierarchikus szerkezet alapelemeinek viszonya. Lényege: egy dologhoz (tulajdonoshoz) több másik dolog (tag) tartozik. A tulajdonost nem szó szerint kell értelmezni. Nevezik még szülő-gyermek (parent-child), apa-fiú, forrás-cél (source-target) kapcsolatnak is.

B) Jelölésrendszer, szimbólumok, konvenciók

Az alábbiakban elsősorban a különböző utasításokban használt jelölésrendszert, szimbólumokat magyarázzuk. Ezeknél igyekeztünk az eredetihez alkalmazkoddni.

Az utasítások általában az alábbi szerkezetet követik:

Utasítás kulcsszó [*<kifejezés-lista>*] [*<hatáskör>*]
[FOR *<feltétel>*] [WHILE *<feltétel>*]
[TO FILE *<filename>*]/TO PRINTER
/TO ARRAY *<tömblista>*/TO *<változó>*]
[ALL [LIKE/EXCEPT *<minta>*]] [IN *<másodnév>*]

A függvények formai szerkezete:

Függvéynév(*[paraméterlista]*)

Utasítás-kulcsszó	Az utasítások kulcsszóval kezdődnek. Ez azonosítja magát az utasítást.
[]	A szögletes zárójelek olyan cikkelyeket, paramétereket foglalnak egybe, amelyek nem kötelezőek, azaz opcionálisak. Az utasításban a szimbólumokat el kell hagyni.
< >	a két szimbólum között megjelölteket (változók, mezők, nevek, stb) kötelező megadni. Az utasításban a szimbólumokat el kell hagyni. A jellel elválasztott opciók közül egyet (de csakis egyet) lehet választani.
<hatáskör >	Az utasítás által elérhető, feldolgozandó rekordok számát határozza meg. A hatáskört az alábbiak szerint lehet megadni: RECORD <i><n></i> esetén az <i>n</i> -ik sorszámú rekordra, NEXT <i><n></i> esetén a következő <i>n</i> számú rekordra, ALL esetén <i>minden</i> rekordra REST az aktuális rekordtól a fájl végéig sorakozó rekordokra terjed ki. A NEXT és REST az aktuális index figyelembevételével jelöli ki a hatáskört.
FOR <i><feltétel ></i>	A <i>feltételnek</i> megfelelő rekordok vesznek részt az utasítás által meghatározott műveletben.

WHILE <feltétel >	Az utasítás által meghatározott műveletek csak addig kerülnek végrehajtásra, amíg a <i>feltétel</i> teljesül.
TO	Az utasítás kimenő adatainak rendeltetési helyét határozza meg. A FILE valamilyen fájlt, a PRINTER a nyomtatót, az ARRAY pedig tömböt jelöl ki.
<változó >	Ideiglenesen a memóriában tárolt adat, amelyre névvel lehet hivatkozni. Létrehozása értékadó utasítással történhet.
<tömb >	Azonos típusú és jelentésű változók sorozata, amelyre egy névvel és sorszámmal (kétdimenziós tömb esetén sor- és oszlopszámmal) lehet hivatkozni. A sor (és oszlopszámot) a tömb nevét követő szögletes zárójelekbe kell foglalni.
[ALL [LIKE /EXCEPT <minta >]]	A műveletben résztvevő fájl/mező/változó <i>név</i> alapján történő kiválasztására szolgál. LIKE esetén a mintával egyező, EXCEPT esetén a mintától különböző nevű fájl/mező/változó kerül kiválasztásra. A mintában a ? egy, a * egyszerre több olyan karaktert helyettesít, amely a kiválasztás szempontjából érdektelen, így a helyén bármi állhat a nevekben.
[IN <másodnév >]	A kiválasztott munkaterület <i>másodneve</i> , amellyel a munkaterületen megnyitott fájlra is lehet hivatkozni. A másodnév helyén állhatnak a munkaterület azonosítójaként az A-J, 1-10 jelek is.
<kifejezés >	Mezők, változók, rendszerváltozók, műveleti jelek (operátorok), függvények, konstansok tetszőlegesen, de értelmezhető (legális) kombinációjából állhat. Végrehajtása egy értéket szolgáltat, amelynek típusa lehet <i>numerikus</i> ez az N-kifejezés (F és N típusú), <i>karakteres</i> ez a C-kifejezés, <i>dátum</i> a D-kifejezés, <i>logikai</i> , azaz <i>feltétel</i> .
...lista	több azonos típusú elemből álló felsorolás rendszerint kulcsszavak után. Ha az elemek típusa névvel azonosítandó, akkor természetesen a nevek listája Pl.: numerikus kifejezéslista: a + b, c**3,... ablaklista: kisablak, nagyablak, vakablak. A listaelemeket vesszővel kell elválasztani.

[...]	a három pont előtt szereplő utasításrész többszöri ismételtetését jelzi.
visszatérési érték	A függvények utasításainak végrehajtása eredményeként keletkező egyetlen érték. Ha a függvényekben a visszatérési érték valamilyen szám, akkor az különösebb utalás nélkül is numerikus adattípust jelent. Ha valamilyen nevet ad vissza egy függvény, az értelemszerűen karakter típusú érték.

Műveleti jelek - operátorok:

()	minden operátortípusnál az egybefoglalást, csoportosítást jelenti, és egyben azt is, hogy a zárójelbe zárt kifejezés kiértékelése történik meg először.
----	---

Matematikai operátorok:

+	összeadás, pozitív előjel
-	kivonás, negatív előjel
*	szorzás
/	osztás
** vagy ^	hatványozás

Összehasonlító operátorok:

<	kisebb mint
>	nagyobb mint
<> vagy #	nem egyenlő
<=	kisebb vagy egyenlő
=>	nagyobb vagy egyenlő
\$	Sztring összehasonlító, amely igaz értéket ad, ha a baloldalán álló sztring előfordul a jobboldalán álló sztringben. Pl: igaz, hogy "Pista" \$ "Dankó Pista népipzenekara"

Logikai operátorok :

AND.	logikai és
.OR.	logikai vagy
NOT.	logikai nem, tagadás

Sztring operátorok :

- + Sztringek összekapcsolása
- Sztringek összekapcsolása úgy, hogy az összekapcsolt sztringek végén lévő szóközök az eredménystring végére kerülnek.

Az utasításmagyarázatok során használt egyéb kifejezések:

mondat : egy utasításban a több kulcsszóból és paraméterből álló utasításrész rövid elnevezése. Előtte mindig ott áll az utasítás alap-kulcsszava.
(Pl: select mondat)

cikkely : Többrészes (többkulcsszavas) utasítás egyik kulcsszavához tartozó rész, amely lehet opcionális, azaz szabadon választható is.

opció : önmagában álló (nem cikkely) választható kulcsszó, paraméter, esetenként menüelem. Alkalmazása nem kötelező.

ndx-fájl : Egyszerű indexfájl .ndx kiterjesztéssel

mdx-fájl : Összetett (integrált) indexfájl .mdx kiterjesztéssel

alapindexfájl : az az összetett indexfájl, amelyik az adatfájl nevével azonos. Összetett indexfájl létrehozásakor az alapindexfájl az alapértelmezés.

sztring : karakterfüzér vagy karakterlánc, azaz több karakterből álló egybefüggő karaktersorozat. A függvényekben a karakteres típusu visszatérési értékre utal.

üres sztring : A nulla hosszúságú, azaz egyetlen karaktert sem tartalmazó sztring.

Karakterpozíció: Az a szám, ahányadik helyen a sztringben (karakterláncban) az adott vagy keresett karakter áll.

felhasználónév : (User ID) Elsősorban az adathozzáférés jogosultságának biztosítására szolgáló utasításokban a felhasználókat névvel, azonosítóval különböztetik meg. A szó erre az azonosítóra utal.

Elnevezési szabályok

Fájlnevek hossza legfeljebb 8 karakter, minden egyéb név 10 karakter lehet. A neveknek angol betűvel kell kezdődniük, angol kis- és nagybetűket, számokat és az aláhúzásjelet "_" tartalmazhatják.

Megjegyzések:

A kulcsszavak mindig nagybetűvel szerepelnek, de az alkalmazás során lehet végyesen kis és nagybetűkkel is írni. Az utasításban a szimbólumokat el kell hagyni.

C) Utasítások

C-1. Adat- és fájlkezelés

APPEND [BLANK]

APPEND [BLANK]

Új rekordot fűz az aktuális adatfájl végére. A rekordmutató az új rekordra áll. A megnyitott indexfájlokat aktualizálja.

Lásd még: BROWSE, EDIT, SET AUTOSAVE, SET CARRY, SET FORMAT, SET WINDOW

APPEND FROM

APPEND FROM <fájlnev>/? [[TYPE] <fajltípus>]
[FOR <feltétel>]

Egy fájlból a *feltétel*nek megfelelő rekordokat másol az aktuális adatfájlba. Interaktív üzemmódban a *fájlnev* helyett ? is szerepelhet, amelynek hatására megjeleníti a fájl-listát, könyvtárakat és a másolni kívánt fájl (a sávkurzor segítségével) kijelölhető. dBASE adatfájlokból törendő másolásakor a logikailag törölt (törlésre kijelölt) rekordokat nem másolja át, ha a SET DELETED ON érvényes.

Másolható fájltypusok:

- DBASEII, dBASE III, dBASE IV adatfájl
- DIF (VisiCalc)
- FW2 (Framework II adatbázis vagy frame)
- RPD (Rapidfile)
- SDF (System data format .txt fájl)
- SYLK (Multiplan)
- WKS (LOTUS 1-2-3 adatfájl)

Lásd még: COPY, DELETED(), IMPORT, SET DELETED

APPEND FROM ARRAY

APPEND FROM ARRAY <tömbnév>/? [FOR <feltétel>]

A megadott tömb minden sorát rekordként a fájl végéhez fűzi. A tömb minden sora egy rekord, egy sor egy elemének értéke az oszlop számának megfelelő sor-számú mezőbe kerül. Ha a FOR cikkely szerepel, akkor csak azokat a sorokat szűri be, amelyek megfelelnek a megadott *feltételnek*. A *tömbnév* helyett ? is szerepelhet, melynek hatására megjeleníti a tömblistát és a másolni kívánt tömb (a sávkurzor segítségével) kijelölhető.

Lásd még: APPEND, COPY TO ARRAY, DECLARE, STORE

APPEND MEMO

APPEND MEMO <memo-mezőnév> FROM <fájlnev> [OVERWRITE]

A megjelölt *memo-mező* aktuális tartalmához hozzáfűzi a megadott *fájl* tartalmát. Ha a *fájl* kiterjesztése nem szerepel ".txt"-t feltételez. Az OVERWRITE esetén nem hozzáfűzés, hanem felülírás történik.

Lásd még: COPY MEMO, MODIFY COMMAND

AVERAGE

AVERAGE [<N-kifejezéslista>] [<hatáskör>]
[FOR <feltétel>] [WHILE <feltétel>]
[TO <változólista>/TO ARRAY <tömbnév>]

Az aktuális adatfájl numerikus mezői értékeinek átlagát számítja ki.

A *N-kifejezéslista* az adatfájl kiválasztott mezőire épített N-kifejezéseket tartalmazhat.

Az egyes N-kifejezések kiszámított átlagai rendre a *változólistában* felsorolt változóba vagy TO ARRAY esetén a *tömbnévvel* azonosított tömb elemeibe kerülnek.

Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Példa:

```
AVERAGE magassag,csipo,derek,mell
      FOR kor = 21 .AND. suly = 55 TO ARRAY idealis
```

Lásd még: CALCULATE, DECLARE, SET HEADING, SET TALK, SUM

BROWSE

BROWSE [NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU] [NOEDIT]
 [NODELETE] [NOCLEAR] [COMPRESS] [FORMAT]
 [LOCK < N-kifejezés >] [WIDTH < N-kifejezés >]
 [FREEZE < mezőnév >] [WINDOW < ablaknév >]
 [FIELDS < mezőnév1 > [/R] [/ < szélességoszlop >]
 [, < mezőnév2 > [/R] [/ < oszlop >] ...]

Táblázatos formában megjeleníti az aktuális adatfájl rekordjait a képernyőn és – ha ezt a megadott paraméterek nem korlátozzák, – lehetővé teszi a billentyűzetről az adatok módosítását, új rekordok létrehozását, törlését. Az adatbevitelnél a szerkesztőbillentyűk használhatók. A megnyitott indexfájlokat módosítások esetén aktualizálja.

Cikkelyek, opciók:

- NOINIT** Az előző BROWSE utasítás megadott jellemzői, paraméterei érvényesülnek.
- NOFOLLOW** Indexelt adatfájl esetén hatásos. Alkalmazása esetén az indexező módosításakor a rekord nem "ugrik" a képernyőn megjelenő táblázatban a módosított index szerinti helyére, hanem megtartja az eredeti helyét.
- NOAPPEND** Megakadályozza új rekord létrehozását.
- NOMENU** A BROWSE táblázat menüsora nem jelenik meg a képernyőn.
- NOEDIT** Megtiltja az adatok módosítását.
- NODELETE** Megtiltja a rekord törlést.
- NOCLEAR** A BROWSE táblázat az utasítás elhagyása után is a képernyőn marad.
- COMPRESS** A táblázat fejlécének tömörítésével két rekorddal több fér el a képernyőn (ablakban).
- FORMAT** Az adatok megjelenítése egy korábban megtervezett (CREATE SCREEN) és aktivizált (SET FORMAT TO) képernyőformátum-fájl szerint történik.
- LOCK** A *N-kifejezés* meghatározza, hogy a képernyő bal szélétől számítva hány oszlop maradjon ki a vízszintes görgetésből (maradjon fixen), ha a megjelenítendő táblázat a képernyőnél szélesebb.
- WIDTH** A *N-kifejezés* meghatározza a megjelenítendő mezők szélességének felső határát.
- FREEZE** Csak a *mezőnév*-vel megjelölt mező tartalma módosítható.

WINDOW Az *ablaknévvel* azt az ablakot határozza meg, amelyben a BROWSE tábla megjelenik.

FIELDS A megjelenítendő mezők és azok sorrendje határozható meg.

Lásd még: APPEND, EDIT, PROTECT, SET MEMOWIDTH, SET REFRESH, SET WINDOW OF MEMO

CALCULATE

CALCULATE [<hatáskör>] <függvénylista>
 [FOR <feltétel>] [WHILE <feltétel>]
 [TO <változólista>]/TO ARRAY <tömbnév>]

Az aktuális adatfájl mezőin elvégezhető többféle, statisztikai jellegű (függvényekkel történő) számításokat integrál.

Az egyes függvények kiszámított értékei rendre a *változólistában* felsorolt változóba vagy TO ARRAY esetén a *tömbnévvel* azonosított tömb elemeibe kerülnek.

Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

A választható függvények:

AVG(<N-kifejezés>) Az *N-kifejezés* átlagát számítja ki.

CNT() Meghatározza a műveletben résztvevő és a feltételeknek megfelelő rekordok létszámát.

MAX(<kifejezés>) Az *N-kifejezés* szerinti értékekből a legnagyobbat választja ki.

MIN(<kifejezés>) Az *N-kifejezés* szerinti értékekből a legkisebbet választja ki..

NPV(<kamatláb>, <N-kifejezés>, <indulóérték>) Kiszámítja az *N-kifejezés* által reprezentált értéksorozat jelenlegi nettó értékét a (leszámítolási) *kamatláb* figyelembevételével. Az *indulóérték* a kiszámított értékhez adja. Képlet: $\sum \text{mező}/\text{kamatláb}^i$, ahol *i* a rekord feldolgozásbeli sorszáma.

STD(<N-kifejezés>) A matematikai statisztikából ismert képlet alapján az *N-kifejezés* **eltérését** számolja ki.

SUM(<N-kifejezés>) Az *N-kifejezés* összegzését végzi.

VAR(<N-kifejezés>) A matematikai statisztikából ismert képlet alapján az *N-kifejezés* **szórását** számolja ki.

Az *N-kifejezés* illetve *kifejezés* általában a mező neve, vagy mező(k)ből képzett kifejezés, amellyel a műveletet el kell végezni.

CHANGE

CHANGE [NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU]
 [NOEDIT] [NODELETE] [NOCLEAR] [sorszám]
 [FIELDS <mezőlista >] [hatáskör]
 [FOR <feltétel >] [WHILE <feltétel >]

Megjeleníti a rekordokat a képernyőn és – ha ezt a paraméterek nem korlátozzák – lehetővé teszi a billentyűzetről az adatok módosítását. **Azonos az EDIT paranccsal.** Az opciókat és cikkelyeket lásd a BROWSE utasításnál.

Lásd még: EDIT, BROWSE

CLOSE

CLOSE ALL/ALTERNATE/DATABASES/FORMAT/INDEX/PROCEDURE

Lezárja az opciók valamelyikével meghatározott fájltypusokat.

Opciók:

ALL Mindenfajta fájl lezár és visszatér az 1-es munkaterületre.
 ALTERNATE Az alternate fájlokat zárja le.
 DATABASES Az adatfájlokat és a hozzájuk tartozó indexfájlokat, formátumfájlokat zárja le.
 FORMAT Csak a formátumfájlokat zárja le.
 INDEX Csak az indexfájlokat zárja le.
 PROCEDURE Az eljárásfájlt zárja le.

COPY FILE

COPY FILE <fájlnév1 > TO <fájlnév2 >

Másolatot készít a *fájlnév1* fájlról a *fájlnév2* fájlba.

A *fájlnév*nek tartalmaznia kell minden esetben a kiterjesztést és szükség esetén a meghajtó és elérési út azonosítóját is. Működése azonos a DOS COPY utasításával.

Lásd még: COPY

COPY INDEXES

COPY INDEXES <ndx-fájl-lista> [TO <mdx-fájlnev>]

Az megadott *ndx-fájl-listában* felsorolt egyszerű indexfájlokat az *mdx-fájlnev* összetett indexfájlba integrálja. Legfeljebb 47 egyszerű indexfájl másolható egybe. Az integrált egyszerű indexekre a későbbiekben indextagként az eredeti fájlnevükkel lehet hivatkozni.

Lásd még: COPY TAG, INDEX, KEY(), MDX(), NDX(), SET INDEX, SET ORDER, TAG(), USE

COPY MEMO

COPY MEMO <memo-mezőnev> TO <fájlnev> [ADDITIVE]

Az aktuális rekord megadott *memo-mezőjének* tartalmát másolja át a *fájlnev* fájlba. Ha a fájl kiterjesztése hiányzik, .TXT kiterjesztést kap.

Ha a fájl már létezik, a tartalmát felülírja hacsak az ADDITIVE opció nem szerepel. Ilyenkor a *memo-mező* tartalmát a megadott *fájl* végéhez fűzi.

Lásd még: APPEND MEMO, COPY, COPY FILE

COPY STRUCTURE

COPY STRUCTURE TO <fájlnev> [FIELDS <mezőlista>]

Létrehoz az aktuális adatfájl szerkezetével azonos struktúrájú üres adatfájlt *fájlnev* megnevezéssel.

A FIELDS alkalmazása esetén az új adatfájl szerkezete csak a *mezőlistában* felsorolt mezőkből épül fel.

Lásd még: APPEND, APPEND BLANK, APPEND FROM, DISPLAY STRUCTURE, REPLACE, SET SAFETY

COPY STRUCTURE EXTENDED**COPY TO <fájlnev> STRUCTURE EXTENDED**

Az aktuális adatfájl szerkezetleírását a *fájlnev* **struktúrafájl**ba tölti. A struktúrafájl valójában egy adatfájl, amelynek egy rekordja az alábbi mezőkből épül fel:

FIELD_NAME mezőnév
 FIELD_TYPE típus
 FIELD_LEN hossz
 FIELD_DEC tizedesjegyek száma
 FIELD_IDX indexkénti alkalmazást jelzi

A struktúrafájlból a CREATE FROM utasítással létrehozható az aktuális adatfájl szerkezetével azonos adatfájl.

Lásd még: APPEND FROM, CREATE FROM, LIST/DISPLAY STRUCTURE

COPY TAG**COPY TAG <indextagnév> [OF <mdxfájlnev>] TO <ndxfájlnev>**

Az *mdxfájlnev* összetett indexfájl *indextagját* az *ndxfájlnev* egyszerű indexfájlba másolja.

Csak aktív (USE) adatfájl aktív mdx-fájljának indextagja másolható. Az új egyszerű indexfájl megnyitásra kerül.

Lásd még: COPY INDEX, INDEX, MDX(), NDX(), SET INDEX, SET ORDER, TAG()

COPY TO

**COPY TO <fájlnev> [[TYPE] <fajltípus>]
 [FIELDS <mezőlista>] [<hatáskör>]
 [FOR <feltétel>] [WHILE <feltétel>]**

Másolatot készít az aktuális adatfájlról. Ha a *fajltípus* elmarad, a másolat dBASE adatfájlba készül. A fájl típusokat lásd az APPEND FROM utasításnál. *Mezőlista* (FIELDS) hiányában a művelet valamennyi mezővel megtörténik. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Lásd még: APPEND FROM, COPY FAJL, COPY STRUCTURE, EXPORT, IMPORT, SET DELETED, SET FIELDS, SET SAFETY

COPY TO ARRAY

COPY TO ARRAY <tömbnév> [FIELDS <mezőlista>]
 [<hatáskör>] [FOR <feltétel>]
 [WHILE <feltétel>]

Feltölti a *tömbnévvel* azonosított – előzőleg deklarált (kijelölt) – tömböt az aktuális adatfájlból. A *mezőlista* (FIELDS) hiányában a művelet valamennyi mezővel megtörténik. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Lásd még: APPEND FROM ARRAY, COPY FILE, COPY STRUCTURE EXPORT, IMPORT, SET DELETED, SET FIELDS

COUNT

COUNT [TO <változó>] [<hatáskör>]
 [FOR <feltétel>] [WHILE <feltétel>]

Az aktuális adatfájl rekordjainak (darab)számát adja. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Lásd még: AVERAGE, CALCULATE, RECCOUNT(), SUM

CREATE vagy MODIFY STRUCTURE

CREATE [<adatfájlnev>]

MODIFY STRUCTURE

A CREATE új adatfájlt hoz létre, a MODIFY STRUCTURE paranccsal az aktuális adatfájl szerkezete módosítható. E különbségtől eltekintve megjelenésük, és működésük azonos.

Ha az *adatfájlnev* hiányzik vagy nincs aktuális adatfájl, "Enter database name:" üzenet jelenik meg, amelyre az adatfájl nevét kell begépelni vagy MODIFY esetén a Shift-F1 billentyűk megnyomásával a képernyőn megjelenített adatfájlok közül lehet (a sávkurzorral és az Enter leütésével) választani.

Az utasítás hatására egy menürendszerrel működő interaktív, párbeszédés felhasználói csatoló lép működésbe.

A képernyőn (többek között) egy menüsor – Layout Organize Append GoTo Exit – és egy táblázat jelenik meg.

CREATE esetén a táblázat üres, MODIFY esetén a mezők adatait tartalmazza és az Organize menüpont rolómenüje is látszik. (Ebből az Esc billentyűvel lehet kilépni.)

A táblázat rovataiba a mezők jellemzőit kell begépelni:

Field Name: mező neve

Field Type: mező típusa (Numeric-numerikus, Character-karakteres, Float-lebegőpontos numerikus, Logical-logikai, Date-dátum, Memo-memo)

Width: hossz

Dec: tizedesjegyek száma a hosszon belül (numerikus típusnál)

Index: a mező indexként használandó = Y, nem használandó = N

A szerkezetmeghatározás illetve a módosítás műveletét a **Ctrl-End** billentyűvel lehet befejezni, amelynek hatására az adatfájl létrejön illetve szerkezete módosul. Módosítás esetén az adatfájlban lévő adatok automatikusan az új szerkezetű fájlba kerülnek. Mindazok az adatok **elvesznek**, amelyek

- törölt mezőben voltak,
- mezőtípusuk (a két numerikus típus közti csere kivételével) megváltozott,
- mezőjük neve és hossza együtt megváltozott,
- azonos nevű de rövidebb mezőbe kerültek. (Ilyenkor csak a rövidebb mezőből "kilógó" adatrész veszik el.)

Ha valamelyik mező indexként szerepelt, akkor automatikusan létrejön illetve karbantartásra kerül az **összetett indexfájl (.mdx)**.

A felhasználói csatoló olyan funkciókat is ellát, amelyek más utasítások segítségével megvalósíthatók. (Pl: indexelés, rendezés, stb) Információkat lásd a megfelelő utasításnál. (A csatoló működéséről részletesebben a Control Center c. fejezet szól.)

Lásd még: APPEND FROM, APPEND MEMO, COPY STRUCTURE, CREATE FROM, SET BLOCKSIZE, SET SAFETY, SET SQL

CREATE

CREATE <adatfájlnev> [FROM <struktúra-fájlnev>]

Létrehoz egy adatfájlt egy struktúrafájl alapján. A struktúrafájl a COPY STRUCTURE EXTENDED utasítással hozható létre.

Lásd még: COPY, COPY STRUCTURE, COPY STRUCTURE EXTENDED

CREATE VIEW .. FROM

CREATE VIEW <fájlnév> FROM ENVIRONMENT

Az utasítással a DBASE III PLUS-szal kompatibilis *.vue* kiterjesztésű *fájlnév* nézetfájl építhető fel.

Az utasítás hatására egy – a CREATE QUERY utasítással azonos – interaktív tervezőprogram lép működésbe, amely vizuális megjelenítéssel, menürendszerű párbeszédés kommunikációval támogatja a megfelelő kapcsolatok és formátum megtervezését.

Részletesebb információk a Control Center c. fejezetben.

Lásd még: SELECT, SET FIELDS, SET FORMAT, SET INDEX, USE, SET RELATION, SET VIEW

DELETE

DELETE [<hatáskör>] [FOR <feltétel>]
[WHILE <feltétel>]

Törlésre jelöli ki a (FOR, WHILE) *feltételeknek* és a *hatáskörnek* megfelelő illetve ezek hiányában az összes rekordot az aktuális adatfájlban. Fizikai törlés nem történik.

Lásd még: DELETED(), PACK, RECALL, SET DELETED, ZAP

DELETE TAG

DELETE TAG <indextagnév1> [OF <mdx-fájlnév>]/<ndx-fájlnév1>
[, <indextagnév2> [OF <mdx-fájlnév>]/<ndx-fájlnév2> ...]

Törli az *indextagnévvel* megjelölt indextagokat az (összetett index) *mdx-fájl*ból illetve *ndx-fájlnév* esetén lezárja (de nem törli!) a megadott egyszerű indexfájlokat.

Lásd még: COPY INDEX, COPY TAG, INDEX, MDX(), NDX(), ORDER(), TAG(), USE, SET INDEX, SET ORDER

EDIT

EDIT [NOINIT] [NOFOLLOW] [NOAPPEND] [NOMENU] [NOEDIT]
 [NODELETE] [NOCLEAR] [< rekordsorszám >]
 [FIELDS < mezőlista >] [< hatáskör >]
 [FOR < feltétel >] [WHILE < feltétel >]

Megjeleníti az aktuális adatfájl rekordjait vagy a *rekordsorszám* szerinti egyetlen rekordot a képernyőn és – ha ezt a megadott paraméterek nem korlátozzák, – lehetővé teszi a billentyűzetről az adatok módosítását, új rekordok létrehozását, törlését. Az adatbevitelnél a szerkesztőbillentyűk használhatóak. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Mezőlista (FIELDS) hiányában a művelet valamennyi mezővel megtörténik. A megnyitott indexfájlokat módosítások esetén aktualizálja.

Lásd még: BROWSE, CHANGE, CREATE, SET AUTOSAVE, SET FIELDS, SET DESIGN, SET FORMAT, SET LOCK, SET REFRESH, SET WINDOW OF MEMO

EXPORT

EXPORT TO < fájlnev > [TYPE] PFS/DBASEII/FW2/RPD
 [FIELD < mezőlista >] [< hatáskör >]
 [FOR < feltétel >] [WHILE < feltétel >]

Az aktuális adatfájl tartalmát átmásolja a lehetséges formátumok (PFS:FILE, DBASE II, FRAMEWORK II, RAPIDFILE) valamelyike szerint a *fájlnev* fájlba.

Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz. *Mezőlista* (FIELDS) hiányában a művelet valamennyi mezővel megtörténik.

Lásd még: COPY, IMPORT, SET FORMAT, SET SAFETY

GO/GOTO

GO/GOTO BOTTOM/TOP [IN < másodnév >]

vagy

GO/GOTO [RECORD] < rekordsorszám > [IN < másodnév >]

vagy

< rekordsorszám >

Az IN *másodnévvel* kiválasztott adatfájl, ennek hiányában az aktuális adatfájl megadott rekordjára állítja a rekordmutatót, és ezzel ez a rekord lesz az aktuális.

Opciók:

TOP Az adatfájl első rekordjára lép.

BOTTOM Az adatfájl utolsó rekordjára lép.

RECORD A *rekordsorszámmal* megadott rekordra lép. A RECORD kulcsszó elhagyható.

A *rekordsorszám* helyett állhat **N-kifejezés**. A rendszer bármely utasítás-sorban *rekordsorszámnak* értelmezi az önmagában álló numerikus kifejezést, és ennek hatására is az adott sorszámú rekordra lép.

Az utasítás végrehajtását semmiféle index nem befolyásolja.

Lásd még: ON ERROR, RECNO(), SET DELETED, SET FILTER, SET RELATION, SKIP

IMPORT

IMPORT FROM < fájlnev > [TYPE] PFS/DBASEII/FW2/RPD/WK1

A *fájlnevvel* megadott fájlból – a fájl típus szerinti konvertáló eljárással – egy új adatfájlt hoz létre. A megengedett típusok: PFS:FILE, DBASE II, FrameWork II, RapidFile, Lotus WK1.

A *fájlnevnek* a kiterjesztést is tartalmaznia kell. Az új adatfájl neve a *fájlnevvel* lesz azonos, de kiterjesztése .dbf lesz.

Lásd még: APPEND FROM, COPY, EXPORT, SET FORMAT, USE

INDEX ON

INDEX ON < kulcskifejezés > TO < ndx-fájlnev > /TAG < indextagnév >
 [OF < mdx-fájlnev >] [UNIQUE] [DESCENDING]

Létrehozza az aktuális adatfájlhoz a *kulcskifejezés* szerinti indexet. A helyes működéshez *kulcskifejezésben* – egy elemként – az adatfájl legalább egy mezőjének szerepelnie kell.

Egyszerű indexfájl létrehozásához elegendő az *ndx-fájlnev* szerepeltetése. Összetett indexfájl indextagjának létrehozásához *TAG* cikkelyt kell alkalmazni. Ilyenkor a TAG kulcsszó után megadott *indextagnévvel* jön létre egy tagindex az *OF mdx-fájlnevvel* jelölt vagy ennek hiányában az adatfájl nevével egyező nevű összetett indexfájlban, az *alapindexfájlban*.

Az utasítás végrehajtása után az indextáblázat felépítése is megtörténik és az *ndx-fájlnevvel* vagy az *indextagnévvel* jelölt index lesz **aktuális**, azaz az adatfájl rekordjai ennek sorrendjében érhetőek el.

Opciók:

UNIQUE Azonos kulcsú rekordok közül csak egy kerül bejegyzésre az indexfájlba.

DESCENDING Az index csökkenő sorrendű lesz.

Lásd még: CLOSE, COPY INDEX, COPY TAG, DELETE TAG, FIND, KEY(), MDX(), NDX(), ORDER(), SET INDEX, SET ORDER, REINDEX, SEEK, SEEK(), SET DELETED, SET FILTER, SET INDEX, SORT, TAG(), USE, SET UNIQUE

INSERT

INSERT [BEFORE] [BLANK]

Új rekordot illeszt az aktuális adatfájlba az aktuális rekord elé vagy mögé. Az üres rekord mezőibe a billentyűzetről adatok vihetők az EDIT működéséhez hasonló módon.

Opciók:

BEFORE Az aktuális rekord elé kerül az új rekord.

BLANK Csak üres rekord beillesztés történik azonnali adatbevitel nélkül.

Lásd még: @, APPEND, CHANGE, EDIT, MODIFY COMMAND, READ, SET AUTOSAVE, SET CARRY, SET FORMAT

JOIN WITH

JOIN WITH < másodnév > TO < fájlnev >
FOR < feltétel > [FIELDS < mezőlista >]

Az aktuális adatfájlt és a *másodnévvel* azonosított adatfájlt egyesíti egy új adatfájlban, amelynek azonosítója a megadott *fájlnev* lesz. Ha a (FOR) *feltétel* nem szerepel, a műveletben minden rekord résztvesz. Az új fájlban a rekordok száma a két fájl – műveletben résztvevő – rekordjai számának összege lesz. *Mezőlista* (FIELDS) hiányában az új fájlban valamennyi mező szerepelni fog.

Lásd még: SET FIELDS, SET RELATION

LOAD

LOAD < fájlnev >

Bináris (gépi kódú) programfájlt tölt be a memóriába. A betöltött program a CALL utasítással vagy a CALL() függvénnel futtatható. Legfeljebb 16 ilyen program lehet a memóriában egyidőben, és darabonként legfeljebb 32000 bájttal hosszúak lehetnek. Ezeket a dBASE IV szubrutinként indítja el.

Lásd még: CALL, CALL(), LIST/DISPLAY STATUS, RELEASE, RUN

PACK

PACK

A törlésre kijelölt rekordokat eltávolítja. A rekordokat tömöríti, így azok sorszáma megváltozik. Karbantartja a megnyitott indexeket.

A törölt rekordokat visszaillesztani nem lehet. Ez alól csak akkor van kivétel, ha a PACK utasítás a BEGIN TRANSACTION és END TRANSACTION között áll.

Lásd még: COPY DELETE, DELETED(), DIR, RECALL, REINDEX, SET AUTOSAVE, ZAP

RECALL

RECALL [<hatáskör>] [FOR <feltétel>]
[WHILE <feltétel>]

A logikailag törölt (törlésre kijelölt) rekordokat visszaállítja. (A törlésjelzőt eltávolítja.). Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Lásd még: DELETE, DELETED(), PACK, SET DELETED, ZAP

REINDEX**REINDEX**

Újraépíti az aktuális adatfájlhoz tartozó összes megnyitott indexfájlt.

Lásd még: INDEX, KEY(), MDX(), NDX(), PACK, SET INDEX, USE, ZAP, SET ORDER, SET UNIQUE

RENAME

RENAME <régi fájlnev> TO <új fájlnev>

A *régi fájl* nevét módosítja a megadott *új* névre. A fájlnevekben a kiterjesztésnek is szerepelni kell.

Lásd még: CLOSE, COPY, COPY FILE, USE

REPLACE

REPLACE <mezőnev> WITH <kifejezés> [ADDITIVE]
[, <mezőnev> WITH <kifejezés> [ADDITIVE] ...]
[<hatáskör>] [FOR <feltétel>] [WHILE <feltétel>]

Felülírja az aktuális adatfájl kijelölt mezőit a megadott kifejezéssel. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Opciók:

ADDITIV Memo-mező esetén az új kifejezést a régebbihez hozzáfűzi.

Lásd még: ORDER, REINDEX, SET INDEX, SET ORDER, STORE

ROLLBACK

ROLLBACK [< adatfájlnev >]

Visszaállítja az adatfájl BEGIN TRANSACTION utasítás végrehajtása előtti állapotát. Alkalmazása elsősorban az áramkimaradás és egyéb rendszerhibák miatt előforduló adatvesztés elkerülése érdekében célszerű. Csak a BEGIN/END TRANSACTION utasításokkal együtt használható.

Lásd még: BEGIN/END TRANSACTION, COMPLETED(), ISMARKED(), RESET, ROLLBACK()

SELECT

SELECT < munkaterületnév/másodnév >

Kiválaszt egy munkaterületet a lehetséges tíz közül. Ez lesz az **aktuális munkaterület**. Ha a munkaterületen korábban megnyitásra került adatfájl, akkor azt aktuálissá teszi. Ebben az esetben *másodnévvel* is lehet hivatkozni. A *munkaterületnév* helyén a 1 - 10 terjedő egész számok, vagy a A - J betűk szerepelhetnek.

Lásd még: ALIAS(), CLOSE, CONTINUE, LOCATE, SET CATALOG, USE, SET FIELDS, SET VIEW

SKIP

SKIP [< N-kifejezés >] [IN < másodnév >]

Az *N-kifejezés* által meghatározott számú és irányú lépést tesz az aktuális – vagy az *IN másodnévvel* kiválasztott aktív – adatfájlban. Az utolsó lépés által kiválasztott rekord lesz az adatfájl aktuális rekordja, erre fog (ennek a rekordszámát fogja) mutatni a rekordmutató. Ha a kiválasztott adatfájlnak van aktuális indexe, akkor a lépések ennek sorrendje szerint történnek.

Lásd még: BOF(), EOF(), GO, RECNO()

SORT TO

SORT TO <adatfájlnev> ON <mezőnev1> [/A] [/C] [/D]
 [, <mezőnev2> [/A] [/C] [/D] ...] [ASCENDING/DESCENDING]
 [<hatáskör>] [FOR <feltétel>]
 [WHILE <feltétel>]

Létrehoz egy új adatfájlt, amelyben az eredeti rekordok a megadott *mezők* szerint fizikailag rendezettek. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Opciók:

/A	A rendezés ASCII kód szerint történik.
/C	A rendezés ASCII kód szerint történik, de az angol abc kis és nagybetűit nem különbözteti meg.
/D	Fordított ASCII sorrend.
ASCENDING/DESCENDING	Növekvő vagy csökkenő sorrendet határoz meg azoknál a mezőknél, amelyeknél az /A, /D opció egyike sincs megadva.

Lásd még: COPY, INDEX

SUM

SUM <N-kifejezéslista>
 [TO <változólista> / TO ARRAY <tömbnev>]
 [<hatáskör>] [FOR <feltétel>]
 [WHILE <feltétel>]

Az aktuális adatfájl numerikus mezői értékeinek összegét számítja ki.

A *N-kifejezéslista* az adatfájl kiválasztott mezőire épített N-kifejezéseket tartalmazhat.

Az egyes N-kifejezések kiszámított összegei rendre a *változólistában* felsorolt változóba vagy TO ARRAY esetén a *tömbnévvel* azonosított tömb elemeibe kerülnek.

Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Példa:

```
SUM fogyasztas
  FOR italtipus = "sör" TO hordo
```

Lásd még: AVERAGE, CALCULATE, DECLARE, FIXED(), FLOAT(), STORE, TOTAL

TOTAL ON

TOTAL ON < kulcsmező > TO < adatfájlnev > [FIELDS < mezőlista >]
[< hatáskör >] [FOR < feltétel >]
[WHILE < feltétel >]

Az aktuális adatfájl minden numerikus mezőjét összegzi azoknál a rekordoknál, amelyeknél a *kulcsmező* értéke azonos. Az összegek az *adatfájlnev* azonosítóval létrehozott új adatfájlba kerülnek, amelynek szerkezete az aktuális adatfájléval azonos lesz. Az összegzendő mezők korlátozhatók a (FIELDS) *mezőlistában* felsorolt mezőkre, ennek megfelelően a *mezőlista* csak **numerikus mezőket** tartalmazhat. Az új adatfájl többi mezője az aktuális adatfájl azonos kulcsú rekordjai közül mindig az elsőnek az értékeit veszik fel.

Az aktuális adatfájlnak a **kulcsmező szerint** – index alkalmazásával, vagy fizikailag – **rendezettnek** kell lennie.

Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Példa:

```
TOTAL ON italtipus TO szeszfogy  
FIELDS italtipus, fogyasztas  
FOR italjelleg < > "alkoholmentes"
```

Lásd még: AVERAGE, CALCULATE, INDEX, MODIFY STRUCTURE, SET SAFETY, SORT, SUM

UPDATE ON

```
UPDATE ON <kulcsmező> FROM <másodnév>  
REPLACE <mezőnév1> WITH <kifejezés1>  
[, <mezőnév2> WITH <kifejezés2> ...] [RANDOM]
```

Az aktuális adatfájl *mezőnévvel* kiválasztott mezőit módosítja a megadott *kifejezésekkel*. A *kifejezések* általában a *másodnévvel* jelölt adatfájl mezőire épülnek, amely mezőnevek előtt a *másodnév* előtag használata elengedhetetlen.

A módosítás csak azoknál a rekordoknál történik meg, amelyeknél a kulcskifejezés egyezik a forrás adatfájl rekordjával.

Az aktuális adatfájlnek a **kulcskifejezés** szerint – index alkalmazásával, vagy fizikailag – **rendezettnek** kell lennie.

Opciók:

RANDOM Jelzi, hogy a másodnévvel jelölt (forrás) adatfájl a *kulcskifejezés* szerint nem rendezett. Ilyenkor a helyes működéshez alkalmazása elengedhetetlen.

Lásd még: INDEX, JOIN, REPLACE, TOTAL, SUM

USE

USE [< adatfájlnev > /?] [IN < munkaterületszám >]
 [[INDEX < ndx-/mdx-fájl-lista >]
 [ORDER < ndx-fájlnev > / < indextagnév > [OF < mdx-fájlnev >]]
 [ALIAS < másodnév >] [EXCLUSIVE] [NOUPDATE]

Az *adatfájlnev*vel azonosított adatfájl megnyitása, használatba vétele. Ha az IN *munkaterületszám* hiányzik, akkor a fájl az éppen aktuális munkaterületen nyílik meg. Megnyithatók az INDEX cikkely *ndx-* és *mdx-fájl-listáiban* felsorolt indexfájlok.

Kijelölhető az ORDER alkalmazásával az **aktuális indexként** használandó *ndx-fájl* vagy *indextag*. Ha *indextagnév* esetén az OF *mdx-fájlnev* hiányzik, akkor az *indextagnak* az **alapindexfájlban** kell lennie.

Az adatfájlra a későbbiekben **másodlagos névvel** is lehet hivatkozni, ha az ALIAS *másodnév* szerepel az utasításban. Ennek hiányában a másodnév az *adatfájlnev*vel lesz azonos, de ilyenkor a *munkaterület számával* (vagy betűjelével) is lehet az adatfájlra hivatkozni.

Opciók:

EXCLUSIVE Többfelhasználós üzemmódban megakadályozza, hogy az adatfájlt bármelyk más felhasználó elérje. A hozzáférés más számára csak a CLOSE, CLEAR ALL vagy USE utasítások valamelyikének (ezen a munkaterületen történő) kiadása után lehetséges.

NOUPDATE Az adatfájl csak olvasható.

Lásd még: CLEAR ALL, CLOSE, COPY INDEX, COPY TAG, DBF(), DELETE TAG, INDEX, SELECT, SET CATALOG, SET INDEX, SET EXCLUSIVE, SET ORDER, SET REPROCESS

ZAP**ZAP**

Fizikailag törli az összes rekordot az aktuális adatfájlból, de az üres adatfájl megmarad. Karbantartja az aktív indexállományokat is. Ha a SET SAFETY ON állásban van a törlés előtt a **Zap <adatfájlnev> ? (Y/N)** üzenet kér jóváhagyást a tévedések megelőzése érdekében.

Lásd még: DELETE, PACK, SET SAFETY

C-2. Kereső utasítások

CONTINUE

CONTINUE

Megtalálja a következő rekordot az aktuális adatfájlban, amely kielégíti a LOCATE utasításban leírt feltételrendszert. A rekordmutató a megtalált rekordra áll és ezzel ez a rekord lesz aktuális. Ha a keresés sikertelen, a rekordmutató az adatfájl végére mutat.

Lásd még: EOF(), FOUND(), FIND, LOCATE, SEEK, SEEK()

FIND

FIND < kulcs >

Megkeresi az aktuális adatfájlban a *kulccsal* egyező – illetve SET NEAR ON beállítás esetén a *kulcshoz* legközelebbi – indexű rekordot. A *kulcs* helyén csak idézőjel nélküli karakteres konstans, vagy & makro előtaggal kiegészített karakteres változó állhat. A rekordmutató a megtalált rekordra áll és ezzel ez a rekord lesz aktuális. Ha a keresés sikertelen, a rekordmutató az adatfájl végére mutat.

Csak aktív index esetén használható és csak az aktuális index szerinti keresésre.

Lásd még: EOF(), FOUND(), INDEX, KEY(), LOCATE, LOOKUP(), MDX(), NDX(), SEEK, SEEK(), SET DELETED, SET FILTER, SET INDEX, SET NEAR, SET ORDER, TAG(), USE

LOCATE

LOCATE [FOR] <feltétel> [<hatáskör>] [WHILE <feltétel>]

Az aktuális adatfájlban a megkeresi megadott *feltételnek* eleget tevő első rekordot, a rekordmutatót erre állítja, és ezzel ez lesz az aktuális rekord. Ha a keresés sikertelen, a rekordmutató a fájl végére mutat.

A keresés nem igényel aktív indexet, ennek következtében lényegesen lassabb a FIND és SEEK utasításoknál. Ha a (WHILE) *feltétel* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Lásd még: CONTINUE, FIND, FOUND(), SEEK

SEEK

SEEK < kifejezés >

Megkeresi az aktuális adatfájlban a *kifejezéssel* egyező – illetve SET NEAR ON beállítás esetén a *kifejezéshez* legközelebbi – kulcsú rekordot. A *kifejezés* tetszés szerinti lehet, de az aktuális index-szel csak azonos típusú értéket szolgáltatathat. A rekordmutató a megtalált rekordra áll és ezzel ez a rekord lesz aktuális. Ha a keresés sikertelen, a rekordmutató az adatfájl végére mutat.

Csak aktív index esetén használható és csak az aktuális index szerinti keresésre.

Egy indexelt adatfájlban keressük azt a rekordot, amelynek az indexkulcsa a megadott kifejezéssel kezdődik. A keresés abban az esetben is sikeres, SET EXACT OFF mellett, ha csak a mező egy részét adjuk meg kulcsként. Mindig csak az első megfelelő helyre pozicionál a pointer. (A további kereséshez program szükséges.)

Lásd még: CONTINUE, EOF(), FIND, FOUND(), INDEX, KEY(), LOCATE, MDX(), NDX(), SET DELETED, SET EXACT, SET FILTER, SET INDEX, SET NEAR, TAG(), SET ORDER, USE

C-3. Adatmegjelenítés, adatbevitel

?/??

?/?? [< kifejezés1 > [PICTURE] < C-kifejezés >]
 [FUNCTION < függvénylista >] [AT < N-kifejezés >]
 [STYLE < karakteres kifejezés > / < N-kifejezés >]
 [, < kifejezés2 > ...] [,]

?/?? egy vagy több kifejezés értékét jeleníti meg. A megjelenítés helye SET PRINTER OFF beállítás esetén – ez az alapértelmezés – a képernyő, egyébként a nyomtató.

A megjelenítés **adatfolyam** (folyamatos) jellegű, azaz a ? mindig a következő sor elején kezdi az adatokat, a ?? pedig az aktuális kurzorpozíciót követő pozíción.

Az önmagában álló ? egy üres sort ír ki, azaz sort emel.

Cikkelyek, opciók:

PICTURE A C-kifejezés megjelenítési mintát határoz meg az adathoz. A @ utasítás PICTURE cikkelyében felsorolt minták használhatóak.

