

**English/FileXARexx**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> English/FileXARexx		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		December 8, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>English/FileXARexx</b>	<b>1</b>
1.1	English/FileXARexx.guide	1
1.2	FileXARexx.guide/In general	1
1.3	FileXARexx.guide/Command overview	2
1.4	FileXARexx.guide/ACTIVATE	5
1.5	FileXARexx.guide/ACTIVATEWINDOW	5
1.6	FileXARexx.guide/APPEND	5
1.7	FileXARexx.guide/ASSIGNCOMMAND	6
1.8	FileXARexx.guide/CHANGEWINDOW	6
1.9	FileXARexx.guide/CLEAR	7
1.10	FileXARexx.guide/COLUMN	7
1.11	FileXARexx.guide/COPY	8
1.12	FileXARexx.guide/CURSOR	8
1.13	FileXARexx.guide/CUT	9
1.14	FileXARexx.guide/DEACTIVATE	9
1.15	FileXARexx.guide/EXECUTECOMMAND	9
1.16	FileXARexx.guide/FAULT	10
1.17	FileXARexx.guide/FILL	10
1.18	FileXARexx.guide/FIND	11
1.19	FileXARexx.guide/FINDCHANGE	12
1.20	FileXARexx.guide/FINDNEXT	13
1.21	FileXARexx.guide/FONT	13
1.22	FileXARexx.guide/GETATTR	14
1.23	FileXARexx.guide/GETBLOCK	14
1.24	FileXARexx.guide/GETBYTES	15
1.25	FileXARexx.guide/GOTOBOOKMARK	15
1.26	FileXARexx.guide/GOTOBYTE	16
1.27	FileXARexx.guide/GOTOCOLUMN	16
1.28	FileXARexx.guide/GOTOLINE	16
1.29	FileXARexx.guide/GRABMEMORY	17

1.30 FileXARexx.guide/HELP . . . . .	17
1.31 FileXARexx.guide/INSERT . . . . .	18
1.32 FileXARexx.guide/LINE . . . . .	18
1.33 FileXARexx.guide/LOADBLOCK . . . . .	18
1.34 FileXARexx.guide/LOADBLOCKCLIP . . . . .	19
1.35 FileXARexx.guide/LOADCOMMANDS . . . . .	19
1.36 FileXARexx.guide/LOADDISPLAYTYP . . . . .	20
1.37 FileXARexx.guide/LOADSETTINGS . . . . .	20
1.38 FileXARexx.guide/MOVEWINDOW . . . . .	20
1.39 FileXARexx.guide/NOP . . . . .	21
1.40 FileXARexx.guide/OPEN . . . . .	21
1.41 FileXARexx.guide/PASTE . . . . .	22
1.42 FileXARexx.guide/POSITION . . . . .	22
1.43 FileXARexx.guide/PRINT . . . . .	22
1.44 FileXARexx.guide/PRINTBLOCK . . . . .	23
1.45 FileXARexx.guide/QUIT . . . . .	23
1.46 FileXARexx.guide/REDO . . . . .	23
1.47 FileXARexx.guide/REQUESTFILE . . . . .	24
1.48 FileXARexx.guide/REQUESTNOTIFY . . . . .	24
1.49 FileXARexx.guide/REQUESTRESPONSE . . . . .	25
1.50 FileXARexx.guide/REQUESTSTRING . . . . .	25
1.51 FileXARexx.guide/RX . . . . .	26
1.52 FileXARexx.guide/SAVE . . . . .	26
1.53 FileXARexx.guide/SAVEAS . . . . .	27
1.54 FileXARexx.guide/SAVEBLOCK . . . . .	27
1.55 FileXARexx.guide/SAVEBLOCKCLIP . . . . .	27
1.56 FileXARexx.guide/SAVECOMMANDS . . . . .	28
1.57 FileXARexx.guide/SAVECOMMANDSAS . . . . .	28
1.58 FileXARexx.guide/SAVESETTINGS . . . . .	28
1.59 FileXARexx.guide/SAVESETTINGSAS . . . . .	29
1.60 FileXARexx.guide/SETATTR . . . . .	29
1.61 FileXARexx.guide/SETBLOCK . . . . .	30
1.62 FileXARexx.guide/SETBLOCKMARK . . . . .	30
1.63 FileXARexx.guide/SETBOOKMARK . . . . .	30
1.64 FileXARexx.guide/SETBYTES . . . . .	31
1.65 FileXARexx.guide/SETSCREEN . . . . .	31
1.66 FileXARexx.guide/SIZEWINDOW . . . . .	32
1.67 FileXARexx.guide/UNDO . . . . .	32
1.68 FileXARexx.guide/WINDOWTOBACK . . . . .	32
1.69 FileXARexx.guide/WINDOWTOFRONT . . . . .	33
1.70 FileXARexx.guide/ZIPWINDOW . . . . .	33
1.71 FileXARexx.guide/Attributes . . . . .	33

---

## Chapter 1

# English/FileXARexx

### 1.1 English/FileXARexx.guide

```
$VER: FileXARexxDocumentation 1.1 (16.10.93)
```

```
FileX - A binary file editor
```

```
Copyright 1993 by Klaas Hermanns
```

```
Shareware, all rights reserved.
```

```
In general  
Command overview  
Attributes
```

### 1.2 FileXARexx.guide/In general

```
In general  
*****
```

FileX has an arexxport which is quite completely based on the Commodore Amiga User Interface Style Guide.