FUNCTION A megjelenítendő adat formátumát a függvénylistában megadott függvények befolyásolják. Ezek azonosak a @ utasításban használhatókkal. Két olyan függvény van, amelyik csak a ?/?? utasításban alkalmazható:

H Ha az adat hosszabb a PICTURE cikkelyben megadottnál, akkor a minta hosszát figyelemmen kívül hagyja és a teljes adattartalom megjelenik. A további adatok az új hosszának megfelelően jobbra tolódnak. Ha az adat rövidebb, akkor a minta hosszát figyelmen kívül hagyja és az adat a megjelenítéskor csak a tényleges hosszának megfelelő helyet foglalja el. A követő adatokat balra igazítja. (Ez a függvény az általunk ismert dBASE IV változatban tapasztalatunk szerint nem működik. *a Szerk.*)

V Ha az adat hosszabb a PICTURE cikkelyben megadottnál, akkor a minta hosszának megfelelő hasáiban jeleníti meg, úgy, hogy az első sorban az összes többi adat és a hasáb első sora, a további sorokban a hasáb többi sorai jelennek meg. A hasáb sorainak számát az (egész szám) megadásával lehet korlátozni. Az így korlátozott hasábból "kilógó" adatrész nem jelenik meg. (Tapasztalatunk szerint ez a függvény sem a leírtaknak megfelelően, hanem a következők szerint működik: A V önmagában hatástalan, a V n szélességű hasábot képez. A sorok számát nem lehet korlátozni. *a Szerk.*)

I A középre igazító függvényt csak azért említjük itt, mert a `?/??` utasításban – azt tapasztaltuk – nem működik helyesen. Balra tolja a következő adatot. (*a Szerk.*)

AT **A** *N-kifejezés* meghatározza azt az oszlopot (pozíciót), amelynél az adat első karaktere kezdődik

STYLE A megjelenő szöveg betűtípusát és -stílusát lehet beállítani a következők szerint:

B félkövér (boldface) betűk

I dőlt (italic) betűk

U aláhúzott (underline) betűk

R felső index

L alsó index

1 - 5 A **Config.Db** fájlban előzőleg a **PRINTER** paranccsal beállított betűtípust azonosítja. A betűtípusokat a képernyőn csak egyes monitortípusoknál lehet megkülönböztetni.

Lásd még: `???,@`, **SET PRINT**

@ ...SAY ...GET

@ < sor > , < oszlop >

[SAY < kifejezés > [PICTURE < C-kifejezés >]
[FUNCTION < függvény lista >]]

[GET < mező/változó > [[OPEN] WINDOW < ablaknév >]
[PICTURE < C-kifejezés >] [FUNCTION < függvénylista >]
[RANGE [< alsó határ >] [, < felső határ >]]

[VALID < feltétel > [ERROR < C-kifejezés >]]
[WHEN < feltétel >] [DEFAULT < kifejezés >]
[MESSAGE < C-kifejezés >]]

[COLOR [< normál színpár >] [, < kiemelt színpár >]]

Az adatokat jeleníti meg és lehetőséget biztosít a billentyűzetről történő adatbevitelre. A megjelenítés helye és formátuma meghatározható, a bevitt adatok érvényessége különböző eszközökkel ellenőrizhető. A megjelenítés képernyőn és nyomtatón, az adatbevitel csak képernyőn történhet (**SET DEVICE TO PRINT/SCREEN**).

A *sor*, *oszlop* tetszőleges N-kifejezés lehet, amelynek felső határértékeit képernyő esetén az *ablaknév*vel jelölt ablak méretei, ennek híján a képernyő méretei befolyásolják. Nyomtató esetén a *sor* 0 - 32767 közé, az *oszlop* pedig 0 és 255 közé eshet.

A *sor*, *oszlop* koordináták mindig viszonylagosak, az aktív *ablakon* belül számítandók.

Az önmagában (**SAY**, **GET** cikkely nélkül) álló *sor*, *oszlop* az adott sort, a meghatározott oszloptól törli.

A SAY kulcsszóval megadott *kifejezés* megjeleníti a képernyőn a koordináták által meghatározott helyen, de az nem módosítható.

A GET kulcsszó megjeleníti a *változót* vagy *mezőt* és lehetővé teszi tartalmának módosítását egy későbbi READ utasítás után. A módosításkor a szerkesztőbillentyűk használhatóak.

A *mezők/változók* módosításakor az azzal összefüggésben lévő és a képernyőn lévő számított változók is módosulása is megjelenik.

Cikkelyek és opciók:

PICTURE A megjelenítendő/beolvasandó adatok megengedett típusát, hosszát, megjelenési formáját határozza meg a *C-kifejezés*, amely gyakorlatilag megjelenítési *mintát* (maszkot) tartalmaz, amelyet idézőjelbe kell zárni. A minták különböző jelekből állhatnak és minden egyes jel a mintában az adat egy karakterét befolyásolja az alábbiak szerint:

- ! Az angol betűket nagy betűvé alakítja át.
- # Csak 0-9,-, + és szóköz karaktert fogad el.
- \$ A szám elé a pénznem szimbólumot helyezi el.
- * A szám elé a csillag szimbólumot helyezi el.
- , A vessző csak akkor jelenik meg, ha a pozíciójától balra számjegy van.
- . A tizedespont helyét jelöli ki.
- 9 Csak 0-9,-, + karaktert fogad el.
- A Csak betűt fogad el.
- L Csak logikai adatot fogad el.
- N Csak betűt és számjegyet fogad el.
- X Bármilyen karaktert elfogad.
- Y. Csak Y, y, N, n betűket fogadja el és nagy betűvé alakítja át.

A PICTURE cikkelyben együtt is írható a minta és a formátumot befolyásoló függvények (lásd lejjebb) a következőképpen:

PICTURE "<@függv1függv2..függvn> <minta>"

FUNCTION

A megjelenítendő adatok formátumát a mintákon kívül és azokkal együtt az alább felsorolt függvényekkel befolyásolhatjuk:

- ! Az angol betűket nagybetűkké alakítja.
- ^ A számokat tudományos formában írja ki
- \$ A szám előtt a pénznem szimbóluma jelenik meg.
- (A negatív számokat zárójelbe kerülnek.
- A Csak betűt fogad el.
- B Az adatot balra igazítja.
- C A pozitív számok után kiírja a CR (credit) jelet.
- D Dátum formátumú megjelenítés
- E Európai dátum formátum.
- I Középre igazít a mezőben.
- J Jobbra igazítva jelenít meg.
- L A számot bevezető nullákkal együtt írja ki.
- M Mögötte felsorolhatóak (,-vel elválasztva) azok az értékek, amelyeket a változó felvehet. Az adatbevitelnél a szokásostól eltérően a szóköz billentyű lenyomására sorra jelennek meg a lista értékei, amelyből az Enter leütésével lehet választani. Csak GET esetében használható. (Pl: FUNCTION "M Egy, Kettő, Három")
- R Hatására a mintában használt mintakaraktereken kívüli karakterek a képernyőn megjelennek, de a változóba/mezőbe nem kerülnek be. (Pl: a személyi szám kötőjelei)
- S<n> A minta hosszúságát meghaladó *n* hosszúságú adatok bevitelét teszi lehetővé. A bevitelnél a képernyőmező végére érve az adat fényujságszerűen gördül balra illetve az elejére érve jobbra.
- T Levágja az adat elején és végén lévő szóközöket.
- X A negatív számok után kiírja a DB jelet.
- Z A bevezető és záró nullák valamint 0 értékű adat esetén az egész adat helyett szóközt ír.

MESSAGE

Az aktuális GET beolvasásakor az itt megadott üzenet jelenik meg a státusz-sorban. Ha a SET STATUS OFF az érvényes, akkor az üzenet nem jelenik meg.

[OPEN] WINDOW	Memo mezők beviteléhez nyitja meg a (korábban már kijelölt) <i>ablaknév</i> ablakot. Ha az <i>OPEN</i> kulcsszó nem szerepel, akkor a beolvasásnál a memo mezőre érve a Ctrl-Home billentyűk lenyomására nyílik meg (válik aktívvá) az ablak, amelyből az adatbevitel végeztével a Ctrl-End billentyűkkel lehet kilépni. Az <i>OPEN</i> alkalmazásakor az ablak nemcsak a memo beolvasásánál nyílik meg, hanem az ablak a többi (egy READ utasításhoz tartozó @ ..GET utasításokkal beolvasott) adat beolvasása alatt is látszik. Az ablakba be- és kilépésre ilyenkor is a Ctrl-Home ill. -End szolgál.
RANGE	a beolvasandó adat értéke <i>alsó</i> illetve <i>felső határának</i> meghatározására használható karakteres, numerikus és dátum típusú változók esetén. Ha a beolvasott érték a <i>határokon</i> kívül esik, akkor a következő adat beolvasása helyett a szóbanforgó beolvasását ismételteti.
VALID	Ha a beolvasott érték nem felel meg a <i>feltételnek</i> , akkor hibaüzenettel jelez, és a következő adat beolvasása helyett a szóbanforgó beolvasását ismételteti. A hibaüzenet vagy az Editing condition not satisfied (A szerkesztési feltételnek nem felel meg.), vagy pedig az ERROR kulcsszónál megadott C-kifejezés szerinti szöveg.
ERROR	Ha a VALID-nál megadott feltétel nem teljesül, akkor hibaüzenetként a <i>C-kifejezés</i> által meghatározott szöveg jelenik meg.
DEFAULT	A <i>kifejezés</i> kezdőértéket (alapértelmezés) ad a <i>mezőnek/változónak</i> . A <i>kifejezés</i> típusának egyeznie kell a <i>mező/változó</i> típusával. A SET CARRY ON beállítás felülbírálja.
WHEN	Ha a <i>feltétel</i> teljesül, a <i>mezőbe/változóba</i> lehet adatot bevinni. Ellenkező esetben a vezérlés a következő utasításra kerül.
COLOR	A GET/SAY utasítások által megjelenített adatok színeit határozza meg. A SAY használja a <i>normál színpárt</i> , a GET pedig a <i>kiemelt színpárt</i> . A színjellemzők csak az aktuális utasításban érvényesek.

A SET DEVICE TO PRINTER utasítás kiadása után valamennyi @ utasítás a nyomtatóra fog írni a GET cikkelyt használók kivételével.

A SAY/GET/READ utasítások összefoghatók egy ún. képformátum fájlba (.fmt), amelyeket tetszőleges szövegszerkesztővel vagy pedig a CREATE/MODIFY SCREEN utasítások segítségével állíthatók elő.

SET STATUS ON esetén a 22. sorban az aktuális állapotot jellemző információk láthatók. A SET STATUS OFF beállítás csak a 0. sorba helyezi át az állapotsort. A teljes képernyő használatát a SET STATUS OFF és a SET SCOREBOARD OFF teszi lehetővé. Ezek a CONFIG.DB fájlban is rögzíthetők.

Lásd még: ?/??, ACTIVATE WINDOW, APPEND, CHANGE, COL(), CREATE/MODIFY SCREEN, EDIT, INSERT, MODIFY COMMAND, PCOL(), PROW(), READ, ROW(), SET COLOR, SET CONFIRM, SET CURRENCY, SET DELIMITERS, SET DEVICE, SET FIELDS, SET FORMAT, SET INTENSITY, SET POINT, SET SEPARATOR, SET WINDOW OF MEMO, TRANSFORM()

@ ...CLEAR

@ < sor1 > , < oszlop1 > CLEAR [TO < sor2 > , < oszlop2 >]

Törli a megadott koordináták által meghatározott téglalap alakú képernyőrészt. A *sor1*, *oszlop1* a téglalap bal felső sarkát, a *sor2*, *oszlop2* a jobb alsó sarkát jelöli ki. A *sor*, *oszlop* lehetséges értékeire, használatára a @ ..SAY/GET utasításnál leírtak érvényesek.

Ha a TO cikkely elmarad, a törlés a képernyő illetve ablak jobb alsó sarkáig történik.

Lásd még: @ ...FILL, CLEAR

@ ...FILL

@ < sor1 > , < oszlop1 > FILL TO < sor2 > , < oszlop2 >
[COLOR < színpár >]

A *színpár* által meghatározott színűre festi a megadott koordináták szerinti téglalap alakú képernyőrészt. A *sor1*, *oszlop1* a téglalap bal felső sarkát, a *sor2*, *oszlop2* a jobb alsó sarkát jelöli ki. A *sor*, *oszlop* lehetséges értékeire, használatára a @ ..SAY/GET utasításnál leírtak érvényesek. A *színpárban* megadható színek és azok szintaktikája azonos a SET COLOR utasításban leírtakkal.

Lásd még: SET COLOR

@ ...TO

@ < sor1 > , < oszlop1 > TO < sor2 > , < oszlop2 >
 [DOUBLE/PANEL/< keret sztring >] [COLOR < színpár >]

A *sorok*, *oszlopok* által megadott elhelyezésű és nagyságú keretet rajzol.

Az alapértelmezés szerint (ha egyik opció sem szerepel) szimpla vonalú keret készül. A *DOUBLE* dupla vonallal rajzol, a *PANEL* esetén a keret kiemelt fényerejű inverz sorok és oszlopok alkotják.

A *keret sztring* saját tervezésű keretre ad lehetőséget. Az idézőjellel határolt sztringben 8 db, egymástól vesszővel elválasztott karakter szerepelhet, amelyek mindegyike a keret egy-egy elemét határozza meg a következő sorrendben:

f, a, b, j, bf, jf, ba, ja

ahol a betűk a következőket jelentik:

f	- felső sor
a	- alsó sor
b	- bal oldali oszlop
j	- jobb oldali oszlop
f	- bal felső sarok
jf	- jobb felső sarok
ba	- bal alsó sarok
ja	- jobb alsó sarok

A keret színét a *színpár* határozza meg.

Lásd még: @, @ ...CLEAR, @ ...FILL, CHR(), SET BORDER, SET COLOR

LABEL FORM

LABEL FORM < címke-fájlnev > /? [< hatáskör >]
 [FOR < feltétel >] [WHILE < feltétel >]
 [SAMPLE] [TO PRINTER/TO FILE < fájlnev >]

Kinyomtatja (vagy képernyőn megjeleníti) az aktuális adatfájl rekordjait a *címke-fájl* által meghatározott címkeformátummal. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz. A *címke-fájlnev* helyett ? is szerepelhet, amelynek hatására megjeleníti a fájl-listát és a használni kívánt fájl (a sávkurzor segítségével) kijelölhető.

Opciók:

SAMPLE Írás előtt egy mintát jelenít meg.

TOPRINTER A megjelenítés nyomtatón történik.

TO FILE A kimenő adatok nyomtató helyett (nyomtatási formátumban) a fájlnevel jelölt fájlba kerül, ahonnan a későbbiekben (a DOS PRINT utasításával) kinyomtatható.

Lásd még: CREATE/MODIFY LABEL, SET PRINTER

LIST/DISPLAY

Az itt felsorolandó LIST és DISPLAY utasítások páronként ugyanazokat a célokat szolgálják. Egyetlen különbség köztük, hogy a DISPLAY utasítások egyszerre csak egy képernyőnyi információt írnak ki, s egy billentyű leütése után folytatják működésüket. A listázás felfüggesztése a CTRL-S vagy BACKSPACE billentyűkkel lehetséges. A listázás megszakítása pedig az ESCAPE billentyűvel lehetséges.

Mindkét típusú utasítás lehetővé teszi az output lemezes fájlba irányítását. A fentiek miatt a két típust egyszerre tárgyaljuk.

```
LIST/DISPLAY [[FIELDS] <kifejezéslista>] [OFF]
               [<hatáskör>] [FOR <feltétel>] [WHILE <feltétel>]
               [TO PRINTER/TO FILE <fájlnev>]
```

Egy adatfájl tartalmát jeleníti meg.

A LIST folyamatosan, a DISPLAY képernyőnkénti megállással jeleníti meg az adatokat. A megjelenítés felfüggesztése a Ctrl-S vagy Backspace billentyűkkel, a megszakítás az Esc billentyűvel lehetséges.

Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz. *Kifejezéslista* (FIELDS) hiányában valamennyi mező megjelenik. A helyes működéshez *kifejezéslistában* a mezőkre alapozott kifejezéseknek kell szerepelniük. A memo mezők tartalma alapértelmezésben nem jelenik meg, csak akkor, ha nevük *kifejezéslistában* szerepel. A memo mezők 50 karakter szélességben jelennek meg, hacsak a SET MEMOWIDTH utasítás mást nem határoz meg.

Opciók:

TOPRINTER A kimenő adatok a helyi nyomtatón jelennek meg.

TO FILE A kimenő adatok a *fájlnevel* azonosított szövegfájlba kerülnek nyomtatási formátumban.

OFF A rekord sorszámok nem jelennek meg.

Lásd még: SET ESCAPE, SET HEADING, SET MARGIN, SET MEMOWIDTH

LIST/DISPLAY FILES

LIST/DISPLAY FILES [LIKE < minta >]
[TO PRINTER/TO FILE < fájlnev >]

Az aktuális könyvtárra vonatkozó információkat jeleníti meg a DOS DIR utasításhoz. Ha *minta* nem szerepel, csak az (.dbf kiterjesztésű) adatfájlokra vonatkozó információk jelennek meg. Működése és opciói azonosak a LIST/DISPLAY utasításnál leírtakkal.

Lásd még: DIR, FILE(), SET AUTOSAVE

LIST/DISPLAY HISTORY

LIST/DISPLAY HISTORY [LAST < N-kifejezés >]
[TO PRINTER/TO FILE < fájlnev >]

A korábban végrehajtott – az utasításpufferban tárolt – (alapértelmezésben 20) utasítást írja ki. Alapértelmezésben az összes ilyen utasítás megjelenik, korlátozni a LAST *N-kifejezés* megadásával lehet. Működése és opciói azonosak a LIST/DISPLAY utasításnál leírtakkal.

Lásd még: SET HISTORY

LIST/DISPLAY MEMORY

LIST/DISPLAY MEMORY [TO PRINTER/TO FILE < fájlnev >]

Az aktuális memóriakiosztást jeleníti meg. Megmutatja az aktív változók nevét, értékét, jellegét (globális, privát), az ablakok, menük helyfoglalását, az összes és a szabad memóriát. Működése és opciói azonosak a LIST/DISPLAY utasításnál leírtakkal.

Lásd még: DECLARE, DEFINE MENU, DEFINE PAD, DEFINE POPUP, DEFINE WINDOW, PRIVATE, PUBLIC, RELEASE, RESTORE, SAVE, STORE

LIST/DISPLAY STATUS

LIST/DISPLAY STATUS [TO PRINTER/TO FILE <fájlnev>]

A megnyitott adatfájlok aktuális információit, a környezet beállított jellemzőit és egyéb környezeti információkat jelenít meg. Néhány az adatfájlok megjelenített információi közül:

- a munkaterület száma;
- fájlnev, meghajtónév, elérési útnév, másodnev;
- csak olvasható (read-only) állapot;
- nyitott indexfájlok, indexkulcs kifejezések, stb;
- nyitott memo fájlok;
- szűrő (filter) kifejezések;
- adatfájl-kapcsolatok (SET RELATION);
- formátum fájlok.

Működése és opciói azonosak a LIST/DISPLAY utasításnál leírtakkal.

Lásd még: ALIAS(), DIR, DISKSPACE(), FILE(), INDEX, KEY(), MDX(), MEMORY(), NDX(), ORDER(), OS(), PROGRAM(), SET, TAG(), VERSION()

LIST/DISPLAY STRUCTURE

LIST/DISPLAY STRUCTURE [IN <másodnev>]
[TO PRINTER/TO FILE <fájlnev>]

Az IN *másodnévvel* megadott, ennek hiányában az aktuális adatfájl rekord-szerkezetét jeleníti meg.

Működése és opciói azonosak a LIST/DISPLAY utasításnál leírtakkal.

Lásd még: COPY STRUCTURE, COPY STRUCTURE EXTENDED, CREATE/MODIFY STRUCTURE, CREATE FROM, SET FIELDS

LIST/DISPLAY USERS

LIST/DISPLAY USERS

Hálózati üzemmódban megjeleníti a bejelentkezett felhasználókat.

Működése azonos a LIST/DISPLAY utasításnál leírtakkal.

Lásd még: LIST/DISPLAY STATUS, NETWORK()

TEXT

TEXT <szöveg> ENDTEXT

A két kulcsszó közé írt *szöveg* megjelenik a képernyőn illetve a nyomtatón.

Lásd még: @, ?/??, ???, LIST/DISPLAY, SET PRINTER

READ

READ [SAVE]

Adatok beolvasását, bevitelét indítja azzal, hogy aktivizálja az összes @ ...GET utasítást, amely a legutolsó CLEAR, CLEAR ALL, CLEAR GETS illetve READ utasítás után került kiadásra. Az utasítás hatására a legelső (GET) képernyőmezőben megjelenik a kurzor, és az adat szerkesztése megkezdhető. A bevitel folyamán a szerkesztőbillentyűk használhatók. A SAVE kiadása az aktuális GET funkciókat elmenti. tehát azok egy következő READ utasítással újra aktivizálhatók. Normál esetben a READ törli az előtte kiadott GET-eket.

Lásd még: @, CLEAR, CLEAR GETS, CLEAR MEMORY, CREATE SCREEN, REPLACE, SET DEVICE, SET FORMAT, STORE

REPORT FORM

```
REPORT FORM <listaformátum-fájlnev>/?
  [PLAIN] [HEADING <C-kifejezés>]
  [NOEJECT] [SUMMARY] [<hatáskör>]
  [FOR <feltétel>] WHILE <feltétel>
  [TO PRINTER/TO FILE <fájlnev>]
```

A *listaformátum-fájlban* (.frm) meghatározott formában listát, jelentést készít az aktuális adatfájlról. Ha a (FOR, WHILE) *feltételek* illetve a *hatáskör* egyike sem szerepel, a műveletben minden rekord résztvesz.

Opciók:

- NOEJECT** A jelentés előtt nem történik lapdobás.
- SUMMARY** Az aktuális adatfájl rekordjait nem jeleníti meg csak az azokból képzett összegfokozatokat.
- HEADING** A *C-kifejezés* a jelentés minden lapjának első sorában – fejlécként – megjelenik. (A formátumfájlon kívüli, extra fejléc.)
- PLAIN** Az első lap kivételével minden lapon letiltja a fejléc és lábléc kiírását.

Lásd még: CREATE/MODIFY REPORT, LABEL FORM

TYPE

TYPE <fájlnev1> [TO PRINTER/TO FILE <fájlnev2>] [NUMBER]

A *fájlnev1* nevű ASCII szövegfájl tartalmát jeleníti meg képernyőn, *TO PRINTER* esetén nyomtatón, vagy *TO FILE fájlnev2* fájlba írja nyomtatási formátumban.

A **NUMBER** opció hatására a kiírt sorok elején megjelenik a sorszám.

Lásd még: SET DATE, SET HEADING, SET PATH, SET PRINTER

C-4. Értékadó utasítások

ACCEPT

ACCEPT [*< C-kifejezés >*] TO *< változó >*

Karakteres *változó* beolvasása a billentyűzetről. A tájékoztatásra szolgáló *C-kifejezés* az utasítás kiadásakor aktuális sort követő sorban jelenik meg a képernyőn, ablakban. A bevitt karakterek ez után jelennek meg. A beolvasást az Enter leütése zárja. Legfeljebb 254 karakter olvasható be. Értékadó utasítás, tehát a *változót* létre is hozza, ha az korábban nem létezett.

Lásd még: @, INPUT, READ, SET ESCAPE, WAIT

INPUT

INPUT [*< C-kifejezés >*] TO *< változó >*

Numerikus és dátum típusú *változó* beolvasása a billentyűzetről. A tájékoztatásra szolgáló *C-kifejezés* az utasítás kiadásakor aktuális sort követő sorban jelenik meg a képernyőn, ablakban. A bevitt karakterek ez után jelennek meg. A beolvasást az Enter leütése zárja. A *változó* típusa a beolvasott kifejezés szintaxisától függ. Dátum bevitelénél a kapcsos {} zárójelet kell használni, numerikus adatok bevitelénél a számokon kívül matematikai műveleti jelek is használhatók. Értékadó utasítás, tehát a *változót* létre is hozza, ha az korábban nem létezett.

Lásd még: @, ACCEPT, PRIVATE, PUBLIC, READ, STORE, WAIT

STORE

STORE *< kifejezés >* TO *< változólista >/< tömbelem-lista >*

vagy

< változó/tömbelem > = *< kifejezés >*

A *kifejezés* értékével feltölti *változólistában* felsorolt változókat vagy a *tömbelemeket*. Ha a *változó* korábban nem létezett, azt létrehozza. A *változó* illetve a tömb felveszi a *kifejezés* típusát.

Lásd még: @, CLEAR ALL, CLEAR MEMORY, DECLARE, LIST/DISP, PARAMETERS, PRIVATE, PUBLIC, RELEASE, RESTORE, SAVE

WAIT

WAIT [*<C-kifejezés>*] [TO *<változó>*]

Általában az aktuális eljárás (program) felfüggesztésére szolgál. Egy billentyű leütésére a felfüggesztett eljárás folytatódik. A tájékoztatásra szolgáló *C-kifejezés* az utasítás kiadásakor aktuális sort követő sorban jelenik meg a képernyőn, ablakban. Ennek hiányában a **Press any key to continue ..** (Folytatáshoz üssön le egy billentyűt ..) üzenet jelenik meg. Ha a TO *változó* szerepel, akkor a leütött billentyű egy karakteres változóba kerül. Értékadó utasítás, tehát a változót létre is hozza, ha az korábban nem létezett.

Lásd még: ACCEPT, INPUT, ON KEY, STORE

C-5. Változók, szimbólumok kezelése

CLEAR

CLEAR [ALL/FIELDS/GETS/MEMORY/MENUS/POPUPS /TYPEAHEAD/WINDOWS]

Opciók nélkül törli a képernyőt, a kurzort a bal felső sarokba helyezi és megszünteti az összes függőben maradt GET-et. (Ha a Config.db másként nem rendelkezik, legfeljebb 128 ilyen lehet.)

Opciók:

ALL	Lezárja az összes adatfájlt, törli a változókat (a rendszerváltozók kivételével), a tömböket, a menüket és az aktuális munkaterület az 1-es lesz. Az adatfájlokkal együtt lezárja a hozzájuk tartozó indexfájlokat, formátum fájlokat és memo fájlokat, sőt még a katalógust is, ha volt ilyen.
FIELDS	Felszabadítja a SET FIELD által megjelölt mezőket és SET FIELD kapcsolót OFF állásba kapcsolja.
GETS	Törli a legutolsó CLEAR ALL, CLEAR GETS vagy READ utasítások óta kiadott GET-eket.
MEMORY	Törli az összes változót és tömböt a rendszerváltozók kivételével. Hatása gyakorlatilag megegyezik a RELEASE ALL utasításával.
MENUS	Törli az összes vonalmenüt a képernyőről és a memóriából is.
POPUPS	Törli az összes rolómenüt a képernyőről és a memóriából is.
TYPEAHEAD	Törli a billentyűpuffert.
WINDOWS	Törli az összes kijelölt ablakot a képernyőről és a memóriából is.

Lásd még: @, @ ...FILL, @ ...TO, CLOSE, CREATE/MODIFY VIEW, READ, RELEASE, SET FIELDS, SET FORMAT, SET TYPEAHEAD, SET VIEW

DECLARE

```
DECLARE <tömbnév1> [< sor1 >, < oszlop1 >]
          [<tömbnév2> [< sor2 >, < oszlop2 >]
```

Egy- vagy kétdimenziós tömb deklarációjának utasítása.

Pl.: DECLARE A[12, 12], B[10].

Lásd még: APPEND, AVERAGE, CALCULATE, CLEAR ALL, CLEAR MEMORY, COPY, COUNT, LIST/DISPLAY MEMORY, PUBLIC, RELEASE, RESTORE, SAVE, SUM

PARAMETERS

```
PARAMETERS <változólista>
```

A *változólista* kijelöli az eljárást hívó utasítás által átadott paraméterek helyi változóinak neveit. Ebből következően ha egy programban eljárásban a PARAMETERS utasítás szerepel, akkor annak az első (ill. a PROCEDURE vagy FUNCTION utasításokat követő) utasításnak kell lennie. A listában a változók sorrendben az átadott paraméterek típusait és értékét veszik fel.

Lásd még: @, DO, PRIVATE, PROCEDURE, PUBLIC, SET PROCEDURE

PRIVATE

```
PRIVATE <változólista>]
```

vagy

```
PRIVATE ALL [LIKE/EXCEPT <minta>]
```

A *változólistában* felsorolt változókat helyi változónak jelöli ki. A helyi változóhoz hozzáférni (névével hivatkozva vele műveletet végezni) csak az adott eljárásban illetve azokban az eljárásokban lehet, amelyek az egymást hívó eljárások hierarchiájában szigorúan az adott eljáráshoz kapcsolódnak, annak alárendeltjei: Más – az adott eljárással nem alárendelt kapcsolatban lévő – eljárásban szabadon használhatók az ugyanilyen nevű változók.

A PRIVATE ALL helyivé tesz minden – illetve LIKE esetén csak a *mintának* megfelelő nevű – változót. Az EXCEPT alkalmazása esetén a *mintának* megfelelő változók kivételével minden változó helyi lesz.

Lásd még: DECLARE, DO, PARAMETERS, PUBLIC, STORE

PUBLIC

PUBLIC <változólista>/[ARRAY <tömblista>]

A *változólistában* felsorolt változókat globális változónak jelöli ki, amelyek ezután minden eljárásból hozzáférhetők lesznek.

Az ARRAY kulcsszóval *tömbök* is globálissá tehetők.

Lásd még: DECLARE, DO, PARAMETERS, PRIVATE, RELEASE, RESTORE, SAVE, STORE

RELEASE

RELEASE <változólista>

vagy

RELEASE [ALL] [LIKE/EXCEPT <minta>]

vagy

RELEASE [MODULE <modullista>]
 [MENUS <vonalmenü-lista>]
 [POPUPS <rolómenü-lista>]
 [WINDOWS <ablaklista>]

Az utasítás törli az összes, illetve a *listákban* kijelölt *változókat*, *modulokat*, *menüket*, stb a memóriából.

Opciók:

- ALL** Törli az összes – illetve *LIKE* esetén csak a *mintának* megfelelő nevű – változót.
 Az *EXCEPT* alkalmazása esetén a *mintának* megfelelő változók kivételével töröl minden változót.
- MODULE** Törli a memóriából a LOAD paranccsal betöltött bináris (assembly szintű) programokat.
- MENUS** A *vonalmenülistában* felsorolt vonalmenük törlődnek a memóriából és a képernyőről.
- POPUPS** Törli a felsorolt *rolómenüket* a memóriából és a képernyőről.
- WINDOWS** Törli az *ablaklistában* kijelölt ablakokat a memóriából és a képernyőről.

Lásd még: CALL, CALL(), CLEAR ALL, CLEAR MEMORY, LOAD, RESTORE, RETURN, SAVE, STORE

RESTORE FROM

RESTORE FROM <memóriafájl-név> [ADDITIVE]

Betölti és aktivizálja a változókat, tömböket a megnevezett *memóriafájl*ból. Az ADDITIVE paraméter megadása esetén az eddig használt változók nem törölődnek, kiegészülnek a .mem fájl tartalmával.

Lásd még: PRIVATE, PUBLIC, SAVE, STORE

SAVE TO

SAVE TO <memóriafájl> [ALL LIKE/EXCEPT <minta>]

Elmenti egy *memóriafájl*ba az összes – illetve *LIKE* esetén csak a *mintának* megfelelő nevű – változót. Az *EXCEPT* alkalmazása esetén a *mintának* megfelelő változók kivételével minden változót.

Lásd még: PARAMETERS, PRIVATE, PUBLIC, RESTORE, SET SAFETY, STORE

C-6. Vezérlő szerkezetek

BEGIN TRANSACTION

```
BEGIN TRANSACTION [< elérési út > ]  
    < tranzakciós utasítások >  
END TRANSACTION
```

A BEGIN/END TRANSACTION utasítások az adatvédelmet szolgálják a fájl módosító műveletek végrehajtása során. Az utasítás kiadása után az adatfájlokban történt változásokat egy .log kiterjesztésű tranzakció- vagy másnéven naplófájlban naplózza, regisztrálja. Ha a tranzakciós utasítások rendellenesen megszakadnak, azaz a vezérlés nem éri el az END TRANSACTION utasítást, akkor az adatfájlok BEGIN TRANSACTION előtti állapota a ROLLBACK utasítás kiadásával – amely a naplófájlt használja fel – helyreállítható. Az END TRANSACTION utasítás végrehajtása a naplófájl törlését és ezzel a helyreállítás lehetőségének megszűnését jelenti.

Lásd még: COMPLETED(), ISMARKED(), RESET, ROLLBACK, ROLLBACK()

CALL

```
CALL < fájlnev > [WITH < kifejezés lista > ]
```

A vezérlést a korábban (a LOAD paranccsal) a memóriába töltött, a *fájlnev*-vel azonosított bináris (gépi kódra lefordított) programmodulra adja. Ha paramétereket kell átadni, akkor azokat a *kifejezéslistában* kell felsorolni.

Lásd még: CALL(), LOAD, RELEASE, RUN/!

CANCEL

```
CANCEL
```

Megszakítja a program futását, lezárja az összes nyitott programfájlt és visszatér az interpreterbe.

Lásd még: RESUME, RETRY, RETURN, SET REPROCESS, SUSPEND

DO

```
DO < program-fájlnev > / < eljárás-fájlnev >  
    [WITH < kifejezéslista > ]
```

Programfájlt vagy eljárást indít. Ha arra szükség van, a futtatás előtt a programot lefordítja (aktivizálja COMPILE utasítást). Ha paraméterek átadására van szükség, akkor azokat (legfeljebb 64-et) a WITH cikkely utáni *kifejezéslistában* kell felsorolni.

Lásd még: CANCEL, COMPILE, CREATE/MODIFY QUERY/VIEW, DEBUG, FUNCTION, MODIFY COMMAND, PARAMETERS, PRIVATE, PROCEDURE, PUBLIC, RESUME, RETURN, SET DEBUG, SET DEVELOPMENT, SET ECHO, SET PROCEDURE, SET TRAP, SUSPEND

DO CASE

```
DO CASE < feltétel >  
    < utasítások >  
[CASE < feltétel >  
    < utasítások > ]  
[OTHERWISE  
    < utasítások > ]  
ENDCASE
```

Többkimenetelű esemény után (a kimeneti) feltételek szerinti választás a lehetséges akciók közül. A kimeneteket jellemző *feltételek* rendre egy-egy CASE kulcsszó után szerepelnek.

Ha egy CASE utáni feltétel teljesül, úgy az azt követő utasítások kerülnek végrehajtásra, majd a vezérlés az ENDCASE utasítás utáni utasításra kerül, tehát a különböző feltételekhez (kimenetekhez) tartozó akciók (utasítások) közül mindig csak egy kerül végrehajtásra.

Ha egyik feltétel sem teljesül, akkor az OTHERWISE utáni utasítások érvényesülnek. Ha ilyenek nincsenek, akkor a vezérlés az ENDCASE utasítás utáni utasításra kerül anélkül, hogy valami történne.

Lásd még: DO, DO WHILE, IF, IIF()

DO WHILE

```
DO WHILE <feltétel >  
    <utasítások >  
    [LOOP]  
    <utasítások >  
    [EXIT]  
    <utasítások >  
ENDDO
```

Akciók (utasítás-sorozatok) ismétlődő végrehajtását teszi lehetővé.

Először a feltételvizsgálat történik. Ha ez igaz az utasítások kerülnek végrehajtásra, majd az ENDDO végrehajtása után ismét a ciklus elejére kerül a vezérlés, ahol a ismét a feltételvizsgálat következik, ... stb, egészen addig, amíg a WHILE mögötti *feltétel* igaz. Ha a *feltétel* nem teljesül, akkor a vezérlés az ENDDO utáni utasításra kerül.

Ha az utasítások között *LOOP* utasítás van, akkor a LOOP utáni utasításokat nem hajtja végre, hanem visszatér a ciklus elejére.

Az *EXIT* utasításra azonnal kilép a ciklusból az ENDDO utáni utasításra.

Lásd még: DO, DO CASE, IF, RETURN, SCAN

FUNCTION

```
FUNCTION <függvénynév >  
    <utasítások >  
RETURN <kifejezés >
```

A felhasználó által konstruált függvény, amely a FUNCTION *függvénynév*vel kezdődik és a függvényérték kiszámítását és egyéb feladatokat teljesítő utasításokkal folytatódik, majd a RETURN *kifejezés* utasítással zárul. A *kifejezés* függvény eredménye, a visszatérési érték.

Lásd még: DO, PARAMETERS, PROCEDURE, RETURN, SET PROCEDURE

IF

```
IF <feltétel >  
    <utasítások >  
[ELSE  
    <utasítások >]  
ENDIF
```

Kétkimenetelű esemény után (a kimeneti) feltétel szerinti választás a lehetséges akciók közül. A kimenetet jellemző *feltétel* az IF kulcsszó után áll.

Ha a *feltétel* teljesül, úgy az IF utasítást követő utasítások kerülnek végrehajtásra, majd a vezérlés az ENDIF utasítás utáni utasításra kerül.

Ha a *feltétel* nem teljesül, akkor az ELSE utáni utasítások érvényesülnek.

Ha ilyen nincs vagy az ELSE kulcsszó is hiányzik, akkor a vezérlés az ENDIF utasítás utáni utasításra kerül anélkül, hogy valami történne.

Lásd még: DO CASE, DO WHILE, IIF(), SCAN

ON ERROR

```
ON ERROR [<parancs >]
```

Eseménycsapda, amely a hibákat figyeli, és azok előfordulásakor lép működésbe.

Ha bárhol a program futása során hiba következik be, a hibaüzenet megjelenése helyett az ON ERROR utáni *parancsra* kerül a vezérlés. A *parancs* végrehajtása után a vezérlés visszakerül a hiba bekövetkezésekor végrehajtott utasítás követő utasításra (hacsak a *parancsban* más intézkedés nem történik). A *parancs* lehet lefordított program, eljárás, függvény hívása.

A parancs nélkül kiadott ON ERROR a hibafigyelést megszünteti.

Lásd még: INKEY(), LASTKEY(), ON READERROR, PROCEDURE, READKEY(), RETRY, RETURN, SETESCAPE, SET, PROCEDURE, UPPER(), WAIT

ON ESCAPE**ON ESCAPE [< parancs >]**

Eseménycsapda, amely az Esc billentyű lenyomását figyeli, és ekkor működésbe lép.

Há bárhol a program futása az Esc billentyű lenyomásra kerül, az ON ESCAPE utáni *parancsra* kerül a vezérlés. A *parancs* végrehajtása után a vezérlés visszakerül az Esc lenyomásakor végrehajtott utasítást követő utáni utasításra (hacsak a *parancsban* más intézkedés nem történik). Bizonyos műveletektől (Pl: indexelés, másolás, stb) nem szakít félbe. A *parancs* lehet lefordított program, eljárás, függvény hívása.

A parancs nélkül kiadott ON ESCAPE a hibafigyelést megszünteti.

Lásd még: INKEY(), LASTKEY(), ON READERROR, PROCEDURE, READKEY(), RETRY, RETURN, SET ESCAPE, SET PROCEDURE, UPPER(), WAIT

ON KEY

ON KEY [LABEL <billentyűnév>] [<parancs>]

Eseménycsapda, amely a *billentyűnévvel* megadott billentyű leütését figyeli, és ekkor lép működésbe.

Ha bárhol a program futása az *billentyűnévvel* megadott billentyű lenyomására kerül, az ON KEY utáni *parancsra* kerül a vezérlés. A *parancs* végrehajtása után a vezérlés visszakerül az Esc lenyomásakor végrehajtott utasítást követő utáni utasításra (hacsak a *parancsban* más intézkedés nem történik). Bizonyos műveleteket (Pl: indexelés, másolás, stb) nem szakít félbe. A *parancs* lehet lefordított program, eljárás, függvény hívása.

A *billentyűnév* a képernyőn is megjeleníthető karaktereknél maga a karakter, a nem megjeleníthető billentyűknél (Pl: F1,F2, stb) általában a billentyű felirata. A nevet nem szabad idézőjelbe zárni. Például:

Billentyű	Név
F1-F10	F1, F2, F3, stb
Enter	Enter
Esc	Esc
Del	Del
PrtSc	PrtSc
Backspace	Backspace
Tab	Tab
Shift-Tab	Shift-Tab
→	Rightarrow
←	Leftarrow
↓	Uparrow
↑	Downarrow
PgUp	PgUp
PgDn	PgDn
Home	Home
End	End
Shift	Shift
Ins	Ins
Ctrl-és más	Ctrl-és más
Alt-és más	Alt-és más

A parancs nélkül kiadott ON KEY a hibafigyelést megszünteti.

Lásd még: INKEY(), LASTKEY(), ON READERROR, PROCEDURE, READKEY(), RETRY, RETURN, SET ESCAPE, SET PROCEDURE, UPPER(), WAIT

ON PAD

ON PAD <menüpont> OF <vonalmenünev>
 [ACTIVATE POPUP <rolómenünev>]

Eseménycsapda, amely a *vonalmenünevvel* megadott menü *menüpontját* figyeli, és akkor lép működésbe, ha arra mutat a sávkurzor. Ilyenkor a aktívál egy *rolómenüt*, amely megjelenik a képernyőn, és a függőleges kurzor-mozgató billentyűvel a rolómenüben is lehet mozogni. Az aktivált rolómenüt a vízszintes kurzormozgató billentyűvel is el lehet hagyni, amivel a másik (*vonalmenünev*) menüben történik mozgás.

Az ACTIVATE POPUP nélkül ON PAD a megjelölt menüpontot megbénítja, azaz nem lehet kiválasztani.

Lásd még: DEFINE MENU, DEFINE PAD, DEFINE POPUP, ON SELECTION PAD

Példa:

ON PAD leves OF vacsora ACTIVATE levesek

ON PAGE

ON PAGE [AT LINE <N-kifejezés> <parancs>]

Eseménycsapda, amely a nyomtatott sor számát figyeli, és akkor lép működésbe, ha az (? , ?? és PAGE EJECT használata esetén) eléri a *N-kifejezésben* megadott értéket. Ilyenkor a *parancs* kerül végrehajtásra, majd ezután az az adott sorszám elérésekor végrehajtott utasítást követő utasítás (hacsak a *parancsban* más intézkedés nem történik). A *parancs* lehet lefordított program, eljárás, függvény hívása.

Leggyakoribb alkalmazás nyomtatás esetén a fejléc elkészítése.

A parancs nélkül kiadott ON PAGE a lapkezelést megszünteti.

Lásd még: PRINTJOB, PROCEDURE, REPORT FORM, SET PRINTER, _plength, _plineno

ON READERROR

ON READERROR [<parancs>]

Eseménycsapda, amely az előforduló adatbeviteli hibákat figyeli, és ezek előfordulása esetén lép működésbe.

Ha a program futása során bármelyik @ ...GET utasításnál adatbeviteli hiba következik be (a RANGE, VALID nem teljesül, vagy rossz a dátum), a hibaüzenet megjelenése helyett az ON READERROR utáni *parancsra* kerül a vezérlés. A *parancs* végrehajtása után a vezérlés visszakerül a hiba bekövetkezésekor végrehajtott @ ...GET utasításra, így a bevitel megismételhető (hacsak a *parancs*-ban más intézkedés nem történik). A *parancs* lehet lefordított program, eljárás, függvény hívása.

A parancs nélkül kiadott ON READERROR a hibafigyelést megszünteti.

Lásd még: @, APPEND, CHANGE, EDIT, INSERT, ON ERROR, READ

ON SELECTION PAD

ON SELECTION PAD <menüpont> OF <vonalmenünev> [<parancs>]

Eseménycsapda, amely a *vonalmenünevvel* megadott vonalmenü *menüpont*-ját figyeli, és akkor lép működésbe, ha arra esik a választás. Ilyenkor a *parancs* kerül végrehajtásra, majd ezután a vezérlés visszatér a *menüpontra* (hacsak a *parancs*-ban más intézkedés nem történik). A *parancs* lehet lefordított program, eljárás, függvény hívása.

A parancs nélkül kiadott ON SELECTION PAD a megjelölt menüpontot megbénítja, azaz kiválasztásakor nem történik semmi.

Lásd még: DEFINE MENU, DEFINE PAD, MENU(), ON PAD, PAD(), PROMPT()

ON SELECTION POPUP

ON SELECTION POPUP <rolómenünev> / ALL [<parancs>]

Eseménycsapda, amely a *rolómenünevvel* megadott – vagy *ALL* kulcszó esetén mindegyik – (POPUP) rolómenüben a választás (Enter lenyomása) megtörténtét figyeli, és ekkor működésbe lép. Ilyenkor a rolómenü eltűnik, és a *parancs* kerül végrehajtásra, majd a rolómenü ismét aktivizálódik (hacsak a *parancsban* más intézkedés nem történik). A *parancs* lehet lefordított program, eljárás, függvény hívása. Az *ALL* alkalmazásak A parancs nélkül kiadott ON SELECTION POPUP a megjelölt menüpontot megbénítja, azaz választásakor nem történik semmi.

Lásd még: BAR(), DEFINE BAR, DEFINE POPUP, POPUP(), PROMPT()

PRINTJOB

PRINTJOB

<utasítások>

ENDPRINTJOB

Egy nyomtatási köteget határoz meg. Az előzőleg – a rendszerváltozók segítségével – beállított nyomtatási paramétereket figyelembevéve a nyomtatást előkészíti (lapemelés, stb), a megadott példányszámnak (*_pcopies*) megfelelően ismétli, majd a kinyomtatás után szükséges tevékenységeket elvégzi (lapemelés, nyomtatóvisszaállítás, stb).

Lásd még: ON PAGE, PROCEDURE, REPORT FORM, RETURN, SET PRINTER, SET TALK, *_pcodes*, *_pcolno*, *_pcopies*, *_pject*, *_plineno*, *_pscodes*

RESUME

RESUME

A SUSPEND utasítással felfüggesztett program folytatása a következő program-utasítással.

Lásd még: BEGIN/END TRANSACTION, CANCEL, CLEAR, DEBUG, RETURN, ROLLBACK, SUSPEND

RETRY

RETRY

Többnyire hibakezelésre használatos (ON ERROR). Befejezi az (hibakezelő) eljárás végrehajtását, és a hívóprogramban nem a hívást követő utasításra lép vissza, hanem magára a hívást előidéző utasításra ezzel megismételve annak végrehajtását.

Lásd még: ERROR(), ON ERROR, RETURN

RETURN

RETURN [TO MASTER / <eljárásnév>] / [<kifejezés>]

A meghívott program, eljárás és függvény befejezésére, és az egymást hívó eljárások (programok) hierarchiájában a visszatérésre szolgál. Alapértelmezésben az éppen végrehajtás alatt lévő eljárást (programot, függvényt) hívó program következő sora kerül végrehajtásra.

Opciók:

- TO MASTER** Visszatérés a hierarchiában legmagasabb szintű programba, a *főprogramba*.
- TO <eljárásnév>** Visszatérés a közvetlenül hívó program helyett a hierarchiában magasabb szintű programba (eljárásba). Az eljárásnév csak aktív (az aktuális eljárás szigorúan fölérendelt) eljárást jelölhet meg.
- <kifejezés>** Függvény meghívásakor használható. A *kifejezés* értéke (visszatérési érték) átadásra kerül a közvetlenül hívó programnak, és a visszatérés is ide történik.

Lásd még: COMPILE, DO, ERROR(), FUNCTION, PARAMETER, PRIVATE, PROCEDURE, PUBLIC, RESUME, SUSPEND

SCAN

```
SCAN [<hatáskör>] [FOR <feltétel>] [WHILE <feltétel>]]  
    [<utasítások>]  
    [LOOP]  
    [<utasítások>]  
    [EXIT]  
    [<utasítások>]  
ENDSCAN
```

Ugyanúgy működik, mint a DO WHILE ciklus az alábbi különbségekkel:

Megkeresi azt a következő, illetve induláskor a legelső rekordot, – hacsak ezt a *hatáskör* másként nem diktálja –, amely megfelel a *FOR* cikkely *feltételének*, és a rekordot aktuálissá téve a ciklusbeli utasítások rendelkezésére bocsátja. A *WHILE* cikkely nem kötelező.

A ciklust befejezi – a vezérlést az ENDSCAN utáni utasításra adja – ha:

- a *WHILE feltétel* nem teljesül (ebben a DO WHILE-lal azonos),
- a keresés elérte az adatfájl végét,
- a megtalált rekord kívül esik a *hatáskörön*.

Lásd még: CONTINUE, DO WHILE, LOCATE

SUSPEND

SUSPEND

Az utasítás a program, eljárás futását felfüggeszti. Folytatni a RESUME utasítással lehet.

Lásd még: CANCEL, COMPILE, DEBUG, DO, RESUME, SET ECHO, SET PROCEDURE, SET STEP, SET TALK, SET TRAP

C-7. Ablak- és menükezelés

ACTIVATE MENU

ACTIVATE MENU <vonalmenünev> [PAD <menüpont>]

Aktivizálja, azaz a képernyőn megjeleníti és használhatóvá teszi a korábban már kijelölt *vonalmenünev* menüt. A PAD cikkely használatakor a sávkurzor a *menüpont* pontra áll, egyébként az első menüpontra.

Lásd még: DEACTIVATE MENU, DEFINE MENU, MENU(), ON PAD, PAD(), SHOW MENU

ACTIVATE POPUP

ACTIVATE POPUP <rolómenünev>

Aktivizálja, azaz a képernyőn megjeleníti és használhatóvá teszi a korábban már kijelölt *rolómenünev* rolómenüt.

Egyidőben csak egy rolómenü lehet aktív.

Lásd még: BAR(), DEACTIVATE POPUP, DEFINE POPUP, POPUP(), PROMPT(), SHOW POPUP

ACTIVATE SCREEN

ACTIVATE SCREEN

A képernyőt aktivizálja. Az adatmegjelenítés – esetleges korábban aktív ablak helyett – az egész képernyőn történik.

Lásd még: ACTIVATE WINDOW, DEACTIVATE WINDOW, DEFINE WINDOW

ACTIVATE WINDOW

ACTIVATE WINDOW <ablaknév> /ALL

Aktivizálja — a képernyőn megjeleníti és használhatóvá teszi — a korábban már kijelölt *ablaknév* ablakot. Az adatmegjelenítés az ablakban történik.

Egyszerre csak egy — a legutoljára aktivált — ablak lehet aktív, így az ALL opció is csak az utoljára aktivált ablakot teszi használhatóvá, bár minden korábban kijelölt ablakot megjelenít a képernyőn.

Lásd még: CLEAR WINDOW, DEACTIVATE WINDOW, DEFINE, WINDOW, MOVE WINDOW, RESTORE WINDOW, SAVE, WINDOW, SET BORDER

DEACTIVATE MENU

DEACTIVATE MENU

Használaton kívül helyezi és törli a képernyőről az éppen aktív vonalmenüt. A megjelenített adatokat és állapotot elmenti a memóriába, így azok újabb aktiváláskor változatlanul jelennek meg.

Az ESC megnyomása hasonló működést eredményez.

Lásd még: ACTIVATE MENU, CLEAR MENUS, DEFINE MENU, ON SELECTION, RELEASE MENUS

DEACTIVATE POPUP

DEACTIVATE POPUP

Használaton kívül helyezi és törli a képernyőről az éppen aktív rolómenüt. A megjelenített adatokat és állapotot elmenti a memóriába, így azok újabb aktiváláskor változatlanul jelennek meg.

AZ ESC megnyomása hasonló működést eredményez.

Lásd még: ACTIVATE POPUP, CLEAR POPUPS, DEFINE POPUP, ON SELECTION, RELEASE POPUPS

DEACTIVATE WINDOW

DEACTIVATE WINDOW <ablaknév> / ALL

Használaton kívül helyezi és törli a képernyőről az *ablaknév* ablakot. Az megjelenített adatokat és állapotot elmenti a memóriába, így azok újabb aktiváláskor változatlanul jelennek meg.

A használaton kívül helyezéssel együtt aktiválja a megelőzően aktív ablakot, vagy ha ilyen nem volt, a képernyőt. Az *ALL* opció ugyanezt teszi minden ablakra, így a teljes képernyő aktív lesz.

Lásd még: ACTIVATE SCREEN,ACTIVATE WINDOW, DEFINE WINDOW, MOVE WINDOW, RESTORE WINDOW, SAVE WINDOW

DEFINE BAR

```

DEFINE BAR <sorszám> OF <rolómenünev>
    PROMPT <C-kifejezés>
    [MESSAGE <C-kifejezés>]
    [SKIP [FOR <feltétel>]]

```

A *rolómenünev* rolómenü egy elemét, menüpontját jelöli ki, annak paramétereit határozza meg. Azt, hogy a menüpont – a felső kerettől számítva – hányadik a sorban, a *sorszám* határozza meg.

A PROMPT kulcsszót követő *C-kifejezés* a menüpont megnevezése, amely a képernyőn megjelenik.

Opciók:

- MESSAGE** A megnevezés magyarázatként kiegészíthető a *C-kifejezéssel*, amely a 24.sor közepére kerül.
- SKIP** Csökkentett fényerővel jelenik meg, nem választható ki, azaz a sávkurzor átlépi.
- SKIP FOR** Ha a *feltétel* teljesül, akkor normál menüpontként viselkedik, ha nem, akkor csökkentett fényerővel jelenik meg, nem választható ki, tehát a sávkurzor átlépi.

Menüelemet csak olyan rolómenühöz lehet meghatározni, amelynek kijelölésénél (DEFINE POPUP) a PROMPT FIELD, PROMPT FILES és a PROMPT STRUCTURE cikkelyek egyike sem szerepel.

Példa:

```

DEFINE POPUP levesek FROM 10,4 TO 18, 40
DEFINE BAR 1 OF levesek PROMPT "Ujházi tyúkhúsleves"
DEFINE BAR 2 OF levesek PROMPT "Ököruszályleves"
DEFINE BAR 3 OF levesek PROMPT "Erőleves tojással"
DEFINE BAR 4 OF levesek PROMPT "Fecskefészekleves"
    MESSAGE "Igazi kínai ínyencség. Import alapanyag!"
... stb

```

Lásd még: ACTIVATE POPUP, BAR(), DEACTIVATE POPUP, DEFINE, POPUP, POPUP(), ON SELECTION POPUP, PROMPT(), SHOW POPUP

DEFINE MENU

```
DEFINE MENU <vonalmenünev> [MESSAGE <C-kifejezés>]
```

Kijelöli a *vonalmenünev* vonalmenüt, amelyhez a menüpontok a DEFINE PAD utasítással kell kiválasztani.

A MESSAGE utáni *C-kifejezés* a 24. sor közepére kerül.

Lásd még: DEFINE PAD, MENU(), ON PAD, ON SELECTION PAD, PAD()

DEFINE PAD

```
DEFINE PAD <menüpontnev> OF <vonalmenünev>
    PROMPT <C-kifejezés> [AT <sor> , <oszlop> ]
    [MESSAGE <C-kifejezés> ]
```

A *vonalmenünev* menü *menüpontnev* elemét, menüpontját jelöli ki, annak paramétereit határozza meg.

A PROMPT kulcsszót követő *C-kifejezés* a menüpont megnevezése, amely a képernyőn megjelenik.

Az előzőleg definiált vonalmenü elemeit választja ki. A <C-kifejezés> a menüpontnev tartalma.

Opciók:

AT A *sor, oszlop* a menüpont megnevezésének koordinátái.

MESSAGE Adott menüponthoz tartozó üzenet, amely a képernyő közepére a 24. sorba kerül.

Példa:

```
DEFINE MENU vacsora
DEFINE PAD elomel    PROMPT "Meleg előételek" AT 2,2
DEFINE PAD leves    PROMPT "Levesek" AT 2,10
DEFINE PAD fozel    PROMPT "Főzelékek, köreték" AT 2,18
DEFINE PAD sal      PROMPT "Saláták" AT 2,18
    MESSAGE "Ingyenes! Tetszés szerint akármennyit választhat!"
... stb
```

Lásd még: DEFINE MENU, ON PAD, ON SELECTION PAD, PAD(), SET SCOREBOARD

DEFINE POPUP

```

DEFINE POPUP <rolómenünev> FROM < sor1 > , < oszlop1 >
      TO < sor2 > , < oszlop2 >
      [PROMPT FIELD <mezőnev>
        /PROMPT FILES [LIKE <minta>
          /PROMPT STRUCTURE]
      [PROMPT MESSAGE <C-kifejezés> ]

```

Kijelöli a *rolómenünev* rolómenüt, amelyhez a menüpontokat a DEFINE BAR utasítással kell kiválasztani.

A rolómenü keretének bal felső sarkát a *sor1*, *oszlop1*, jobb alsó sarkát a *sor2,oszlop2* határozza meg (a @ ...TO utasításhoz hasonlóan). Ha a menüelemek száma nagyobb annál mint amennyi a menüablakban elfér, a kurzorbillentyűk hatására az elemek az ablakban "fel-le gördülnek".

Opciók:

- FIELD** Az aktuális adatfájl valamennyi rekordja *mezőnev* mezőjének értékei jelennek meg, amelyből választani lehet.
- FILES** Az aktuális alkönyvtárban lévő valamennyi fájl, illetve a LIKE esetén csak a *mintával* egyező fájlok jelennek meg, amelyből választani lehet.
- STRUCTURE** Megjelenik a rolómenüben az aktuális adatfájl szerkezete.
- MESSAGE** A magyarázó *C-kifejezés* a 24. sor közepére kerül.

A PROMPT FIELD, PROMPT FILES, PROMPT STRUCTURE cikkelyekkel kijelölt rolómenükhöz nem lehet menüelemeket meghatározni (DEFINE BAR). Ilyenkor (egyébként is) kiválasztott menüelemekhez a PROMPT() függvénnyel férhetünk hozzá.

Példa:

```

DEFINE POPUP levesek FROM 10,4 TO 18, 40

```

Lásd még: ACTIVATE POPUP, BAR(), DEFINE BAR, ON SELECTION, POPUP, POPUP(), PROMPT()

DEFINE WINDOW

```
DEFINE WINDOW <ablaknév> FROM < sor1 > , < oszlop1 >
      TO < sor2 > , < oszlop2 > [DOUBLE/PANEL/NONE/<keret sztring >]
      [COLOR [<normál színpár >] [, <kiemelt színpár >]
      [, <keret színpár >
```

Kijelöli az *ablaknév* ablakot és beállítja annak paramétereit jellemzőit. A bal felső sarkát a *sor1, oszlop1*, jobb alsó sarkát a *sor2, oszlop2* határozza meg (a @ ...TO utasításhoz hasonlóan).

Keretet rajzol az ablak köré, a *DOUBLE/PANEL/keret sztring* opciókat lásd a @ ...TO utasításnál.

NONE opció esetén nem rajzol semmilyen keretet.

A COLOR kulcsszó mögött külön megadható az ablakban megjelenő adatok és a háttér színe a *normál színpárral*, a kiemelten megjelenítendő adatok (GET, stb) színe a *kiemelt színpárral* és a keret színei a *keret színpárral*.

Lásd még: @ ...TO, ACTIVATE WINDOW, DEACTIVATE WINDOW, RESTORE WINDOW, SAVE WINDOW, SET BORDER, SET COLOR

MOVE WINDOW

```
MOVE WINDOW <ablaknév> TO < sor > , < oszlop >
      / BY <delta sor > , <delta oszlop >
```

A kijelölt ablakot áthelyezi.

A *sor, oszlop* az ablak bal felső sarkának új helye lesz a képernyőn. Az ablak relatív (az aktuális helyzetéhez viszonyítot) mozgathatásához a *BY delta sor, delta oszlop* cikkelyt kell alkalmazni.

Lásd még: ACTIVATE WINDOW, DEACTIVATE WINDOW, DEFINE WINDOW, RESTORE WINDOW, SAVE WINDOW, SET WINDOW OF MEMO

RESTORE WINDOW

```
RESTORE WINDOW <ablaklista>/ALL FROM <fájlnev >
```

Betölti a memóriába az *ablaklistában* felsorolt, vagy *ALL* esetén az összes ablakdefiníciót, amelyeket a *fájlnev* .win kiterjesztésű fájl tartalmaz. A betöltött ablakokat nem kell kijelölni, elég csak aktiválni.

Lásd még: DEFINE WINDOW, SAVE WINDOW

SAVE WINDOW

SAVE WINDOW <ablaknév> lista / ALL TO <fájlnév>

Elmenti az *ablaklistában* felsorolt, vagy *ALL* esetén az összes ablakdefiníciót a *fájlnév* elnevezésű fájlba, amely automatikusan *.win* kiterjesztést kap.

Visszatöltés a RESTORE WINDOW utasítással lehetséges.

Lásd még: DEFINE WINDOW, RESTORE WINDOW

SHOW MENU

SHOW MENU <vonalmenünév> [PAD <menüpontnév>]

Aktíválás nélkül megjeleníti a képernyőn a *vonalmenünév* menüt. A *PAD* *menüpontnév* által kijelöl menüpontot kiemeli. A vonalmenü nem aktív, tehát nem használható.

Lásd még: ACTIVATE MENU, CLEAR, CLEAR MENU, DEFINE MENU, DEFINE PAD

SHOW POPUP

SHOW POPUP <rolómenünév>

Aktíválás nélkül megjeleníti a képernyőn a *rolómenünév* rolómenüt. A menü nem aktív, tehát nem használható.

Lásd még: ACTIVATE POPUP, CLEAR, DEFINE BAR, DEFINE POPUP(), POPUP()

C-8. Tervező utasítások

CREATE/MODIFY APPLICATION

CREATE/MODIFY APPLICATION <fájlnev>/?

A CREATE új alkalmazást hoz létre a megadott *fájlnev*vel, a MODIFY utasítással egy létező alkalmazás módosítható.

Ha MODIFY esetén a ? cikkelyre a létező alkalmazások megjelennek a képernyőn, amelyek közül lehet (a sávkurzorral és az Enter leütésével) választani.

Az utasítás hatására az Alkalmazásgenerátor (Application Generator) lép működésbe. Ez programrendszerek tervezését támogató, vizuális megjelenítéssel és menürendszerrel működő interaktív, párbeszédés felhasználói csatoló, amellyel az utasításleírások ismerete nélkül (de kellő angol tudással) a felhasználói célokat szolgáló, működő rendszer készíthető úgy, hogy forrásprogramlista is keletkezik. Részletesen lásd az Alkalmazásgenerátor c. fejezetet.

CREATE/MODIFY LABEL

CREATE/MODIFY LABEL <fájlnev>/?

Az utasítással *fájlnev* új címkeformátum-fájl (.lbl) hozható létre, vagy a már létező módosítható. A *fájlnev* helyett ? is szerepelhet, amelynek hatására megjeleníti a fájl-listát és a használni kívánt fájl (a sávkurzor segítségével) kijelölhető.

A címkeformátum-fájl alkalmazásával (LABEL FORM) az adatfájl adatai címke formájában jeleníthetők meg, amelyek leggyakrabban áruk felcímkézésére, öntapadós levélboríték-címzések készítésére használatosak.

Az utasítás hatására egy interaktív tervezőprogram lép működésbe, amely vizuális megjelenítéssel, menürendszerű párbeszédés kommunikációval támogatja a megfelelő formátum megtervezését.

Részletesebb információk a Control Center c. fejezetben.

Lásd még: LABEL FORM, SET CATALOG, SET SAFETY, SET VIEW

CREATE/MODIFY QUERY/VIEW

CREATE/MODIFY QUERY/VIEW <fájlnev>/?

Az utasítással *fájlnev* új nézetfájl (.qbe, .upd) hozható létre, vagy a már létező módosítható. A *fájlnev* helyett ? is szerepelhet, amelynek hatására megjeleníti a fájl-listát és a használni kívánt fájl (a sávkurzor segítségével) kijelölhető.

A nézetfájl alkalmazásával (SET VIEW) egy vagy több egymáshoz kapcsolt adatfájl szűrőfeltételek által meghatározott rekordjai válnak hozzáférhetővé a nézetfájlnak megfelelő mezők, kapcsolatok szerint.

Az utasítás hatására egy interaktív tervezőprogram lép működésbe, amely vizuális megjelenítéssel, menürendszerű párbeszédés kommunikációval támogatja a megfelelő feltételek és formátum megtervezését.

Részletesebb információk a Control Center c. fejezetben.

Lásd még: SET CATALOG, SET FILTER, SET SAFETY, SET VIEW

CREATE/MODIFY REPORT

CREATE/MODIFY REPORT <fájlnev>/?

Az utasítással *fájlnev* új listaformátum-fájl (.frm) hozható létre, vagy a már létező módosítható. A *fájlnev* helyett ? is szerepelhet, amelynek hatására megjeleníti a fájl-listát és a használni kívánt fájl (a sávkurzor segítségével) kijelölhető.

A listaformátum-fájl alkalmazásával (REPORT FORM) az adatfájl rekordjaiból a formátumfájlnak megfelelő fejlécekkel, formátummal és összegfokozatokkal rendelkező lista (jelentés) készíthető.

Az utasítás hatására egy interaktív tervezőprogram lép működésbe, amely vizuális megjelenítéssel, menürendszerű párbeszédés kommunikációval támogatja a megfelelő lista megtervezését.

Részletesebb információk a Control Center c. fejezetben.

Lásd még: REPORT FORM, SET CATALOG, SET DESIGN, SET VIEW

CREATE/MODIFY SCREEN

CREATE/MODIFY SCREEN <fájlnev>/?

A CREATE új képernyőformátum-fájlt (.scr) hoz létre a megadott *fájlnev*vel, a MODIFY utasítással egy létező formátum módosítható.

Ha MODIFY esetén a ?-re a létező formátumfájlok megjelennek a képernyőn, amelyek közül (a sávkurzorral és az Enter leütésével) választani lehet. Az utasítás hatására a képernyők tervezését támogató, menürendszerrel működő interaktív, párbeszédés felhasználói csatoló lép működésbe.

Részletesen lásd a Control Center c. fejezetet.

Lásd még: @, APPEND, BROWSE, EDIT, INSERT, SET CATALOG, SET FORMAT, SET SAFETY, SET VIEW

MODIFY COMMAND/FILE

MODIFY COMMAND/FILE <fájlnév> [WINDOW <ablaknév>]

A MODIFY COMMAND új forrásprogram-fájlok létrehozására, és már létezők módosítására szolgál.

A MODIFY FILE a *fájlnév* szöveges jellegű (.frm, .txt, stb) új fájl létrehozására vagy ha az létezik, annak módosítására szolgál. A *fájlnév*nek tartalmaznia kell a kiterjesztést is.

Az utasítás hatására az dBASE IV aktualizált szövegszerkesztője lép működésbe mindig az éppen aktív ablakban, illetve a WINDOW cikkelyben megadott *ablaknév* ablakban.

Bővebb információk a Control Center c. fejezetben.

Nem az eredeti szövegszerkesztő is használható, ha az a CONFIG.DB fájlban szerepel.

Lásd még: COMPILE, CREATE, DO, NOTE/*/&&, RENAME, SET DEVELOP, TYPE, UPDATE

C-9. Különleges utasítások

???

??? < C-kifejezés >

Az installált nyomtatóvezérlő kikerülésével, közvetlenül a nyomtatónak küldi a *C-kifejezést*, amely általában olyan nyomtatóvezérlő kód(sorozat), amelyet az installált nyomtatómeghajtó nem kezel, de a nyomtató igen.

A nyomtatóvezérlő karakterek nyomtatónként változnak, és ezek a nyomtató felhasználói kézikönyvében található meg. A nyomtatóvezérlő karakterek egy jelentős része nem megjeleníthető karakter. Ezeket a különleges karaktereket vagy az ASCII decimális kódjukkal és a CHR() függvény alkalmazásával, vagy a karaktereket előállító billentyűkombinációk nevével lehet meghatározni. A kódokat vagy a billentyűfelirat kombinációit a { } zárójelbe kell zárni. Az alábbi táblázat a nem nyomtatható karakterek kódjait és a megfelelő kombinációk leírasi módját sorolja fel:

ASCII kódok	Leírás módja
0	{CTRL-@} vagy {NULL}
1	{CTRL-A}
2	{CTRL-B}
3	{CTRL-C}
4	{CTRL-D}
5	{CTRL-E}
6	{CTRL-F}
7	{CTRL-G} vagy {BELL}
8	{CTRL-H} vagy {BACKSPACE}
9	{CTRL-I} vagy {TAB}
10	{CTRL-J} vagy {LINEFEED}
11	{CTRL-K}
12	{CTRL-L}
13	{CTRL-M} vagy {RETURN}
14	{CTRL-N}
15	{CTRL-O}
16	{CTRL-P}
17	{CTRL-Q}
18	{CTRL-R}
19	{CTRL-S}
20	{CTRL-T}
21	{CTRL-U}
22	{CTRL-V}
23	{CTRL-W}
24	{CTRL-X}

ASCII kódok	Leírás módja
25	{CTRL-Y}
26	{CTRL-Z}
27	{CTRL-[} v. {ESC} v. {ESCAPE}
28	{CTRL-\}
29	{CTRL-]}
30	{CTRL-^}
31	{CTRL-}
127	{DEL} vagy {DELETE}

Példa:

```
.??? CHR(27) + "E"
.??? "{ESC}E"
.??? "{27}E"
```

Lásd még: ?/?, @, CHR()

ASSIST

ASSIST

A Control Centert üzembe helyezi.

Lásd még: Control Center c. fejezet.

COMPILE

COMPILE <fájlnev> [RUNTIME]

Létrehoz egy futtatható programfájlt (.dbo). RUNTIME opció esetén a fordítási hibák kiírásra kerülnek.

Lásd még: DEBUG, DO, FUNCTION, MODIFY COMMAND, PROCEDURE, SET DEBUG, SET DEVELOPMENT, SET EXACT, SET PATH, SET PROCEDURE, SET TRAP

CONVERT

CONVERT [TO <N-kifejezés >]

Hozzáfűzi a `_dbaselock` nevű mezőt az aktuális adatfájlhoz. A mező a többfelhasználós üzemmódban rendszeradminisztrációs célokat szolgál. Az *N-kifejezés* a mező hosszát határozza meg.

Alapértelmezés: 16. Értéktartomány: 8 – 24.

Lásd még: `CHANGE()`, `FLOCK()`, `LKSYS()`, `LOCK()`, `NETWORK()`, `SET LOCK`, `SET REPROCESS`, `UNLOCK`

DEBUG

DEBUG <fájlnev >/<eljárásnev > [WITH <paraméter lista >]

Betölti a dBASE IV beépített, interaktív hibakereső, tesztelő rendszerét (`DEBUGGER`), és elindítja a *fájlnev/eljárásnev* program futását és tesztelését.

A képernyő egyik ablakában megjelennek a *fájl/eljárás* utasításai, más ablakokban különféle információkat szolgáltat részben automatikusan, részben kérdésekre. A hibakeresést, tesztelést az alábbi egybetűs utasításokkal lehet vezérelni. (Ez látszik a help-ablakban is.):

- B A program megszakításának feltételei vihetők be a billentyűzetről.
- D A változók, mezők határozhatók meg, amelyeknek értékeit folyamatosan megjeleníti.
- E A forrásprogram sorai javíthatóak, szerkeszthetőek lesznek.
- L A megadott sortól folytatja a program futását.
- [sorszám] N Az adott sorszámnál befejezi a tesztelést.
- P Információt ad a program futásának pillanatnyi állapotáról.
- Q Kilép a DEBUG utasításból.
- [szám] S A megadott számú lépést hajtja végre.
- X Kilép a hibakeresésből.

Lásd még: `COMPILE`, `DO`, `MODIFY COMMAND`, `SET DEBUG`, `SET ECHO`, `SET PROCEDURE`, `SET STEP`, `SET TALK`, `SET TRAP`.

DEFINE BOX

```
DEFINE BOX FROM <oszlop1> TO <oszlop2>
          HEIGHT <N-kifejezés>
          [AT LINE < sor > ][SINGLE/DOUBLE/<keret sztring >]
```

Nyomtatott dokumentum egy részét emeli ki azzal, hogy keretet rajzol köré. Csak nyomtatás esetén működik.

Az *oszlop1* a keret baloldali határa, az *oszlop2* a jobboldali.

Az *N-kifejezés* a keret magassága.

Az AT LINE *sor* a keret felső határa. Ennek hiányában a keret mindig az utasítás kiadásakor aktuális nyomtatási sorban kezdődik.

A *SINGLE/DOUBLE/keret sztring* opciókat lásd a SET BORDER illetve a @ ...TO utasításoknál.

Lásd még: SET BORDER

DIR

```
DIR [[ON] < meghajtó > :] [[LIKE][ < elérési út > ] < minta >]
```

Paraméter nélkül az aktuális alkönyvtár összes adatfájljának neve és adatai jelennek meg a képernyőn.

Opciók:

ON Az aktuálistól eltérő *meghajtó* kiválasztása.

LIKE A *mintával* bármilyen típusu fájl – vagy ezek egy csoportja – kiválasztható. A mintában a ? és * helyettesítő karakterek szabadon alkalmazhatók. Az *elérési út* szerepeltetésével az aktuális alkönyvtár helyett más alkönyvtár is kijelölhető.

Lásd még: LIST/DISPLAY FILES, SET AUTOSAVE

EJECT

EJECT

Lapdobás. A következő lap elejére áll a nyomtató. A PROW(), és PCOL() függvények értéke 0 lesz.

Lásd még: ???, EJECT PAGE, ON PAGE, PCOL(), PRINTSTATUS(), PROW(), SET PRINTER

EJECT PAGE**EJECT PAGE**

A következő lap elejére áll, vagy pedig az ON PAGE utasításban meghatározott módon halad előre a nyomtatófej.

Lásd még: ?/??, EJECT, ON PAGE, SET ALTERNATE, SET CONSOLE, SET PRINTER

ERASE**ERASE <fájlnév >**

Törli a *fájlnévvel* azonosított fájlt. A *fájlnévnek* a kiterjesztést is tartalmaznia kell.

Lásd még: CLOSE, DELETE, DELETE TAG(), FILE(), USE

HELP**HELP <utasítás/függvény >]**

Megjelennek az *utasítás/függvény* használatára vonatkozó segítőinformációk.

Lásd még: SET HELP, SET INSTRUCT

LOGOUT**LOGOUT**

Az aktuális felhasználó kilép a hálózati rendszerből. A felhasználó által megnyitott adat-, index-, és egyéb fájlok lezárulnak. Ha a PROTECT utasítás érvényben van, megjelenik a belépési (log-in) képernyő, lehetőséget adva egy új felhasználó belépésére. Egyébként visszatér a rendszer az interpreterhez.

Lásd még: PROTECT

NOTE

NOTE < szöveg >

vagy

* < szöveg >

vagy

< utasítás > && < szöveg >

Az utasítás végrehajtáskor figyelmen kívül hagyott, elsősorban a program áttekinthetőségét javító, utasításokat magyarázó *szöveg* helyezhető el a forrásprogramban. A NOTE és * használatakor külön sorban, a && jelek alkalmazása esetén az utasítással egy sorba írható a magyarázó szöveg.

Lásd még: MODIFY COMMAND, PROCEDURE

PLAY MACRO

PLAY MACRO < makronév >

Egy korábban összeállított, az aktuális (a memóriában lévő) makro-könyvtárban *makronévv*vel azonosított billentyű-leütés-sorozatot hajt végre. Programban a végrehajtás nem akkor történik, amikor a PLAY MACRO utasításra kerül vezérlés, hanem csak akkor, amikor az első olyan utasítás jön sorra, amely billentyűzetről fogad el bemenő adatot (Pl: APPEND, EDIT, READ, stb). Ha eddig az utasításig nemcsak egy, hanem több PLAY MACRO utasítás is előfordult a programban, akkor azok "lejátszása" fordított sorrendben történik (először az utolsó, majd az utolsó előtti, stb).

A billentyű-makro létrehozása és tárolása a Control Center segítségével történhet.

Leggyakoribb felhasználása egy-egy billentyű hatásának megváltoztatása.

Lásd még: RESTORE MACROS, SAVE MACROS

PROTECT

PROTECT

Menürendszerű felhasználói csatoló lép működésbe, amellyel beállíthatóak (felhasználókhöz, jelszókhoz köthető) az adata hozzáférési, -módosítási jogok, sőt a dBASE használata is korlátozható. Minden üzemmódban működik.

Lásd még: LOGOUT, SET ENCRYPTION

RESET

RESET [IN < másodnév >]

Az aktuális, vagy a *másodnévvel* jelölt adatfájlban visszaállítja a fájl sértetlenségét jelző **mutatót**, amelyet a **BEGIN TRANSACTION** utasítás állított át. Enélkül az adatfájlhoz nem lehet hozzáférni. A sértetlenség mutatója automatikusan az **END TRANSACTION** illetve a sikeres **ROLLBACK** utasítás hatására áll helyre.

Az utasítás használata akkor indokolt, ha a **ROLLBACK** nem sikeres, de az adatfájl adataira még részlegesen "sérült" állapotban is szükség van.

Programban nem használandó.

Lásd még: **BEGIN/END TRANSACTION**, **ISMARKED()**, **ROLLBACK**, **ROLLBACK()**

RESTORE MACROS FROM

RESTORE MACROS FROM < fájlnev >

Betölti a memóriába, és ezzel aktuálissá (használhatóvá) teszi a *fájlnev* makro fájl (kiterjesztése: *.key*).

Lásd még: **PLAY MACRO**, **SAVE MACRO**

RUN

RUN/! < DOS utasítás/program >

Az utasítás végrehajt egy *DOS utasítást*, vagy elindít egy *.com*, *.exe*, *.bat* kiterjesztésű *programot*. *DOS utasítás* esetén a **DOS command.com** fájl-nak az aktuális alkönyvtárban kell lennie, vagy a **SET COMSPEC DOS** utasítással meg kell határozni az elérési útját.

A *DOS utasítás/program* végrehajtása után a vezérlés visszatér a **RUN-t** követő utasításra.

Nagyobb programok végrehajtása gyakran a kevés szabad memória miatt megghi-úsul, amelyet az **Insufficient memory** hibaüzenet jelez.

SAVE MACROS

SAVE MACROS < macro-fájl >

Elementi az aktív makrókat (makro-könyvtárat) a memóriából egy *.key* kiterjesztésű fájlba.

Lásd még: **PLAY MACRO**, **RESTORE MACRO**

UNLOCK

UNLOCK [ALL/IN < másodnév >]

Felszabadítja *ALL* esetén az összes, *IN* esetén pedig a *másodnévvel* kijelölt munkaterületen zárolt fájlt illetve rekordot, és ezzel lehetővé teszi, hogy azokhoz más felhasználó is hozzáférhessen.

Lásd még: FLOCK(), RLOCK(), LOCK(), SET RELATION

D) Set utasítások

A SET utasításoknál az **on/OFF** (bekapcsolva/KIKAPCSOLVA) párban a nagybetűvel szedett az alapértelmezés. Ha az külön nem kerül említésre, a SET <kulcsszó> TO <....> típusú utasításokban SET <kulcsszó> TO argumentum nélkül az alapértelmezés visszaállítását jelenti.

SET

SET

A SET utasítások beállítási értékeinek megjelenítése, módosítása.

Az utasítás hatására menürendszerrel működő interaktív, párbeszédés felhasználói csatoló lép működésbe.

A menürendszer az alábbi elemekből épül fel:

OPTIONS	megjeleníti és módosításra kínálja számos Set utasítás beállított (induláskor az alapértelmezési) értékét
DISPLAY	megjeleníti és módosításra kínálja a képernyő szín, státusz-sor, stb jellemzőit.
KEY	megjeleníti és módosításra kínálja a funkció billentyűkhöz és -kombinációkhoz rendelt parancsokat.
DISC almenü	lehetővé teszi a lemezegység alapértelmezés szerinti azonosítójának és az elérési útnak a megváltoztatását.
FILES almenü	lehetővé teszi alternate, format, device vagy indexfájlok kijelölését.

Lásd még: Control Center c. fejezet

SET ALTERNATE

SET ALTERNATE on/OFF

SET ALTERNATE TO [<fájlnev> [ADDITIVE]]

A teljes képernyőn dolgozó beviteli és megjelenítési utasításokon – úgy mint @...SAY, EDIT, BROWSE, APPEND, stb – minden a képernyőn megjelenő és begépett adatot (a megjelenési formátumban) – a megőrzéshez hasonló módon – a *fájlnev* ASCII típusú szövegfájlba is rögzíti. (Ez későbbi hibakeresésre, stb használható). A szövegfájl bármilyen szövegszerkesztővel megjeleníthető és ki is nyomtatható.

SET ALTERNATE TO létrehozza és megnyitja a *fájlnev* szövegfájlt. Ha ilyen névvel már létezett fájl, azt felülírja, hacsak az ADDITIVE opció nem szerepel. Ilyenkor az új adatokat a létező fájl végéhez fűzi.

SET ALTERNATE ON bekapcsolja a megnyitott fájlba történő rögzítést (a felvételt).

SET ALTERNATE OFF felfüggeszti a fájlba írást anélkül, hogy lezárná a fájlt.

A fájlnev nélkül kiadott SET ALTERNATE TO lezárja a fájlt.

Lásd még: CLOSE

SET AUTOSAVE

SET AUTOSAVE on/OFF

Módosításnál vagy felvitelnél azonnal elmenti a rekordot a lemezre. Így csökken az adatok elvesztésének lehetősége és minden változás azonnal rögzítésre kerül.

SET AUTOSAVE OFF esetén csak akkor írja a lemezre a módosításokat, ha a memóriapuffer megtelt. Config.db fájlban is beállítható.

SET BELL

SET BELL ON/off

A hibákra és egyéb eseményekre figyelmeztető hangjelzés OFF = ki- és ON = bekapcsolása. Ez nem vonatkozik a CHR(7) karakter használatára.

SET BELL TO [<frekvencia> , <időtartam>]

A figyelmeztető hangjelzés jellemzőinek beállítása. Ez az utasítás viszont a CHR(7) karakterre is vonatkozik. A hangmagasságot a *frekvencia*, az időtartamot az *időtartam* határozza meg. A hangmagasság mértékegysége a Herz, amely 19 és 10000 közé eshet. Az időtartamé 0.0549 másodperc (dBASE terminológia:tick), amely 2 és 19 közé eshet.

Alapértelmezés: 512,2. Ezt eredményezi a SET BELL TO paraméterek nélkül.

SET BLOCKSIZE

SET BLOCKSIZE TO <N-kifejezés>

Memo mezők és összetett indexfájlok blokkméretének beállítása. Mértékegység: 512 bájt. Lehetséges értékek: 1-32. A túl nagy blokkméret kijelölés lassítja az írás/olvasás műveleteket.

Lásd még: COPY, CREATE/MODIFY STRUCTURE

SET BORDER

SET BORDER TO [SINGLE/DOUBLE/PANEL/NONE/<keret sztring >]

A menük, ablakok és a @ ... utasítások keretjellemezőinek alapértelmezését állítja be.

A *SINGLE/DOUBLE/PANEL/<keretsztring>* opciókat lásd a @ ...TO utasításnál.

A *NONE* opció esetén nincs keretrajzolás.

Lásd még: @...SAY, @...TO

SET CARRY

SET CARRY on/OFF

ON állásban az APPEND és az INSERT utasításoknál az új rekord mezőit az előző rekordbeli értékekkel tölti fel.

SET CARRY TO [<mezőlista > [ADDITIVE]]

Az APPEND, BROWSE és az INSERT utasításoknál csak a *mezőlistában* megadott mezők előző rekordbeli tartalmát viszi át az új rekordba.

Az ADDITIVE opció esetén az új mezőlista hozzáadódik az előzőleg kiadott SET CARRY utasításban (ha volt ilyen) felsorolt mezőkhöz.

A mezőlista nélkül kiadott utasítás minden mező áttöltését eredményezi, hacsak ezután a SET CARRY OFF utasítás nem kerül kiadásra.

Lásd még: APPEND, BROWSE, EDIT, INSERT, READ, SET FIELDS, SET FORMAT

SET CATALOG

SET CATALOG on/OFF

ON állásban a korábban SET CATALOG TO utasítással kijelölt katalógusban a (használatbavett, létrehozott, módosított) fájlokról a nyilvántartás vezetését ismét elindítja. Az OFF opció a nyilvántartás vezetését, karbantartását felfüggeszti.

SET CATALOG TO [<fájlnev >/?]

A 10-es munkaterületen megnyitja vagy (ha még nincs) létrehozza és megnyitja a *fájlnev* katalógust. A ? megjeleníti a létező katalógusokat, felkínálva a választási lehetőséget.

A katalógus megnyitásától kezdve minden újonan létrehozott vagy használatbavett fájl a katalógus nyilvántartásába kerül. Ez az utasítás automatikusan ON-ra

állítja a SET CATALOG paramétert is. A *fájl*névhez a *.cat* kiterjesztést a rendszer automatikusan hozzáteszi. A katalógus lezárása és ezzel a 10-es munkaterület felszabadítása a fájlnev nélküli SET CATALOG TO utasítással történik.

A katalógus *.cat* kiterjesztése ellenére adatfájl szerkezetű az alábbiak szerint:

Mező	Mezőnév	Típus	Hossz	Magyarázat
1	Path	Character	70	elérési út
2	File_name	Character	12	fájlnev
3	Alias	Character	8	másodnev
4	Type	Character	3	fajltípus (kiterjesztés)
5	Title	Character	80	magyarázat
6	Code	Numeric	3	generált azonosító
7	Tag	Character	4	nem használt
**	Összesen	**	181	

A katalógus létrehozásakor a Tag és a Title mezők kivételével a mezők automatikusan kapnak értéket.

Ha SET CATALOG ON állásban van a katalógusban mindig új bejegyzéssel (rekorddal) bővül a következő utasítások végrehajtásakor:

COPYSTRUCTURE	COPYSTRUCTUREEXTENDED
CREATE	CREATE FROM
CREATE/MODIFYLABEL	CREATE/MODIFYQUERY
CREATE/MODIFYREPORT	CREATE/MODIFYSCREEN
CREATE/MODIFYVIEW	IMPORT FROM
INDEX	JOIN
SET FILTER TO FILE	SET FORMAT
SET VIEW	SORT
TOTAL	USE

Lásd még: SET TITLE

SET CENTURY

SET CENTURY on/OFF

ON állásban a dátumokban az évezred, évszázad is szerepel.

Lásd még: CTOD(), DATE(), DTOC(), DTOS(), SET DATE, YEAR()

SET CLOCK**SET CLOCK on/OFF**

Ha *ON*, a rendszeridő látható a képernyőn.

SETCLOCKTO [< sor > , < oszlop >]

A rendszeridő alapértelmezés szerint a 0, 69 koordinátáknál jelenik meg. A *sor*, *oszlop* meghatározásával ez megváltoztatható.

SET COLOR**SET COLOR ON/OFF**

Egyszínű és színes monitorok közti váltásra használható. Az alapértelmezés a *dbase IV* indításakor áll be.

SET COLOR TO [[< normál színpár >] [, [< kiemelt színpár >]]]

A színek és egyéb jellemzők (aláhúzás, villogás) egész képernyőre történő beállítás.

A színek összeállítása előre megadható a *config.db* fájlban is.

A *normál színpár* a megjelenített (?, SAY) adatok színe, előtér/háttér formában.

A *kiemelt színpár* a beolvasandó (GET, INPUT, stb) adatok, a menük aktuális sorának színe előtérszín/háttérszín formában.

Színtáblázat

Szín	Jel	Szín	Jel
Fekete	N vagy szóköz	Piros	R
Kék	B	Lila	RB
Zöld	G	Barna	GR
Üres	X	Sárga	GR +
Szürke	N +	Fehér	W
Ciánkék	BG		

Villogó szín a betű előtti csillaggal, az emelt fényerő a plusz jellel érhető el.

Pl: +G/N, *RB/W, * +BG/B.

A rejtett karakterekhez az X betű használható előtér vagy háttér színeként. Pl: jelszó begépeléséhez X/B. Az X kódú színnel kijelölt karakter nem jelenik meg a képernyőn.

Egyszínű monitor esetén az U és I betű használható, U az aláhúzást jelenti, I pedig az inverz megjelenítést.

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

Külön állítható a képernyőn megjelenő különféle alakzatok színe. Az alakzatok alábbi csoportjainak színei állíthatók be egymástól függetlenül:

NORMAL	normál adatok (Pl: SAY, ? utasítások)
MESSAGES	üzenetek, menüpontok
TITLES	különböző fejlécek, azonosítók stb
BOX	keretek
HIGHLIGHT	aktuális menüsorok, stb (sávkurzor)
INFORMATION	státusz-sor, óra
FIELDS	beolvasandó adatok (GET, INPUT, stb)

Lásd még: SET, SET DISPLAY

SET CONFIRM

SET CONFIRM on/OFF

Adatok beolvasásakor az egyik mezőről a másikra való kurzormozgás automatikusan megtörténik, ha a mező betelt és a SET CONFIRM értéke OFF. ON esetén minden mező kitöltése után még egy **Enter**-t is kell ütni ahhoz, hogy a kurzor a következő mezőre ugorjon.

SET CONSOLE

SET CONSOLE ON/off

OFF állásban az adatfolyam (?/??, stb utasítások) nem jelenik meg a képernyőn. Általában a nyomtatás esetén alkalmazható a képernyő tartalmának megőrzésére. (A nyomtatott szöveg automatikusan megjelenik a képernyőn is alapértelmezésben).

Hatástalan az utasítás a hibaüzenetekre és a @ ... SAY ... GET -re.

SET CURRENCY

SET CURRENCY TO [<C-kifejezés>]

Saját pénznem szimbólum határozható meg a *C-kifejezéssel*, amely szükség esetén a számok előtt/mögött jeleníthető meg (*PICTURE*). Pénzügyi jellegű feladatoknál lehet hasznos a pénznem rövidítésének megjelenítése a számok mellett. A szimbólum legfeljebb 9 karakter hosszú lehet.

Példa:

SET CURRENCY TO "Forint"

SET CURRENCY LEFT/RIGHT

SET CURRENCY LEFT/right

A pénznem szimbólum megjelenési helyét határozza meg. Az elhelyezés *LEFT* esetén a számtól balra, *RIGHT* esetén pedig jobbra történik.

SET DATE

SET DATE [TO]

AMERICAN/ansi/british/french/german/italian
/japan/usa/mdy/dmy/ymd

Meghatározza a dátum típusú adatok megjelenési formáját az alábbi módon:

AMERICAN	mm/dd/yy
ANSI	yy.mm.dd
BRITISH/FRENCH	dd/mm/yy
GERMAN	dd.mm.yy
ITALIAN	dd-mm-yy
JAPAN	yy/mm/dd
USA	mm-dd-yy
MDY	mm/dd/yy
DMY	dd/mm/yy
YMD	yy/mm/dd

ahol "yy" az évtizedet és az évet, "mm" a hónapot, "dd" a napot jelöli.
Lásd még: DATE(), DMY(), MDY(), SET CENTURY

SET DEBUG

SET DEBUG on/OFF

Beállítja, hogy a SET ECHO által meghatározott kimenő adatok a képernyőre (*OFF*) vagy a nyomtatóra (*ON*) kerüljenek.

Lásd még: DEBUG, SET ECHO, SET TALK

SET DECIMALS

SET DECIMALS TO <N-kifejezés>

A matematikai műveleteknél a megjelenő tizedesjegyek számát határozza meg. A *N-kifejezés* értéke 0-18 közé eshet.

SET DEFAULT

SET DEFAULT TO <meghajtó> [:]

Ha az adatok nem az aktuális meghajtóban lévő lemezen vannak, megadható egy másik meghajtó azonosító. A rendszer ezentúl ott keresi az adatokat.

Lásd még: SET PATH

SET DELETED

SET DELETED on/OFF

Meghatározza, hogy a törlésre kijelölt rekordok más utasítások által hozzáférhetőek legyenek (*OFF*) vagy ne (*ON*).

Az INDEX, REINDEX utasítások minden esetben az összes rekordra érvényesek függetlenül a SET DELETED állásától.

Lásd még: DELETED(), SET FILTER

SET DELIMITERS

SET DELIMITERS on/OFF

ON állásban megjelöli a bemenő adatok képernyőmezőinek két végét egy megadott karakterpárral.

SET DELIMITERS TO <C-kifejezés>/DEFAULT

A *C-kifejezés* meghatározza a bemenő adatok képernyőmezőit határoló karaktereket. Legfeljebb 2 karakter lehet, ahol az első a kezdő – ha nincs második, egyben befejező is –, a második a befejező. Ezek csak a SET DELIMITERS ON esetén jelennek meg.

A DEFAULT opció visszaállítja az alapértelmezést, amely a":"

Lásd még: @, APPEND, CHANGE, EDIT, INSERT, READ, SET INTENSITY

SET DESIGN

SET DESIGN ON/off

OFF állásban letiltja a Shift-F2-vel kiváltható (adatfájl, nézetfájl, stb) tervezési funkciókat. Megakadályozható az illetéktelen és várhatóan károkat okozó hozzáférés a szerkezetekhez.

SET DEVELOPMENT

SET DEVELOPMENT ON/off

ON állásban futtatás előtt a rendszer megvizsgálja, hogy a lefordított programhoz (.dbo) (és egyéb fordított fájlhoz) képest a forrásfájl (.prg, stb) megváltozott-e. Ha igen, újrafordítja (COMPILE).

Lásd még: COMPILE, MODIFY COMMAND

SET DEVICE

SET DEVICE TO SCREEN/printer/file <fájlnev >

Meghatározza, hogy a @ ... SAY utasítások eredménye a képernyőre (*SCREEN*), nyomtatóra (*PRINTER*), vagy a *fájlnev* fájlba kerüljön (*FILE*).

Lásd még: @, EJECT, SET FORMAT

SET DISPLAY

SET DISPLAY TO MONO/COLOR/EGA25/EGA43/MONO43

Átkapcsolást tesz a megjelenítési módok között, ha a számítógépben lévő megjelenítést vezérlő hardverelem (kártya) több módon képes működni. Az opciók sorra: MONO: egyszínű / COLOR: színes / EGA25: 25 soros EGA (Enhanced Graphic Adapter) / EGA43: 43 soros EGA / MONO43: 43 soros egyszínű.

Lásd még: SET, SET COLOR OF, SET COLOR TO

SET ECHO

SET ECHO on/OFF

ON esetén a program utasításai futás közben megjelennek a képernyőn illetve SET DEBUG ON esetén a nyomtatón.

Lásd még: DEBUG, SET DEBUG, SET PRINT, SET STEP, SET TALK

SET ENCRYPTION

SET ENCRYPTION on/OFF

ON esetén az újonnan keletkező fájlok akár CREATE akár JOIN, COPY vagy TOTAL hozta létre azokat, titkosított formában jönnek létre. Titkosított fájlhoz egy legális felhasználónév, jó jelszó és csoportnév segítségével lehet csak hozzáférni, amelyeket a bejelentkező képernyőnél (log-in) kell megadni.

Lásd még: PROTECT

SET ESCAPE

SET ESCAPE ON/off

ON esetén az Esc billentyű lenyomására megszakad az aktuális felhasználói program futása. OFF esetén a futó programot csak rendszerhívással lehet megszakítani (Ctrl-Alt-Del, reset gomb), de a képernyőn futó adatok (LIST, stb) sem függeszthetők fel a Ctrl-S vagy BACKSPACE billentyűkkel.

Lásd még: INKEY(), ON KEY, ON ERROR, READKEY()

SET EXACT**SET EXACT on/OFF**

ON esetén csak akkor tekint két összehasonlított sztringet egyenlőnek, ha azok hossza is egyenlő. *OFF* esetén ha az egyik sztring karakterei megegyeznek a másik sztring első karaktereivel, az egyenlőség fennáll.

SET EXCLUSIVE**SET EXCLUSIVE on/OFF**

Az *ON* állásban lehetővé teszi azt, hogy többfelhasználós környezetben egy felhasználó az összes általa használt adatfájl automatikusan kizárólagos hozzáférési jogokkal nyissa meg.

Lásd még: COPY INDEXES/TAG, INDEX ON, PROTECT, SET ENCRYPTION

SET FIELDS**SET FIELDS on/OFF**

OFF esetén az aktuális adatfájl minden mezője hozzáférhető függetlenül a SET FIELDS TO utasítás korlátozásaitól.

SET FIELDS TO [<mező> [/R]/<számított mező> ...]
[,<mező> [/R]/<számított mező> ...]

SET FIELDS TO ALL, [LIKE/EXCEPT <minta>]

Az aktuális adatfájl megadott *mezői* érthetők el az ezt követő utasításokban. A *számított mező* más mezőkből képzett tetszőleges kifejezés lehet. A */R* megtiltja az aktuális mező felülírását.

Az utasításnak *SET FIELDS TO ALL* formája több hasonló nevű mező kijelölésére használható. A *mintában* szabadon használhatóak a "*" és a "?" karakterek, hasonlóan az DOS fájlneveinél megismertekhez.

Az *ALL LIKE* azokat a mezőket jelöli ki, amelyek illenek a megadott *mintához*, az *ALL EXCEPT* pedig azokat, amelyek nem.

A SET FIELDS TO által kiválasztott mezők érvényesek az alábbi utasításokra:

APPEND	AVERAGE
BROWSE	CALCULATE
CHANGE	CREATE/MODIFY
VIEWFROMENVIRONMENT	COPY TO
COPYSTRUCTURE	DISPLAY

EDIT
JOIN
SUM

EXPORT
LIST
TOTAL

Lásd még: CLEAR FIELDS, CREATE/MODIFY VIEW, CREATE VIEW FROM ENVIRONMENT, SET RELATION, SET SKIP, SET VIEW

SET FILTER

SET FILTER TO [FILE <fájlnev>/?] / [<feltétel>]

Egy adatfájlban csak azokat a rekordokat teszi hozzáférhetővé, amelyek kielégítik a megadott (szűrő)feltételt.

A *feltétel* helyett a *FILE* kulcsszóval megadható a *fájlnev* szűrőfeltételeket tartalmazó fájl is.

Minden nyitott adatfájlhoz meghatározható szűrőfeltétel, ezek az adatfájl lezárásáig, vagy az adatfájl munkaterületén kiadott praméter nélküli SET FILTER TO utasításig érvényesek.

A szűrő csak akkor lesz aktív, ha kiadása után a rekordmutató elmozdul, amelyet legegyszerűbben a GO TOP utasítással lehet elérni.

Lásd még: CREATE/MODIFY QUERY, SET DELETED

SET FORMAT

SET FORMAT TO [<formátum fájlnev>/?]

Lehetővé teszi, hogy a READ, EDIT, APPEND, INSERT, CHANGE utasításoknál az adatok beolvasása egy előre elkészített formátumfájl segítségével történjék. Az esetlegesen csak forrásnyelven létező *.fmt* fájlokat *.fmo* fájlokra lefordítja.

(A formátumfájlok a CREATE/MODIFY SCREEN utasításokkal hozhatók létre.)

Minden munkaterületen egyszerre csak egy formátumfájl lehet nyitva.

A ? hatására megjelennek az *.fmt* kiterjesztésű fájlok nevei, amelyből a megfelelőt a sávkurzorral és az Enter lenyomásával lehet kiválasztani.

Lásd még: @, APPEND, CHANGE, CLOSE, EDIT, INSERT, READ, SET DEVICE

SET FULLPATH

SET FULLPATH on/OFF

A fájlneveket visszaadó függvények alapértelmezésben csak a meghajtó és a fájlnevét szolgáltatják. *ON* esetén a teljes fájlnevet szolgáltat a rendszer, beleértve az elérési utat is.

SET FUNCTION

SET FUNCTION < N-kifejezés > / < C-kifejezés1 >
/ < billentyű felirat > TO < C-kifejezés2 >

Segítségével a klaviatúrán található funkciós (F1-F10) billentyűkhöz tetszőleges dBASE IV utasítássorozatot rendelhetők.

A *C-kifejezés2* tartalmazza az utasítássorozatot, amely legfeljebb 238 karakter hosszú lehet, és az egyes utasításokat benne a ;-vel kell lezárni.

Programozható az F2 - F10, a Shift-F1 - Shift-F9, a Ctrl-F1 - Ctrl-F10 billentyű. A billentyűk azonosíthatók az *N-kifejezéssel*, amely 2 - 10-ig terjedő szám lehet; a *C-kifejezéssel*, amely karakteres változó is lehet és a kombinált billentyűk feliratát tartalmazhatja a kötőjellel elválasztva (pl: "Shift-F6"); de megadható ugyanez idézőjelek nélkül a *billentyű felirataként*.

A funkciós billentyűk alapértelmezése:

Billentyű	Utasítás	Billentyű	Utasítás
F1	HELP;	F6	DISPLAYSTATUS;
F2	ASSIST;	F7	DISPLAYMEMORY;
F3	LIST;	F8	DISPLAY;
F4	DIR;	F9	APPEND;
F5	DISPLAY STRUCTURE;	F10	EDIT;

Az F1, a Shift-F10 és az összes Alt kombináció nem programozható. A SET FUNCTION utasítást felülbírálja az ON KEY utasítás.

Lásd még: FKLABEL ()

SET HEADING

SET HEADING ON/off

OFF esetén a DISPLAY, LIST, SUM, AVERAGE utasítások listáinál a mezőnevek (fejléc) és a rekordsorszámok nem jelennek meg, illetve TYPE utasítás esetén az oldalak automatikus oldalsorszámozása elmarad.

SET HELP

SET HELP ON/off

OFF esetén a hibás utasítás begépelése után nem jelenik meg automatikusan a help ablak.

Lásd még: HELP, SET INSTRUCT

SET HISTORY

SET HISTORY ON/off

ON esetén a kiadott utasításokat pufferben tárolja a rendszer és ezeket szükség esetén a fel- és lefele mutató nyilakkal lehet újból megjeleníteni a képernyőn és Enter leütésével megismételni. Csak az interpreterben kiadott utasításokat őrzi.

SET HISTORY TO <N-kifejezés>

Az utasítástároló puffer méretét határozza meg a *N-kifejezés*, amely 0 – 16000-ig terjedhet. Alapértelmezés: 20 (utasítás).

Lásd még: DISPLAY HISTORY, LIST HISTORY

SET HOURS

SET HOURS TO [12/24]

A rendszeridő 12 illetve 24 órás megjelenítése váltható. Alapértelmezés: 12

Lásd még: SET CLOCK

SET INDEX

```
SET INDEX TO [?/<fájlnévlista >
  [ORDER [TAG] <ndx-fájlnev >/<indextagnév >
  [OF <mdx-fájlnev > ]]]
```

Az aktuális munkaterületen (az aktuális adatfájlhoz) megnyitja a *fájlnévlistá-*ban megadott indexfájlokat akár egyszerű akár összetett fájlokról van szó. Az *ORDER* cikkelyben kijelölhető az aktuális index az *ndx-fájlnev* vagy az *indextagnév* megadásával. Ez utóbbinál – ha az szükséges – az *OF mdx-fájlnevvel* az az összetett indexfájl is megadható, amelyhez az indextag tartozik. rendezettséget i szempontnak megfelelő fájl a felsoroltak közül. Az aktuális index a rekordok elérési sorrendjét határozza meg. Az aktuális index a SET ORDER TO utasítással cserélhető. A rendszer a megnyitott indexfájlok mind-egyikét karbantartja az adatfájl módosulásakor.

A SET INDEX TO paraméterek nélkül az adott munkaterületen lezárja az összes indexfájlt az mdx-fájlok kivételével.

Egy munkaterületen legfeljebb 10 egyszerű indexfájl és 1 db összetett indexfájl lehet nyitva. Egy összetett indexfájl maximálisan 47 db indextagot tartalmazhat.

Lásd még: CLOSE, DELETE TAG, DISPLAY INDEXES, INDEX, KEY(), MDX(), NDX(), ORDER(), REINDEX, SET ORDER, TAG(), USE

SET INSTRUCT

SET INSTRUCT ON/off

OFF esetén (elvileg!) nem jelennek meg a menürendszerrel működő teljesképernyős utasítások (APPEND, BROWSE, EDIT, stb) promptjai.

SET INTENSITY

SET INTENSITY ON/off

OFF esetén a GET utasítások is normál színnel jelennek meg a képernyőn. (A GET adatok színeit alapértelmezésben az érvényes kiemelt színpár határozza meg.)

Lásd még: SET, SET COLOR, SET DISPLAY

SET LOCK

SET LOCK ON/off

OFF esetén kikapcsolja az automatikus rekord- vagy fájlzárolást. Csak többfelhasználós környezetben hatásos.

SET MARGIN

SET MARGIN TO <N-kifejezés>

Az *N-kifejezés* meghatározza a nyomtatásnál a bal **alpmargót**. Az alapértelmezés 0.

A képernyőn történő megjelenítésre nincs hatással. Hatása azonos az *_ploffset* rendszerváltozóéval.

Lásd még: Rendszerváltozók, *_lmargin*, *_ploffset*, *_wrap*

SET MARK

SET MARK TO <C-kifejezés>

A *C-kifejezés* a dátum típusú adatok elválasztó karaktereit határozza meg.

Lásd még: *DATE()*, *DMY()*, *MDY()*, *SET CENTURY*, *SET DATE*

SET MEMOWIDTH

SET MEMOWIDTH TO <N-kifejezés>

A *N-kifejezés* a memo mezők megjelenítési szélességét határozza meg. Az alapértelmezés 50, lehetséges értéktartomány 8 – 32000.

Lásd még: *MEMLINES()*, *MLINE()*, *@V PICTURE FUNCTION*, *SET()*

SET MESSAGE

SET MESSAGE TO [<C-kifejezés>

A státusz-sor alatti sor (23) közepére igazítva megjeleníti a *C-kifejezést*. *SET STATUS OFF* esetén is működik.

A *C-kifejezés* legfeljebb 79 karakter hosszú lehet.

Lásd még: *SET STATUS*

SET NEAR**SET NEAR on/OFF**

ON esetén a kereső utasítások (SEEK, FIND, LOCATE) sikertelen keresés esetén nem a fájl végére, hanem a fájl rendezettsége szerinti következő, a keresési kulcshoz legközelebb álló rekordra állnak illetve oda állítják a rekordmutatót.

Lásd még: EOF(), FIND, FOUND(), LOCATE, SEEK, SEEK()

SET ODOMETER**SET ODOMETER TO <N-kifejezés>**

Az *N-kifejezés* az – általában rekordsorozatokat feldolgozó (REPLACE ALL, DELETE, PACK, stb) – utasításoknál a feldolgozás közbeni rekordszám-megjelenítés lépésközét határozza meg, ha a SET TALK ON állásban van. Így például 100 esetén csak 100 rekordonként jelzi ki a rekordsorszámot. Tapasztalatunk szerint nem működik.

Az alapértelmezés 1, értéktartomány 1 – 200.

SET ORDER

SET ORDER TO [TAG]

<ndx-fájlnev>/<.mdx indextagnév> [OF <mdx-fájlnev>]

Kiválasztja a megnyitott indexek közül az aktuálisat, másnéven főindexet. Az utasítás hatására adatfájl rekordjai az aktuális index által meghatározott sorrendben érhetők el.

Az aktuális index az *ndx-fájlnev*vel vagy az *indextagnév*vel jelölt index lesz. Ez utóbbinál – ha az az egyértelműséghez szükséges – az *OF mdx-fájlnev*vel az az összetett indexfájl is megadható, amelyhez az indextag tartozik.

SET ORDER TO <N-kifejezés>

Kiválasztja a megnyitott indexek közül az aktuális indexet.

Csak akkor használható, ha csak egyszerű indexfájlok kerültek megnyitásra. Az *N-kifejezés* a SET INDEX TO vagy a USE utasításokban felsorolt egyszerű indexfájlok listában elfoglalt helyére utal. Értéke 0 és 10 között lehet összefüggésben azzal, hogy egy adatfájlhoz egy időben legfeljebb 10 egyszerű indexfájl nyitható meg. Az utasításnak ez a formája elsősorban a dBASE III PLUS kompatibilitását biztosítja.

SET ORDER TO

Lezárja a megnyitott indexfájlokat, és visszatér a természetes, azaz rekordsorszám szerinti rendezettséghez.

Lásd még: DISPLAY, INDEX, KEY(), MDX(), NDX(), ORDER(), REINDEX, SET INDEX, TAG(), USE

SET PATH

SET PATH TO [<elérési utak listája>]

Meghatározza azokat az elérési utakat, ahol a rendszer az aktuális könyvtárban nem létező fájlokat keresse. Az utasítás független, de hasonló a DOS PATH utasításához.

Az elérési utakat vesszővel kell elválasztani. A lista legfeljebb 60 karakter hosszú lehet.

A SET PATH TO visszaállítja az eredeti aktuális könyvtárat

Lásd még: DIR, SET DEFAULT

SET PAUSE**SET PAUSE on/OFF**

Az SQL SELECT utasításának működési módját állítja be.

ON esetén úgy működik, mint a DISPLAY, azaz a képernyő megtelése után egy billetyű leütését várja.

OFF esetén a működés analóg a dBASE IV LIST utasításáéval.

SET POINT**SET POINT TO [<C-kifejezés>]**

A *C-kifejezéssel* a tizedespontként használt karakter jelölhető ki. Bármely karakter felhasználható a számjegyek és a szóköz kivételével. Az alapértelmezés a pont ".".

SETPRECISION**SET PRECISION TO [<N-kifejezés>]**

A *N-kifejezés* meghatározza, hogy a matematikai számítások a memóriában hány számjegy pontossággal történjenek N típusú számok esetén.

Alapértelmezés: 16. Értéktartomány: 10 – 20.

Lásd még: SET DECIMALS

SET PRINTER

Az utasítás első 5 formája arra szolgál, hogy a nyomtatandó outputokat átirányíthassuk a nyomtatóról egyéb eszközökre, míg a 6. forma a nyomtatandó szövegek textfájlba írását teszi lehetővé.

SET PRINTER on/OFF

ON esetén minden nem formázott (nem @...SAY utasításokkal megtervezett) kimenő adat a helyi (lokális) nyomtatón is megjelenik.

SET PRINTER TO <DOS eszköznév >

A kimenő adatok helyi (lokális) *DOS eszköznév* jelölt egységre (Device) irányíthatóak. A *DOS eszköznév* lehetséges értékei: PRN,LPT1,LPT2,LPT3, COM1, COM2

Az alapértelmezés PRN.

```
SET PRINTER TO \\számítógépnév > \ <printernév > = <céleszköz név >
```

```
SET PRINTER TO \\SPOOLER
```

```
SET PRINTER TO \\CAPTURE
```

A kimenő adatok megjelenítése a hálózatban lévő fájlkiszolgáló egységek (nem lokális) eszközein.

A *számítógépnév* a SERVER számítógép hálózatbeli neve. A *printernév* a nyomtató hálózatbeli neve. A *céleszköz neve* a közös nyomtató eszköz neve (pl.: LPT1,LPT2, LPT3).

A *SPOOLER* és *CAPTURE* opciók a Novell hálózat alatt használhatók, és ott a nyomtatásra várakozó sorba (queue) irányítják a kimenetet.

SET PRINTER TO FILE <fájlnév >

A formázott (@ ...SAY) kimenő adatok is a *fájlnév* által megjelölt szabványos ASCII szövegfájlba kerülnek nyomtatási formátumban (nyomtatóvezérlő kódokkal együtt). A DOS PRINT utasításával kinyomtathatók.

Lásd még: ?/??, ???, LABEL FORM, LIST/DISPLAY, REPORT FORM, SET DEBUG, SET DEVICE

SET PROCEDURE

SET PROCEDURE TO [<fájlnév>]

Megnyitja a *fájlnév* eljárásfájlt. A *fájlnév*nek tartalmaznia kell a ki-terjesztést, ha az nem *.db*, és az elérési utat is, ha nem az aktuális alkönyvtárban van. Egy eljárás fájl legfeljebb 1170 eljárást (procedure, function) tartalmazhat

SET PROCEDURE TO fájlnev nélküli lezárja az aktuálisan nyitott eljárásfájlt.

Lásd még: COMPILE, DO, PARAMETERS

SET REFRESH

SET REFRESH TO <N-kifejezés>

Az *N-kifejezés* meghatározza azt az időszakot másodpercben, amely szakaszonként a BROWSE vagy egyéb adtfajlmódosító utasítások által megjelenített adatokban felfrissíti a hálózat más pontjain történt változásokat.

Az alapértelmezés: 0. Értéktartomány: 1 – 3600.

Az utasítás csak a CONVERT utasítással hálózati üzemre adaptált adatfájloknál hatásos.

Lásd még: CONVERT

SET RELATION

```
SET RELATION TO <kulcskifejezés1>/<N-kifejezés1>
                INTO <másodnév1>
                [[,<kulcskifejezés2>/<N-kifejezés2>
                INTO <másodnév2>] ...]
```

Kapcsolatot teremt az aktuális – tulajdonos – adatfájl és a *másodnév*vel azonosított másik adatfájl – továbbiakban **kapcsolt fájl** – között egy olyan **kulcs**on keresztül, amely mindkét fájlban szerepel.

A *kulcskifejezés* olyan kifejezés, amely mindkét adatfájiban értelmezhető, tehát az aktuális és a kapcsolt fájl valamely mezőire épül. A kapcsolat léterjöt-téhez a **kapcsolt fájl**t a *kulcskifejezés* szerint indexelni kell, és ennek kell lenni az **aktuális index**nek is (ORDER).

Ha a kapcsolt fájl **nem indexelt**, azaz naturális (rekordsorszám szerinti) rendezettségű, akkor a *N-kifejezés* a kapcsolt fájl rekordsorszámára utal, és ezen keresztül létesít kapcsolatot.

Egy adatfájlhoz több kapcsolt fájl is köthető.

Minden mozgás az aktuális fájlban mozgást okoz a kapcsolt fájlokban is, mégpedig a *kulcskifejezés-/N-kifejezés*nek megfelelő első rekordra áll a kapcsolt fájlok rekordmutatója. Ha ilyen nincs, akkor a rekordmutató a fájl végére mutat még akkor is, ha a SET NEAR ON állásban van.

SET RELATION TO

Megszünteti az aktuális adatfájl és a kapcsolt fájlok kapcsolatait.

Lásd még: CREATE/MODIFY VIEW, SET FIELDS, SET SKIP, SET VIEW

SETREPROCESS

SET REPROCESS TO <N-kifejezés>

Az *N-kifejezés* többfelhasználós üzemmódban meghatározza a fájl- vagy rekordzárolási (-foglalási) kísérletek maximális számát. Hibajelzést csak ennyi sikertelen kísérlet után ad a rendszer.

A fájl- vagy rekordfoglalás akkor sikertelen, ha a hálózatban egy másik felhasználó azt már lefoglalta és foglalva tartja.

Alapértelmezés: 0. Értéktartomány: -1 – 32000.

0 esetén az Esc billentyű lenyomásáig próbálkozik, és minden sikertelen kísérletnél hibát jelez.

0-nál nagyobb érték a megadott számú ismétlés után a rendszer abbahagyja a próbálkozást, és ekkor küld hibaüzenetet.

-1 esetén végtelenszer próbálkozik, hibaüzenetet nem küld, és a folyamat nem szakítható meg az Esc leütésével.

A mínusz 1 értéket csak programból célszerű használni, és mellette feltételt kell szabni a végtelen számú próbálkozás elkerülésére.

Lásd még: FLOCK(), NETWORK(), RLOCK()/LOCK, UNLOCK

SET SAFETY

SET SAFETY ON/off

Biztonsági védelem a véletlen adattörlések, felülírások ellen. ON esetén az ilyen műveletek végrehajtása előtt a rendszer külön jóváhagyást kér.

Lásd még: COPY, COPY FILE, INDEX, JOIN, SAVE, SORT, TOTAL, UPDATE, ZAP

SET SCOREBOARD**SET SCOREBOARD ON/off**

ON esetén a képernyőn láthatóak az alábbi információk:

Del	ha az aktuális rekord törlésre kijelölt
Ins	ha az insert (beszúró) üzemmód érvényes
Caps	ha a Caps Lock billentyű be van nyomva
Num	ha a Num Lock billentyű be van nyomva
Exlock	ha az aktuális adatfájl kizárólagos használatra van megnyitva
Readonly	ha az aktuális adatfájl csak olvasható

Ha SET STATUS OFF állásban van, akkor a fenti információk a 0-dik sorban, egyébként pedig a státusz-sorban jelennek meg.

Lásd még: SET MESSAGE, SET STATUS

SET SEPARATOR**SET SEPARATOR TO [<C-kifejezés>]**

A *C-kifejezés* a megjelenített számokban a három helyiértékenkénti elválasztáshoz új karaktert jelöl ki. Mindössze egyetlen karakterből állhat, az alapértelmezés a vessző.

Lásd még: SET POINT

SET SKIP**SET SKIP TO [<másodnév1> [, <másodnév2>]...]**

Lehetővé teszi a hozzáférést a *másodnévvel* megjelölt kapcsolt fájlok (SET RELATION) minden olyan rekordjához, amely a kapcsoló kulcskifejezés szerint azonos anélkül, hogy az aktuális (**tulajdonos**) adatfájl rekordmutatója elmozdulna. Ez az igény akkor fordulhat elő, ha a tulajdonos adatfájl egy rekordjához a kapcsolt fájl több rekordja kapcsolódik. Ilyenkor **alapértelmezésként** a kapcsolt fájlnek **csak az első rekordja** érhető el. (Pl: egy vevőhöz tartozó rendelések közül nemcsak az első, hanem az összes megjelenítése szükséges.)

Lásd még: SET RELATION

SET SPACE

SET SPACE ON/off

ON esetén a ??? utasításokkal nyomtatott, egymástól vesszővel elválasztott ki-fejezések között automatikusan egy szóközt iktat be.

SET SQL

SET SQL on/OFF

Az *ON* bekapcsolja SQL üzemmódot. SQL módban az SQL utasítások és a dBASE IV csak bizonyos (SQL-lel konfliktusban nem lévő) utasításai használhatók. (Lásd: SQL c. fejezet)

Programban nem használható.

SET STATUS

SET STATUS ON/off

ON esetén a státusz-sor látható a képernyő alsó részén. Tartalma a következő:

- az aktuális parancs(program) neve;
- az aktuális adatfájl neve;
- az aktuális rekord sorszám/az összes rekordjok száma.

SET SCOREBOARD *ON* esetén az ott leírt információk is a státusz-sorban jelennek meg.

Lásd még: SET MESSAGE, SET SCOREBOARD

SET STEP

SET STEP on/OFF

ON állásban minden utasítás után felfüggeszti a program futását. A programok nyomonkövetésre, hibafelderítésre használható.

Lásd még: COMPILE, DEBUG, DO, SET, SET DEBUG, SET DEVELOPMENT, SET ECHO, SET TALK

SET TALK**SET TALK ON/off**

ON esetén bizonyos dBASE IV utasítások eredményét jeleníti meg a képernyőn (pl. rekord sorszám, változók értékei, APPEND, COPY... stb. utasítások eredménye). Elsősorban interaktív módban használatos.

Lásd még: SET DEBUG, SET ECHO, SET STEP

SET TITLE**SET TITLE ON/off**

OFF esetén az új fájlok katalógusba történő felvételekor (SET CATALOG ON) nem szükséges megadni a fájl szöveges leírását.

Lásd még: EDIT, REPLACE, SET CATALOG

SET TRAP**SET TRAP on/OFF**

ON állásban bekövetkező programhiba vagy az Esc billentyű lenyomása esetén automatikusan működésbe lép a beépített hibakereső (debugger).

Az ON ERROR utasítás felülbírálja a SET TRAP utasítást.

Lásd még: DEBUG, ON ERROR, SET DEBUG, SET DEVELOPMENT

SET TYPEAHEAD**SET TYPEAHEAD TO <N-kifejezés>**

Az *N-kifejezés* meghatározza, hogy hány lenyomott billentyűt jegyez meg a rendszer. (billentyűpuffer).

Alapértelmezés: 20. Értéktartomány: 0 – 32000.

Csak akkor működik, ha SET ESCAPE ON állásban van. Az EDIT és APPEND utasításoknál a méret 20 függetlenül a beállított értéktől.

Lásd még: INKEY(), ON, SET BELL, SET ESCAPE

SET UNIQUE

SET UNIQUE on/OFF

ON esetén az azonos kulcsú rekordok közül indexelés során csak az első kerül bele az indexfájlba, *OFF* esetén pedig az összes.

Ha az indexfájl létrehozásakor a **UNIQUE** opció volt érvényes, egy későbbi újraindexelés is így fog történni, függetlenül a **SET UNIQUE** utasítástól.

Lásd még: **FIND**, **INDEX**, **REINDEX**, **SEEK()**, **SEEK**, **SET INDEX**, **SET ORDER**, **USE**

SET VIEW

SET VIEW TO <fájlnev>/?

Aktuálissá teszi a *fájlnev* nézetfájlt (.qbe, .qbo).

A *?* megjeleníti és választásra kínálja a létező nézetfájlokat.

Lásd még: **CREATE/MODIFY VIEW/QUERY**, **SET CATALOG**, **SET FIELDS**, **SET FILTER**, **SET FORMAT**, **SET INDEX**, **SET ORDER**, **SET RELATION**, **SET TITLE**

SET WINDOW

SET WINDOW OF MEMO TO <ablaknev>

Az *ablaknev* ablakot az **APPEND**, **BROWSE**, **CHANGE**, **EDIT**, **READ** utasítások alapértelmezés szerinti memo-mező szerkesztő ablakakának jelöli ki. Ha a felsorolt utasításokban más intézkedés nem történik, memo mező szerkesztésénél az *ablaknev* ablak nyílik meg a képernyőn.

Lásd még: **ACTIVATE SCREEN**, **ACTIVATE WINDOW**, **CLEAR WINDOW**, **DEFINE WINDOW**, **MOVE WINDOW**

E) Függvények

E-1. Egyszerű matematikai függvények

ABS

ABS(*<N-kifejezés>*)

A *N-kifejezés* abszolútértékét adja meg.

CEILING

CEILING(*<N-kifejezés>*)

A *N-kifejezéssel* egyenlő vagy annál éppen (ahhoz legközelebb álló) nagyobb egész számot adja vissza.

Lásd még: FLOOR(), ROUND().

FLOOR

FLOOR(*<N-kifejezés>*)

A *N-kifejezéssel* egyenlő vagy annál éppen (ahhoz legközelebb álló) kisebb egész számot adja vissza.

Lásd még: CEILING(), INT(), ROUND().

FIXED

FIXED(*<N-kifejezés>*)

Az F (lebegőpontos) típusú *N-kifejezést* N (fixpontos) típusúvá alakítja.

Lásd még: FLOAT().

FLOAT

FLOAT(*<N-kifejezés>*)

Az N (fixpontos) típusú *N-kifejezést* F (lebegőpontos) típusúvá alakítja.

Lásd még: FIXED().

INT

INT(<N-kifejezés >)

A *N-kifejezés* tört részét elhagyja és az így kapott számot adja vissza.

Lásd még: CEILING(), FLOOR(), ROUND().

MAX

MAX(<kifejezés1 >, <kifejezés2 >)

A *kifejezés1* és a *kifejezés2* közül a nagyobbbat adja vissza. Mindkét kifejezésnek azonos típusúnak kell lennie. Megengedett típusok: numerikus, dátum és karakter.

Lásd még: CALCULATE, MIN().

MIN

MIN(<kifejezés1 >, <kifejezés2 >)

A *kifejezés1* és a *kifejezés2* közül a kisebbet adja vissza. Mindkét kifejezésnek azonos típusúnak kell lennie. Megengedett típusok: numerikus, dátum és karakter.

Lásd még: CALCULATE, MAX().

MOD

MOD(<N-kifejezés1 >, <N-kifejezés2 >)

A *N-kifejezés1* és a *N-kifejezés2* hányadosának maradékát adja vissza. (Moduló függvény).Képlete:

$\langle N\text{-kifejezés1} \rangle - \text{FLOOR}(\langle N\text{-kifejezés1} \rangle / \langle N\text{-kifejezés2} \rangle) * \langle N\text{-kifejezés2} \rangle$

Lásd még: FLOOR(), INT().

RAND

RAND([<N-kifejezés >])

0 és 0.999999 közötti véletlenszámot ad vissza. Minden ismételt végrehajtás egy következő véletlenszámot ad. A *N-kifejezéssel* a kiindulóérték adható meg. Ha ez negatív, akkor a rendszeridőt tekinti kiindulóértéknek. Ha kiindulóérték nincs megadva, alapértelmezésként 100001-ről indul.

ROUND

ROUND(<N-kifejezés1>, <N-kifejezés2>)

A *N-kifejezés1*-et kerekíti a kerekítés szabályai szerint. A *N-kifejezés2* meghatározza, hogy a kerekítés hány tizedesjegyre történjen.

Lásd még: CEILING(), FLOOR(), INT(), STR(), VAL().

Példa:

```
.? ROUND(14.74682,2)
```

```
14.75
```

```
.? ROUND(14911,-3)
```

```
15000
```

SIGN

SIGN(<N-kifejezés>)

A *N-kifejezés* előjelét adja vissza.

értékei:

+ 1 : <N-kifejezés > > 0

0 : <N-kifejezés > = 0

- 1 : <N-kifejezés > < 0

Lásd még: ABS().

VAL

VAL(<C-kifejezés>)

A számjegyeket (és tizedespontot) tartalmazó *C-kifejezést* számmá alakítja át.

A bevezető szóközoeket figyelmen kívül hagyja és az átalakítást az első nem szám karakterig végzi el.

Lásd még: SET DECIMALS, STR().

E-2. Bonyolultabb matematikai függvények

ACOS

ACOS(<N-kifejezés>)

Arkusz koszinusz függvény.

Lásd még: ASIN(), ATAN(), ATN2(), COS(), DTOR(), FIXED(), FLOAT(), PI(), RTOD(), SET DECIMALS, SET PRECISION, SIN(), TAN().

ASIN

ASIN(<N-kifejezés>)

Arkusz szinusz függvény. A *N-kifejezésnek* a (-1, 1) intervallumba kell esnie.

Lásd még: ACOS(), ATAN(), ATN2(), COS(), DTOR(), FIXED(), FLOAT(), PI(), RTOD(), SET DECIMALS, SET PRECISION, SIN(), TAN().

ATAN

ATAN(<N-kifejezés>)

Arkusz tangens függvény. A *N-kifejezésnek* az (-PI/2, PI/2) intervallumba kell esnie, és mértékegysége a radián.

Lásd még: ACOS(), ASIN(), ATN2(), COS(), DTOR(), FIXED(), FLOAT(), PI(), RTOD(), SET DECIMALS, SET PRECISION, SIN(), TAN().

ATN2

ATN2(<N-kifejezés1>, <N-kifejezés2>)

Arkusz tangens függvény, amely a szög szinusza – *N-kifejezés1* – és koszinusza – *N-kifejezés2* – alapján a szög nagyságát adja vissza radiánban.

Lásd még: ACOS(), ATAN(), ASIN(), COS(), DTOR(), FIXED(), FLOAT(), PI(), RTOD(), SET DECIMALS, SET PRECISION, SIN(), TAN().

COS

COS(<N-kifejezés >)

A radiánban megadott szögnek kiszámítja a koszinuszát.

Lásd még: ACOS(), ATAN(), ATN2(), ASIN(), DTOR(), FIXED(), FLOAT(), PI(), RTOD(), SET DECIMALS, SET PRECISION, SIN(), TAN().

DTOR

DTOR(<N-kifejezés >)

A *N-kifejezés* fokos szög értékét adja vissza radiánban.

Lásd még: ACOS(), ATAN(), ATN2(), RTOD(), SET DECIMALS, SIN().

EXP

EXP(<N-kifejezés >)

A természetes logaritmus alapszámának hatványfüggvénye.

Képlet: $y = e^x$, ahol x a *N-kifejezés*, és y a visszatérési érték.

Lásd még: LOG(), SET DECIMALS, SET PRECISION.

LOG

LOG(<N-kifejezés >)

A *N-kifejezés* természetes alapú logaritmusát adja vissza.

Példa:

```
.? EXP(LOG(12))
12
```

Lásd még: EXP(), LOG10().

LOG10

LOG10(<N-kifejezés >)

A *N-kifejezés* tízes alapú logaritmusát adja vissza.

Lásd még: EXP(), LOG().

PI

PI()

A Ludolf-féle számot (a kör területének és átmérőjének hányadosa: 3.14159...) adja vissza.

RTOD

RTOD(<N-kifejezés>)

A *N-kifejezés* radián számítja át fokra.

Lásd még: ACOS(), ASIN(), ATAN(), ATN2(), COS(), DTOR(), SIN(), TAN().

SINSIN(*N-kifejezés*)

A *N-kifejezés* radián nagyságú szög szinuszát adja vissza.

Lásd még: ACOS(), ASIN(), ATAN(), ATN2(), COS(), DTOR(), PI(), RTOD(), TAN().

SQRT

SQRT(<N-kifejezés>)

A *N-kifejezés* Négyzetgyökével tér vissza.

Lásd még: SET DECIMALS.

TAN

TAN(<N-kifejezés>)

A *N-kifejezés* radián nagyságú szög tangensével tér vissza.

Tangens függvény. A <N-kifejezés>, a szög értéke radiánban megadva.

Lásd még: ACOS(), ASIN(), ATAN(), ATN2(), COS(), SIN().

E-3. Az adatkezelés függvényei

ALIAS

ALIAS([< N-kifejezés >])

A *N-kifejezés* sorszámú munkaterület nevével (sztring) tér vissza. A *N-kifejezés* értéktartománya: 1 – 10.

BOF

BOF([< másodnév >])

.T. (igaz) értéket ad, ha a *másodnévvel* jelölt, illetve ennek hiányában az aktuális adatfájl elején (első rekord előtt) áll a rekordmutató.

Lásd még: EOF().

COMPLETED

COMPLETED()

.T. (igaz) értékkel tér vissza, ha a BEGIN TRANSACTION utasítással megkezdődött tranzakciós utasítássorozat befejeződött, azaz az END TRANSACTION utasítás végrehajtásra került. .T. (igaz) értéket ad akkor is, ha a ROLLBACK utasítás sikeresen befejeződött.

Lásd még: BEGIN/END TRANSACTION, CHANGE(), CONVERT(), LKSYS(), ROLLBACK, ROLLBACK().

DBF

DBF([< másodnév >])

Megadja a *másodnévvel* kijelölt munkaterületen használt adatfájl nevét.

Lásd még: ALIAS(), FIELD(), KEY(), MDX(), NDX(), TAG().

DELETED

DELETED([< másodnév >])

.T. (igaz) értéket ad, ha a *másodnévvel* jelölt vagy ennek híján az aktuális adatfájl aktuális rekordja törölt (törlésre kijelölt) rekord.

Lásd még: PACK, RECALL, SET DELETED.

EOF

EOF([< másodnév >])

.T. (igaz) értéket ad, ha a *másodnévvel* jelölt, illetve ennek hiányában az aktuális adatfájl végén (utolsó rekord után) áll a rekordmutató.

(RECNO() = RECCOUNT() + 1)

.F. (hamis) értéket ad akkor is, ha az adatfájl nem volt megnyitva.

Lásd még: BOF(), ERROR(), FOUND(), RECCOUNT(), RECNO().

FIELD

FIELD(< N-kifejezés > [, < másodnév >])

A *másodnévvel* jelölt, ennek hiányában az aktuális adatfájl (a fájlstruktúra szerinti) *N-kifejezés* sorszámú mezőjének nevét adja vissza. (sztring)

Lásd még: DBF(), LUPDATE(), RECCOUNT(), RECSIZE().

FILE

FILE(< C-kifejezés >)

.T. (igaz) értékkel tér vissza, ha a *C-kifejezésben* nével és kiterjesztéssel (szükség esetén meghajtóval, és elérési úttal is) azonosított fájl létezik.

Lásd még: SET DEFAULT, SET PATH.

FOUND

FOUND([< másodnév >])

.T. (igaz) értéket szolgáltat, ha a legutolsó keresési utasítás (FIND, SEEK) sikeres volt.

Lásd még: CONTINUE, EOF(), FIND(), LOCATE, LOOKUP(), SEEK, SEEK(), SET NEAR.

ISMARKED

ISMARKED([< másodnév >])

.T. (igaz) értékkel tér vissza, ha folyamatban lévő (még nem befejezett) tranzakciós utasítások operálnak a *másodnévvel* jelölt, vagy ennek híján az aktuális adatfájlon. (BEGIN és END TRANSACTION közti utasítások)

Lásd még: BEGIN/END TRANSACTION, COMPLETED(), ROLLBACK.

KEY

KEY([< mdx-fájlnev > ,] < N-kifejezés > [, < másodnév >])

A *másodnévvel* jelölt, ennek hiányában az aktuális adatfájlhoz tartozó *mdx-fájlnev* összetett indexfájl sorravett indextagjai közül a *N-kifejezéssel* sorszámú index kulcskifejezését adja a visszatérő sztring.

Ha az *mdx-fájlnev* hiányzik, akkor a függvény sorra veszi az adott munkaterületen megnyitott összes indexfájlt és indextagot és azokból a *N-kifejezés-edik* indexkifejezését adja vissza.

Lásd még: MDX(), NDX(), ORDER(), TAG().

LOOKUP

LOOKUP(<eredmény mező>, <kereső kifejezés>
<kereső mező>)

Az aktuális – illetve, ha a mezők előtt a másodnemek is szerepelnek, a másodnévvel jelölt – adatfájlban megkeresi azt az első rekordot, amelyben a *kereső kifejezés* értéke egyezik a *kereső mező* értékével. Az *eredménymező* helyén megnevezett mezőnek a megtalált rekordban lévő értékével tér vissza.

Ha az adatfájl a *kereső mezőre* indexelt, akkor a függvény a SEEK utasításhoz hasonlóan működik, azaz a *kereső kifejezéssel* egyező indexkulcsú rekordra áll a rekordmutató.

Ha nincs érvényes index, akkor a LOCATE utasításhoz hasonlóan szekvenciális keresés történik, és a *kereső kifejezésnek* feltétel jellegű logikai kifejezésnek kell lennie. Ilyenkor a függvény a feltételnek megfelelő első rekordig keres.

Ha a keresés sikertelen, a rekordmutató a fájl végére áll.

LUPDATE

LUPDATE([<másodnév>])

A *másodnévvel* jelölt, ennek hiányában az aktuális adatfájl legutolsó módosításának keltét adja meg dátum típusú adatként.

Lásd még: DBF(), DTOC(), FIELD(), MDX(), NDX(), ORDER(), RECCOUNT(), RECSIZE(), TAG().

MDX

MDX(<N-kifejezés> [, <másodnév>])

Karakter típusú adatként adja vissza a *másodnévvel* jelölt, ennek hiányában az aktuális adatfájl *N-kifejezés* sorszámú összetett indexfájljának nevét. A *N-kifejezés* az index megnyitásokor (SET INDEX, USE) az indexlistában elfoglalt helyre utal.

Üres sztringgel tér vissza, ha nincs megnyitott mdx-fájl, vagy az indexlistában nincs ilyen sorszámú.

Lásd még: KEY(), NDX(), ORDER(), TAG().

NDX

NDX(<N-kifejezés> [, <másodnév>])

Karakter típusú adatként adja vissza a *másodnévvel* jelölt, ennek hiányában az aktuális adatfájl *N-kifejezés* sorszámú egyszerű indexfájljának nevét. A *N-kifejezés* az index megnyitásakor (SET INDEX, USE) az indexlistában elfoglalt helyre utal.

Üres sztringgel tér vissza, ha egyáltalán nincs megnyitott ndx-fájl vagy ilyen sorszámú nem létezik.

Lásd még: ALIAS(), DBF(), FIELD(), LUPDATE(), MDX(), ORDER(), RECCOUNT(), RECSIZE(), SET INDEX, SET ORDER.

ORDER

ORDER([másodnév])

A *másodnévvel* jelölt, ennek hiányában az aktuális adatfájl aktuális (elsődleges, érvényes) indextag vagy ndx-fájl nevét adja vissza.

Lásd még: KEY(), MDX(), NDX(), SET ORDER, TAG().

RECCOUNT

RECCOUNT([<másodnév>])

A *másodnévvel* jelölt, ennek hiányában az aktuális adatfájlban lévő rekordlétszámát adja vissza.

Ez 0, ha nincsen megnyitva a megjelölt, vagy megjelölés hiányában egyetlen adatfájl sem.

Lásd még: DBF(), DISKSPACE(), RECSIZE().

RECNO

RECNO([<másodnév>])

Az aktuális rekord sorszámát adja vissza a *másodnévvel* jelölt, ennek hiányában az aktuális munkaterületen.

Ez 0, ha nincsen megnyitva a megjelölt, vagy megjelölés hiányában egyetlen adatfájl sem, de akkor is ha a rekordmutató a fájl elején (BOF() = .T.) vagy fájl végén (EOF() = .T.) áll.

Lásd még: SET STATUS.

RECSIZE

RECSIZE([< másodnév >])

A *másodnévvel* jelölt, ennek hiányában az aktuális adatfájl rekordjának hosszát (bájt) adja vissza.

Ez 0, ha nincsen megnyitva a megjelölt, vagy megjelölés hiányában egyetlen adatfájl sem.

Lásd még: DBF(), DISKSPACE(), RECCOUNT().

ROLLBACK

ROLLBACK()

.T. (igaz) értéket ad, ha a legutoljára végrehajtott ROLLBACK utasítás sikeres volt.

Lásd még: BEGIN/END TRANSACTION, COMPLETED(), ISMARKED(), ROLLBACK.

SEEK

SEEK(< kifejezés > [, < másodnév >])

.T. (igaz) értéket ad, ha *másodnévvel* jelölt, ennek hiányában az aktuális adatfájlban található egy *kifejezéssel* egyező **indexkulcsú** rekord. Ennek megfelelően **csak indexelt adatfájlok** esetében működik. A *kifejezésre* a SEEK utasításnál említettek vonatkoznak.

Tekintettel arra, hogy a keresést végre is hajtja, a rekordmutató a megtalált rekordra, vagy ha ilyen nincs a fájl végére áll.

Lásd még: FIND, FOUND(), LOOKUP(), SEEK.

SELECT

SELECT()

Visszaadja a szabad munkaterületek közül a legnagyobb számát.

Lásd még: ALIAS().

TAG

TAG([<mdx-fájl > ,] <N-kifejezés > [, <másodnév >])

Sztringként visszaadja a *másodnévvel* jelölt, ennek hiányában az aktuális adatfájlhoz tartozó *mdx-fájlban* a *N-kifejezés* sorszámú indextag, vagy ha ilyen nincs a *N-kifejezésnek* megfelelő *ndx-fájl* nevét. A *N-kifejezés* az összetett indexfájlban az indextag helyére, illetve a nyitott *ndx-fájlok* listabeli (SET INDEX, USE) sorrendjére utal.

Ha ilyen sorszámmal nincs index, a visszatérési érték üres sztring.

Lásd még: DBF(), KEY(), MDX(), NDX(), ORDER().

E-4. Sztringfüggvények

ASC

ASC(<C-kifejezés >)

A *C-kifejezés* első karakterének (decimális) ASCII kódszámát adja vissza.

AT

AT(<C-kifejezés1 > , <C-kifejezés2 > / <memo mező >)

Visszaadja annak a karakterpozíciónak a számát, ahányadik helyen a *C-kifejezés1* a *C-kifejezés2*ben vagy *memo mező*ben kezdődik. Ez 0, ha a *C-kifejezés1* nem fordul elő a *C-kifejezés2*ben.

Példa:

```
.? AT("kétség", "Egységben a kétség.")
13
.? AT("demokrácia", "Egypártrendszer")
0
```

Lásd még: LEFT(), RIGHT(), SUBSTR().

CHR

CHR(<N-kifejezés>)

Az ASCII (decimális) kódtáblázat szerint a *N-kifejezésnek* megfelelő karaktert adja vissza.

Lásd még: ???, ASC().

ISALPHA

ISALPHA(<C-kifejezés>)

.T. (igaz) értéket ad, ha a *C-kifejezés* az angol abc-ben lévő betűvel kezdődik.

Lásd még: ISLOWER(), ISUPPER(), LOWER(), UPPER().

ISLOWER

ISLOWER(<C-kifejezés>)

.T. (igaz) értéket ad, ha a *C-kifejezés* az angol abc-ben lévő kisbetűvel kezdődik.

Lásd még: ISALPHA(), ISUPPER(), LOWER(), UPPER().

ISUPPER

ISUPPER(<C-kifejezés>)

.T. (igaz) értéket ad, ha a *C-kifejezés* az angol abc-ben lévő nagybetűvel kezdődik.

Lásd még: ISALPHA(), ISLOWER(), LOWER(), UPPER().

LEFT

LEFT(<C-kifejezés>/<memo mező>,
<N-kifejezés>)

A *C-kifejezés* vagy *memo mező* első (balról számított) *N-kifejezés* darab karakterét adja vissza.

Lásd még: AT(), LTRIM(), RIGHT(), RTRIM(), STUFF(), SUBSTR(), TRIM().

LEN

LEN(<C-kifejezés>/<memo mező>)

A *C-kifejezés* vagy *memo mező* hosszával tér vissza.

Lásd még: TRIM().

LOWER

LOWER(<C-kifejezés>)

Átalakítja a *C-kifejezés* angol nagybetűit kisbetűkké, és az így képzett sztringgel tér vissza.

Lásd még: ISALPHA(), ISLOWER(), ISUPPER(), UPPER().

LTRIM

LTRIM(<C-kifejezés>)

A *C-kifejezés* elejéről (bal oldaláról) levágja a bevezető szóközöket, – ha volt ilyen –, és az így megrövidített sztringgel tér vissza.

Lásd még: LEFT(), RIGHT(), RTRIM(), STR(), SUBSTR().

MEMLINES

MEMLINES(<memo mező>)

Visszadja azt, hogy hány – a SET MEMOWIDTH utasításban meghatározott hosszúságú – sorban fér el a *memo mező*.

Lásd még: MLINE(), SET MEMOWIDTH.

MLINE

MLINE(< memo mező >, < N-kifejezés >)

Sztringként adja vissza a *memo mező* *N-kifejezés* sorszámú sorát, amelyet a SET MEMOWIDTH utasításban meghatározott sorhossz alapján keres meg.

Példa:

```
SET MEMOWIDTH TO 13
```

```
.? m m
```

```
Katinak volt
```

```
kis báránya,
```

```
gyapja fehér,
```

```
mint a hó.
```

```
.? MLINE(m m,3)
```

```
gyapja fehér,
```

Lásd még: MEMLINES(), SET MEMOWIDTH.

REPLICATE

REPLICATE(< C-kifejezés >, < N-kifejezés >)

Olyan sztringgel tér vissza, amelyben a *C-kifejezés* a *N-kifejezés-szer* ismétlődik.

Példa:

```
.? REPLICATE("mindent kétszer mond, ",2)
```

```
mindent kétszer mond, mindent kétszer mond,
```

RIGHT

RIGHT(< C-kifejezés >/< memo-mező >, < N-kifejezés >)

A *C-kifejezés* vagy *memo-mező* utolsó (jobbról számított) *N-kifejezés* darab karakterét adja vissza.

Lásd még: AT(), LEFT(), LTRIM(), RTRIM(), STUFF(), SUBSTR(), TRIM().

RTRIM

RTRIM(<C-kifejezés >)

vagy

TRIM(<C-kifejezés >)

A *C-kifejezés* végéről (jobb oldaláról) levágja a befejező szóközöket, – ha volt ilyen –, és az így megrövidített sztringgel tér vissza.

Lásd még: LEFT(), LTRIM(), RIGHT().

SPACE

SPACE(<N-kifejezés >)

N-kifejezés darab szóközből álló sztringgel tér vissza.

A *N-kifejezés* legfeljebb 254 lehet.

STR

STR(<N-kifejezés > [, <hossz >] [, <tizedes >])

A *N-kifejezés* értékét – számként – a megadott *hossz* hosszúságú sztringgé alakítja, amely során csak a *tizedes* számú tizedesjegyet veszi figyelembe. A visszatérési érték az így képzett sztring.

Alapértelmezés: *hossz* = 10, *tizedes* = 0.

Ha a szám nem fér el a megadott *hosszon*, akkor csillagsorozatot ad vissza.

Lásd még: VAL().

STUFF

STUFF(<C-kifejezés1>, <N-kifejezés1>,
<N-kifejezés2>, <C-kifejezés2>)

A *C-kifejezés1* egy részét megváltoztatja.

A *C-kifejezés1*-ből a *N-kifejezés1* karakterpozíciótól kezdve a *N-kifejezés2* darabszámú karaktert törli.

Az így keletkezett sztringbe – ugyancsak a – *N-kifejezés1* karakterpozíció elé beszúrja a *C-kifejezés2*-t, és ez lesz a visszatérő érték (sztring).

Bármelyik *N-kifejezés* lehet 0, és a *C-kifejezés2* is lehet üres sztring. A *N-kifejezések* értékétől és a *C-kifejezések* hosszától és ezek kombinációjától függően a visszakapott sztring hossza több is kevesebb is lehet az eredeti *C-kifejezés1* hosszánál, és ettől függően tartalma is különbözőképpen változhat az eredetihez képest.

Lásd még: LEFT(), RIGHT(), SUBSTR().

SUBSTR

SUBSTR(<C-kifejezés>, <N-kifejezés1>
[, <N-kifejezés2>)

A *C-kifejezés* *N-kifejezés1* karakterpozíciójától *N-kifejezés2* pozíciójáig tartó részével tér vissza.

Ha *N-kifejezés2* hiányzik, akkor alapértelmezése a *C-kifejezés* hossza.

Lásd még: AT(), LEFT(), LTRIM(), RIGHT(), STR(), STUFF().

TRANSFORM

TRANSFORM(<kifejezés>, <C-kifejezés>)

A visszatérő érték a *kifejezés* *C-kifejezés* szerinti formában, ahol a *C-kifejezés*-ben a PICTURE cikkelyben (lásd: @ ...SAY utasítás) megadható minták szerepelhetnek.

A visszatérő érték típusa a *kifejezés* típusától függ. (Jobbára visszatérő sztring.)

Példa:

```
?.? TRANSFORM("Kovács", "@R X-X-X-X-X-X" )
K-o-v-á-c-s
```

Lásd még: @...PICTURE.

UPPER

UPPER(<C-kifejezés>)

Átalakítja a *C-kifejezés* angol kisbetűit nagybetűkké, és az így képzett sztringgel tér vissza.

Lásd még: ISALPHA(), ISLOWER(), ISUPPER(), LOWER().

E-5. Dátumfüggvények

CDOW

CDOW(<D-kifejezés>)

A visszatérő sztring a hét azon napjának angol nevét tartalmazza, amelyre a *D-kifejezés* dátum esik.

Lásd még: CTOD(), DATE(), DAY(), DOW(), DTOC().

CMONTH

CMONTH(<D-kifejezés>)

A *D-kifejezés* hónapjának angol nevét adja vissza sztringként.

Lásd még: DATE(), MONTH().

CTOD

CTOD(<C-kifejezés>) vagy { <C-kifejezés> }

A *C-kifejezés* dátum típusú megfelelőjével tér vissza.

Lásd még: DATE(), DTOC(), DTOS(), SET CENTURY, SET DATE.

DATE

DATE()

Az aktuális rendszerdátumot adja vissza.

Lásd még: CTOD(), DTOC(), DTOS(), SET CENTURY, SET DATE.

DAY

DAY(<D-kifejezés >)

A *D-kifejezés* hónapjának napját adja vissza. (numerikus)

Lásd még: CDOW(), DOW(), DATE().

DMY

DMY(<D-kifejezés >)

A *D-kifejezést* adja vissza sztringként a következő formátumban: NN angol hónapnév ÉÉ (ahol ÉÉ az év utolsó két számjegye, NN a nap száma)

Példa:

```
.? DMY({02/29/88})  
29 February 88
```

Lásd még: CDOW(), CMONTH(), MDY(), MONTH(), DATE(), DOW(), SET CENTURY, SET DATE, YEAR().

DOW

DOW(<D-kifejezés >)

A visszatérő numerikus érték a hét azon napjának számát tartalmazza, amelyre a *D-kifejezés* dátum esik.

A számozás vasárnapkal kezdődik, amelynek sorszáma az 1.

Lásd még: CDOW(), DATE(), DAY().

DTOC

DTOC(<D-kifejezés >)

A *D-kifejezés* sztring megfelelőjével tér vissza.

Példa:

```
.? DTOC({02/29/88})  
02/29/88
```

Lásd még: CTOD(), SET CENTURY, SET DATE.

DTOS

DTOS(<D-kifejezés >)

A *D-kifejezés* indexelésre is alkalmas formátumú sztring megfelelőjével tér vissza. A formátum: **ÉÉÉÉHHNN** (ahol **ÉÉÉÉ** az évszám, **HH** a hónap száma, **NN** a nap száma)

Példa:

```
.? DTOS({02/29/88})  
19880229
```

Lásd még: **CTOD()**, **DATE()**, **DTOC()**, **SET CENTURY**, **SET DATE**.

MDY

DMY(<D-kifejezés >)

A *D-kifejezést* adja vissza sztringként a következő formátumban: **angol hónapnév NN ÉÉ** (ahol **ÉÉ** az év utolsó két számjegye, **NN** a nap száma)

Példa:

```
.? DMY({02/09/88})  
February 29, 88
```

Lásd még: **DMY()**, **SET CENTURY**, **SET DATE**, **SET MARK**.

MONTH

MONTH(<D-kifejezés >)

A *D-kifejezés* hónapjának számát adja vissza.

Lásd még: **CMONTH()**, **DAY()**, **YEAR()**.

TIME

TIME()

A visszatérő sztring a rendszeridőt tartalmazza **óó:pp:mm** formában (ahol **óó** az óra, **pp** a percek száma, **mm** a másodpercek száma).

Példa:

```
.? TIME()  
23:59:59
```

Lásd még: **DATE()**.

YEAR

YEAR(<D-kifejezés>)

A *D-kifejezés* dátum évszámát adja 4 számjeggyel.

E-6. Képernyős adatbevitel függvényei

BAR

BAR()

Az utolsó aktív rolómenü kiválasztott menüpontjának számát adja vissza.

Ez 0, ha nincs aktív, vagy nem is volt kijelölt rolómenü, vagy ha mégis volt, abból az Esc billentyűvel történt kilépés.

Lásd még: ACTIVATE POPUP, DEFINE POPUP, POPUP(), PROMPT().

INKEY

INKEY([n-másodperc])

A billentyűpufferben levő első karakter kódját adja vissza.

A függvény a Ctrl-S és Esc billentyűket nem kezeli. A vizsgálattal együtt kiüríti a puffert.

Ha a *n-másodperc* időtartam szerepel, akkor a függvény legfeljebb *n-másodpercig* várakozik egy billentyű leütésére. Időtartam híján várakozás nincs.

A visszatérési érték a normál billentyűknél általában az ASCII kódtáblázat szerinti, a billentyűkombinációknál pedig az alábbi táblázat szerinti:

Billentyű nevek	Kódszám (decimális)
Ctrl-A Home	1
Ctrl-B	2
Ctrl-C PgDn	3
Ctrl-D →	4
Ctrl-E ↑	5
Ctrl-F Ctrl-END	6
Ctrl-G Del	7
Ctrl-H	8
Ctrl-I Tab	9
Ctrl-J	10
Ctrl-K	11
Ctrl-L	12
Ctrl-M	13
Ctrl-N	14
Ctrl-O	15
Ctrl-P	16
Ctrl-Q	17
Ctrl-R PgUp	18

Billentyű nevek	Kódszám (decimális)
Ctrl-S ←	19
Ctrl-T	20
Ctrl-U	21
Ctrl-V Ins	22
Ctrl-W Ctrl-End	23
Ctrl-X ↓	24
Ctrl-Y	25
Ctrl-Z	26
ESC Ctrl-[27
F1 Ctrl-\	28
Ctrl-] Ctrl-Home	29
Ctrl-^ Ctrl-PgDn	30
Ctrl- Ctrl-PgUp	31
Space	32
Vissza Tab	-400
Ctrl-Del	-401
Ctrl-Enter	-402
F2	-1
F3	-2
F4	-3
F5	-4
F6	-5
F7	-6
F8	-7
F9	-8
F10	-9
Ctrl-F1	-10
Ctrl-F2	-11
Ctrl-F3	-12
Ctrl-F4	-13
Ctrl-F5	-14
Ctrl-F6	-15
Ctrl-F7	-16
Ctrl-F8	-17
Ctrl-F9	-18
Ctrl-F10	-19
Shift-F1	-20
Shift-F2	-21
Shift-F3	-22
Shift-F4	-23
Shift-F5	-24
Shift-F6	-25
Shift-F7	-26
Shift-F8	-27
Shift-F9	-28
Shift-F10	-29

Billentyű nevek	Kódszám (decimális)
Alt-a	-435
Alt-b	-434
Alt-c	-433
Alt-d	-432
Alt-e	-431
Alt-f	-430
Alt-g	-429
Alt-h	-428
Alt-i	-427
Alt-j	-426
Alt-k	-425
Alt-l	-424
Alt-m	-423
Alt-n	-422
Alt-o	-421
Alt-p	-420
Alt-q	-419
Alt-r	-418
Alt-s	-417
Alt-t	-416
Alt-u	-415
Alt-v	-414
Alt-w	-413
Alt-x	-412
Alt-y	-411
Alt-z	-410
Alt-1	-451
Alt-2	-450
Alt-3	-449
Alt-4	-448
Alt-5	-447
Alt-6	-446
Alt-7	-445
Alt-8	-444
Alt-9	-443
Alt-0	-442

Lásd még: CHR(), LASTKEY(), ON KEY, READ, READKEY(), SET TYPEAHEAD.

LASTKEY

LASTKEY()

A billentyűpufferben levő **utolsó** karakter kódját adja vissza.

Visszatérési értékei egyeznek az INKEY() függvényével.

Lásd még: INKEY(), READKEY().

MENU

MENU()

Visszaadja az aktív vonalmenü nevét.

Lásd még: ACTIVATE MENU, DEFINE MENU.

PAD

PAD()

Az aktív vonalmenü kiválasztott menüpontjának szövegét adja vissza.

Példa:

Ha egy vonalmenüben a TÁMADÁS, VÉDEKEZÉS, FUTÁS lehetséges menüpontok közül a felhasználó a FUTÁS-t választotta, akkor

```
?.? PAD()  
FUTÁS
```

Lásd még: DEFINE BAR, DEFINE MENU, DEFINE PAD, DEFINE POPUP, MENU(), ON PAD, ON SELECTIO PAD, POPUP().

POPUP

POPUP()

Az aktív rolómenü nevét adja vissza.

Lásd még: ACTIVATE POPUP, BAR(), DEFINE POPUP, MENU(), PAD(), PROMPT()

PROMPT**PROMPT()**

Az éppen kiválasztott menüpont szövegét adja vissza függetlenül attól, hogy az vonal- vagy rolómenühöz tartozik.

Üres sztringgel tér vissza, ha nincs aktív menü, vagy az felhasználó az Esc billentyűt nyomta meg.

Lásd még: BAR(), DEFINE POPUP, ON SELECTION PAD, ON SELECTION POPUP, POPUP().

READKEY**READKEY()**

A teljes-képernyős adatbevitel kilépő (befejező) billentyűleütésének kódszámát (lásd INKEY()) adja vissza. A teljes-képernyős adatbevitel az APPEND, BROWSE, CHANGE, CREATE, EDIT, INSERT, MODIFY, READ utasításokkal történhet. A visszatérő értékből az is megállapítható, hogy az adat módosításra került e vagy sem. Ha nem volt módosítás, akkor a visszatérési érték 0 – 36 közé esik, ellenkező esetben 256-nál nagyobb. Az adatbevitelt, szerkesztést befejező kilépőbillentyűk (terminátor) visszatérési kódjait tartalmazza a következő táblázat:

Módosítás történt	Módosítás nem történt	Billentyű
0	256	Ctrl-S Ctrl-H ←
-	256	Del
1	257	Ctrl-D Ctrl-L →
4	260	Ctrl-E Ctrl-K ↑
5	261	Ctrl-J Ctrl-X ↓
6	262	Ctrl-R PgUp
7	263	Ctrl-C PgDn
12	-	Ctrl-Q Ctrl-\ ESC
-	270	Ctrl-W Ctrl-End
15	271	Ctrl-M Enter
16	-	Ctrl-M Enter
33	289	Ctrl-Home
34	290	Ctrl-PgUp
35	291	Ctrl-PgDn
36	292	F1

VARREAD

VARREAD()

Az éppen szerkesztés, módosítás alatt álló változó vagy mező nevét adja vissza.
Lásd még: APPEND, DEFINE WINDOW, EDIT, READ.

E-7. Környezeti függvények

COL

COL()

A kurzor aktuális oszlopának számát adja vissza.
Lásd még: @, PCOL(), PROW(), ROW().

DISKSPACE

DISKSPACE()

Az (alapértelmezés szerinti) lemezegység szabad kapacitását adja vissza bájtban.
Lásd még: RECCOUNT(), RECSIZE().

FKLABEL

FKLABEL(<N-kifejezés>)

A *N-kifejezéssel* megadott (kódú) funkcióbilletterűhöz rendelt utasításssorozatot adja vissza.

A *N-kifejezésnek* az 1,28 intervallumba kell esnie, 1 = F2.

Lásd még: FKMAX(), SET, SET FUNCTION.

FKMAX

FKMAX()

A programozható funkciógombok létszámát adja vissza.

Lásd még: FKLABEL(), SET, SET FUNCTION.

GETENV

GETNEV(<C-kifejezés>)

A C-kifejezéssel megadott DOS rendszerváltozó értékét adja vissza.

Példa:

```
.? GETENV("path")  
C:\DOS;C:\DBASE
```

Lásd még: OS()

ISCOLOR

ISCOLOR()

.T. (igaz) értéket ad, ha a számítógép rendelkezik színes grafikus kártyával.

Lásd még: SET COLOR, SET DISPLAY.

MEMORY

MEMORY()

A szabad (felhasználható) memória méretét adja vissza kilobájtokban.

Lásd még: DISKSPACE(), GETENV(), OS().

OS

OS()

Az operációs rendszer nevével tér vissza.

Lásd még: GETENV(), VERSION().

PCOL

PCOL()

A nyomtatófej aktuális oszlopának számát adja vissza.

Lásd még: @, COL(), PROW(), ROW(), SET MARGIN, _pcolno, _ploffset.

PRINTSTATUS

PRINTSTATUS()

.T. (igaz) értékkel tér vissza, ha a nyomtató üzemkész.

Lásd még: SET DEVICE, SET PRINTER.

PROW

PROW()

A nyomtatófej aktuális sorának számát adja vissza.

PROW() = 0, ha lapdobás történt.

Lásd még: @, COL(), EJECT, ROW(), SET DEVICE, SET PRINTER, _plineno.

ROW

ROW()

A kurzor aktuális sorának számát adja vissza.

Lásd még: @, COL(), PCOL(), PROW().

SET

SET(<C-kifejezés>)

A *C-kifejezéssel* megadott SET utasítás állapotát adja vissza. A visszatérési érték "ON", "OFF" vagy szám.

A *C-kifejezés* csak olyan SET kulcsszót tartalmazhat, amelynek paramétere ON, OFF vagy egy szám. Ha a kulcsszóhoz ON/OFF és szám is tartozhat, akkor a függvény "ON"-t vagy "OFF"-ot ad vissza.

Lásd még: LIST/DISPLAY STATUS, SET, SET STATUS.

TYPE

TYPE(<C-kifejezés >)

A C-kifejezéssel megadott kifejezés típusát adja vissza.

A következő karaktereket adja vissza a függvény:

"C" = karakteres típus

"N" = numerikus típus

"L" = logikai típus

"M" = memo típus

"D" = dátum típus

"F" = lebegőpontos típus

"U" = meghatározatlan, nem létező

Példa:

```
.a = 2
```

```
.? TYPE ("a")
```

```
N
```

```
.c = "karakteres "
```

```
.kif = "kifejezés"
```

```
.? TYPE ("c + kif")
```

```
C
```

```
.? TYPE (c)
```

```
U
```

VERSION

VERSION()

A dBASE változatának nevét adja vissza.

Példa:

```
.? VERSION()
```

```
dBASE IV 1.0
```

Lásd még: OS().

E-8. Hálózati üzemmód függvényei

ACCESS

ACCESS()

Az aktuális felhasználó adathozzáférési szintjének számát adja vissza.

Lásd még: FLOCK(), NETWORK(), PROTECT(), RLOCK(), UNLOCK(), USER().

CHANGE

CHANGE()

.T. (igaz) értékkel tér vissza, ha a hálózati üzemmódba történt belépés óta változás történt az aktuális rekordban.

Csak akkor működik helyesen, ha az adatfájlt a CONVERT utasítással hálózati üzemmódra előkészítették.

Lásd még: CONVERT(), FLOCK(), LKSYS(), RLOCK(), SET REFRESH.

FLOCK

FLOCK(< [másodnév >])

Kizárólagos használatra foglalja le az aktuális, illetve a *másodnévvel* jelölt adatfájlt. .T. (igaz) értékkel tér vissza, ha a művelet sikeres volt.

Lásd még: ACCESS(), RLOCK(), SET LOCK, SET REPROCESS, UNLOCK.

LKSYS

LKSYS(n)

A zárolt rekordról vagy adatfájlról szolgáltat karakteres típusú adatokat.

n = 0 a zárolás időpontját adja vissza.

n = 1 a zárolás dátumát adja vissza.

n = 2 annak a felhasználónak a bejelentkezési nevét adja vissza, aki zárolta a rekordot.

Ha zároláskor nem használták a CONVERT parancsot, akkor a függvény null sztringet ad vissza.

NETWORK**NETWORK()**

.T. értékkel tér vissza, ha a rendszer hálózati üzemmódban működik.

Lásd még: GETENV(), OS().

RLOCK/LOCK**RLOCK/LOCK([<C-kifejezés > ,] [< másodnév >])**

Zárolja a *C-kifejezésben* rekordszámú rekordokat a *másodnévvel* megjelölt munkaterületen. A *C-kifejezésben* a zárolni kívánt rekordok sorszáma sorolható fel vesszővel elválasztva. Ha a rekordlista hiányzik, akkor az aktuális rekordot zárolja.

.T. (igaz) értéket ad vissza, ha az összes zárolás sikeres volt.

Példa:

```
.? RLOCK("1,2,5",kilenc)  
.T.
```

Lásd még: FLOCK(), SET LOCK, SET RELATION, UNLOCK.

USER**USER()**

Az éppen bejelentkezett felhasználó nevét adja vissza, ha a védelmi rendszer (PROTECT) érvényben van. Egyébként üres sztringgel tér vissza.

Lásd még: PROTECT.

E-9. Különleges függvények

&

&< karakteres változó > [.]

A karakteres változó tartalmát konstansként értelmezi egy utasításban.

Ha a változót még kiegészíti más konstans is, akkor az elkülönítésre a "." használható.

Példa:

```
.csok = "puszi"
.csokit = "mogyoróst"
.ropit = "ropogóst"
```

```
.makro = "csok"
.? makro
csok
.? &makro
puszi
.? &makro.it
mogyoróst
.makro = "rop"
.? &makro.it
ropogóst
```

CALL

CALL(<C-kifejezés1 >, <C-kifejezés2 >
/ <változó >)

A memóriában lévő *C-kifejezés1* nevű bináris (gépi kódú) program végrehajtása során keletkezett értéket szolgáltat vissza.

A *C-kifejezés2* az átadandó paramétereket tartalmazhatja (legfeljebb hetet) a fogadóprogram által követelt szintaktika szerint. Paraméter átadható *változóban* is. Tömb nem, de tömbelem (egyenként) átadható. Így a *C-kifejezés2* változóra, mezőre, tömbelemre való hivatkozást is tartalmazhat.

Lásd még: DECLARE, LOAD, STORE.

DIFFERENCE

DIFFERENCE(<C-kifejezés1 >, <C-kifejezés2 >)

A visszatérési érték 0 és 4 közé esik, amely azt jellemzi, hogy angol fonetika alapján hangzás szerint mennyi közös van a két *C-kifejezés*ben. 0 az érték, ha nincs közös hangzó bennük, 4 ha azonosan hangzanak. A hangzást a SOUNDEX() függvény segítségével vizsgálja.

Lásd még: SOUNDEX().

ERROR

ERROR()

az előforduló hiba kódját adja vissza.

Csak akkor használható, ha előtte az ON ERROR utasítás is kiadásra került.

Lásd még: DEBUG, LINENO(), MESSAGE(), ON ERROR, PROGRAM(),
RETRY, RETURN.

FV

FV(<befizetés >, <kamatláb >, <periódus-szám >)

A rendszeres *befizetések*nek a *periódus-szám* letelte utáni, a *kamatláb* szerinti kamatos kamattal számított összértékét adja vissza. A *kamatláb* egy periódusnyi időre vonatkozik.

Lásd még: CALCULATE, PV().

IIF

IIF(<feltétel >, <kifejezés1 >, <kifejezés2 >)

Ha a *feltétel* teljesül, a *kifejezés1* visszatérési érték, ha nem, akkor a *kifejezés2*.

LIKE

LIKE(< minta >, < C-kifejezés >)

.T. (igaz) értéket ad, ha a *minta* illeszkedik a *C-kifejezésre*. A mintában két helyettesítő (Joker) szimbólum állhat:

* Egyszerre több karaktert helyettesít, azaz a helyén több tetszés szerinti karakter is lehet.

? Egy tetszés szerinti karaktert helyettesít.

Példa:

```
.? LIKE("*a*","abc")  
.T.  
.? LIKE("?a*","abc")  
.F.  
.? LIKE("*BASE*","dBASE IV")  
.T.  
.? LIKE("*BASE*","FOXBASE")  
.T.  
.? LIKE("*BASE*","dBASE III")  
.F.
```

Lásd még: SOUNDINDEX().

LINENO

LINENO()

Az éppen végrehajtott programsor számával tér vissza. Programtesztelésre (DEBUG), vagy az ON ERROR utasítással együtt hibadetektálásra használható.

Lásd még: PROGRAM(), SET DEVELOPMENT.

MESSAGE

MESSAGE()

Az ON ERROR-ral kezelhető hibák hibaüzenetét adja vissza.

Lásd még: ERROR(), ON ERROR.

PAYMENT

PAYMENT(< kölcsön > , < kamatláb > , < periódus-szám >)

A megadott *kamatlábbal* felvett *kölcsön* *periódus-szám* idő alatt (vagy lépésben) tervezett törlesztéséhez szükséges rendszeres törlesztőrészlet értékét adja vissza. A *kamatláb* egy periódusnyi időre vonatkozik.

Lásd még: CALCULATE, FV(), PV().

PROGRAM

PROGRAM()

Annak a programnak vagy eljárásnak a nevét adja vissza, amelyiknek éppen a hiba előfordulásakor történt a végrehajtása.

Lásd még: DEBUG, LINENO(), RESUME, SET DEVELOPMENT, SUSPEND.

PV

PV(< befizetés > , < kamatláb > , < periódus-szám >)

A *periódus-szám* óta rendszeresen megtörtént *befizetés* jelenlegi összértékét adja vissza beleszámítva a *kamatláb* szerinti kamatos kamatokat is. A *kamatláb* egy periódusnyi időre vonatkozik.

Lásd még: FV(), PAYMENT().

SOUNDEX**SOUNDEX(< C-kifejezés >)**

Visszatérési értéként olyan kódot szolgáltat, amely a hasonló, azonos kiejtésű szavaknál azonos kódértéket szolgáltat. A függvény az angol fonetikát veszi alapul, ezért hazai felhasználása várhatóan ritkaságszámba megy. Csupán az érdekesség kedvéért álljon itt néhány szóban a működése:

A függvény négy karakteres kódot állít elő a következő algoritmus szerint.

1; Veszi a C-kifejezés első betűjét.

2; Elhagyja az a, e, h, i, o, u, w, y betűket, kivéve ha az a legelső pozíción van.

3; A következő számokat rendeli a maradék betűkhöz (azzal helyettesítve őket):

b,f,p,v	= 1
c,g,j,k,q,s,x,z	= 2
d,t	= 3
l	= 4
m,n	= 5
r	= 6

4; Ha két vagy több szomszédos betűnek ugyanaz a kódja, akkor az első kivételével a többit elhagyja.

5; Előállítja a "betű számjegy számjegy számjegy" formátumot úgy, hogy ha három számjegynél kevesebb maradt, akkor 0-kal egészíti ki, ha 3-nál több maradt, akkor a többi számjegyet elhagyja.

6; Leáll az első nem betű karakternél.

7; Átlépi a bevezető szóközöket.

8; "0000"-át ad vissza, ha az első nem szóköz karakter nem betű.

Lásd még: DIFFERENCE(), LOCATE.

F) Rendszerváltozók

A rendszerváltozók általában az adatfolyamként történő megjelenítés, módját, formáját, pillanatnyi állapotát vezérlő információkat tartalmazzák, amelyek értékadással megváltoztathatóak. Adatfolyamként jelenít meg adatokat a @, @ ...TO, EJECT utasítások kivételével minden adatot is megjelenítő utasítás (pl:??/? utasítások).

Ritkább esetben hatásosak teljes-képernyős megjelenítésnél is. Ez a magyarázatoknál külön is megemlítesre kerül.

_alignment

```
_alignment = "LEFT"/"center"/"right"
```

A megjelenített szöveg igazításának éppen érvényes módját tartalmazza, amely értékadással megváltoztatható. *Left* esetén balra, *Center* esetén középre, *Right* esetén jobbra történik az igazítás.

Csak akkor van hatása, ha a `_wrap` változó értéke `.T.`

Lásd még: `_lmargin`, `_rmargin`, `_wrap`, @ (PICTURE B,I,J paraméterek), STR().

_box

```
_box = <feltétel >
```

A DEFINE BOX utasítással kijelölt keretek megjelenítését vezérli. Ha a `_box = .T.`, akkor a keret éppen soronkövetkező része megjelenik, egyébként nem.

Alapértelmezés: `_box = .T.`

Lásd még: DEFINE BOX, SPACE().

_indent

`_indent` = < N-kifejezés >

A bekezdéseknél a "behúzás" mindenkor értékét tartalmazza, amely értékadással megváltoztatható. A *N-kifejezés* meghatározza, hogy soronkövetkező bekezdés első sora hol kezdődjön a bal margóhoz (`_lmargin`) képest, de csak a jobb és a bal margó közé eshet.

Alapértelmezés : `_indent` = 0.

Értéktartomány: $0 \leq \textit{_indent} \leq (\textit{_rmargin} - \textit{_lmargin})$.

Csak akkor van hatása ha a `_wrap` változó értéke `.T.`

Lásd még: `_lmargin`, `_ploffset`, `_rmargin`, `_wrap`, ???.

_lmargin

`_lmargin` = < N-kifejezés >

Az éppen érvényes bal margó értékét tartalmazza, amely megváltoztatható. Értékadásnál a *N-kifejezés* lesz az új bal margó.

Alapértelmezés : 0. Értéktartomány: 0 – 254.

Csak akkor van hatása, ha a `_wrap` változó értéke `.T.`

Lásd még: `_indent`, `_ploffset`, `_rmargin`, `_wrap`, ???.

_padvance

`_padvance` = "FORMFEED"/"linefeeds"

A nyomtatási lapemelés éppen érvényes módját tartalmazza. Nyomtatásnál *formfeed* esetén szabályos lapdobás történik, *linefeeds* esetén pedig a hátralévő sorok kiszámított értéke szerinti soremelések történnek.

Lásd még: `_peject`, `_plength`, `_plineno`, `EJECT`, `PROW()`.

_pageno

_pageno = < N-kifejezés >

A mindenkori lapszámot tartalmazza, amely a megjelenítés során változik. Értékadás után az aktuális oldalszám az *N-kifejezés* lesz.

Alapértelmezés : 1. Értéktartomány: 1 – 32767. Értéke a nyomtatás során változik és függ a `_plineno` és `_plenght` értékeitől.

Lásd még: `_pbpage`, `_pepage`, `_plenght`, `_plineno`, `on page`.

_pbpage

_pbpage = < N-kifejezés >

A nyomtatási köteg (PRINTJOB, ENDPRINTJOB) kezdő oldalszámát tartalmazza.

Alapértelmezés : `_pbpage = 1`.

Értékadással a kezdő oldalszám *N-kifejezésre* módosítható.

Értéktartomány: 1 – 32767, de kisebb mint a `_pepage` értéke.

Lásd még: `_pageno`, `_pepage`, PRINTJOB/ENDPRINTJOB.

_pcolno

_pcolno = < N-kifejezés >

Az éppen megjelenített karakter oszlopszámát tartalmazza, amely a megjelenítés során változik.

Értékadás után a *N-kifejezéssel* jelölt oszlopra pozicionál az aktuálisan megjelenítendő sorban.

Értéktartomány: 0 – 255.

Ha `_wrap = .T.` és SET PRINTER ON, akkor az előzőleg kinyomtatott szövegre is rá lehet nyomtatni, ha `_pcolno` az aktuálisnál kisebb értéket kap.

A `@ ..`, `@...TO`, EJECT utasításoknál hatástalan.

Lásd még: `_plineno`, `_rmargin`, `LEN()`, `PCOL()`, `REPLICATE()`, `SET PRINTER`, `TRIM()`.

_pcopies

`_pcopies = <N-kifejezés>`

A nyomtatási köteg (PRINTJOB, ENDPRINTJOB) példányszámát tartalmazza. A nyomtatás automatikusan a megadott példányszámban történik.

Alapértelmezés : `_pcopies = 1`.

Értékkadással a példányszám a *N-kifejezésre* módosítható.

Értéktartomány: 1 – 32767.

Lásd még: @, PRINTJOB/ENDPRINTJOB.

_pdriver

`_pdriver = "<nyomtatómeghajtó>"`

Az aktuális nyomtatómeghajtó nevét tartalmazza. Értékkadással ez megváltoztatható.

A nyomtatómeghajtókra a dBASE IV bőséges választékot kínál. Ezekből az üzembehelyezéskor (installáláskor) lehet kiválasztani azt a néhányat (a használt nyomtatótípusoknak megfelelőt), amelyre várhatóan szükség lesz. Ehhez az installálóprogram segítséget nyújt. A rendszerváltozóval az így kiválasztottakból lehet valamelyiket aktiválni.

Lásd még: `_ppitch`, `_pquality`, `?/??`, `set printer`.

_pecode

`_pecode = <C-kifejezés>`

A nyomtatási köteg végén a nyomtatón megjelenítendő illetve nyomtatóvezérlő karaktersorozatot tartalmaz.

Alapértelmezés : `_pecode = ""`.

Értékkadásra a nyomtatási köteg végén a *C-kifejezést* "küldi" a nyomtatóra és az ha nyomtatható karaktereket tartalmaz, ott megjelenik, illetve vezérlő karakterek esetén a nyomtató a megfelelő tevékenységet elvégzi. (Pl: állapotállítás)

A *C-kifejezés* hossza legfeljebb 255 karakter lehet.

Példa:

`_pecode = "{27}{69}" && ESC E`

Lásd még: `_pscode`, `?/??` (`STYLE` paraméter), `???`, PRINTJOB/ENDPRINTJOB.

_peject

_peject = "BEFORE"/"after"/"both"/"none"

A nyomtatási köteg előtti, utáni lapdobások vezérlőinformációit tartalmazza. Lapemelés történik *before* esetén a nyomtatási köteg előtt, *after* esetén utána, *both* esetén előtte és utána is. *None* esetén se előtte, se utána nem történik lapdobás.

Lásd még: **_padvance**, **EJECT**, **EJECT PAGE**, **PRINTJOB/ENDPRINTJOB**.

_pepage

_pepage = < N-kifejezés >

A nyomtatási köteg utolsó kinyomtatandó oldalának számát tartalmazza. Alapértelmezés : minden oldal, tehát az értéktartomány felső határa 32767. Értékadással ez korlátozható, azaz az utolsó kinyomtatandó oldal száma a *N-kifejezés* lesz.

Értéktartomány: 1 – 32767.

Lásd még: **_pageno**, **_pbpage**, **PRINTJOB/ENDPRINTJOB**.

_pform

_pform = "< nyomtatási formátumfájl >"

Az érvényes nyomtatási formátumfájl nevét tartalmazza.

Alapértelmezés : nincs ilyen, azaz **_pform** = ""

Értékadással aktivizálható egy *nyomtatási formátumfájl*, amely egy **.prf** kiterjesztésű bináris fájl, és a lényegében a nyomtatásvezérlő rendszerváltozók, nevezetesen a **_padvance**, **_pageno**, **_pbpage**, **_pcopies**, **_pdriver**, **_pecode**, **_peject**, **_pepage**, **_plength**, **_ploffset**, **_ppitch**, **_pquality**, **_pscode**, **_pspacing**, **_pwait** korábbi alkalommal beállított értékeit tartalmazza.

Lásd még: **CREATE/MODIFY LABEL**, **CREATE/MODIFY REPORT**, **LABEL FORM**, **REPORT FORM**.

_plength

_plength = < N-kifejezés >

A megjelenítendő oldal hosszát tartalmazza.

Alapértelmezés : **_plength = 66 (sor)**.

Értékadással az *N-kifejezés* lesz az új oldalhossz.

Értéktartomány: 1 – 32767.

Lásd még: **_padvance, EJECT, EJECT PAGE, ON PAGE.**

_plineno

_plineno = < N-kifejezés >

Az éppen nyomtatandó sor sorszámát tartalmazza.

Alapértelmezés : **_plineno = 0.**

Értékadással ez a *N-kifejezésre* átállítható, amely nem lehet nagyobb az oldalhossznál.

Értéktartomány: 0 – (**_plength-1**)

Lásd még: **_plength, EJECT PAGE, ON PAGE, PCOL(), PROW().**

_ploffset

_ploffset = < N-kifejezés >

Nyomtatáshoz a papír bal széléhez viszonyított vízszintes nullpontot, azaz bal alampargót tartalmazza. A margók, behúzás ettől a ponttól értelmezett.

Alapértelmezés : **_ploffset = 0.** Értéktartomány: 0 – 255.

A **_ploffset** beállításának hatása ugyanaz mint a **SET MARGIN** utasításnak, de különbözik és független a **_lmargin** változótól.

Lásd még: **_lmargin, SET MARGIN.**

_ppitch

`_ppitch = "pica"/"elite"/"condensed"/"DEFAULT"`

A nyomtatott betűk alapszélességének, sűrűségének értékét tartalmazza. Értékadással az alábbiak szerint változtatható:

<i>pica</i>	= 10 karakter/inch
<i>elite</i>	= 12 karakter/inch
<i>condensed</i>	= 16,17 karakter/inch
<i>default</i>	= a nyomtató alapértelmezési karakterszélessége

Lásd még: `_pdriver`, `_pquality`.

_pquality

`_pquality = <feltétel >`

A mindenkori nyomtatási minőségre utal.

Két minőséget különböztet meg: a normál (draft) és a szépírás minőség (Letter vagy Near Letter Quality).

Ha a *feltétel* .T. (igaz) értéket tartalmaz, akkor szépírás a nyomtatási minőség.

Alapértelmezés : `_pquality = .F.`

Lásd még: `_pdriver`, `_ppitch`.

_pscode

`_pscode = <C-kifejezés >`

A nyomtatási köteg előtt a nyomtatón megjelenítendő illetve nyomtatóvezérlő karaktersorozatot tartalmaz.

Alapértelmezés : `_pscode = ""`.

Értékadás esetén a nyomtatási köteg előtt a *C-kifejezést* "küldi" a nyomtatóra és az ha nyomtatható karaktereket tartalmaz, ott megjelenik, illetve vezérlő karakterek esetén a nyomtató a megfelelő tevékenységet elvégzi. (Pl: állapotállítás)

A *C-kifejezés* hossza legfeljebb 255 karakter lehet.

Példa:

`_pscode = "{27}{38}{108}{49}{79}" && HP-LaserJet`

Lásd még: `_pcode`, `???` (`STYLE` paraméter), `???`, `PRINTJOB/ENDPRINTJOB`.

_pspacing

`_pspacing = 1/2/3`

A nyomtatandó sorok közötti üres sorok számát tartalmazza.

Alapértelmezés : `_pspacing = 1`

Értékadással ez megváltoztatható az alábbiak szerint:

1 = nincs üres sor

2 = 1 üres sor

3 = 2 üres sor.

Lásd még: `???`, `list/display`.

_pwait

`_pwait = <feltétel>`

Ha `.T.` (igaz) értéket tartalmaz, akkor minden nyomtatott oldal után (papírbefűzéshez) várakozik.

Alapértelmezés : `_pwait = .F.`

Lásd még: `_plength`, `EJECT`, `EJECT PAGE`.

_rmargin

`_rmargin = <N-kifejezés>`

Az éppen érvényes jobb margó értékét tartalmazza, amely megváltoztatható. Értékadásnál a *N-kifejezés* lesz az új jobb margó, amelynek nagyobbak kell lennie, mint a bal margó illetve ha behúzás (`_indent`) is van akkor ezek összegénél.

Alapértelmezés : 80.

Értéktartomány: `_lmargin + 1` vagy `_lmargin + _indent + 1 - 254`.

Csak akkor van hatása, ha a `_wrap` változó értéke `.T.`.

Lásd még: `_indent`, `_lmargin`, `_ploffset`, `_wrap`.

_tabs

_tabs = <C-kifejezés>

Az aktuális tabulátorpozíciókat tartalmazza.

Értékadáskor a *C-kifejezés* tartalma vesszővel elválasztott növekvő számsorozat lehet, amely mindegyike egy-egy tabulátorpozíciót jelent. Az elemek egyike sem lehet 255-nél nagyobb.

Lásd még: **_indent**.

_wrap

_wrap = <feltétel>

Ha **.T.** (igaz) értéket tartalmaz, akkor a szövegeket csak szavanként választja el a sor végén, tehát szó közben elválasztás nem történik.

Alapértelmezés : **_wrap = .F.**

Az **_align**, **_lmargin**, **_rmargin** változók csak akkor hatnak, ha **_wrap = .T.**

Lásd még: **_align**, **_indent**, **_lmargin**, **_rmargin**.

G) SQL utasítások és függvények

G-1. SQL utasítások

ALTER TABLE

```
ALTER TABLE <táblanév>  
  ADD (<oszlopnév> <adattípus>  
  [, <oszlopnév> <adattípus> ...]);
```

Az aktuális adatbázishoz tartozó SQL tábla strukturáját módosítja. Segítségével csak új oszlopokat lehet a meglévőkhöz adni. Az új oszlop az utolsó már létező oszlop után kerül. A táblához csatolt oszlopokat később már nem lehet törölni. Az utasítás hatására az SQL rendszertáblákba bejegyzés kerül az oszloproól. PROTECT utasítással védett (titkosított) adatbázisok esetén az oszlophoz csak a rendszergazda és a tábla tulajdonosa (a létrehozó) férhet hozzá. Egy oszlopnév a táblán belül csak egyszer fordulhat elő. Az adattípus bármilyen a CREATE TABLE utasításnál leírt dBASE IV/SQL típus lehet.

Példa:

```
ALTER TABLE regitabla  
  ADD (afa NUMERIC(5,2),leiras CHAR(30));
```

Lásd még: CREATE TABLE, DROP TABLE, CREATE SYNONYM, GRANT, REVOKE.

CLOSE

```
CLOSE <sql-kurzornév>;
```

Lezár egy korábban OPEN-nel aktivizált sql-kurzort. Törli az aktivizáláskor keletkezett munkafájlt és felszabadítja a sql-kurzor által használt munkaterületet. Sikeres végrehajtása független az sql-kurzor helyétől, azaz attól, hogy az sql-kurzor végighaladt-e minden soron.

Csak programban használható.

Példa:

```
CLOSE mutato1;
```

Lásd még: DECLARE, OPEN, FETCH.

CREATE DATABASE

```
CREATE DATABASE [ <út> ] <adatbázisnév>;
```

Létrehoz egy üres adatbázist. Az adatbázis neve legfeljebb 8 karakteres lehet és általában a DOS fájlnévre vonatkozó szabályainak kell eleget tennie. Hatására az aktuális könyvtárban <adatbázisnév> néven létrejön egy alkönyvtár, amely az adatbázis leendő tábláit fogja tartalmazni. Ha kijelölésre került az <út> is, akkor az adatbázis az ennek megfelelő könyvtárban létesül. Az adatbázisnév és az út között nem állhat szóköz. Az utasítás sikeres végrehajtása esetén a SYSDBSQL rendszer-főkatalógusba kerül az adatbázis neve, a létrehozás dátuma, az adatbázis elérési útvonala, és PROTECT használata esetén az adatbázis gazdájának a neve. Az utasítás végrehajtása után az adatbázis automatikusan aktivizálódik. Csak interaktív üzemmódban használható.

Lásd még: START DATABASE, STOP DATABASE, DROP DATABASE, CREATE TABLE.

CREATE INDEX

```
CREATE [UNIQUE] INDEX <indexnév>
    ON <táblanév>
    (<oszlopnév> [ASC/DESC]
    [, <oszlopnév> [ASC/DESC] ...] );
```

A megadott adatbázisban a megadott táblához létrehozza az utasításban szereplő indexet. Nézettáblák (Virtuális tábla, view) nem indexelhetők. Az utasítás hatására a tábla sorai a megadott oszlopok értékei szerint növekvő vagy csökkenő sorrendben érhetők el. Egy táblához legfeljebb 47 index tartozhat. Az elsődleges rendezési szempontot az első oszlop határozza meg, az azon belüli rendezési szempontot a második oszlop és így tovább. Az ASC kulcsszó határozza meg a növekvő sorrendet, a DESC kulcsszó pedig a csökkenő ellenkező rendezettséget definiál. Az alapértelmezés az ASC. Az ASC és DESC kulcsszavak egy utasításon belül vegyesen nem használhatók. A rendezési kulcs, azaz az oszlopok együttes hossza a 100 karaktert nem haladhatja meg. A UNIQUE kulcsszó kizárja, hogy két vagy több azonos indexű (rendezési kulcsú) sor előfordulhasson. előírja a dBASE IV/SQL számára, hogy megadott rendezési szempont szerint a tábla minden egyes sora egyedi legyen. Ha a tábla már az utasítás kiadásakor duplikált sorokat tartalmaz, akkor az indexelést nem lehet végrehajtani.

Példa:

```
CREATE UNIQUE INDEX nevircim
    ON lakas
    (nev ASC , iranyasz ASC , cim ASC);
```

Lásd még: DROP INDEX, SELECT ORDER BY.

CREATE SYNONYM

```
CREATE SYNONYM < másodnév >  
FOR < táblanév >;
```

Az utasítás másodnévvel látja el a FOR kulcszó után megadott táblát, nézet-táblát. A másodnév bárhol használható a tábla eredeti neve helyett. Másodnévre az eredeti név formai szabályai érvényesek.

Lásd még: DROP SYNONYM, GRANT.

CREATE TABLE

```
CREATE TABLE <táblanév>
  (<oszlopnév> <adattípus>
  [, <oszlopnév> <adattípus> ...]);
```

Tábla létrehozása az aktuális adatbázisban. A tábla neve legfeljebb 8 karakteres lehet, betűvel kell kezdődnie és betűket, számokat, aláhúzás-jelet tartalmazhat. A tábla fizikailag az aktuális adatbázis által kijelölt könyvtárban jön létre. A tábla szerkezetét a zárójelek között felsorolt oszlopok határozzák meg.

Az oszlopnév legfeljebb 10 karakter hosszú lehet, a többi szabályok megegyeznek a táblanévre vonatkozokkal. Egy táblán belül két azonos nevű oszlop nem fordulhat elő. A létrehozás után a tábla oszlopait törölni, típusát módosítani nem lehet.

A dBASE IV/SQL adattípusai:

SMALLINT	Kis egész számok. Legfeljebb 6 számjegyből állhat, beleértve az előjelet. A legkisebb szám a -99999, a legnagyobb a 999999.
INTEGER	Egész számok. Legfeljebb 11 számjegy az előjellel együtt. Intervalluma: -9999999999 - 9999999999.
DECIMAL(n,m)	n hosszúságu, m tizedesjegyű fixpontos decimális számok. A szám hosszába beleszámít az előjel is). n 1-től 19-ig változhat, m 0 és 18 között lehet. Például DECIMAL(5,2) értéke -99.99 és 999.99 között lehet.
NUMERIC(n,m)	n hosszúságu, m tizedesjegyet tartalmazó fixpontos decimális adattípus. A szám hosszába beleszámít az előjel és a tizedespont is. n 1 és 20, m pedig 0 és 18 között állhat. Például NUMERIC(5,2) értéke -9.99 és 99.99 lehet.
FLOAT(n,m)	Lebegőpontos számok. Az adat n hosszban, m tizedesjeggyel jelenik meg. A szám megjelenési hosszába beleszámít az előjel és a tizedespont. Az n 1 és 20, az m pedig m 0 és 18 között állhat. A lebegőpontos számok $0,1E-307$ és $0,9E+308$ között lehetnek. Ha a szám hagyományos módon nem jeleníthető meg n pozícióban akkor a kijelzés lebegőpontos formában történik meg. Ha lebegőpontos formában sem jeleníthető meg, akkor helyette csillagokat láthatunk a mezőben.
CHAR(n)	Karakteres adat n hosszal. n 1-től 254-ig terjedhet. Nagy és kis betűket, számjegyeket, írásjeleket (például !,;,., stb.) és általában bármely nyomtatható karaktert tartalmazhat.

DATE	Naptári dátumok. Hossza 8 bájt. A megjelenési formája hó/nap/év, például 12/31/89 illetve 12/31/1989. A megjelenési formát a SET DATE, SET CENTURY, SET MARK utasítások szabályozzák. Dátum típusú adatot a CTOD() függvény illetve a {} zárójel pár segítségével lehet előállítani, például {"12/31/89"}, CTOD("12/31/89"). Értékadásnál ha az évszázad hiányzik, akkor a dBASE IV/SQL automatikusan 19-et szúr be az év elé.
LOGICAL	Logikai (kétértékű) adat. Értékei lehetnek: .T., .t., .Y., .y. (igaz); .F., .f., .N.,.n. (hamis). A mezőbe pont nélkül csak a jellemző betűk kerülnek.

Példa:

```
CREATE TABLE ujtabla
(kodszam SMALLINT,
fonokkod SMALLINT,
irszam NUMERIC(4,0),
mhely CHAR(30),
nev CHAR(30),
fizetes DECIMAL(8,2),
fizdat DATE,
fizetve LOGICAL);
```

```
CREATE TABLE mhely
(kodszam SMALLINT,
irany NUMERIC(4,0),
teleph CHAR(25),
osztaly CHAR(20),
osztkod NUMERIC(3,0),
szoba DECIMAL(3,0));
```


CREATE VIEW

```
CREATE VIEW <nézettáblanév>
  [(<oszloplista>)]
  AS <select mondat>
  [WITH CHECK OPTION];
```

Egy új nézettáblát (virtuális táblát) állít elő. A nézettáblában az AS kulcsszó után szereplő select mondatból kiválasztott sorok vesznek részt. A nézettábla oszlopai a select mondatban felsorolt oszlopokkal lesznek azonosak. Az oszloplistával új nevet lehet adni a nézettábla oszlopainak. A WITH CHECK OPTION esetén minden művelet (INSERT, UPDATE) esetén ellenőrzésre kerülnek a kiválasztó feltételek, és a feltételnek nem megfelelő sorok nem kerülhetnek a táblába.

A select mondat nem tartalmazhatja az INTO, UNION, ORDER BY, FOR UPDATE OF, SAVE TO TEMP cikkelyket.

Példa:

```
CREATE VIEW nagyfizu
  (nagynev, nfizetes)
  AS SELECT nev fizetes
  FROM ujtabla
  WHERE fizetes > 5000
  WITH CHECK OPTION;
```

Lásd még: SELECT.

DBCHECK

```
DBCHECK [<táblanév>];
```

Ellenőrzi a dBASE IV/SQL rendszer táblák tartalmának helyességét az aktuális adatbázisra vonatkozóan. Ha a táblák fizikai állapota eltér a rendszer táblákban levő információktól akkor hibajelzést ad. Ha táblanév szerepel az utasításban, akkor a konzisztenciavizsgálat csak az adott táblára történik meg.

DBDEFINE

```
DBDEFINE [<adatfájlnév>];
```

Már létező dBASE IV adatfájlt tesz elérhetővé SQL módban. Hatására az aktuális adatbázis könyvtárában lévő .dbf és a hozzátartozó .mdx fájlokat csatolja az aktuális adatbázishoz és a rendszertáblákban ezt adminisztrálja. Ha szerepel a .dbf fájlnev az utasításban, akkor csak az adott adatfájl lesz az SQL adatbázis része.

DECLARE CURSOR

```
DECLARE <sql-kurzornév> CURSOR
  FOR <select mondat>
  [FOR UPDATE OF <oszloplista>/ORDER BY <cikkely>];
```

A rekordmutató (sql-kurzor) feltételrendszerét határozza meg. A rekordmutató csak a meghatározott feltételeknek megfelelő rekordra mutathat. Az ORDER BY cikkely esetén a sql-kurzor a tábla sorain a megadott rendezési szempont szerint fog végigjárni. Az ORDER BY cikkely leírását lásd SELECT ORDER BY részben. Ha a FOR UPDATE OF cikkely hiányzik az utasításból, akkor a kiválasztott sorok nem módosíthatók. A későbbiekben csak azok az oszlopok változtathatóak meg UPDATE utasítással amelyek neve szerepel a kulcsszó után. Az oszlopok neveit vesszővel kell elválasztani. A FOR UPDATE OF és az ORDER BY cikkelyek kölcsönösen kizárják egymást.

A select mondatban nem szerepelhet SAVE TO TEMP és INTO cikkely. Csak programban használható.

Példa:

```
DECLARE mutato CURSOR
  FOR SELECT nev, fizetes, fizetve
  WHERE fizetes > 10000
  FOR UPDATE OF fizetes, fizetve ;
```

Lásd még: OPEN, CLOSE, DELETE FROM, UPDATE.

DELETE

Több sor törlése

```
DELETE FROM <táblanév>  
  [WHERE cikkely];
```

Törli a táblából a WHERE cikkelyben megadott feltételnek eleget tevő sorokat. Alap esetben a kijelölt sorok végleg eltűnnek a táblából. Törölt sorok helyreállítása a ROLLBACK utasítással lehetséges. A WHERE cikkely leírása a SELECT utasításnál található meg.

Példa:

```
DELETE FROM ujtabla  
  WHERE fizetve;
```

Lásd még: ROLLBACK, SELECT WHERE, UPDATE.

Egy kijelölt sor törlése

```
DELETE FROM <táblanév>  
  WHERE CURRENT OF <sql-kurzornév>;
```

Az sql-kurzor által kijelölt sort törli a táblából. Az utasítás kiadásakor az sql-kurzornak mutatnia kell valamelyik sorra, ezért a DELETE utasítást meg kell előznie egy DECLARE CURSOR, egy OPEN, és egy vagy több FETCH utasításnak. Ha a DECLARE CURSOR utasítás nem tartalmazza a FOR UPDATE OF cikkelyt a sorok törlése nem lehetséges. A sor törlése után a sql-kurzor nem mutat egyik sorra sem, ezért egy újabb FETCH utasítással aktualizálni kell. A törölt sor csak BEGIN TRANSACTION használata esetén állítható helyre. Csak programban használható.

Lásd még: DECLARE CURSOR, OPEN, FETCH, CLOSE, ROLLBACK.

DROP DATABASE

```
DROP DATABASE <adatbázisnév>;
```

Törli a megadott adatbázist. Az adatbázisnak az utasítás kiadásakor nem lehet aktív. Az adatbázis törlése az adatbázishoz tartozó összes tábla, index, nézettábla (view), másodnév törlését is jelenti. Ennek megfelelően módosulnak a rendszertáblák is. Ha a SET SAFETY ON állásban van, akkor a rendszer a törlés végrehajtása előtt külön jóváhagyást kér.

DROP INDEX

DROP INDEX <indexnév>;

Törli a megadott nevű indexet az adatbázisból. A törlés az utasítás kiadása után azonnal megtörténik.

DROP SYNONYM

DROP SYNONYM <másodnév>;

Az utasításban megadott másodnevet kitörli az adatbázisból.

DROP TABLE

DROP TABLE <táblanév>;

Törli az utasításban megadott táblát az adatbázisból. A törlés az utasítás kiadása után azonnal megtörténik.

DROP VIEW

DROP VIEW <nézettáblanév>;

Törli az utasításban megadott nézettáblát az adatbázisból. A nézettábla törlése azonnal megtörténik.

FETCH

FETCH <sql-kurzornév>
INTO <változólista>;

Az előzőleg **DECLARE CURSOR** és **OPEN** utasításokkal aktivizált sql-kurzort a következő – a kurzorfeltételeknek megfelelő – sorra viszi. Az így aktuálisá vált sor oszlopainak tartalmát az **INTO** kulcsszó után felsorolt változókba teszi. A változók számának és az oszlopok számának egyeznie kell.

Példa:

```
FETCH mutato
      INTO msorszam, mnev1, mfizetes, mfizdat, mfizetve;
```

Lásd még: **DECLARE CURSOR**, **OPEN**, **CLOSE**, **DELETE WHERE CURRENT OF**

GRANT

```
GRANT ALL [PRIVILEGES]/<jogosultságlista >
  ON [TABLE] <táblalista >
  TO PUBLIC/<felhasználólista >
  [WITH GRANT OPTION];
```

A táblákra vonatkozó hozzáférési jogokat határozza meg. A meghatározott jogok az alaptáblákból készült nézettáblákra is vonatkoznak függetlenül attól, hogy a hivatkozás azokra eredeti, vagy másodnevükön történik.

Csak akkor használható, ha előtte a PROTECT utasítás végrehajtása megtörtént.

Az ALL opció az összes jog átadását jelenti. A PRIVILEGES kulcsszó elhagyható.

A jogosultságlista segítségével leszűkíthető az átadott jogok köre. A jogosultságlista lehetséges elemei:

kulcsszó	jogkör
ALTER	ALTER utasítás használatával új oszlopok hozzáadása a táblához.
DELETE	Sorok törlése a táblából vagy nézettáblából.
INDEX	Index létrehozása a táblán.
INSERT	Új sorokat bevitele (hozzáadása) a táblához.
SELECT	A tábla vagy nézettábla használata SELECT utasítással.
UPDATE [(\langle oszloplista \rangle)]	A tábla sorainak módosítása. Ha az oszloplista hiányzik, akkor a tábla összes oszlopát lehet módosítani, egyébként csak az oszlop listában szereplő oszlopokat.

G-1. táblázat

Az ON TABLE kulcsszó után sorolhatók fel a táblák, amelyekre a jogok vonatkoznak. Egy táblára használatára csak a tábla létrehozója és a rendszergazda adhat jogokat. A táblák neveit vesszővel kell elválasztani.

A TO kulcsszó után a jogokkal felruházott felhasználók névsora következik, vagy a PUBLIC opció, amely nyilvánossá teszi a felsorolt jogokat, tehát azokat bárki gyakorolhatja.

A WITH GRANT OPTION lehetőséget ad az éppen feljogosított felhasználónak, hogy jogait más felhasználókra is átruházhassa.

Példa:

```
GRANT ALL
  ON TABLE ujtabla
  TO joska
  WITH GRANT OPTION;
```

```
GRANT INDEX, SELECT, INSERT, UPDATE(fizetes,fizetve,fizdat)
  ON TABLE ujtabla
  TO PUBLIC;
```

Lásd még: REVOKE, PROTECT.

INSERT**Sor hozzáadása a táblához értéklistánál**

```
INSERT INTO <táblanév>
    [(<oszloplista>)]
    VALUES (<értéklista>);
```

Egy új sort ad a táblához. A sor oszlopainak tartalmát az értéklistán felsorolt kifejezések – amelyekben konstansok, változók és dBASE IV függvények egyaránt szerepelhetnek – eredményértékével tölti fel. Ha az oszloplista elmarad, akkor a tábla összes oszlopa értéket kap. A felsorolt értékeknek az oszloplista, illetve ennek hiányában a tábla oszlopainak sorrendjét kell követniük. Az oszlopok számának és az értéklista elemei számának egyeznie kell.

Oszloplista esetén csak a listában felsorolt oszlopok lesznek értékkel feltöltve. Azok az oszlopok, amelyek a listában nem szerepelnek, a következő (üres) értékeket kapják:

típus	érték
CHAR (n)	SPACE(n)
SMALLINT	0
INTEGER	0
DECIMAL()	0
NUMERIC()	0
FLOAT()	0
DATE	{ / / }
LOGICAL	.F.

G-2. táblázat

Ahol

n a karakter típusú oszlop hossza.

{, } a dátum típust jelölő kapcsos-zárójel pár, de nem része az értéknek.

Példa:

```
INSERT INTO ujtabla
    (sorszam, nev)
    VALUES (1111,"Nagy Sandor");
```


Sor hozzáadása a táblához más táblából

```
INSERT INTO <táblanév>
  [(<oszloplista>)]
  <select mondat>;
```

A select mondat által kijelölt sorokat illeszti a táblába végéhez. A sor oszlopainak tartalmát a select mondatban felsorolt kifejezések – amelyekben konstansok, változók és dBASE IV függvények egyaránt szerepelhetnek – eredményértékével tölti fel. Ha az oszloplista elmarad, akkor a tábla összes oszlopa értéket kap. A select mondatban felsorolt értékeknek az oszloplista, illetve ennek hiányában a tábla oszlopainak sorrendjét kell követniük. Az oszlopok számának és a select mondat értékelemei számának egyeznie kell. Azok az oszlopok, amelyek a listában nem szerepelnek, a korábbi táblázatnak megfelelő (üres) értékeket kapnak. A select mondatban nem állhat INTO TEMP, ORDER BY cikkely.

Lásd még: SELECT.

LOAD DATA

```
LOAD DATA FROM [<út>] <fájlnev>
  INTO <táblanév>
  [[TYPE] SDF/DIF/WKS/SYLK/FW2/RPD/DBASEII
  /DELIMITED [WITH BLANK/ <határoló>];
```

Külső - nem dBASE IV/SQL - fájlból tölt rekordokat SQL táblába. Ha a TYPE cikkely elmarad, dBASE IV adatfájlt olvas. dBASE adatfájlok esetén csak a tábla oszlopneveivel egyező nevű mezők kerülnek átmásolásra. A memo mezőket figyelmen kívül hagyja. Más típusú fájlknál az oszlopok számának egyeznie kell a forrásfájl oszlopainak (mezőinek) számával. Az utasítás működése azonos dBASE IV APPEND FROM utasítással.

Lásd még: APPEND FROM, UNLOAD DATA.

OPEN

```
OPEN <sql-kurzornév>;
```

A DECLARE utasítással kijelölt sql-kurzort aktivizálja. Hatására az sql-kurzor a DECLARE utasítás feltételeivel egyező első sor elé áll, de arra csak a FETCH utasítás után mutat.

Csak programban használható.

Lásd még: DECLARE, FETCH, CLOSE

REVOKE

```
REVOKE ALL [PRIVILEGES]/ <jogosultságlista >  
ON [TABLE] <táblalista >  
FROM PUBLIC/ <felhasználólista >;
```

Jogosultságokat von vissza a felhasználotól. A GRANT ellentétpárja. A jogok visszavonása az ON kulcsszó után felsorolt táblákra vonatkozik. Az ALL PRIVILEGES kulcsszó az összes jogot, a jogosultságlista csak a listában felsorolt jogokat szünteti meg. A jogok és a jogokat reprezentáló kulcsszavak a lásd a GRANT utasításnál.

PUBLIC esetén mindenkitől, felhasználólista esetén csak a megemlített felhasználotól kerülnek visszavonásra a különböző hozzáférési jogok. A visszavont jogok azok számára is megszűnnek, akikre a felhasználó azokat átruházta (WITH GRANT OPTION). Csak akkor működik, ha a dBASE IV védelmi rendszere aktív (aktivizálás: PROTECT utasítás).

Példa:

```
REVOKE ALL PRIVILEGES  
ON TABLE ujtabla  
FROM joska ;  
  
REVOKE INDEX, SELECT, INSERT  
ON TABLE ujtabla  
FROM PUBLIC;
```

Lásd még: GRANT, PROTECT.

ROLLBACK

```
ROLLBACK [WORK];
```

Az adatbázist visszaállítja a BEGIN TRANSACTION utasítás kiadásakor állapotába. A WORK kulcsszónak nincs funkciója. Működése azonos a ROLLBACK dBASE IV utasítással.

Lásd még: dBASE IV: BEGIN TRANSACTION, END TRANSACTION, ROLLBACK, COMPLETED()

RUNSTATS

RUNSTATS [<táblanév >];

Aktualizálja az adatbázis statisztikai információit a rendszertáblákban. Az aktualizálás ajánlatos, ha új táblát vagy indexet jön létre, vagy a tábla sorainak tíz százaléka megváltozik. Ellenkező esetben az adatbázisműveletek végrehajtási ideje lecsökken.

Ha táblanév szerepel, akkor csak a megadott tábla információit frissíti fel.

Lásd még: CREATE TABLE, CREATE INDEX.

SELECT**Szintaktika:**

```
SELECT [ALL/DISTINCT] {*/<kifejezéslista >}
      [INTO <változólista >]
      FROM <táblalista >
      [WHERE <feltétel >]
      [GROUP BY <oszloplista >
        [HAVING <feltétel >]]
      [UNION <select mondat >]...
      [ORDER BY <oszlopnév >/<oszlop sorszám > [ASC/DESC]
        [, <oszlopnév >/<oszlopsorszám > [ASC/DESC]...]]
      [FOR UPDATE OF <oszloplista >]]
      [SAVE TO TEMP <táblanév > [( <oszloplista > )]] [KEEP]]
```

A táblákból a kívánt információk kiválasztására szolgál.

Ha önmagában áll, akkor a kiválasztott információkat a képernyőn is megjeleníti.

Más utasítások és utasításcikkelyek részeként is alkalmazható. Ilyenkor csak a kívánt adatokat bocsátja a befoglaló utasítás rendelkezésére, de a képernyőn az adatokat nem jeleníti meg. (Más utasítások részeként a "select mondat" kifejezéssel utalunk rá.)

A CREATE VIEW, WHERE vagy UNION részeként nem tartalmazhatja az INTO, UNION, ORDER BY, FOR UPDATE OF, SAVE TO TEMP cikkelyeket.

Egyaránt használható interaktív üzemmódban és programban.

Cikkelyei működését lásd a továbbiakban.

SELECT - SELECT cikkely

SELECT [ALL/DISTINCT] {*/<kifejezéslista> }

Kiválasztja az utasításban résztvevő oszlopokat, illetve kifejezések esetén úgynevezett logikai oszlopokat határoz meg. A * alkalmazása esetén a tábla minden oszlopa kiválasztásra kerül. A kifejezésekben csak olyan oszlopokra történhet hivatkozás, amelyek a FROM cikkelyben felsorolt táblákhoz tartoznak.

Gyűjtőfüggvények (AVG, SUM, stb) és oszlopok nem szerepelhetnek vegyesen a SELECT cikkelyben. Ez alól a szabály alól kivétel, ha a SELECT utasítás GROUP BY cikkelyt is tartalmaz.

Ha a GROUP BY cikkely része az utasításnak, akkor csak azokra az oszlopokra történhet hivatkozás a kifejezésekben, amelyek a GROUP BY cikkelyben is szerepelnek. Kivételt képeznek a gyűjtőfüggvények (AVG, SUM, stb), amelyekben más oszlopokra lehet hivatkozni. Ilyenkor a kifejezésekben alkalmazott gyűjtőfüggvények nem a tábla (táblák) összes sorain, hanem csak a GROUP BY cikkely által csoportosított duplikált sorokon operálnak, ott is a csoporton belül. Ha két táblában azonos nevű oszlopok vannak - az oszlop nevét ki kell egészíteni a tábla nevével vagy másodnevével. A táblanevet az oszlop neve elé kell írni és közéjük pontot kell tenni.

Példa:

```
SELECT ujtabla.oszlopa, regitabl.oszlopa;
```

A DISTINCT opcióval a duplikált (a kiválasztott oszlopok mindegyikében azonos értékeket tartalmazó) sorok nem kerülnek kiválasztásra.

ALL esetében az utasítás minden sort figyelembe vesz. Ez az alapértelmezés.

Példa:

```
SELECT * FROM ujtabla;
```

```
SELECT DISTINCT kodsza, nev, "Havi fizetese:",  
                fizetes, "      Eves fizetese:", fizetes*12  
FROM ujtabla;
```

Lásd még: SELECT FROM cikkely, SELECT WHERE cikkely, SQL függvények.

SELECT - INTO cikkely**INTO <változólista >**

A kiválasztott sor tartalmát a változólistában felsorolt változóba tölti. Az oszlopok és a változók számának egyeznie kell. A változók felveszik a hozzájuk tartozó oszlop adattípusát.

Csak akkor működik helyesen, ha a keresési feltétel (WHERE cikkely) egyetlen sort választ ki.

Csak programban használható.

Példa:

```
SELECT DISTINCT kodsza, nev, fizetes, fizetve
      INTO mkod, mnev, mfizu, mokey
      FROM ujtabla
      WHERE kodsza = 71976;
```

Lásd még: SELECT SELECT cikkely, SELECT WHERE cikkely, SELECT FROM cikkely.

SELECT - FROM cikkely**FROM <táblanév > [<másodnév >] [, <táblanév > [<másodnév >] ...]**

A résztvevő táblákat határozza meg. Csak ezeknek a tábláknak az oszlopaire lehet hivatkozni az utasítás más részeiben.

Ugyanaz a tábla többször is szerepelhet a cikkelyben, ilyenkor az ismétlődő táblát mindig más és más másodnévvel kell megkülönböztetni egymástól. A másodnév a tábla nevét követi és egy szóköz választja el őket. A másodnevek a táblanevek szabályai szerint képezhetők, de nem kezdődhetnek A - J betűkkel. Ez a másodnév nem azonos a SYNONIM utasításban kijelölt másodnevekkel!

Példa:

```
SELECT ujtabla.nev ujtabla.mhely
      FROM ujtabla;

SELECT u1.nev, " főnöke: ", u2.nev
      FROM ujtabla u1, ujtabla u2
      WHERE u1.fonokod = u2.kodsza
```

Lásd még: SELECT SELECT cikkely, SELECT WHERE cikkely.

SELECT - WHERE cikkely

WHERE <feltétel >

A táblák a megadott feltételnek megfelelő sorait választja ki.

Ha a cikkely elmarad, az utasítás a táblák összes sorát figyelembe veszi.

Opcionális (elhagyható) cikkely.

Példa:

```
SELECT *  
  FROM ujtabla  
  WHERE fizetve;
```

```
SELECT *  
  FROM ujtabla u1  
  WHERE u1.fizetes >  
        (SELECT AVG(u2.fizetes)  
         FROM ujtabla u2  
         WHERE u2.fizetve)  
  ORDER BY u1.kodszam;
```

```
SELECT nev, ujtabla.kodszam, teleph, osztaly, szoba  
  FROM ujtabla, mhely  
  WHERE ujtabla.kodszam = mhely.kodszam;
```

Lásd még: SELECT - GROUP BY cikkely, SELECT - FROM cikkely

SELECT - GROUP BY cikkely

GROUP BY <oszloplista> [HAVING <feltétel>]

Az oszloplistában megadott oszlop(ok) szerint duplikált (mindegyik oszlopban azonos értéket tartalmazó) sorokat egy sorrá vonja össze. Az oszloplistában számított (logikai) oszlop nem szerepelhet. Csak akkor működik helyesen, ha a tábla a cikkelyben szereplő oszlopok szerint rendezett.

Az oszloplistában szereplő oszlopoknak a SELECT cikkelyben is szerepelniük kell, és a SELECT cikkelyben más oszlophivatkozás nem is szerepelhet a gyűjtőfüggvények (AVG, SUM, stb) kivételével. A SELECT cikkelyben lévő gyűjtőfüggvények a GROUP BY által csoportosított duplikált sorok adataival operálnak, a műveleteket egy-egy csoporton belül (a csoport sorain) végzik el. A SELECT cikkelyben a fentiekén kívül csak a a GROUP BY által képzett csoportoktól független kifejezések állhatnak.

A HAVING cikkely a duplikált sorok csoportosításából keletkezett új sorok közül választja, szűri ki a megadott feltételnek megfelelőket. Működése hasonló a WHERE cikkelyéhez.

Példa:

```
SELECT teleph, "telephelyen dolgozók száma:", COUNT(*)
      FROM mhely
      GROUP BY teleph
      HAVING NOT (COUNT(*) = 0)
      ORDER BY teleph;
```

Lásd még: SQL függvények, SELECT - SELECT cikkely, SELECT - WHERE cikkely, SELECT - ORDER BY cikkely.

SELECT - UNION cikkely

UNION <select mondat > [UNION <select mondat > ...]

Egy táblába egyesít két vagy több különböző táblát. Az esetleges duplikált sorokat kihagyja az eredmény táblából.

Az új tábla és az egyesítendő táblák oszlopai számának, típusának és hosszának egyeznie kell. A lebegőpontos típus kivételével a numerikus típusok keverhetők, ha a hosszuk és a tizedesjegyek száma azonos.

Az UNION-hoz tartozó select mondat nem tartalmazhatja az INTO, ORDER BY, FOR UPDATE OF, SAVE TO TEMP és UNION cikkelyeket. Az ORDER BY cikkelynél oszlophivatkozásként csak egész szám alkalmazható.

Példa:

```
SELECT irszam, mhely
   FROM ujtabla
      UNION
SELECT irany, teleph
   FROM mhely
      UNION
SELECT irányitosz, varos
   FROM rendeló
ORDER BY 2;
```

Lásd még: SELECT ORDER BY cikkely.

SELECT - ORDER BY cikkely

ORDER BY <oszlopnév >/<oszlop sorszám > [ASC/DESC]
[, <oszlopnév >/<oszlop sorszám > [ASC/DESC]...]

Az eredmény táblát a megadott oszlopok szerint rendezi.

Az oszlop neve helyett szerepelhet az oszlop száma. A szám az oszlopnak a SELECT cikkelyben elfoglalt helyét jelenti. Az ASC opció növekvő sorrendet, a DESC pedig csökkenőt határoz meg. Az alapértelmezés az ASC. A CREATE INDEX utasítástól eltérően megengedett a csökkenő és a növekvő sorrend vegyes alkalmazása.

A rendezési kulcsként megadott oszlopok összes hossza nem haladhatja meg a 100 karaktert. Logikai típusú oszlopot tilos rendezni.

Példa:

```
SELECT kodsza, nev, fonokod
   FROM ujtabla
ORDER BY kodsza ASC, 3 DESC;
```

Lásd még: CREATE INDEX.

SELECT - FOR UPDATE OF cikkely**FOR UPDATE OF <oszloplista >**

A DECLARE utasítás select mondatában meghatározza a módosítható oszlopokat illetve lehetővé teszi sorok törlését. A módosítás és törlés csak az sql-kurzor aktív állapotában azon a soron lehetséges, amelyikre éppen mutat.

Csak az sql-kurzor kijelölésénél (DECLARE CURSOR) hatásos. A FOR UPDATE OF cikkellyel nem szerepelhetnek együtt az ORDER BY és SAVE TO TEMP cikkelyekkel.

Példa:

```
DECLARE mutato CURSOR
      FOR SELECT *
          FROM ujtabla
      FOR UPDATE OF fizetes, fizdat, fizetve.
```

Lásd még: DECLARE CURSOR, OPEN, FETCH, CLOSE.

SELECT - SAVE TO TEMP cikkely**SAVE TO TEMP <táblanévv> [(<oszloplista >)] [KEEP]**

Kitölti (elmenti) a SELECT utasítással meghatározott táblát egy munkafájlba. A táblára a TEMP kulcsszó után álló névvel lehet a továbbiakban hivatkozni.

Oszloplista használata esetén csak a felsorolt oszlopok kerülnek az ideiglenes táblába, egyébként az összes oszlop.

A munkafájl az SQL üzemmódban kikapcsolásakor törlésre kerül, ha a KEEP opció a cikkelyben nem szerepel.

Az ideiglenes tábla dBASE IV programban adatfájlként (.dbf) használható. A megőrzött (KEEP) munkatábla SQL módban ismételten csak a DBDEFINE utasítás segítségével válik használhatóvá.

Példa:

```
SELECT *
      FROM ujtabla
      WHERE fonokkod < 100
      SAVE TO TEMP fejesek (kodsam, nev, fonokkod);
```

Lásd még: DBDEFINE.

SHOW DATABASE

```
SHOW DATABASE;
```

Kilistázza a képernyőre a létező adatbázisokat.

Lásd még: CREATE DATABASE, DROP DATABASE.

START DATABASE

```
START DATABASE <adatbázisnév>;
```

Aktivizálja a megadott adatbázist és ezzel egyidejűleg lezárja a korábban aktív adatbázist (ha volt ilyen). Törli az előző adatbázishoz tartozó ideiglenes munkafájlokat.

Az utasítás végrehajtása után csak az aktív adatbázishoz tartozó táblák, nézet-táblák, oszlopok, stb. használhatók.

STOP DATABASE

```
STOP DATABASE;
```

Lezárja az aktuális adatbázist az ideiglenes fájlok törlésével együtt.

UNLOAD DATA

```
UNLOAD DATA TO [<út>] <fájlnev>  
FORM <táblanév>  
[[TYPE] SDF/DIF/WKS/SYLK/FW2/RPD/DBASEII  
/DELIMITED [WITH BLANK/<határoló>]];
```

A megadott táblát megadott formátumúvá átalakítja át.

Ha a formátum hiányzik dBASE adatfájlt hoz létre. dBASE fájlok esetén a tábla oszlopainak megfelelő mezőtípusokat hoz létre. Az UNLOAD DATA utasítás működése azonos dBASE IV COPY TO utasítással.

Példa:

```
UNLOAD DATA TO \VISICALC\DATA\SQLDB4  
FROM ujtabla  
TYPE DIF;
```

Lásd még: COPY TO, LOAD DATA.

UPDATE**Tábla több sorának módosítása**

```
UPDATE <táblanév>
  SET <oszlopnév> = <kifejezés>
  [, <oszlopnév> = <kifejezés> ...]
  [WHERE <feltétel>];
```

A tábla sorainak tartalmát módosítja. Csak azok a sorok kerülnek módosításra, amelyek a WHERE cikkelynél szereplő feltételt kielégítik. A kifejezés és a hozzátartozó oszlop típusának egyezni kell.

Példa:

```
UPDATE ujtabela
  SET fizetes = fizetes + m->fizu, fizdat = DATE()
  WHERE fizetes < 4000;
```

Lásd még: SELECT WHERE.

Sql-kurzor által kijelölt sor módosítása

```
UPDATE <táblanév>
  SET <oszlopnév> = <kifejezés>
  [, <oszlopnév> = <kifejezés> ...]
  WHERE CURRENT OF <sql-kurzornév>;
```

A tábla egy sql-kurzor által kijelölt sorának tartalmát módosítja. Csak azok az oszlopok módosíthatók, amelyek a DECLARE CURSOR utasítás FOR UPDATE OF cikkelyében szerepelnek. A kifejezés és a hozzátartozó oszlop típusának egyezni kell.

Csak programban használható.

Példa:

```
UPDATE ujtabela
  SET fizetes = fizetes + m->fizu,
  fizdat = DATE(),
  fezetve = .T.
  WHERE CURRENT OF mutato ;
```

Lásd még: FETCH, OPEN, DECLARE CURSOR.

G-2. SQL függvények

AVG

AVG([ALL/DISTINCT] < numerikus oszlopnév >
/[ALL] < numerikus oszlop kifejezés >)

A kiválasztott sorok egy adott oszlopának az átlagát adja vissza.

ALL az alapértelmezés, ekkor az összes kijelölt sor résztvesz a számításban. DISTINCT esetén az ismétlődő értékek kimaradnak a műveletből. Ha a SELECT cikkelyben áll, akkor minden más a SELECT cikkelyben szereplő kifejezésnek csak gyűjtőfüggvénynek vagy (a GROUP BY cikkelyben is szereplő) oszlop lehet.

SELECT és HAVING kulcsszavak után használatos.

Példa:

```
SELECT AVG(fizetes), nev
      FROM ujabla
      GROUP BY nev;
```

Lásd még: SELECT GROUP BY cikkely, COUNT, SUM, MIN, MAX.

COUNT

COUNT(*/[DISTINCT] < oszlopnév >)

A kiválasztott sorok számát adja vissza. Csillag vagy oszlop DISTINCT nélküli használata esetén az összes kijelölt sor résztvesz a számításban. DISTINCT esetén az ismétlődő értékeket tartalmazó sorok kimaradnak a műveletből.

Ha a SELECT cikkelyben áll, akkor minden más a SELECT cikkelyben szereplő kifejezésnek csak gyűjtőfüggvénynek vagy (a GROUP BY cikkelyben is szereplő) oszlop lehet.

Csak SELECT és HAVING kulcsszavak után használatos.

Példa:

```
SELECT COUNT(*), nev
      FROM ujabla
      GROUP BY nev;
```

Lásd még: SELECT GROUP BY cikkely, AVG, SUM, MIN, MAX.

MAX

MAX([ALL/DISTINCT] <oszlopnév>/[ALL] <oszlop kifejezés>)

A kiválasztott sorok közül az oszlop vagy oszlopkifejezés legnagyobb értékét adja vissza. A függvény argumentuma karakter, dátum, és numerikus típus lehet. Az ALL és a DISTINCT kulcsszavak azonos módon viselkednek, mert az ismétlődő értékek léte vagy hiánya a legnagyobb értékre nincs befolyással.

Ha a SELECT cikkelyben áll, akkor minden más a SELECT cikkelyben szereplő kifejezésnek csak gyűjtőfüggvénynek vagy (a GROUP BY cikkelyben is szereplő) oszlop lehet. Csak SELECT és HAVING kulcsszavak után használható.

Példa:

```
SELECT MAX(fizetes), nev
      FROM ujtabla
      GROUP BY nev
      HAVING MAX(fizetes) > 7000;
```

Lásd még: SELECT GROUP BY cikkely, AVG, SUM, MIN, COUNT.

MIN

MIN([ALL/DISTINCT] <oszlopnév>/[ALL] <oszlop kifejezés>)

Működése azonos MAX függvénnyel, azzal a különbséggel, hogy a kiválasztott értékek közül a legkisebbet adja vissza.

Lásd még: MAX (előző rész).

SUM

SUM([ALL/DISTINCT] <numerikus oszlopnév>/[ALL] <numerikus oszlop kifejezés>)

A kiválasztott sorok adott oszlopának összegét adja vissza. Az ALL opció esetén az összes sor résztvesz a számításban. DISTINCT esetén az ismétlődő értékeket tartalmazó sorok kimaradnak a műveletből. Alapértelmezés: ALL.

Ha a SELECT cikkelyben áll, akkor minden más a SELECT cikkelyben szereplő kifejezésnek csak gyűjtőfüggvénynek vagy (a GROUP BY cikkelyben is szereplő) oszlop lehet. Csak SELECT és HAVING kulcsszavakban használható.

Példa:

```
SELECT SUM(fizetes*12), nev
      FROM ujtabla
      GROUP BY nev;
```

Lásd még: SELECT GROUP BY cikkely, AVG, SUM, MIN, COUNT.

H) Hibaüzenetek

Ez a függelék ABC sorrendben tartalmazza az összes dBASE IV hibaüzenetet. Minden üzenet mellett feltüntettük a hozzá kapcsolódó számot is. Ezek a számok nem jelennek meg a képernyőn, de programok nyomkövetésekor hasznosak lehetnek az ON ERROR parancs illetve az ERROR() függvény argumentumai-ként.

A szám nélküli hibaüzenetek a Control Center-ből erednek, s ezeket nem lehet nyomkövetéskor használni.

A kövéren szedett sorok tartalmazzák az eredeti angol nyelvű üzeneteket, alattuk pedig rövid, magyar nyelvű magyarázatuk olvasható.

A replace query must have at least one WITH operator [370]

Módosító nézetet (Update query) WITH cikkely nélkül nem lehet elindítani.

A UNIQUE aggregate must be the only aggregate in a query [324]

Több fájlból kapcsolt nézetnél csak egy UNIQUE (egyedi, nem duplikált) meghatározást tartalmazhat.

ACOS(): Out of range [293]

Az arkusz koszinusz függvény argumentuma kívül esik a [-1, +1] intervallumon.

ALIAS expression not in range [232]

Rossz munkaterületre történt hivatkozás. A kifejezés kívül esik az 1 – 10 intervallumon.

ALIAS name already in use [24]

A másodnévvel már másik munkaterületen létezik megnyitott adatfájl, vagy a másodnév tiltott, azaz az A – J betűk valamelyike.

ALIAS not found [13]

Nem létező másodnév, vagy az A – J munkaterület azonosítókon kívüli hivatkozás.

All allowed slots have been filled

10 mdx- és/vagy ndx-fájl, vagy egy mdx-fájlon belül 47 indextag már megnyitva. Többet nem lehet.

All database files must be closed before using PROTECT [173]

A PROTECT utasítás aktivizálása előtt minden adatállományt le kell zárni.

ASIN(): Out of range [291]

Az arkusz szinusz függvény argumentuma kívül esik a [-1, +1] intervallumon.

- ATAN(): Out of range** [294]
Az arkusz tangens függvény argumentuma kívül esik a $[-\pi/2, +\pi/2]$ intervallumon.
- Bad array dimension(s)** [230]
A kijelölt tömb elemeinek száma rossz. Kívül esik az 1 – 1023 intervallumon.
- Bad EXPRESSION** [40]
Rossz szintakszisú kifejezés a DO WHILE, IF vagy CASE utasításokon belül.
- Bad PROCEDURE name** [32]
A meghívott eljárás nem található. Célszerű ellenőrizni kell az eljárásfájl teljes DOS specifikációját.
- Bar position must be a positiv number** [167]
Menük kijelölésekor pozitív számokat kell használni.
- Beginning of file encountered** [38]
Visszalépési kísérlet történt, amikor a rekordmutató már az adatfájl elején, azaz az első rekord előtt volt.
- Branching must end before the first @Command** [308]
A formátum állományok első részében az első @ utasítás kiadása előtt minden elkezdett vezérlő szerkezetet (mint például IF) le kell zárni.
- Calculated field requires an expression**
Számított mezőt akart létrehozni, de nem adta meg a szükséges kifejezést.
- Calculated fields in VIEW must not be empty** [323]
Egy nézet számított mezőjét érvényes kifejezés nélkül nem lehet létrehozni.
- Cannot add an empty or erroneous calculated fields to the view**
Üres vagy hibás számított mezőt hozzáadni a nézethez nem lehet.
- Cannot add a group band to this type of band**
Ezen a listasávon belül (pl: fejléc, intro, stb) nem lehet csoportszávon kijelölni.
- Cannot add fields to the view in an update query**
Módosító nézethez mezők nem illeszthetők.
- Cannot append in column order** [147]
MultiPlan táblázatból nem lehet az adatfájlhoz rekordokat fűzni, ha a táblázat sorai nincsenek növekvő sorrendben.
- Cannot clear menu in use** [176]
A képernyőn lévő aktív menü nem törölhető.

- Cannot clear popup in use** [177]
A képernyőn lévő aktív rolómenü nem törölhető.
- Cannot close database when transaction is in process** [185]
Tranzakció alatt lévő adatfájlt nem lehet lezárni. Először ki kell adni az END TRANSACTION, vagy egy ROLLBACK utasítást.
- Cannot close index files when transaction is in process** [185]
Tranzakció alatt lévő adatfájlhoz tartozó indexfájlt nem lehet lezárni. Először ki kell adni az END TRANSACTION, vagy egy ROLLBACK utasítást.
- Cannot create a link in the pothandle**
Rossz helyen állt a kurzor, amikor az adatfájlok kapcsolására (Link by point) kísérlet történt. (Nézettervezés).
- Cannot create more than 20 calculated fields**
Legfeljebb 20 számított mező határozható meg.
- Cannot create! SQL table exists with same name** [374]
Az adatfájlt nem lehet létrehozni ezzel a névvel, mert ilyen nevű SQL tábla már létezik.
- Cannot delete SQL created TAG:<indextag-név>** [376]
Csak interaktív SQL módban törölhető az SQL által létrehozott indextag.
- Cannot enter SUM or AVERAGE in a character field**
A SUM és AVERAGE utasítások nem használhatók karakteres típusú mezőkre.
- Cannot erase open file** [89]
Nyitott fájl nem törölhető.
- Cannot erase a read-only file** [336]
Csak olvasható (READ ONLY) állomány nem törölhető.
- Cannot execute this command while transaction is in process** [186]
Ilyen utasítás tranzakciós feldolgozás közben nem adható. Lásd a BEGIN TRANSACTION utasításnál felsorolt tiltott utasításokat.
- Cannot go to Browse/Edit if errors exist in the query design** [344]
Amíg a nézetfájl tervezésében hibák vannak, nem lehet az adatokat a BROWSE/EDIT utasításokkal megjeleníteni.
- Cannot have more than one aggregate followed by UNIQUE**
Az UNIQUE kulcsszóval csak egyetlen csoportosító operátor állhat.

Cannot have more than one aggregate operator in a column [380]
SQL hibaüzenet, egy oszlophoz csak egyetlen egy csoportosító, gyűjtő operátor tartozhat.

Cannot have more than one GROUP BY in a column [339]
Egy oszlophoz kétszer nem adható meg a GROUP BY cikkely.

Cannot insert. Expression length at limit
A mezőnél hosszabb kifejezés nem vihető a mezőbe.

Cannot JOIN a file with itself [139]
A JOIN utasítással az adatfájlt nem lehet saját magával egyesíteni.

Cannot link a file skeleton to itself [345]
Az adatfájlokat saját magukkal nem lehet összekapcsolni. A kapcsoláshoz másik fájl mezőjét kell kijelölni.

Cannot load more than eight files in query design
Legfeljebb 8 állományt használható egy nézetfájl tervezésekor.

Cannot modify SQL table [375]
Az SQL tábla nem módosítható dBASE IV üzemmódban.

Cannot move empty field
Üres mező nem mozgatható. Adatfájl létrehozásakor a mozgatni kívánt mező (F7) még nem volt meghatározva.

Cannot MOVE or COPY without a defined selection
Kijelölés (blokk) nélkül nem lehet másolni, vagy áthelyezni.

Cannot move the cursor beyond bottom of window
Nem lehet az aktív ablak alsó széle alá vinni a kurzort. Legközelebbi kísérletre az ablak lezárásra kerül.

Cannot move the cursor beyond left edge of window
Nem lehet az aktív ablak bal széle elé vinni a kurzort. Legközelebbi kísérletre az ablak lezárásra kerül.

Cannot move the cursor beyond right edge of window
Nem lehet az aktív ablak jobb széle mögé vinni a kurzort. Legközelebbi kísérletre az ablak lezárásra kerül.

Cannot move the cursor beyond top of window
Nem lehet az aktív ablak fölé vinni a kurzort. Legközelebbi kísérletre az ablak lezárásra kerül.

- Cannot nest transactions** [198]
Tranzakciókat nem lehet egymásba ágyazni. Előbb az éppen folyamatban lévő az END TRANSACTION utasítással be kell fejezni.
- Cannot open <fájlnev> resource file** [398]
A fájlnev nevű, .RES kiterjesztésű fájl valószínűleg megsérült. Enélkül a dBASE IV működésképtelen. Meg kell kísérelni az eredeti lemezről visszatölteni.
- Cannot overwrite SQL TAG:<tag-nev>** [377]
Csak SQL módban írható felül az SQL által létrehozott indextag.
- Cannot place a link in a field that has an error** [346]
Hibás mezővel nem hozható létre kapcsolat adatfájlok között.
- Cannot recover damaged index file** [364]
Az indexfájl megsérült, egy új INDEX utasítás kiadása célszerű.
- Cannot re-define menu in use** [174]
Aktív menüt nem lehet újradefiniálni. Először a DEACTIVATE MENU utasítást kell kiadni.
- Cannot re-define popup in use** [175]
Aktív rolómenüt nem lehet újradefiniálni. Először egy DEACTIVATE POPUP utasítást kell kiadni.
- Cannot release memvar in use: <változónév>** [395]
Aktív változót nem lehet felszabadítani (törölni). (Pl.: GET, PROCEDURE, FUNCTION által használt változók).
- Cannot select requested database** [17]
Érvénytelen munkaterület azonosító. Az érvényesek: 1-10 vagy A-J.
- Cannot use a sort priority number higher than nine**
Ez a Control Center üzenete. Akkor fordul elő, ha egy adatfájlt 9-nél több szempont szerint kísérünk rendezni.
- Cannot write to a read-only file** [111]
Csak olvasható fájlt nem lehet felülírni.
- Cannot write to database due to incomplete transaction** [201]
Félbeszakadt tranzakció miatt nem lehet az adatfájlba írni.
- Cannot write to transaction log file** [188]
A napló fájl nem hozzáférhető. Ki kell adni az END TRANSACTION utasítást.
- Catalog has not been established** [122]
A katalógust használatbavétel előtt létre kell hozni.

- CHANGE(), not enough memory** [161]
Kevés a memória a CHANGE végrehajtásához.
- CHANGE(), record locked by another** [160]
A CHANGE által használni kívánt rekordot más felhasználó zárolta.
- CHR(): Out of range** [57]
Az argumentum vagy kisebb mint 0, vagy nagyobb, mint 255.
- Column is full (255 character maximum)**
Az oszlop megtelt. (Legfeljebb 255 karakter lehet).
- Column number must be between 0 and either right margin or 255** [223]
A nyomtatott lapon kívülre esne a megadott oszlopszám.
- Command not allowed in programs** [306]
Az utasítás programban nem használható.
- Command not allowed in SQL** [307]
Az utasítás SQL-ben nem használható.
- Command not allowed in a user defined function** [384]
Az utasítás felhasználói függvényben nem használható.
- Command not functional in dBASE IV** [93]
Az utasításnak a dBASE IV-ben nincs hatása, csak a kompatibilitás miatt maradt meg.
- Command not supported by RunTime** [290]
A rendszer az utasítást programfutás közben nem támogatja. (Pl: CREATE SCREEN, stb)
- Command not valid in RunTime environment** [348]
Az utasítás programfutás alatt nem használható.
- Command only valid in programs** [260]
Az utasítás csak programban használható.
- Command too long, press CTRL-HOME to edit zoom window** [402]
Az utasítássor legfeljebb 254 karakter hosszú lehet. Ennél hosszabb utasítás beírásához a CTRL-HOME billentyűvel lehet az ablakot bővíteni.
- Command will never be reached** [369]
Az utasításra sohasem kerül a vezérlés, mert vagy RETURN vagy EXIT után áll a programban.

- Compilation error** [360]
Fordítási hiba. (Szintaktikus hibára utal.)
- Condition box cannot be extended any further**
A feltételablak nem tágítható tovább.
- Condition box has not been created yet**
A feltételablakot először ki kell jelölni.
- Conflicting data types**
Adattípus konfliktus nézettervezés közben.
- CONTINUE without LOCATE** [42]
CONTINUE utasítás fordult elő megelőző LOCATE nélkül.
- Control code exceeds 255** [401]
A nyomtatóvezérlő kódok nem lehetnek 255-nél nagyobbak.
- Coordinates are off the screen** [30]
A koordináták kívül esnek a képernyőn.
- Could not be opened** [72]
A CONFIG.DB-ben egy hibás ALTERNATE fájl került kijelölésre.
- Current printer driver does not support quality** [331]
Az aktuális nyomtatómeghajtó nem képes levélminőségű nyomtatásra. Más meghajtót kell kijelölni. (_pdriver rendszerváltozót).
- Cyclic relation** [44]
Közvetve önmagára mutató fájlkapcsolat jött létre.
- Data error**
Hardver hiba. A lemezen az éppen olvasott szektorhibás. Adatvesztéssel járhat.
- Data type mismatch** [9]
A kifejezésen belül nem összeférhető adattípusok vannak.
- Database encrypted** [131]
Titkosított adatfájl. A PROTECT utasítás használatával lehet visszaállítani, ha a titkosító jelszó ismert.
- Database not indexed** [26]
Az adatfájl nincs indexelve.
- Date and time cannot be sized**
A dátum és az idő mérete nem változtatható a Shift-F7-tel.

- DBF file in VIEW is not in current directory**
A nézetfájlhoz kijelölt adatfájl nincs az aktuális alkönyvtárban.
- DBT file cannot be opened** [41]
A memo fájl nem sikerült megnyitni.
- Debugger already in use** [347]
A debugger program már üzemel.
- Disk full when writing file: < fájlnev >** [56]
Az < fájlnev > fájl írása közben betelt a lemez.
- Disk I/O error**
Hardver hiba olvasás vagy írás közben.
- Disk is write protected**
A lemez írásvédett.
- Display mode not available** [216]
A kiválasztott képernyőmód nem elérhető. A SET utasítással állítható be.
- Display width exceeds 255** [400]
A megjelenítés szélessége nem haladhatja meg a 255 karaktert.
- Drive not ready on < meghajtó >**
A meghajtó nem üzemkész. A meghajtó fülét le kell hajtani.
- Duplicate field name**
A fájlszerkezet meghatározása közben már létező mezőnév került ismét kijelölésre.
- Duplicate production MDX file** [211]
Az alapindexfájlt ismételten, kétszeresen nem lehet megnyitni.
- Duplicate sort number** [322]
A szemantikus ellenőrzés során a rendszer két különböző oszlopban ugyanazt a rendezési kulcsot találta (nézetfájl).
- Editing condition not satisfied** [171]
A @ ...SAY ...GET utasítás VALID vagy RANGE cikkelyében meghatározott feltételeknek nem felel meg a beolvasott adat.
- Empty structure will not be saved**
Üres fájlszerkezetet nem lehet elmenteni.

- End of file encountered** [4]
Előrelépési kísérlet történt, amikor a rekordmutató már az adatfájl végén, azaz a utolsó rekord mögött volt.
- End of file or error on keyboard input** [51]
Amikor a billentyű input egy fájlból(makró) jön, és a fájl váratlanul befejeződik, ez az üzenet jelentkezik.
- Environment not correct for rollback** [193]
Kiadott ROLLBACK utasítás, amelyet nem előzött meg félbeszakadt tranzakció.
- Error during processing of ON ERROR command** [394]
Az ON ERROR utasítás végrehajtása közben történt hiba.
- Error in configuration value** [145]
A CONFIG.DB fájlban egy hibás érték szerepel.
- Error in query definition - cannot execute**
Hibásan tervezett nézetfájl. Nem hozható létre, nem futtatható.
- Error in reading log file** [192]
A ROLLBACK utasítás nem hajtható végre, mert a naplófájl meghibásodott.
- Error on line <szám>** [96]
Fordítási hiba. A szám a program sorszámra utal, ahol a hiba előfordult.
- Errors in query definition - cannot be executed** [352]
Hibás nézet. Nem futtatható.
- EVERY allowed with only one of example variable pair** [318]
Kapcsolat létrehozása esetén csak egyetlen változópárhoz lehet megadni az EVERY kulcsszót.
- Exceeded maximum compiler nesting level** [248]
Az eljárás lefordított kódjának mérete túllépte a megengedett 64 kilobájtos határt, illetve a 32 kilobájtos határt, amely egy elágazás lefordított kódjának felső határa lehet. Az eljárás sorait csökkenteni kell.
- Exceeded maximum number of compile time symbols** [274]
Túl sok változó, mező van használatban a fordítás alatt. Ezek együttes száma nem haladhatja meg az 500-at, hacsak a CONFIG.DB fájlban CTMAXSYMS értéke más felső határt nem jelöl ki.
- Exceeded maximum number of compile time symbol references**[273]
Túl sok változó, mező van használatban a fordítás alatt. Ezek együttes száma nem haladhatja meg az 500-at, hacsak a CONFIG.DB fájlban CTMAXSYMS értéke más felső határt nem jelöl ki.

- Exceeded maximum number of procedures** [397]
Túl sok eljárás van használatban egy .dbo fájlban belül.
- Exceeded maximum number of runtime symbols** [272]
Túl sok változó, mező van használatban a futás alatt. Ezek együttes száma nem haladhatja meg az 500-at, hacsak a CONFIG.DB fájlban RTMAXBLKS és RTBLKSIZE értéke más felső határt nem jelöl ki.
- Extra characters ignored at end of command** [151]
Az utasítássor végén a rendszer a számára értelmezhetetlen karaktereket figyelmen kívül hagyja.
- <mező> data type does not match <mező>**
A képernyő- lista-, címkeformátumok és az adatfájl mezői között adattípus konfliktus van.
- <mező> not found in <fájlnév>**
A képernyő- lista-, címkeformátumok meghatározásai szerinti mező az adatfájlban nem található.
- Field must be a memo field** [350]
A mező csak memo-mező lehet.
- Field name already exist**
A mezőnév már létezik.
- Field name already exist in the view**
A mezőnév már létezik a nézetben.
- Field name already in use**
A mezőnév már foglalt.
- Field name not found**
Nincs ilyen mezőnév.
- Field name required**
Mezőnevet vár a rendszer.
- Field name too long**
A mezőnév túl hosszú.
- Field not found** [48]
Nincs ilyen mező a fájlban.
- Field requires a name**
A számított és összegző mezőknek is nevet kell adni.

H) Hibaüzenetek

Field type must be A,B,C,D,L or M

A megengedett mezőtípusok: A,B,C,D,L vagy M

Field type must be Y or N

A mezőbe Y vagy N írható.

Fields list too complicated

[141]

Túl sok mező van a mezőlistában.

File already exist

[7]

A fájl már létezik.

File already in catalog

[286]

A fájl már benne van a katalógusban.

File already open

[3]

A fájl már megnyitva.

File cannot be modified because design mode is set to OFF

[314]

A SET DESIGN OFF állásban van, ezért az adatmegjelenítésből tervezési módba nem lehet váltani.

File catalog empty

A katalógus üres.

File does not exist

[1]

A keresett fájl nem létezik.

File has been deleted

[49]

A fájl törölve.

File in use by <felhasználó név>

[372]

A fájlt a <felhasználó név> felhasználó zárta.

File in use by another

[108]

A fájl zárva, valaki más használja.

File must be opened in exclusive mode

[110]

A fájlt csak kizárólagos (exclusive) módon lehet megnyitni.

File not accessible

[29]

A fájl nem elérhető.

File not an MDX index file

[206]

A fájl nem mdx-fájl.

- File not in transaction log** [194]
A naplófájlban az adatfájl nem szerepel, ezért a ROLLBACK utasítással nem lehet visszaállítani.
- File not linked** [326]
Az adatfájl nincs összekapcsolva a nézetfájlban.
- File skeleton cannot be extended any further**
Több fájl minta már nem határozható meg.
- File was not loaded** [91]
CALL utasítással csak előzőleg LOAD-dal memóriába töltött bináris programot lehet hívni.
- Find not successful** [14]
A keresés sikertelen volt.
- FIRST allowed with only one example variable pair**
Kapcsolat létrehozása esetén csak egyetlen változó párhoz lehet megadni az FIRST kulcsszót.
- Format files must be in row major order to be exported**
A formátum-fájlban a @ koordinátáknak sor szerint növekvő sorrendben kell lenniük, mielőtt PFS:FILE fájlba kerülnek.
- FROM database must be in one of the unselected work areas** [189]
COPY FROM vagy APPEND FROM utasítások esetén a forrásfájl csak az aktuálistól eltérő munkaterületen lehet nyitva.
- Function prohibited in SQL mode: <függvénynév>** [340]
A <függvénynév> nevű függvény nem használható SQL módban.
- General failure**
Operációs rendszer általános hibaüzenete. Az esetek többségében hardver hibát jelent.
- GROUP BY used without an aggregate operator** [327]
A csoportosító, gyűjtő operátor hiányzik a GROUP BY után.
- Group name has not been defined** [155]
A csoport név nincs meghatározva.
- Group name has not been established** [158]
A csoport név még nincs létrehozva.
- Hidden field requires a name**
A rejtett mezőknek is nevet kell adni.

H) Hibaüzenetek

- Illegal call to the compiler** [379]
Illegális fordítóprogram hívás.
- Illegal character data length**
A karakter típusú adat hossza hibás.
- Illegal data length**
Hibás adathossz.
- Illegal decimal length**
Hibás a tizedesjegyek száma.
- Illegal field name**
Hibás mezőnév.
- Illegal field width**
Hibás mezőhossz.
- Illegal Key Expression** [386]
Hibás kulcskifejezés.
- Illegal macro usage** [266]
Nem megengedett makróhasználat.
- Illegal numeric data length**
A numerikus adat hossza hibás.
- Illegal operation. Database structure is empty or has been changed** [362]
Nem megengedett művelet. Az adatfájl szerkezet üres vagy megváltozott.
- Illegal value** [46]
Hibás érték.
- Index damaged. Do REINDEX before using data** [114]
Az indexsorrend felborult. REINDEX utasítással állítható helyre.
- Index expression is too large (220 char maximum)** [112]
Az indexkifejezés túl hosszú (220 kar. max.).
- INDEX failed - key is not DISTINCT** [236]
Rossz indexkulcs, a fájl duplikált rekordokat tartalmaz és ez nem megengedett. Másikat kell választani.
- Index field must be Y or N**
Az indexmező csak Y-t vagy N-t tartalmazhat.

- Index file does not match database** [19]
Az indexfájl nem az adatfájlhoz tartozik.
- INDEX interrupted. Index will be deleted if not completed** [113]
Az indexelés az Esc lenyomása miatt félbeszakadt. Ha nem fejeződik be, az indexfájl törlődik.
- Index is too large (100 char maximum)** [23]
Az indexkulcs túl hosszú (100 kar. max.).
- Index TAG already exists** [205]
Az indextag már létezik.
- Insufficient memory** [43]
Nincs elegendő memória.
- Insufficient space on row**
Nincs elég hely a sorban.
- Insufficient space on row, field truncated**
Nincs elég hely a sorban. A mező csonkul.
- Insufficient space on row, position field with cursor keys**
Nincs elég hely a sorban. A mezőt a kurzor mozgatásával kell elhelyezni.
- Internal editor error. Editor buffer may be damaged**
Belső fatális hiba. Ha előfordul lehet az ASHTON - TATE-hez fordulni.
- Internal error: CMDSET():** [66]
Fatális belső hiba. Ez a hiba nem fordulhat elő.
- Internal error: EVAL work area overflow** [67]
Belső hiba. A verem túlcsondult.
- Internal error: Illegal opcode** [68]
Belső hiba. Hibás műveleti kód.
- Internal error: Internal table overflow** [69]
Belső hiba. Túl bonyolult kifejezések kiértékelésekor fordulhat elő. Próbálja meg egyszerűsíteni a kifejezést, vagy növelje a config.db fájlban az EXPSIZE értéket.
- Internal error: Unknown command code:** [65]
Belső hiba. Hibás utasításkód.
- Internal SQL code generation error** [311]
Fatális belső hiba. Ha előfordul, nincs mit tenni.

H) Hibaüzenetek

Internal: Virtual Stack Overflow	[199]
Belső hiba. Túl mélyen egymásbaágyazott függvények egy szélsőségesen bonyolult kifejezésben.	
Invalid box dimensions	[227]
Hibás keret koordináták.	
Invalid date	[81]
Hibás dátum.	
Invalid DIF character	[118]
Érvénytelen DIF karakter. DIF formátumú fájlból történő beolvasás során következhet be.	
Invalid DIF file header	[115]
A beolvasandó DIF fájl fejrésze sérült.	
Invalid DIF type indicator	[117]
A beolvasandó DIF fájl egy hibás típusjelzőt tartalmaz.	
Invalid DIF vector - DBF field mismatch	[116]
A beolvasandó DIF fájl fejrésze ellentmondásban van a tartalmával.	
Invalid DOS SET option	[99]
Az autoexec.bat fájlban egy hibás utasítás van.	
Invalid file extension: <kiterjesztés>	[333]
A fájl kiterjesztése a célnak nem megfelelő.	
Invalid function argument	[11]
Hibás függvény argumentum.	
Invalid function name	[87]
Hibás függvénynév.	
Invalid group name for file. Please re-enter	[159]
A PROTECT programmal használt csoport név rossz. Újat kell megadni.	
Invalid index number	[106]
Hibás indexhivatkozás a SET ORDER TO utasításban.	
Invalid key label	[317]
Nem megengedett a billentyű programozása.	
Invalid Lotus 1-2-3 version 2.0 spreadsheet	[297]
A beolvasandó Lotus fájl verziója nem 2.0	

Invalid macro file header Hibás makrófájl fejrész.	[356]
Invalid operator Hibás művelet.	[107]
Invalid password.Please re-enter Hibás jelszó. Újat kell megadni.	[153]
Invalid path or file name: <fájlnév>/<elérési út> Hibás fájlnev vagy elérési út.	[202]
Invalid printer port Nem létező nyomtató kimenet.	[123]
Invalid printer redirection Hibás nyomtató átirányítás.	[124]
Invalid SET expression Hibás SET kifejezés.	[231]
Invalid SYLK file dimension bounds Hibás dimenzió határok SYLK fájlnál.	[120]
Invalid SYLK file format 121] Hibás SYLK fájl formátum.	
Invalid SYLK file header Hibás SYLK fájl fejrész.	[119]
Invalid TAG name Hibás indextag-név.	[284]
Label field invalid Érvénytelen címkemező.	
Label file invalid Érvénytelen címkefájl.	[54]
Left margin and indentation must be less than right margin A bal margó nem lehet nagyobb, mint a jobb.	[221]
Line exceeds maximum of 1024 characters A sor meghaladja a megengedett 1024 karakteres határt.	[18]
Line number must be between 0 and page length 222] A sorszámnak 0 és a laphossz között kell lennie.	

H) Hibaüzenetek

Link is currently pending

Új CREATE LINK kijelölés anélkül, hogy az előző befejeződött volna.

Lock table is full

[217]

A LOCK tábla megtelt. Legfeljebb 50 db lehet egyidőben.

Log file corrupted

[195]

A naplófájl megsérült.

Log file not found

[191]

A naplófájlt nem található.

Log record does not match database record

[196]

A naplófájl nem illeszkedik az adatfájlhoz. END TRANSACTION kidása szükséges.

LOG(): Zero or negative

[58]

A természetes logaritmus argumentuma 0 vagy negatív.

LOG10(): Zero or negative

[292]

A tízes alapú logaritmus argumentuma 0 vagy negatív.

Macro library does not exist

[354]

A makró-könyvtár nem létezik.

Macro not found in the current library

[355]

A makró nincs az aktuális könyvtárban.

Macro recording is in process

[357]

A makró visszajátszása folyamatban van.

Macros are not allowed in user defined functions

[393]

Felhasználói függvényekben nem használhat makrókat.

Macros cannot expand flow-of-control commands

[388]

Makrót nem használható vezérlő utasításokban, mint az IF, SCAN, stb.

Macros nested too deep

[361]

A makrók egymásbaágyazása túl mély.

Maximum field width exceeded

Túllépte a megengedett mezőhosszúságot.

Maximum number of GET commands exceeded

[310]

Túllépte a megengedett GET-ek számát.

Maximum number of fields reached	[296]
Túllépte a mezők megengedett számát.	
Maximum number of index tags already reached	
Az indextagok megengedett száma túllépve.	
Maximum number of tab stops exceeded	
A tabulátorok megengedett száma túllépve.	
Maximum popup nesting exceeded	[182]
A rolómenük egymásbaágyazásának megengedett száma túllépve (40).	
Maximum print width of 255 characters exceeded	
A megengedett nyomtatási szélesség túllépve.	
Maximum record length exceeded	[137]
A megengedett rekordhossz túllépve.	
Maximum record length exceeded in WK1 file	[392]
A megengedett rekordhossz a WK1 fájlban túllépve.	
MDX file does not match database	[207]
Az mdx-fájl nem illeszkedik az aktív adatfájlhoz.	
MDX file full	[203]
Az mdx-fájl megtelt.	
Memo fields cannot be prompted	[280]
Memo-mező nem szerepelhet egy rolómenü soraként.	
Memory error	
Memória hiba.	
Memory variable already defined-cannot make PUBLIC	[271]
A korábban már kijelölt változót most nem lehet PUBLIC típusúvá tenni.	
Memory variable cannot be defined here as PRIVATE	[270]
Korábban már kijelölt változót még egyszer nem lehet PRIVATE típusúvá tenni.	
Memory variable file invalid	[55]
A változókat tartalmazó fájl hibás.	
Memos are not available	[172]
A memo-adatokat nem lehet elérni. Akkor fordul elő, ha az aktív adatfájlhoz nem sikerült megnyitni a hozzá tartozó .dbt fájlt.	

H) Hibaüzenetek

MENU has not been activated A menü még nem aktív.	[178]
MENU has not been defined A menüdefiníció nem létezik.	[168]
Menu is already in use A menü már használatban.	[181]
Minimum field width reached A mezőhossz kisebb 1-nél. Ez nem megengedett.	
Missing END PRINTJOB for previous PRINTJOB Hiányzó END PRINTJOB az előzőleg kiadott PRINTJOB után.	[300]
Missing ENDTEXT for previous TEXT Hiányzó ENDTEXT az előzőleg kiadott TEXT után.	[341]
Missing END TRANSACTION for previous BEGIN TRANSACTION Hiányzó END TRANSACTION az előzőleg kiadott BEGIN TRANSACTION után.	[301]
Missing ENDCASE for previous DO CASE command Hiányzó ENDCASE az előzőleg kiadott DO CASE után.	[247]
Missing ENDDO for previous DO WHILE command Hiányzó ENDDO az előzőleg kiadott DO WHILE után.	[246]
Missing ENDIF for previous IF command Hiányzó ENDIF az előzőleg kiadott IF után.	[244]
Missing -ENDIF for previous IF/ELSE commands Hiányzó ENDIF az előzőleg kiadott IF/ELSE után.	[245]
Missing ENDSCAN for previous SCAN command Hiányzó ENDSCAN az előzőleg kiadott SCAN után.	[269]
Missing EXPRESSION Kifejezés hiányzik.	[152]
Missing index for ARRAY reference Tömbindex nélküli tömbhivatkozás.	[299]
Modify group only modify group bands A csoportot módosítani csak a csoportszámban lehet.	
More than one sort option in a column Egynél több rendezési előírás van egy oszlopban.	[325]

Must be a valid dBASE expression. Press any key to continue
Itt csak érvényes dBASE kifejezés lehet. Folytatáshoz egy billentyűt kell lenyomni.

NDX index limit reached [204]
Nem nyitható meg több index fájl.

NDX index may not be DESCENDING [235]
Egy ndx-fájl nem lehet csökkenő sorrendben.

Network server busy [148]
A hálózatvezérlő egység foglalt.

No bars have been defined for this popup [166]
Nem létezik menüpont ehhez a rolómenühöz.

No database in use [52]
Az aktuális munkaterületen nincs adatfájl használatban.

No database in use. From contains the following fields:
Az aktuális munkaterületen nincs adatfájl használatban.

No DBF files are present
Nincs adatfájl az aktuális könyvtárban. (Control Center üzenet)

No fields of the requested type are present. Press any key to continue
A kívánt típusú mező nincs. Folytatáshoz egy billentyűt kell lenyomni.

No fields to process [47]
Nincs feldolgozható mező.

No fields were found to copy [138]
Nincs másolható mező.

No files of the requested type are cataloged
A kívánt típusú fájlból egy sem szerepel a katalógusban.

No files of the requested type are cataloged.
Press any key to continue
A kívánt típusú fájlból egy sem szerepel a katalógusban.
Üssön le egy billentyűt a folytatáshoz.

No files of the requested type are present
Press any key to continue
A kívánt típusú fájl nem létezik.
Üssön le egy billentyűt folytatáshoz.

H) Hibaüzenetek

- No files of the requested type in this drive or catalog** [53]
A kívánt típusú fájl nem létezik ezen a lemezen vagy a katalógusban. Üssön le egy billentyűt a folytatáshoz.
- No format files in current directory**
Nincs formátumfájl az aktuális könyvtárban.
- No format files in the catalog for the current database**
Az aktuális adatfájlhoz nincs formátumfájl a katalógusban.
- No index files for current database in the catalog**
Az aktuális adatfájlhoz nincs indexfájl a katalógusban.
- No label forms in catalog file for current database**
Az aktuális adatfájlhoz nincs címkeformátum a katalógusban.
- No more windows available** [213]
Több ablak nem nyitható meg.
- No previous BEGIN TRANSACTION to match this command** [303]
Nincs megelőző BEGIN TRANSACTION.
- No previous DO CASE to match this command** [250]
Nincs megelőző DO CASE.
- No previous DO WHILE to match this command** [251]
Nincs megelőző DO WHILE.
- No previous DO WHILE/SCAN/PRINTJOB to match this command**[304]
Nincs megelőző DO WHILE/SCAN/PRINTJOB.
- No previous IF to match this command** [249]
Nincs megelőző IF.
- No previous PRINTJOB to match this command** [305]
Nincs megelőző PRINTJOB.
- No previous SCAN to match this command** [302]
Nincs megelőző SCAN.
- No printjob is in progress** [282]
Nincs nyomtatási köteg folyamatban.
- No program file found for this application** [334]
Nincs programfájl az aktuális alkalmazáshoz.

No records selected	[365]
Nincs kiválasztott rekord a szűrés után (SET FILTER).	
No report forms in catalog file for current database	
Az adatfájlhoz nincs listaformátum a katalógusban.	
No search string specified	
Keresési kulcs hiányzik.	
No selection made for MOVE	
Nem választotta ki F6-tal a mezőt, amelyet mozgatni akar.	
No view files in the catalog for current database	
Az adatfájlhoz nincs nézetfájl a katalógusban.	
Not a character expression	[45]
Nem karakteres kifejezés.	
Not a dBASE database	[15]
Nem dBASE adatfájl.	
Not a DBO file: fájlnev	[253]
Nem érvényes dbo (tárgyprogram) fájl. Újrafordítás szükséges.	
Not a defined error	
Felderíthetetlen hiba.	
Not a logical expression	[37]
Nem logikai kifejezés.	
Not a numeric expression	[27]
Nem numerikus kifejezés.	
Not a valid dBASE II database	[257]
Érvénytelen dBASE II adatbázis.	
Not a valid expression	[233]
Érvénytelen kifejezés.	
Not a valid Framework II database/spreadsheet	[256]
Érvénytelen Framework II adatbázis/táblázat.	
Not a valid PFS file	[140]
Érvénytelen PFS fájl.	
Not a valid QUERY file	[134]
Érvénytelen nézetfájl.	

H) Hibaüzenetek

Not a valid RapidFile database Érvénytelen RapidFile adatbázis.	[255]
Not a valid VIEW file Érvénytelen VIEW fájl.	[127]
Not a valid window file Érvénytelen ablakfájl.	[358]
Not an array Nem tömb. Egy egyszerű változót nem lehet tömbként használni.	[299]
Not enough disk space for operation Nincs elég hely a lemezen a művelet elvégzéséhez.	[275]
Not enough disk space for SORT Nincs elég hely a lemezen a rendezéshez.	
Not enough records to sort Nincs elég rekord a rendezéshez.	[277]
Not suspended Nem történt programfutás felfüggesztése (SUSPEND).	[101]
Numbers are not allowed in the CURRENCY symbol A CURRENCY meghatározásánál számokat nem szabad használni.	[295]
Numeric overflow (data was lost) Túlcsordulás. Az adat elveszett.	[39]
ON PAD already defined for this prompt pad Nem lehet ismételten ON PAD utasítást kijelölni ehhez a menüponthoz.	[170]
ON SELECTION already defined for this prompt pad Nem lehet ismételten ON SELECTION utasítást kijelölni ehhez a menüponthoz.	[169]
Only one update operation allowed per query Egy módosító nézetben csak egyetlen módosító művelet megengedett.	[328]
Only windows, boxes, and fields can be sized Csak ablakok, keretek és mezők méreteit lehet változtatni.	
Operation not allowed for calculated fields Ez a művelet számított mezőkre nem megengedett.	[371]
Operation with Logical field invalid Ez a művelet logikai mezőkre nem megengedett.	[90]

Operation with Memo field invalid	[34]
Ez a művelet memo-mezőkre nem megengedett.	
Operation not allowed in the calculated field skeleton	
Ez a művelet számított mezőkre nem megengedett.	
ORDER must be specified by index TAG or file name	[329]
Az indexsorrendet indextagnévvel vagy fájlnevével kell megadni.	
ORDER TAG not found	[208]
Az indextag nem található.	
Out of memory variable memory	[21]
Nincs több memória a változók részére.	
Out of memory variable slots	[22]
Nincs több memória a változók részére.	
PAD has not been defined	[164]
A PAD-ot még nem definiálta.	
PAD has not been defined for this menu	[180]
Még nincs menüpont ehhez a menühez.	
PARAMETERS command must be at top of procedure	[243]
A PARAMETERS utasítás csak az első lehet az eljárásban.	
Password and confirmation mismatch	[154]
A jelszó és az ellenőrzésként kért ismétlés eltér.	
Password file is in use by another	[349]
A jelszófájlt valaki más használja.	
Password has not been defined	[157]
A jelszó nincs meghatározva.	
PFS does not allow row numbers higher than 20	
A PFS fájl nem enged meg 20 sornál többet.	
Place fields in the VIEW skeleton first	[321]
A mezőt először a VIEW mintában kell elhelyezni.	
Please put a database file or view into use first	[281]
Előbb az adat- vagy nézetfájlt kell megnyitni.	
Please put view or dbf in use first	
Előbb az adat- vagy nézetfájlt kell megnyitni.	

H) Hibaüzenetek

POPUP has not been activated	[179]
A rolómenü még nem aktív.	
POPUP has not been defined	[165]
A rolómenüdefiníció hiányzik.	
Popup is already is use	[182]
A rolómenü már aktív.	
POPUP is too small	[287]
A rolómenü túl kicsi.	
Position out of window	[288]
A pozíció kívül esik az ablakon.	
Printer is either not connected or turned off	[126]
Vagy nincs nyomtató a géphez kötve, vagy ki van kapcsolva.	
Printer not ready	[125]
A nyomtató nem üzemkész.	
Printjobs cannot be nested	[337]
A PRINTJOB utasításokat nem lehet egymásbaágyazni.	
PROCEDURE command is required	[242]
A PROCEDURE kulcsszó hiányzik.	
Procedure is too large (64 K)	[258]
Az eljárás túl nagy.	
Procedure not found: <eljárásnév>	[252]
Az <eljárásnév> nevű eljárás nem létezik.	
PROCEDURES cannot return a value	[385]
Az eljárások nem adnak vissza értéket.	
Production MDX file is damaged	[289]
Az mdx-fájl megsérült. Újraindexelés (REINDEX) célszerű.	
Production MDX file not found	[210]
Az alapindexfájl nem található.	
PROMPTS for this popup have already been defined	[279]
Ehhez a rolómenühöz már van kijelölt menüszöveg (PROMPT).	
Query not executable. Saved anyway	
A nézet nem futtatható, mert hibás.	

- Query not valid for this environment** [143]
A nézet nem használható ebben a környezetben. Olyan mezőhivatkozás van benne, amely egyetlen nyitott fájlban sem szerepel.
- Query too complex** [338]
A nézet túl bonyolult. Túl sok mezőhivatkozás van benne vagy a generált utasítássor hosszabb, mint 1024 karakter.
- Quick Report is not available while a form, report, or label is being modified** [351]
Módosításkor nem használható a QUICK REPORT opció.
- Read error** [254]
Olvasási hardverhiba.
- Read error on drive <meghajtó>**
Olvasási hiba a <meghajtó> meghajtón.
- READ or @Command cannot follow FMT termination commands** [359]
A formátumfájl befejező része után nem állhat READ vagy @ utasítás.
- Record in use by <felhasználó név>** [373]
A rekordot a <felhasználó név> nevű felhasználó lefoglalta.
- Record in use by another** [109]
A rekordot valaki lefoglalta.
- Record not in index** [20]
A rekord nem szerepel az indexfájlban. Újraindexelés szükséges.
- Record not inserted** [25]
A rekord beillesztése nem sikerült.
- Record out of range** [5]
A rekord nem létezik.
- Records do not balance (program error)**
Belső hiba. A rendezés eredményadatai megsérültek.
- Recursive keystroke macro call:macro név** [363]
Rekurzív makróhívás. (A makró saját magát hívta).
- Relation record in use by another** [142]
A SET RELATION -nal az aktuális fájlhoz kapcsolt fájl rekordja foglalt.
- Remove group only removes group bands**
A REMOVE GROUP csak csoporsávokat távolít el.

H) Hibaüzenetek

- REPLICATE(): String too large** [88]
A sztring túl hosszú.
- Report field invalid**
A betöltött listaformátumfájl mezője sérült.
- Report file invalid** [50]
A listaformátum sérült.
- Restricted command: not allowed in this context** [389]
Az utasítás nem használható ebben az összefüggésben.
- RETURN TO is invalid in user defined function** [383]
Felhasználói függvényből nem lehet RETURN TO-val visszatérni.
- Right margin must be less than or equal to 255** [225]
A jobb margó nem lehet nagyobb 255-nél.
- Rollback database cannot be executed inside a transaction** [197]
A ROLLBACK egy fájlra csak hardver hiba után adható ki. Az üzenetet egy tranzakció közben kiadott ROLLBACK eredményezi.
- Screen file invalid** [283]
A képernyőformátum nem érvényes.
- Select box with F6 only**
Csak az F6-tal választható ki a keret.
- Sort order type must be A or D**
A rendezés csak növekvő-A vagy csökkenő-D lehet.
- Source does not correspond to the object** [95]
A forrásprogram nem a tárgyprogramnak megfelelő. Módosítás után az újrafordítás elmaradt.
- SPACE(): Negative** [60]
A függvény argumentuma 0-nál kisebb szám.
- SPACE(): Too large** [59]
A függvény argumentuma túl nagy. Legfeljebb 254 lehet.
- SQL run-time error** [335]
SQL futási hiba.
- SQL System Tables may not be USED** [218]
Az SQL rendszertáblái nem nyithatók meg USE utasítással.

SQRT(): Negative	[61]
Negatív számnak nem lehet négyzetgyökét találni.	
STORE: String too large	[79]
A sztring túl hosszú.	
STR(): Out of range	[63]
A harmadik paraméter nagyobb, mint a második.	
String not found	
A keresett sztring nem létezik.	
Structure invalid	[33]
Hibás struktúra.	
STUFF(): String too large	[102]
A sztring túl hosszú.	
SUBSTR(): Start point out of range	[62]
A második paraméter nagyobb, mint a sztring hossza.	
Syntax error	[10]
Szintaktikus hiba.	
Syntax error in contents expression	
Szintaktikus hiba a kifejezésben.	
Syntax error in query definition	
Szintaktikus hiba a nézetdefinícióban.	
Tab stops must be in ascending order	[226]
A tabulátorpozíciókat növekvő sorrendben kell megadni.	
Tabstop value exceeds 255	[399]
A tabulátorpozíció nem haladhatja meg a 255-t.	
Table full	[105]
A táblázat megtelt. Itt a CALL-lal hívható bináris fájlok táblázatáról van szó.	
TAG not found	[209]
Az indextag nincs az mdx-fájlban.	
Target file not found in catalog	[343]
A cél fájl nincs benne a katalógusban.	

H) Hibaüzenetek

There are no files of the type requested in the current dirctory or catalog [53]

Nincs ilyen típusú fájl az aktuális könyvtárban vagy a katalógusban.

This file cannot be modified [342]

Ezt a fájlt nem lehet módosítani.

Too many @Commands on one page of format file

Túl sok @ utasítás szerepel a formátum fájl egy oldalán. Legfeljebb 200 lehet.

Too many fields in WK1 file [391]

A WK1 fájlban túl sok mező van.

Too many files are open [6]

Túl sok fájl van nyitva. A Config.sys-t kell átírni.

Too many indexes [28]

Túl sok index van nyitva.

Too many merge steps [278]

Túl sok rendezési lépés kell. Ez igen nagy fájlok rendezésekor fordulhat elő.

Too many pages in format file

Túl sok oldal van a formátum-fájlban. Legfeljebb 32 lehet.

Too many sort key fields [276]

A rendezési szempontnak kijelölt mezők száma túl sok. Legfeljebb 9 lehet.

Too many WITH operators [319]

Túl sok WITH kulcsszó.

Total label width exceeds maximum size

Az összes címkeszélesség meghaladja a megengedett maximumot.

Total size of fields in view too large for UNIQUE query [381]

A mezők összmérete túl nagy a nézetben.

TRAP cannot be turned OFF while Debugger is active [390]

A TRAP nem lehet OFF állásban, mialatt a debugger program aktív.

Unable to load Command.com [92]

Nem sikerült betölteni a DOS utasításértelmezőt. Memóriafelszabadítás után ismétlés szükséges.

Unable to load file

Nem sikerült betölteni a fájlt.

Unable to LOCK	[129]
A zárolás nem sikerült.	
Unable to SKIP	[128]
A lépés a fájlban nem sikerült. Foglalt rekordra történt volna.	
Unassigned file number	[2]
Belső hiba. Normál használat során nem fordulhat elő.	
Unauthorized access level	[133]
Jogosulatlan elérési szint. Olyan fájlhoz történt a hozzáférési kísérlet, amelyhez a jogosultság hiányzik.	
Unauthorized login	[132]
Jogosulatlan bejelentkezés. A jelszó nem volt megfelelő.	
Unbalanced parenthesis	[8]
Hibás zárójelpárok. Valamelyik zárójel párja hiányzik.	
Undefined box border	[228]
A keretkarakterek nem kerültek meghatározásra.	
Unknown function key	[104]
Ismeretlen funkciós billentyű.	
Unknown system memory variable	[220]
Ismeretlen rendszerváltozó.	
Unrecognized phrase/keyword in command	[36]
Ismeretlen kulcsszó az utasításban.	
Unterminated string	[35]
A sztring végéről hiányzik a lezáró jel.	
Unterminated transaction file exists, cannot start new transaction	[183]
Amíg van egy befejezetlen tranzakció, addig nem indítható új.	
User defined functions must return a value	[382]
Hiányzik a felhasználói függvény visszatérési értéke.	
User name has not been defined	[156]
A megadott felhasználónév még ismeretlen.	
Variable not found	[12]
A változó nem található.	

H) Hibaüzenetek

- View cannot hold any more fields (255 already)** [353]
A nézet nem tartalmazhat több mezőt. (Eddig már 255 van.)
- WARNING: File definition has changed** [320]
A nézethez tartozó valamelyik fájl szerkezete megváltozott.
- Warning: key expression uses ALIAS or MEMVAR** [387]
Figyelmeztetés: Környezetváltáskor az indexkulcs másodnevet vagy változónevet tartalmazott, és ez érvénytelenné válhat.
- WARNING on line <sorszám> <hibaüzenet>** [315]
Figyelmeztetés. A <sorszám> sorszámú programsor hibás. A hibát a <hibaüzenet> magyarázza.
- WINDOW coordinates outside of allowable screen space** [332]
Az ablakkoordináták a képernyőn kívülre esnek.
- WINDOW has not been defined** [214]
Az ablakdefiníció hiányzik.
- WINDOW is too small** [285]
Az ablak túl kicsi.
- WINDOW not presently on the screen** [215]
Az ablak pillanatnyilag nincs a képernyőn.
- Work area already used in a relation** [298]
A munkaterületet már egyszer felhasználva a RELATION utasítás során.
- Work area reserved by SQL** [219]
A munkaterületet az SQL foglalta le.
- Write error on <meghajtó>**
Írási hiba a <meghajtó> meghajtón.
- Wrong number of parameters** [94]
Hibás a paraméterek száma.
- Zoom window command editor, CTRL-END exits to dot prompt**
Nem hibaüzenet, csak tájékoztatás a bővített utasítássor szerkesztéséből történő kilépéshez. (Ctrl-End)
- ** Not found **** [82]
A megadott mező nem található.

- ** WARNING ** Data will probably be lost.** [70]
Confirm (Y/N)
 Az adatok elveszhetnek. Jóváhagyja ?
 Akkor jelenik meg, ha egy művelet végrehajtása közben megtelik a lemez. *N* válasz esetén az adatok elvesznek. *Y* válasz esetén a felkínált fájlok törlésével szabad helyet lehet teremteni a lemezen. Így a megkezdett művelet befejezhető.
- ** WARNING ** Indexes could not be updated** [378]
 Figyelmeztetés: az indexek karbantartása valamilyen akadály miatt nem történhet meg.
- ** WARNING ** Uncompleted transaction found** [190]
 Figyelmeztetés: befejezetlen tranzakció található.
- *** Unrecognized command verb** [16]
 Ismeretlen utasításszó.
- *** Execution error on <függvéynév>** [80]
 Rosszul megadott argumentum a <függvéynév> függvényben.
- : Concatenated string too large** [76]
 Az összefűzött sztring túl hosszú.
- +: Concatenated string too large** [77]
 Az összefűzött sztring túl hosszú.
- , Database in use:** [71]
 Állapotüzenet, ami a DISPLAY STATUS utasításnál jelenik meg. Kiírja a munkaterület számát, az ott megnyitott fájl nevét és másodnevét.
- ^ or **: Negative base, fractional exponent** [78]
 Negatív alap vagy törtkitevő. Trigonometrikus és aritmetikai függvények argumentumai nem lehetnek negatív számok. A kitevő csak egész szám lehet.
- ^__ Keyword not found** [86]
 A CONFIG.DB valamelyik kulcsszava hibás.
- ^__ Out of range** [75]
 A kijelzett függvény argumentumainak valamelyike hibás.
- ^__ Truncated** [74]
 A CONFIG.DB egyik sora túl hosszú, ezért megcsönkult, s így értelmezhetetlenné vált.
- ^ ^ Expected ON or OFF** [73]
 A kiadott SET utasításból hiányzik az ON/OFF kulcsszavak valamelyike.

1) Mozgás és szerkesztés a képernyőn

A szerkesztőbillentyűk a beépített szövegszerkesztőben, a tervező utasítások által aktivizált felhasználói csatolók működése során teljeskörűen, a teljes-képernyős adatbeviteli utasításoknál pedig részlegesen használhatók.

Vezérlőbillentyű	Magyarázat
→, Ctrl-D, Ctrl-L	Jobbra mozgatja a kurzort 1 pozícióval.
←, Ctrl-H, Ctrl-S	Balra mozgatja a kurzort 1 pozícióval.
↓, Ctrl-J, Ctrl-X	Lefele mozgatja a kurzort 1 sorral.
↑, Ctrl-E, Ctrl-K	Felfele mozgatja a kurzort 1 sorral.
PgDn, Ctrl-C	Lejjebb megy egy képernyővel.
PgUp, Ctrl-R	Feljebb megy egy képernyővel.
End	A sor végére áll.
Home	A bal margóra áll a sorban.
Backspace	Letörli a megelőző karaktert.
Tab	A következő tabulátorpozícióra lép.
Shift-Tab	A megelőző tabulátorra lép.
Return	Sort tör és a következő sorba lép.
Esc, Ctrl-Q	A módosításokat figyelmen kívül hagyja, és befejezi a szerkesztést.
Del, Ctrl-G	Törli a kiválasztott részt.
Ins, Ctrl-V	Átvált beszúró és átíró üzemmód között.
Ctrl- →	A következő szó elejére áll.
Ctrl- ←	A megelőző szó elejére áll.
Ctrl-PgDn	A szöveg végére áll.
Ctrl-PgUp	A szöveg elejére áll.
Ctrl-End, Ctrl-W	Kilép és elmenti a munkát.
Ctrl-Return	Elmenti a munkát.
Ctrl-Backspace	Törli a megelőző szót.
F6	Szövegblokk és egyéb képernyőobjektum kijelölésére szolgál a navigáló követő használatával. Csak szövegszerkesztés (memo-mező, programírás, stb) vagy tervezés (nézet, címke, lista, stb) során használható.

Vezérlőbillentyű	Magyarázat
F7	A kijelölt (F6) szövegblokk és egyéb képernyőobjektum áthelyezésére szolgál a navigáló billentyűk követő használatával. Csak szövegszerkesztés (memo-mező, programírás, stb) vagy tervezés (nézet, címke, lista, stb) során használható.
F8	A kijelölt (F6) szövegblokk és egyéb képernyőobjektum másolására szolgál a navigáló billentyűk követő használatával. Csak szövegszerkesztés (memo mező, programírás, stb) vagy tervezés (nézet, címke, lista, stb) során használható.
Shift-F5	Megkeresi a megadott szöveg első előfordulását.
Shift-F4	Megkeresi a Ctrl-K-F billentyűkkel megtalált szöveg következő előfordulását
Ctrl-K-R	A szerkesztett fájlba beolvas egy másik fájlt.
Ctrl-K-W	A szerkesztett fájlt - illetve ha volt kijelölt szöveg, akkor azt - kiírja egy másik fájlba.
Ctrl-N	A sort ketté töri az aktuális kurzorpozícióban.
Ctrl-T	Minden karaktert töröl az adott pozíciótól a következő szó elejéig.
Ctrl-Y	Törli az egész sort.

J) Fájltypusok és -kiterjesztések

Kiterjesztés	Leírás
.\$\$\$	Munkafájl, a művelet nem fejeződött be rendben.
.app	Alkalmazás-generátor meghatározásait tartalmazó fájl.
.bak	Program, eljárás vagy adatfájl elmentett változata.
.bar	Vonalmenü meghatározásait tartalmazó fájl, amely az Alkalmazás-generátorral készül.
.bch	Eljárásköteg (Batch) meghatározásait tartalmazó fájl, amely az Alkalmazás-generátorral készül.
.bin	Gépi kódú (bináris) fájl.
.cat	Katalógusfájl.
.cht	CHART-MASTER fájl, a dBASE IV/CHART-MASTER Bridge program használja.
.cod	Adatmegjénítési minta (PICTURE) forráskódja.
.cpt	Titkosított memo-fájl, a jelszó állománnyal (.crp) együtt használható.
.crp	Jelszó fájl, amit a PROTECT hoz létre.
.cvt	A többfelhasználós rendszernek megfelelően átalakított (kiegészített) adatfájl (.dbf). Létrehozása a CONVERT TO paranccsal történik.
.db	Konfigurációs fájl, a dBASE IV indulásakor ez alapján állítja be a fontosabb paramétereit.
.db2	Átnevezett régi dBASE III PLUS adatfájl (.dbf). Az IMPORT, EXPORT utasításoknál használható.
.dbf	Adatfájl.
.dbo	Tárgykódra lefordított program- vagy eljárásfájl.
.dbt	Memo-fájl. Mindig egy adatfájlhoz tartozik.
.def	Kiválasztást leíró fájl.
.dif	Adatcsere-formátum (Data Interchange Format) vagy VisiCalc fájl, APPEND FROM / COPY TO utasításoknál használható.
.doc	Dokumentáció fájl, amely az Alkalmazás-generátorral készül.
.fil	Fájl-lista meghatározásait tartalmazó fájl.
.fmo	Tárgykódra lefordított (.fmt) képernyőformátum-fájl.
.fmt	Képernyőformátum meghatározásokból (.scr) generált forrásprogram-utasításokat tartalmazó képernyőformátum fájl.

Kiterjesztés	Leírás
.fr3	Átnevezett régi dBASE III PLUS listaformátum fájl (.frm).
.frg	Listaformátum meghatározásokból (.frm) generált forrásprogram-utasításokat tartalmazó listaformátum fájl.
.frm	Listaformátum-meghatározásokat tartalmazó fájl.
.fro	Tárgykódra lefordított (.frg) listaformátum-fájl.
.fw2	Framework számolótábla-fájl. Az IMPORT, EXPORT utasításoknál használható.
.gen	Adatmegjelenítési (PICTURE) minta fájl (lásd: .cod).
.key	Elmentett billentyű-makro könyvtár.
.lb3	Átnevezett régi dBASE III PLUS címke meghatározásokat tartalmazó fájl (.lbl).
.lbg	Címkeformátum meghatározásokból (.lbl) generált forrásprogram-utasításokat tartalmazó címkeformátum-fájl.
.lbl	Címkeformátum-meghatározásokat tartalmazó fájl.
.lbo	Tárgykódra lefordított (.lbg) címkeformátum-fájl.
.log	Tranzakció során keletkező naplófájl.
.mdx	Összetett indexfájl.
.mem	Elmentett változókat tartalmazó fájl.
.ndx	Egyszerű indexfájl.
.npi	Lista-, képernyő, címkeformátum-fájlhoz tartozó minta-értelmező.
.pop	Rolómenü meghatározásait tartalmazó fájl, amely az Alkalmazás-generátorral készül.
.pr2	Nyomtatómeghajtó (printer driver) fájl. Nyomtatómeghajtókat a dBASE IV installációs lemeze tartalmaz, azok onnét másolhatóak át.
.prf	Nyomtatási jellemzőket (pl: laphossz, betű típus) tartalmazó fájl.
.prg	Forráskódú dBASE IV program- vagy eljárásfájl.
.prs	Forráskódú dBASE IV/SQL program- vagy eljárásfájl.
.prt	Nyomtató helyett fájlba irányított adatokat tartalmaz. Az adatok az ASCII típusú fájlban nyomtatási formátumban vannak. A DOS Print utasításával ki lehet nyomtatni.
.qbe	Nézetfájl forráskódban.
.qbo	Tárgykódra lefordított (.qbe) nézet forrásfájl.
.qry	Lekérdezőfájl.

J) Fájltypusok és -kiterjesztések

Kiterjesztés	Leírás
.rpd	Az IMPORT, EXPORT utasításoknál mentésre használható gyorsfájl(RapidFile).
.sc3	Átnevezett régi dBASE III PLUS képernyőformátum-fájl (.scr).
.scr	Képernyőformátum meghatározásait tartalmazó fájl.
.str	Programstruktúra meghatározását tartalmazó fájl, amely az Alkalmazás-generátorral készül.
.t44/w44	Belső munkafájl, amelyet a SORT és INDEX utasítás használ.
.tbk	Memo-fájl elmentett változata.
.txt	Szabványos ASCII szövegfájl.
.upd	Módosító nézetfájl forráskódban.
.upo	A tárgykódra lefordított módosító nézetfájl (.upd).
.val	Érték meghatározásokat tartalmazó fájl, amely az Alkalmazás-generátorral készül.
.vue	Elmentett adatfájl-környezet.
.win	Elmentett ablak.
.wks	Lotus 1-2-3 fájl, amely az APPEND FROM / COPY TO utasításoknál használható.

J-1. táblázat

ÁLLOMÁNYKAPCSOLATOK ÉS KOMPATIBILITÁS

Típus	dBASE IV kiterjesztés	Leford. fájl	Alulról kompa- tibilitás	Átnevezett dBASE III PLUS fájl
katalógus	.cat		igen	
adatfájl	.dbf		igen	
memo	.dbt		igen	
képernyőform. meghatározások	.scr		nincs	.sc3
képernyő- formátum	.fmt	.fmo	nincs	
listaformátum meghatározások	.frm		nincs	.fr3
listaformátum	.frg	.fro	nincs	
címkeformátum meghatározások	.lbl		nincs	.lb3
címkeformátum	.lbg	.lbo	nincs	
nézet	.qbe	.qbo	nincs	
módosító nézet	.upd	.dbo	nincs	
SQL program	.prs	.dbo	nincs	
minta(PICT)	.gen		nincs	
makro	.key		nincs	
program	.prg	.dbo	nincs	

J-2. táblázat

K) Az adatfájlok belső szerkezete

Az adatfájlok fejrészből, rekordokból, törlés-jelzőkből és fájlvége jelből állnak. A fejrész az fájl szerkezetéről tartalmaz információt, a rekordok az aktuális adatokat tartalmazzák. Rekordonként egy-egy bájt a törlés-jelző számára foglalt.

K-1. A fejrész szerkezete

Bájt	Tartalom	Értelmezés
0	1 bájt	dBASE IV fájl-azonosító, 0-2 bitek jelzik a verziószámot, 3-5 bitek foglaltak az SQL számára, 6-7 bitek jelzik, hogy az adatfájlhoz memofájl tartozik e.
1-3	3 bájt	Az utolsó módosítás dátuma ÉÉHHNN formában.
4-7	32 bites szám	Az adatfájlban lévő rekordok száma.
8-9	16 bites szám	A fejrészben levő bájtok száma.
10-11	16 bites szám	Egy rekordban levő bájtok száma.
12-13	2 bájt	Fenntartott.
14	1 bájt	Befejezetlen tranzakciót jelző bájt. Értéke 00, ha tranzakció nincs folyamatban az adatfájlban, egyébként 01.
15	1 bájt	Titkosítást jelző bájt. A 01 érték jelzi, ha az adatfájl titkosított. Ha az értéket mesterségesen 00 állítjuk, akkor a dBASE IV nem jelez hibát illegális hozzáférési kísérletnél, de az illegális felhasználó továbbra sem tudja az eredeti (nem titkosított) információt visszanyerni.
16-27	12 bájt	Többfelhasználós alkalmazásra fenntartott.
28	1 bájt	Összetett indexfájl jelző bájt. Értéke 01, ha az adatfájlhoz létrehozásakor létrejött ilyen indexfájl, illetve 00, ha nem.
29-31	3 bájt	Fenntartott.
32-m	32 * n bájt	Mező leíró tömb. n = a mezők létszáma, m = (n + 1) * 32 . Lásd a következő táblázatot.
m + 1	1 bájt	Mezőterminátor bájt, értéke hexadecimális 0D.

K-1. táblázat

K-2. A mezőleíró tömb

Bájt	Tartalom	Értelmezés
0-10	11 bájt	A mező neve (a nem használt rész nullákkal feltöltve).
11	1 bájt	A mező típusának ASCII kódja. Legális értékek : "C", "D", "F", "L", "M", "N".
12-15	4 bájt	Fenntartott.
16	1 bájt	A mező hossza bináris formában.
17	1 bájt	A mező tizedesjegyeinek száma bináris formában.
18-19	2 bájt	Fenntartott.
20	1 bájt	A munkaterület azonosítója.
21-22	2 bájt	Fenntartott.
23	1 bájt	Jelzi, hogy a mező SET FIELDS TO utasítással engedélyezett vagy tiltott.
24-31	8 bájt	Fenntartott.

K-2. táblázat

Az adatbázisban az adatrekordok folyamatosan követik a fejrészt. A rekordok között nincs elválasztó karakter. Egy rekord a következőképpen épül fel:

- Minden egyes rekord elején megtalálható a törlés-jelző bájt. Tartalma 2AH (csillag karakter), ha az adott rekord törölt, illetve 20H (space karakter), ha nem.
- A rekordban a törlés-jelző bájt után folyamatosan következnek a mezők, abban a sorrendben, ahogy az a mezőleíró tömbben található.
- A mezőket semmilyen jel sem választja el egymástól. A mezők helyét a rekordban a mezőleíró tömb alapján lehet meghatározni.
- Az adatbázis végén mint különleges rekord, a fájlvége jel található. Hossza 1 bájt, tartalma mindig 1AH (ASCII 26).

K-3. Memo-mezők és memofájlok

A memofájlok blokkok sorozatából állnak. Az egyes blokkok méretét a fájl létrehozásakor érvényben lévő SET BLOCKSIZE TO utasítás határozza meg.

A blokkokat sorszámuk alapján azonosíthatjuk. A számozás nullával kezdődik és egyesével növekszik. A nullás blokknak speciális funkciója van. Fejrészként ebben a blokkban tárolja a dBASE IV a memofájlból lévő összes (beleértve a fejrészt is) lefoglalt blokkok számát. Ez a szám a fejrész elején 4 bájt helyre kerül el, bináris formában.

Amikor egy memo-mezőt használunk, akkor az információt nem magában a memo-mezőben tároljuk, hanem a memo-mezőhöz tartozó blokkban. E helyett a memo-mező csak egy hivatkozást tartalmaz a megfelelő blokkra. A hivatkozás természetesen a blokk számának segítségével történik. A blokkszám ASCII formában szerepel a memo-mezőben. Ha egy memo-mező nem tartalmaz adatot — és így nem is tartozik hozzá blokk —, akkor szóközöket (20H) tartalmaz.

Ha egy memo-mező több mint 512 bájtból áll, akkor 1 blokk nem elegendő a tárolására. Ekkor a dBASE IV automatikusan a memo-mezőhöz tartozó blokkoktól kezdődően folyamatosan foglalja le a további blokkokat. Mezővége-jelként a mezőhöz tartozó utolsó blokkban két darab 1AH bájt zárja le az adatokat. Például: Ha egy memo-mező 520 bájtból épül fel és a hozzátartozó első blokk sorszáma a 2, akkor a 2-es blokk teljesen ki van töltve, míg a 3-as blokkból 8 bájt van felhasználva. A 3-as blokk 8. és 10. bájtjának tartalma 1AH és a többi 502 bájt pedig kihasználatlan. Lásd a következő táblázatot.

Ha a memo-mező tartalma megváltozott, akkor ennek megfelelően a blokkszáma is megváltozhat, ami maga után vonja a hivatkozási szám módosulását. Ha a memofájlból van törölt adat, a dBASE IV képes azt újra felhasználni amikor új adat kerül az fájlba.

Blokk	Bájt	Tartalom
	1	04H Blokkszám
	2	00H
	3	00H
	4	00H
	.	
	.	
	.	
	512*	
1	1	Első memo-mező 312 bájt adata
	.	
	.	
	.	
	312	
	313	1AH Memo-mező vége
	314	1AH
	315	Nem használt
	.	
	.	
	.	
	512*	
2	1	Második memo-mező 520 bájt adatának első 512 bájtja
	.	
	.	
	.	
	512*	
3	1	Második memo-mező 520 bájt adatának utolsó 8 bájtja
	.	
	.	
	.	
	8	
	9	1AH Memo-mező vége
	10	1AH
	11	Nem használt
	.	
	.	
	.	
	512*	

K-3. táblázat

* Feltéve, hogy a memofájl létrehozásakor a SET BLOCKSIZE TO utasítás így állította be.

L) A dBASE IV konfigurálása

L-1. Konfiguráló utasítások

A konfigurálás formája :

< konfiguráló kulcsszó > = < érték >

Példa:

FILES = 20.

Rövidítések:

Alapért. = alapértelmezés

Me. = mértékegység

Kulcsszó	Érték	Jelentés
BUCKET	1-31 közötti szám M.e.: Kbájt Alapért.: 2K	A PICTURE és a RANGE paraméterek által felhasználható memóriaterület mérete.
COMMAND	dBASE IV utasítás Alapért.: ASSIST	Automatikusan végrehajtásra kerül a megadott utasítás a dBASE IV indításakor. DO utasítással más programok is elindíthatók.
<i>Példa:</i>		
COMMAND = ASSIST esetén a CONTROL CENTER jelentkezik be.		
COMMAND = DO <i>berszamf</i> esetén a bérszámfejtő programrendszer indul el.		
DO	1-256 közötti szám Alapért.: 20	A DO utasítással meghívható egymásba ágyazott eljárások maximális száma.
EEMS	ON/off	ON esetén a kibővített memóriát is használatba veszi a dBASE IV.
EXPSIZE	100-2000 közötti szám M.e.: bájt Alapért.: 100	A kifejezések kiszámításához lefoglalt memóriaterület nagysága. Bonyolultabb kifejezések kiszámításakor ez szűknek bizonyulhat.

Kulcsszó	Érték	Jelentés
FASTCRT	ON/off	ON állásban kiküszöbölhető a képernyőn tapasztalható esetleges vibrálás. IBM PC színes grafikus adapter (CGA) használata esetén kikapcsolható (OFF), mert az nem vibrál.
FILES	15-99 közötti szám Alapért.: 99	Az egyidejűleg használható fájlok maximális száma. Nem haladhatja meg a DOS <i>config.sys</i> fájlban meghatározott értéket.
GETS	35-1023 közötti szám Alapért.: 128	Az egyszerre aktivizálható (egy READ utasításhoz tartozó) GET utasítások száma.
INDEXBYTES	2-128 közötti szám M.e.: Kbájt Alapért.: 2K	Kijelölheti az indexfájlokhoz tartozó információk (pl: kulcskifejezés) által lefoglalt memóriaterület mértékét.
PDRIVER	Fájlnév Alapért.: GENERIC.PR2	Kiválasztja a nyomtatómeghajtó fájlt, ami a nyomtató karakterkészletét és az egyéb nyomtatóvezérlő karaktereket tartalmazza. Csak a dBSETUP futtatásakor megjelölt nyomtatómeghajtók közül lehet választani.

Kulcsszó	Érték	Jelentés
PRINTER	Alapért.: Irányító fájl GENERIC.PR2 DEVICE=LPT1	Nyomtatók kijelölése az alábbi szintakszis szerint. Egyszerre négy különböző nyomtató jelölhető ki, amelyhez egyenként öt-öt betűtípus rendelhető.

PRINTER <nyomtató száma> = <nyomtatómeghajtó neve>
[NAME <nyomtatónév>] [DEVICE <egység neve>]

A nyomtató száma 1-4 közötti értéket vehet fel.

A *nyomtatónév* tetszés szerint választott megnevezés lehet, csak a későbbi tájékoztatást szolgálja.

Példa:

```
PRINTER 1=GENERIC.PR2 NAME "EPSON FX 850" DEVICE COM1
```

```
PRINTER <nyomtató száma>  
FONT <betűtípus száma> = <kezdő kód> , <záró kód>  
[ NAME <betűtípus neve> ]
```

A betűtípus száma 1-5 közötti értéket vehet fel. A <kezdő kód> és a <záró kód> értékeket a nyomtató üzemeltetési kézikönyve alapján lehet meghatározni.

A betűtípus neve tetszés szerinti, a későbbi tájékoztatás célját szolgálja.

Példa:

```
PL PRINTER 1 FONT 1 = {ESC}(8U,{ESC}({#@, NAME "Times 8 pontos"
```

Ezeket a specifikációkat a dBSETUP program segítségével is létre lehet hozni. Lásd még ??? utasítás STYLE cikkelyét.

PROMPT	C-kifejezés Hossz: max. 19	Az interpreter készenlétét jelző (prompt) pont karakter cserélhető más, akár több karakterre is. Pl: PROMPT = Mi a parancs ?
RESETCRT	ON/off	ON állásban a meghívott (CALL) programok által esetlegesen átállított képernyő-üzemmód visszaáll a hívás előtti állapotba. Pl: Híváskor: EGA43, a hívott program futása alatt EGA25, utána ismét EGA43.

Kulcsszó	Érték	Jelentés
SQLDATABASE	SQL adatbázis neve	Indításkor, ha az SQL üzemmód érvényes (SQL=ON) a rendszer a megjelölt SQL adatbázist aktiválja.
SQLHOME	Elérési út	Kijelöli az SQL adatbázis helyét. (alkönyvtár)
<i>Példa:</i>		
<pre>SQL=ON SQLHOME=C:\DBASE\SQL SQLDATABASE=ADATBANK</pre>		
TEDIT	editor neve	Kijelölhető a MODIFY COMMAND utasításban használni kívánt szövegszerkesztő. A MODIFY COMMAND utasítás kiadásakor (és a Control Center egyes pontjain is) ez a szövegszerkesztő lép működésbe. Az <i>editornév</i> csak az operációs rendszer által végrehajtható fájlnev (.bat, .com, .exe) lehet.
WP	editor neve	Kijelölhető a memo-mezők szerkesztéséhez használni kívánt szövegszerkesztő. A memo-mezők adatbevitelkor ez a szövegszerkesztő lép működésbe. Az <i>editornév</i> csak az operációs rendszer által végrehajtható fájlnev (.bat, .com, .exe) lehet.
MVMAXBLKS	1-150 közötti szám Alapért.: 50	Beállítható a változók által elfoglalható memóriablokkok maximális száma.

Kulcsszó	Érték	Jelentés
MVBLKSIZE	25-100 közötti szám Alapért.: 50	Meghatározza, hogy egy blokkon belül hány változó lehet. Egy változó 56 bájtot foglal el a memóriából.
<i>Példa:</i>		
MVMAXBLKS = 10 és MVBLKSIZE = 50 esetén 500 változó használható egyidőben, és ez a memóriából 28000 bájt helyet foglal el.		
RTMAXBLKS	1-150 közötti szám Alapért.: 10	Programok futása alatt a változók és szimbólumok számára használható memóriablokkok száma jelölhető ki. Hasonló az MVMAXBLKS működéséhez, de programfutás idejére vonatkozik.
RTBLKSIZE	25-100 közötti szám Alapért.: 50	Programok futása alatt a memóriablokkon belüli változók és szimbólumok száma állítható be.
CTMAXSYMS	1-5000 közötti szám Alapért.: 500	Fordítás során használható szimbólumok maximális számát állítja be.

L-2. Funkciós billentyűk

Meghatározható a funkciós billentyűkhöz rendelt sztring, amely dBASE utasításokat is tartalmazhat. Felhasználható billentyűk (billentyűkombinációk) F2...F10, Shift-F1..Shift-F9, Ctrl-F1...Ctrl-F10. Az F1 funkciós billentyű a HELP, a Shift-F10 funkciós billentyű a makrók számára foglalt.

Szintakszis :

< Funkciós billentyű > = < kifejezés >

Példa:

Ctrl-F5 = CREATE APPLICATION

Alapértelmezés

Billentyű	Aktuális tartalom
F1	HELP / lefoglalt /
F2	ASSIST
F3	LIST
F4	DIR
F5	DISPLAYSTRUCTURE
F6	DISPLAYSTATUS
F7	DISPLAYMEMORY
F8	DISPLAY
F9	APPEND
F10	EDIT
Shift-F10	Makró-menü (foglalt)
Ctrl...	Nincs meghatározva
Shift..	Nincs meghatározva

L-3. Környezeti jellemzők, szolgáltatások beállítása

A kulcsszavak és működésük (néhány kivétellel) egyezik a megfelelő SET utasításokéval. Részletesebb információkat lásd ott.

Kulcsszó	Érték	Jelentés
ALTERNATE	on/OFF fájlnev	A teljes-képernyős megjelenítő utasítások kivételével, minden ki-menő adatot a <i>fájlnev</i> . ASCII szövegfájlba irányít. Ha az nem létezik, létrehozza. A létezőt felülírja. Ha a fájlnev kijelölésre került, automatikusan ON állásba kapcsol az interpreter.
AUTOSAVE	on/OFF	Adatkezelés után azonnal elmenti a rekordot.
BELL	Frekvencia, időtartam.	Hangjelzés beállítása mezővégnél illetve hibüzenetnél.
BLOCKSIZE	1-32 közötti szám. Alapért.: 1 M.e.: 32 bájt	Memo-mezők és összetett index-fájlok blokkméretének beállítása. Pl: 3*32 bájt.
BORDER	SINGLE/DOUBLE/PANEL /NONE/<keretsztring> Alapért.: SINGLE.	Keretező karakterek beállítása.
CARRY	on/OFF	APPEND és INSERT utasítások esetén az új rekord mezőit az előző rekord mezőértékeivel tölti fel.
CATALOG	on/OFF fájlnev	Katalógusnyilvántartás bekapcsolása. Megnyitja a <i>fájlnev</i> katalógust, vagy ha az nem létezik, létrehozza.
CENTURY	on/OFF	ON állásban az évezred, évszázad feltüntetésre kerül a dátumokban.
CLOCK	on/OFF sor, oszlop	Rendszeridő megjelenítésének bekapcsolására illetve képernyőn való elhelyezésére szolgál.

Kulcsszó	Érték	Jelentés
COLOR	on/off Alapért.: installáláskor.	Színes és monochrom monitor közötti váltás kapcsolója.
COLOR OF NORMAL TITLES MESSAGES BOX INFORMATION HIGHLIGHT FIELDS	előtér/háttér	Normál szövegek színe. Fejlécek színe. Üzenetek színe. Keretek színe. Óra, státusz sor színe. Menüsorok színe. Beolvasandó adatok színe.
CONFIRM	on/OFF	ON állásban adatbevitel esetén az Enter billentyűvel kér jóváhagyást.
CONSOLE	ON/off	OFF esetén, a képernyőn nem jelennek meg a nyomtatóra kerülő kimenő adatok.
CURRENCY	C-kifejezés Alapért.: \$	Pénznemet jelző sztring meghatározása. Pl: "Ft"
	left/RIGHT	A pénznem helyének meghatározása.
DATE	AMERICAN/ANSI /BRITISH/FRENCH /GERMAN/ITALIAN /JAPAN/USA /MDY/DMY/YMD	A dátum megjelenési formájának beállítása.
DEBUG	on/OFF	ON esetén a SET ECHO kimenő adatait nyomtatóra küldi.
DECIMALS	0-18 közötti szám Alapért.: 2	Tizedesjegyek számát határozza meg.
DEFAULT	lemezmeghajtó neve	Az alapértelmezés szerinti lemezmeghajtót jelöli ki.
DELETED	on/OFF	OFF esetén a törlésre kijelölt rekordokhoz nem lehet hozzáférni.

Kulcsszó	Érték	Jelentés
DELIMITERS	on/OFF C-kifejezés Alapért.: ":"	Input mezők határait jelölő karakter kijelölése, illetve a határoló karakterek megjelenítésének ki- és bekapcsolása.
DESIGN	ON/off	OFF esetén a tervező modulok (CREATE QUERY, stb) nem aktivizálhatók.
DEVELOPMENT	ON/off	ON esetén a forrásprogramot összehasonlítja a megfelelő tárgyprogrammal, és ha az szükséges újrafordítja.
DEVICE	SCREEN/PRINTER /FILE < fájlnev >	Kimenő adatok rendeltetési helyének beállítása.
DISPLAY	MONO/COLOR/EGA25 /EGA43/MONO43	Megfelelő monitortípus, képernyőüzemmód kiválasztása.
ECHO	on/OFF	ON állásban a program futása közben megjelennek a forrásprogram sorai.
ENCRYPTION	on/OFF	Létrehozandó fájlok titkosítására szolgál. Csak akkor működik, ha a védelmi rendszer be van kapcsolva.
ESCAPE	ON/off	ON esetén a programok futása az Esc billentyűvel megszakítható.
EXACT	on/OFF	A sztring-összehasonlítás szigorúságát állítja be.
EXCLUSIVE	on/OFF	ON esetén az adatfájlokat kizárólagos használatra nyitja meg.
FULLPATH	on/OFF	ON esetén a fájlokat az elérési úttal együtt írja ki.
FUNCTION	N-kifejezés, C-kifejezés	Funkciós billentyűkhöz kifejezés hozzárendelése. (lásd funkciós billentyűk). Pl: FUNCTION =1, chr(143)

Kulcsszó	Érték	Jelentés
HEADING	ON/off	OFF állásban a DISP, LIST, SUM, AVERAGE utasítások esetén a fejléc nem jelenik meg.
HELP	ON/off	OFF állásban, hiba esetén nem jelenik meg automatikusan a help ablak.
HISTORY	ON/off	OFF esetén, az utasításokat nem tárolja az utasításpufferben, így azok nem kereshetők vissza.
	0-16000 közötti szám Alapért.: 20	Az utasításpuffer méretének meghatározása.
HOURS	12/24 Alapért.: 12	A rendszeridő megjelenési formájának beállítása.
INSTRUCT	ON/off	A menürendszerrel működő utasítások (APPEND, BROWSE, stb) promptjainak megjelenítését kapcsolja ki. (Tapasztalatunk szerint elvileg.)
INTENSITY	ON/off	OFF állásban a GET utasítások adatai is normál színnel jelennek meg.
LOCK	ON/off	OFF esetén kikapcsolja az automatikus zárolást.
MARGIN	0-254 közötti szám Alapért.: 0	A nyomtatás alampargója állítható be.
MEMOWIDTH	8-32000 közötti szám Alapért.: 0	Memo-mezők nyomtatási szélességének beállítása.
MENUS	ON/off	A dBASE III PLUS-szal való kompatibilitás miatt szerepel. A dBASE IV-ben nincs hatása.
NEAR	on/OFF	ON állásban a kereső utasítások sikertelensége esetén a rekordmutató a fájl vége helyett a legjobban hasonlító kulcsú rekordra áll.

Kulcsszó	Érték	jelentés
ODOMETER	1-200 közötti szám Alapért.: 1	A csoportos rekordműveleteknél a kijelzett rekordszám-megjelenítés lépésközét állítja be.
PATH	Elérési utak listája (Max. 60 karakter).	A fájlok keresési helyeit határozza meg, ha azok nem az aktuális alkönyvtárban vannak.
PAUSE	on/OFF	SQL üzemmódban az ON állás az adatmegjelenítésnél képernyőnként szünetet tart.
POINT	C-kifejezés Alapért.: "."	A tizedespont karakter beállítása.
PRECISION	10-20 közötti szám Alapért.: 16	Meghatározza, hogy a számítások hány számjegy pontossággal történjenek.
PRINTER	on/OFF	ON esetén az adatfolyam a nyomtatón is megjelenik.
REFRESH	0-3600 közötti szám M.e.: másodperc Alapért.: 0	Hálózati alkalmazás esetén a képernyőn megjelenő adatok felfrísítésének időintervalluma.
REPROCESS	-1-32000 közötti szám Alapért.: 0	Hálózati alkalmazás esetén a zárolt rekordhoz való hozzáférés próbálkozásainak száma.
SAFETY	ON/off	Fájlok véletlen törlése elleni védelem.
SCOREBOARD	ON/off	ON esetén néhány tájékoztató állapotinformáció jelenik meg.
SEPARATOR	C-kifejezés Alapért.: ","	Háromhelyiértékenkénti elválasztó karakter meghatározása
SPACE	ON/off	ON esetén a ??? parancsoknál a megjelenített adatok között automatikusan egy szóközt hagy.
SQL	on/OFF	ON állásban induláskor SQL üzemmódba lép a dBASE IV.
STATUS	on/OFF	OFF esetén a státusz-sor nem jelenik meg.

Kulcsszó	Érték	Jelentés
STEP	on/OFF	ON esetén a programokat utasításonkénti lépésekben hajtja végre.
TABS	C-kifejezés Alapért.: " "	_tabs rendszerváltozó értékét, azaz a tabulátorpozíciókat határozza meg.
TALK	ON/off	ON esetén megjeleníti a dBASE IV utasítások eredményét a képernyőn.
TRAP	on/OFF	ON esetén hiba előfordulásakor a debugert aktíválja.
UNIQUE	on/OFF	ON esetén a duplikált indexkulcsú rekordokból csak az elsőt veszi figyelembe.
VIEW	fájlnév	Aktuálissá teszi a <i>fájlnév</i> nézetfájlt.

M) A dBASE IV technikai jellemzői

Adatfájl (.dbf)

Rekordok száma:	1 milliárd
Bájtok száma:	1 milliárd
Rekord méret:	4000 bájt
Mezők száma:	255

Indexfájl

Indexek száma többszörös indexfájl (.mdx) esetén:	47
Blokkméret:	16384 bájt, (alapértelmezés: 2048 bájt)

Mező méretek

Karakter mező:	254 bájt
Dátum mező:	8 bájt
Logikai mező:	1 bájt
Numerikus (N típusú) mező:	20 számjegy
Numerikus (F típusú) mező:	20 számjegy
Mező neve:	10 karakter

Tömbök

Dimenziók száma:	2
Tömb elemeinek száma (sor * oszlop):	1170

Többfelhasználós eljárások

Zárolások száma:	max. 50 (fájl és rekord együtt)
Újra próbálkozások száma:	max. 32000
Felfrissítési idő intervalluma:	max. 3600 másodperc

Fájlműveletek

Összes megnyitott fájl:	99
Megnyitott adatfájl (.dbf):	10
Megnyitott memofájl aktív adatfájlonként:	1
Megnyitott indexfájl aktív adatfájlonként:	10
Megnyitott képernyőformátum- fájl aktív adatfájlonként:	10
Megnyitott eljárásfájl futásonként:	1

(Megjegyzés: A felsorolt értékek csökkenhetnek a config.sys, config.db és a rendelkezésre álló memória függvényében)

Numerikus pontosság

Lebegőpontos (F típusú) számoknál:	15.9 számjegy
Legnagyobb ábrázolható szám:	0.9.E + 308
Legkisebb ábrázolható szám:	0.1.E-307
Decimális (N típusú) számoknál:	10-től 20 számjegy. A SET PRECISION TO < kifejezés > utasítástól függően.
Legnagyobb ábrázolható szám:	0.9.E + 308
Legkisebb ábrázolható szám:	0.1.E-307

Szimbólumok

Memória változók:	max.15000
Memória változók alapértelmezés:	500
Futás alatti szimbólumok:	max. 15000
Futás alatti szimbólumok alapértelmezés:	max. 500
Fordítás alatti szimbólumok:	max. 5000
Fordításidő szimbólumok alapértelmezése:	max. 500

(Megjegyzés: A felsorolt értékek csökkenhetnek a rendelkezésre álló memória függvényében)

Szövegszerkesztő

Sorok száma:	32000
Sorhossz:	
MODIFYCOMMAND	1024
MODIFY FILE	1024
MEMO FIELDS	1024
SQL (F9)	1024
HISTORY	1024
REPORT	255

Képernyőformátumok

Sorok száma: 32767
Szélesség: 80 karakter

Listaformátum

Szélesség: 80 karakter
Adatfájl (.dbf): 9
Indexfájl adatfájlanként: 10
Kapcsolatok (relációk) száma: 9
Listaformátumok száma adatfájlanként: korlátlan
Oldalak száma: 32767
Egymásbaágyazott csoportok száma: 44
Mezők száma: az elérhető memória korlátozza
Másolatok száma: 32767

Címkék

Szélesség: 255 karakter
Hossz: 255 sor
Egy sorban lévő címkék száma: 15
Másolatok száma: 32767

QBE

Összekapcsolt fájlok: 8

SQL

Egymással összekapcsolható táblák száma: csak a rendelkezésre álló memória korlátozza.
sql-kurzorok száma: 10
Indexek száma táblánként: 47
SQL sorhossz: 1024 karakter

Alkalmazás-generátor

A munkafelületen lévő komponensek száma: a rendelkezésre álló memória és a komponensek bonyolultsága korlátozza
Az editor területe: 4096 bájt

Egyéb korlátok

Parancssor hossza:	255 bájtt az interpreternél, 1024 bájtt a szerkesztő ablakban.
Egymásba ágyazható eljárások max. száma:	a DO paraméter határozza meg a CONFIG.DB-ben.
Eljárások száma programonként:	963
Lefordított eljárás mérete:	65520 bájtt
Aktív eljárások száma:	a rendelkezésre álló memória korlátozza.
Egyszerre megadható rendezési szempontok száma a SORT-nál:	16
GET-ek száma egy .fmt fájlban:	2000
Kinyomtatott oldalak száma:	32767
Billentyű makrók száma:	35
Egyszerre betölthető binárisfájlok száma:	16
Konfigurált nyomtatók száma:	4
Betűtípusok száma nyomtató vezérlőnként:	5
Munkaterületek száma:	10
Programozható funkcióbillentyűk száma:	29
Rekurzív hívások száma:	a rendelkezésre álló memória korlátozza.

Tárgymutató



&	F152
.prs	210,251
?/??	F45
???	F85
@ ...CLEAR	F50
@ ...FILL	F50
@ ...SAY ...GET	F46
@ ...TO	F51
_align	F157
_box	F157
_indent	F158
_lmargin	F158
_padvance	F158
_pageno	F159
_pbpage	F159
_pcolno	F159
_pcopies	F160
_pdriver	F160
_pecode	F160
_peject	F161
_pepage	F161
_pform	F161
_plength	F162
_plineno	F162
_ploffset	F162
_ppitch	F163
_pquality	F163
_pscode	F163
_pspacing	F164
_pwait	F164
_rmargin	F164
_tabs	F165
_wrap	F165



ablak	78
Ablak-kezelés	78-80
ABS	F119
ACCEPT	F57

ACCESS	F150
ACOS	F122
ACTIVATE MENU	F74
ACTIVATE POPUP	F74
ACTIVATE SCREEN	F74
ACTIVATE WINDOW	F75
adatbázis	33,211
adatbiztonság	258
adatbiztonsági szint	268
adatfájl	33
Lásd: adatbázis-fájl	
adatfolyam	F45,64,F157
Adatkatalógus	68-70
adatok titkosítása	272
SQL adattípusok	213
adattípus kódok	29
adminisztrátori jelszó	274
aktuális adatfájl	F38
aktuális index	F35
aktuális munkaterület	F38
aktuális rekord	F34,F38,F43
alapindexfájl	F22,F35,F42,115
alaplmargin	65
alaptábla	209
alias	F125
Lásd: másodnév	
alkalmazás	F15,162
alkalmazási program	F15
ALL	225,238-239
ALTER TABLE	F167
ANY	238-239
APPEND FROM	F23
APPEND FROM ARRAY	F24
APPEND MEMO	F24
argumentum	F15
aritmetikai operátorok	
Lásd: operátorok	
ASC	F131
ASCII szövegfájl	F15
ASIN	F122
ASSIST	F86
AT	F131
ATAN	F122
ATN2	F122

Tárgymutató

AVERAGE F24
AVG 229,F190

B

bájt F15
BAR F141
bar menu 81
BCD - Binary Coded Decimal 30
BEGINTRANSACTION F63
BEGIN/ENDTRANSACTION 53
bejelentkezési képernyő 264
bejelentkezési név 259
BETWEEN 230
billentyűmakrók 103
BOF F15,F125
BROWSE F25

C

CALCULATE F26
Call F63,75,F152
CANCEL F63
CDOW F137
CEILING F119
CHANGE F27,F150
CHAR(n) 214,F170
CHR F132
cikkely F22
Ciklus 73
CLEAR F59
CLOSE F27,F167
CMONTH F137,F139
COL F146
collen 242-243
colname 242-243
coltype 242-243
COMPILE F86
COMPLETED F125
CONTINUE F43
Control Center menüsor 96
CONVERT F87
COPY FILE F27

COPYINDEXES F28
COPY MEMO F28
COPYSTRUCTURE F28
EXTENDED F29
COPY TAG F29
COPY TO F29
COPY TO ARRAY F30
COS F123
COUNT F30,229,F190
CREATE F31
CREATEDATABASE F168
CREATEINDEX F168
CREATESYNONYM F169
CREATETABLE 213,F170
CREATE vagy MODIFY
STRUCTURE F30
CREATE VIEW F172
CREATE VIEW .. FROM F32
CREATE/MODIFY
APPLICATION F82
CREATE/MODIFY
LABEL F82
CREATE/MODIFY
QUERY/VIEW F82
CREATE/MODIFYREPORT F83
CREATE/MODIFYSCREEN F83
CREATOR 243
csoport 264
CTOD F137

D

DATE F137,214,F171
DAY F138
DBCHECK 248,F172
DBDEFINE 247-248,F172
DBF F125
DEACTIVATE MENU F75
DEACTIVATE POPUP F75
DEACTIVATE WINDOW F76
DEBUG F87
debugger F15,91

Tárgymutató

DECIMAL(n,m)	213,F170
decimális	F16
DECLARE	F60
DECLARECURSOR	F173
DEFINE BAR	F77
DEFINE BOX	F88
DEFINEMENU	F78
DEFINE PAD	F78
DEFINE POPUP	F79
DEFINE WINDOW	F80
deklaráció	
Lásd: tömb	
DELETE	F32,F174
DELETE TAG	F32
DELETED	F126
DIFFERENCE	F153
dimenziók	
Lásd: tömb	
DIR	F88
DISKSPACE	F146
DISTINCT	225,F182
DMY	F138
Do	F64,75
Do Case	F64,73
Do While	F65,73
dot prompt	28
DOW	F138
DROPDATABASE	F174
DROPINDEX	F175
DROPSYNONYM	F175
DROPTABLE	F175
DROP VIEW	F175
DTOC	F138
DTOR	F123
DTOS	F139

E

EDIT	F33
egyszerű indexfile	
Lásd: index	
EJECT	F88
EJECT PAGE	F89

előjog-táblázat	269
Elágazás	71
elemi adatműveletek	53
elérési szint	269
elérési út	F16
eljáráshívás	F16,75
ementés	F17
EOF	F15
ERASE	F89
ERROR	F153
értékadó utasítások	F57-F58
Eseménycsapdák	71
EXISTS	240
EXP	F123
EXPORT	F33

F

főprogram	F16
fájl	23,33
fájl eleje	F15
fájlvége	F15,F232
FAT	F16
felhasználó	264
felhasználói csatolót	78
felhasználói profil	273
FETCH	F175
FIELD	F126
FILE	F126
file server	257
Lásd: hálózatvezérlő	
FIND	F43
FIXED	F119
FKLABEL	F146
FKMAX	F146
FLOAT	F119
FLOAT(n,m)	213,F170
FLOCK	F150
FLOOR	F119
forrásprogram	F16
FOUND	F127
FUNCTION	F65
FV	F153

Tárgymutató

G

GETENV	F147
GO/GOTO	F34
GRANT	244,F176
GRANTOR	243
gyűjtő függvények	228

H

hálózati erőforrás	263
hálózatvezérlő	257
háttértároló	F16
HAVING	233-234,F185
HELP	F89
holtpont	265

I

If	F66,71
IIF	F153
IMPORT	F34
IN	230-231,238-239
index	
aktuális index	F110
egyszerű indexfájl	37
keresési kulcs	37
összetett indexfájl	37
INDEX ON	F35
INKEY	F141
INPUT	F57
INSERT	F35,F178
INT	F120
INTEGER	213,F170
integritás	258
interaktív üzemmód	F16
interpreter	28
ISALPHA	F132
ISCOLOR	F147
ISLOWER	F132
ISMARKED	F127

ISUPPER	F132
IXNAME	243

J

SQL Jogok	
ALTER	244,F176
DELETE	244,F176
INDEX	244,F176
INSERT	244,F176
SELECT	244,F176
UPDATE	244,F176
WITH GRANT OPTION	244
JOIN WITH	F36
SQL joker-karakterek	231

K

kapcsolt menü	195
katalógusrendszer	68
KEEP	F187
keresési kulcs	
Lásd: index	
KEY	F127
kiterjesztés	F17
környezet	F16
kulcsszó	F17
kurzor	
CLOSE	252
DECLARE	252
FETCH	252
OPEN	252
sql-kurzor	251
WHERE CURRENT OF	253

L

LABEL FORM	F51
LASTKEY	F144
lebegőpontos számábrázolás	F17
LEFT	F132
LEN	F133

Tárgymutató

LIKE	F154,230-231
LINENO	F154
LIST/DISPLAY	F52
LIST/DISPLAY FILES	F53
LIST/DISPLAY HISTORY	F53
LIST/DISPLAY MEMORY	F53
LIST/DISPLAY STATUS	F54
LIST/DISPLAY STRUCTURE	F54
LIST/DISPLAY USERS	F54
LKSYS	F150
Load	F36,75
LOAD DATA	F179
LOCATE	F43
LOG	F123
LOG10	F123
LOGICAL	214,F171
logikai operátorok	
Lásd: operátorok	
logikai oszlop	
Lásd: számított oszlop	
login	264
LOGOUT	F89
lokális hálózatok	257
LOOKUP	F128
LOWER	F133
LTRIM	F133
LUPDATE	F128

M

M- > előtag

 Lásd: változó

makró	74
SQL másodnév	F42,217,237
MAX	F120,229,F191
MDX	F128
mdx-fájl	F22
MDY	F139
MEMLINES	F133
memo-mező	F233
memóriaváltozó	
Lásd: változó	
MEMORY	F147

mentés	F17
menü	81,F144
menürendszer felépítése	81-86
MESSAGE	F154
mező	23
MIN	F120,229,F191
minta	F20,F154
minták	F47
MLINE	F134
MOD	F120
MODIFY COMMAND/FILE	F84
módosító nézetfájl	
Lásd: nézetfájl	
mondat	F22
MONTH	F139
MOVE WINDOW	F80
munkafelület	96
munkaterület	F17

N

naplófájl	53
NDX	F129
ndx-fájl	F22
NETWORK	F151
nézetfájl	121-128
módosító nézetfájl	127
nézettábla	209,221
NOTE	F90
NUMERIC(n,m)	213,F170
nyomtatási köteg	65,F71
nyomtatómeghajtó	62

O

ON ERROR	F66,71
ON ESCAPE	F67
ON KEY	F68
ON PAD	F69
ON PAGE	F69
ONREADERROR	F70
ON SELECTION PAD	F70
ON SELECTION POPUP	F71

Tárgymutató

opció	F22
OPEN	F179
operátorok	
aritmetikai operátorok	227
logikai operátorok	227
relációs operátorok	226,239
ORDER	F129
ORDERING	243
OS	F147
összetett indexfile	
Lásd: index	

P

PACK	F36
PAD	F144
paraméter	F17
PARAMETERS	F60
PAYMENT	F155
PCOL	F147
PI	F124
PLAY MACRO	F90
pont prompt	28
popup	81,F144
printer driver	62
PRINTJOB	F71
PRINTSTATUS	F148
PRIVATE	F60
PRIVILEGES	F176
PROGRAM	F155
Programok tesztelése	87-93
PROMPT	F145
PROTECT	F90
PROW	F148
PUBLIC	F61
puffer	F17
pull-down	81
PV	F155

R

RAND	F120
READ	F55

READKEY	F145
RECALL	F37
RECCOUNT	F129
RECNO	F129
RECSIZE	F130
REINDEX	F37
rekord	23
rekordmutató	F38
rekurzív hívás	78
relációs operátorok	
Lásd: operátorok	
RELEASE	F61
RENAME	F37
rendszergazda	243,258,263
rendszerkatalógusok	
Lásd: rendszertáblák	
rendszertáblák	211,241
rendszer-főkatalógus	211
SYSAUTH	241
SYSCOLAU	241
SYSCOLS	241-242
SYSDBS	241
SYSIDX	241
SYSKEYS	241
SYSSYNS	241
SYSTABLS	241
SYSTIMES	241
SYSVDEPS	241
SYSVIEWS	241
UPDATE jogosultság	242
rendszerváltozó	
Lásd: változó	
Rendszerváltozók	55,F157-F165
REPLACE	F37
REPLICATE	F134
REPORT FORM	F55
RESET	F91
RESTORE FROM	F62
RESTORE MACROS FROM	F91
RESTORE WINDOW	F80-F81
RESUME	F71
RETRY	F72
Return	F72,75
REVOKE	245,F180

Tárgymutató

RIGHT	F134
RLOC FLOCK	F151
ROLLBACK	F38,F130,F180
rolómenü	83
ROUND	F121
ROW	F148
RTOD	F124
RTRIM	F135
Run	75
RUN	F91
RUNSTATS	248,F181

S

SAVE MACROS	F91
SAVE TO	F62
SAVE WINDOW	F81
sávkurzor	96,188
SCAN	F73
SEEK	F44,F130
SQL SELECT	F38,F130
FOR UPDATE OF cikkely	F187
FROM cikkely	F183
GROUP BY cikkely	F185
INTO cikkely	F183
ORDER BY cikkely	F186
SAVE TO TEMP cikkely	F187
belső SELECT	237
SELECT cikkely	F182
UNION cikkely	F186
WHERE cikkely	F184
SELECT utasítás	F181
SET	F93,F148
SET ALTERNATE	F93
SET AUTOSAVE	F94
SET BELL	F94
SET BLOCKSIZE	F94
SET BORDER	F95
SET CARRY	F95
SET CATALOG	F95
SET CENTURY	F96
SET CLOCK	F97
SET COLOR	F97

SET CONFIRM	F98
SET CONSOLE	F98
SET CURRENCY	F99
SET CURRENCY LEFT/RIGHT	F99
SET DATE	F99
SET DEBUG	F100
SET DECIMALS	F100
SET DEFAULT	F100
SET DELETED	F100
SET DELIMITERS	F101
SET DESIGN	F101
SET DEVELOPMENT	F101
SET DEVICE	F101
SET DISPLAY	F102
SET ECHO	F102
SET ENCRYPTION	F102
SET ESCAPE	F102
SET EXACT	F103
SET EXCLUSIVE	F103
SET FIELDS	F103
SET FILTER	F104
SET FORMAT	F104
SET FULLPATH	F105
SET FUNCTION	F105
SET HEADING	F106
SET HELP	F106
SET HISTORY	F106
SET HOURS	F106
SET INDEX	F107
SET INSTRUCT	F107
SET INTENSITY	F107
SET LOCK	F108
SET MARGIN	F108
SET MARK	F108
SET MEMOWIDTH	F108
SET MESSAGE	F108
SET NEAR	F109
SET ODOMETER	F109
SET ORDER	F110
SET PATH	F110
SET PAUSE	F111
SET POINT	F111
SET PRECISION	F111
SET PRINTER	F112

Tárgymutató

SETPROCEDURE	F113	STOPDATABASE	F188
SETREFRESH	F113	STORE	F57
SETRELATION	F114	STR	F135
SETREPROCESS	F114	struktúrafájl	F29
SETSAFETY	F114	STUFF	F136
SETSCOREBOARD	F115	SUBSTR	F136
SETSEPARATOR	F115	SUM	F39,229,F191
SET SKIP	F115	SUSPEND	F73
SET SPACE	F116	sysauth	242
SET SQL	F116	syscolau	242
SETSTATUS	F116	syscols	242
SET STEP	F116	SYSDBS	F168
SET TALK	F117	SYSDBS.DBF	211
SET TITLE	F117	systabls	242
SET TRAP	F117	sysviews	242
SETTYPEAHEAD	F117	számított mező	126
SETUNIQUE	F118	számított oszlop	228
SET VIEW	F118	szintaktika	F17
SET WINDOW	F118	sztring	F22
SHOWDATABASE	F188		
SHOWMENU	F81	T	
SHOWPOPUP	F81		
SIGN	F121	táblagazda	243
SIN	F124	TAG	F131
SKIP	F38	TAN	F124
SMALLINT	213-214,F170	TBCREATOR	243
SORT TO	F39	tbname	242-243
SOUNDEX	F156	TBTYPE	243
SPACE	F135	TEXT	F55
SQL	209	TIME	F139
SQL függvények	228	TNAME	243
SQL hálózati alkalmazás	210	tömb	
SQL index		deklaráció	40
index	216	dimenziók	38
SQL terminológia	209	Lásd: változó	
SQL-ben tiltott		TOTAL ON	F40
dBASE IV utasítások	249	TRANSFORM	F136
SQL-ben tiltott függvények	250	Tranzakció	53
SQLCNT	252	trustee	259
SQLCODE	252	tulajdonos-tag viszony	F18
SQLDBA	243	TYPE	F56,F149
SQLTEXT	243		
SQRT	F124		
STARTDATABASE	F188		

Tárgymutató

U

UNLOAD DATA	F188
UNLOCK	F92
UPDATE jogosultság	
Lásd: rendszertáblák	
UPDATE ON	F41
UPPER	F137
üres sztring	
Lásd: sztring	
USE	F42
USER	F151
User ID	F22
ütközés	265

V

VAL	F121
változó	38
hatásköre	40
rendszerváltozó	41
M- > előtag	39,295
VARREAD	F146
védelmi rendszer	268
védelmi szint	268
VERSION	F149
Vezérlő szerkezetek	71-77
VIEWNAME	243
virtuális tábla	
Lásd: nézettábla	
vonalmenü	81

W

WAIT	F58
------	-----

Y

YEAR	F140
------	------

Z

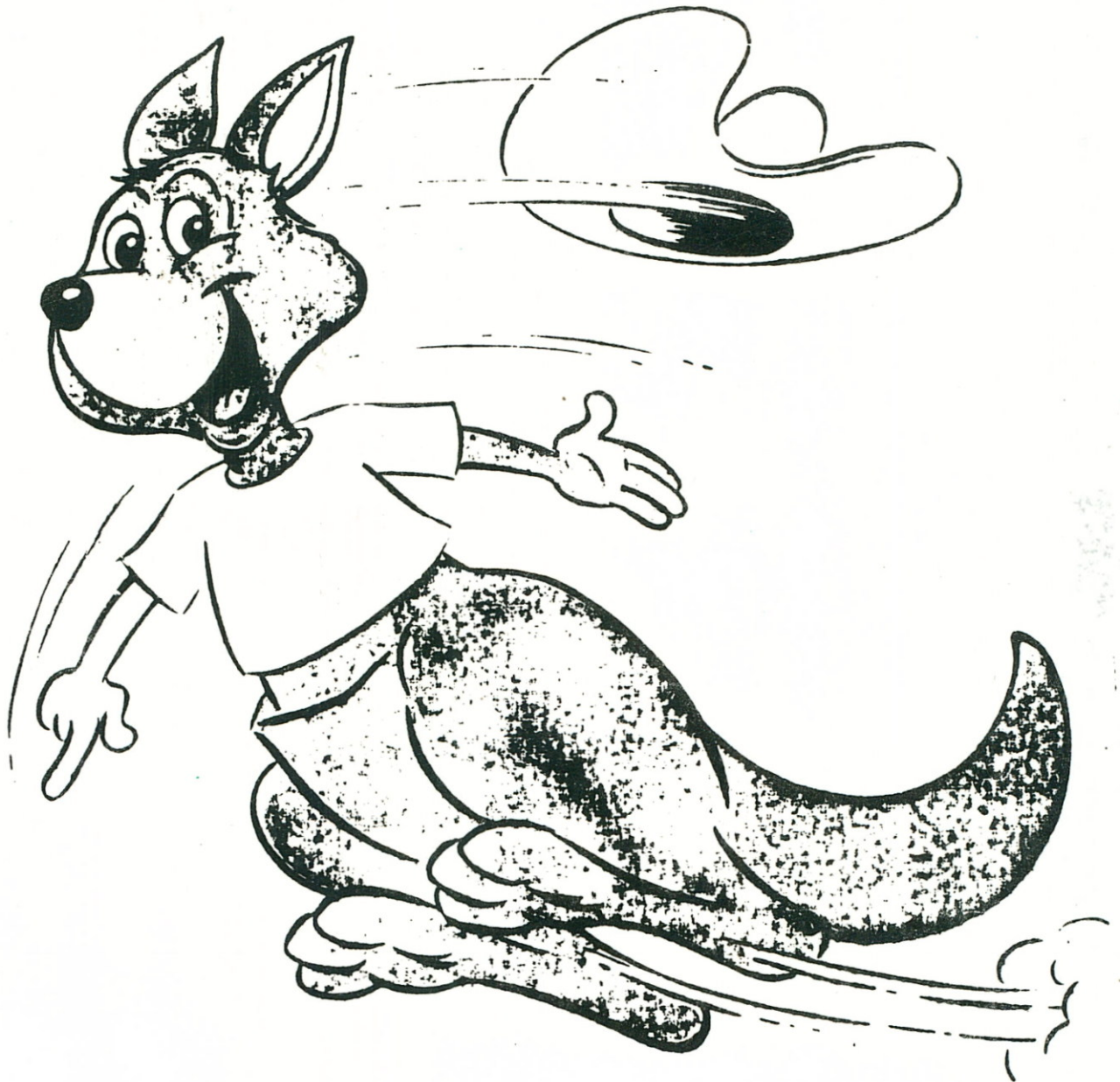
ZAP	F42
zárolás	265

1-2. kötet ára: 488 Ft



aPLUS

SZÉLESKÖRŰ TEVÉKENYSÉGEINKRŐL KÉRJEN TÁJÉKOZTATÓT !!!



MEGBÍZHATÓSÁG

GYORSASÁG

aPLUS Informatikai és Műszaki Korlátolt Felelősségű Társaság

Levelezési cím: 1428 Budapest Pf. 64. Telefon: 18-22-55 Telefax: 144-448 (aPLUS Kft.)