If there is nothing different specified for the PORTNAME at the beginning, the port will be named 'FILEX'. In case that this portname does already exist, a number is being added to the name and increased until a not already used name is found. ('NAME' -> 'NAME.1' -> 'NAME.2')

The arguments will be evaluated in the way AmigaDOS does. If the different argument types are unknown, the following list can be used as a reference.

<arg>

A simple string. It's not necessary to specify the argument's name.

---

<arg>/S  
Switch. Will be set through declaring an argument.

<arg>/K  
Key word. This argument must be given with an argument's name.  
(`<arg> = value' oder `<arg> value'.

<arg>/N  
Number. This argument must be a decimal number.

<arg>/T  
Changing switch. Through specification of this argument a value  
will change its condition.

<arg>/A  
This argument must be given.

<arg>/F  
The rest of the line will be understood as an argument, although if  
there are some more keywords

<arg>/M  
Serveral values are possible.

For all commands that return something it's possible to set a  
variable to the return value by specifying the name of it with the  
argument VAR. The same applies to the argument STEM and its  
stemvariables.

### 1.3 FileXARexx.guide/Command overview

Command overview (alphabetical order)

\*\*\*\*\*

Name:	Arguments => return values:
-----	
ACTIVATE	
ACTIVATEWINDOW	
APPEND	
ASSIGNCOMMAND	NUMBER/N/K, NAME/K
CHANGEWINDOW	LEFTEDGE/N/K, TOPEDGE/N/K, HEIGHT/N/K, WIDTH/N/K
CLEAR	FORCE/S
COLUMN	DELTA/N/A
COPY	
CURSOR	UP/S, DOWN/S, LEFT/S, RIGHT/S
CUT	
DEACTIVATE	
EXECUTECOMMAND	NUMBER/N/K, NAME/K
FAULT	VAR/K, STEM/K, NUMBER/N/A => DESCRIPTION
FILL	
FIND	PROMPT/S, HEX/S, BACKWARDS/S, QUIET/S, TEXT/F
FINDCHANGE	PROMPT/S, HEX/S, ALL/S, BACKWARDS/S, QUIET/S, FIND, CHANGE
FINDNEXT	BACKWARDS/S, QUIET/S
FONT	NAME/A, SIZE/A/N

GETATTR	OBJECT/A, FIELD, STEM/K, VAR/K
GETBLOCK	VAR/K, STEM/K => BLOCK
GETBYTES	VAR/K, STEM/K, POS/N/A, NUMBER/N/A => HEXSTRING
GOTOBOOKMARK	NUMBER/N
GOTOBYTE	POS/N/A
GOTOCOLUMN	COLUMN/N/A
GOTOLINE	LINE/N/A
GRABMEMORY	START/A/N, END/A/N, FORCE/S
HELP	VAR/K, STEM/K, COMMAND => COMMANDDESC, COMMANDLIST/M
INSERT	
LINE	DELTA/N/A
LOADBLOCK	NAME
LOADBLOCKCLIP	
LOADCOMMANDS	NAME
LOADDISPLAYTYP	NAME
LOADSETTINGS	NAME
MOVEWINDOW	LEFTEDGE/N, TOPEDGE/N
NOP	
OPEN	FILENAME, FORCE/S
PASTE	
POSITION	SOF/S, EOF/S, SOL/S, EOL/S
PRINT	
PRINTBLOCK	
QUIT	FORCE/S
REDO	NUMBER/N
REQUESTFILE	VAR/K, STEM/K, TITLE/K, PATH/K, FILE/K, PATTERN/K => FILENAME
REQUESTNOTIFY	PROMPT
REQUESTRESPONSE	BUTTONS/K, PROMPT/K
REQUESTSTRING	VAR/K, STEM/K, PROMPT/K, DEFAULT/K => STRING
RX	VAR/K, STEM/K, CONSOLE/S, ASYNC/S, COMMAND/F => RC/N, RESULT
SAVE	
SAVEAS	NAME
SAVEBLOCK	NAME
SAVEBLOCKCLIP	
SAVECOMMANDS	
SAVECOMMANDSAS	NAME
SAVESETTINGS	
SAVESETTINGSAS	NAME
SETATTR	OBJECT/A, FIELD, FROMSTEM/K, FROMVAR/K
SETBLOCK	BLOCK/A
SETBLOCKMARK	
SETBOOKMARK	NUMBER/N
SETBYTES	POS/N/A, HEXSTRING/A
SETSCREEN	NAME/K, OWN/S
SIZEWINDOW	HEIGHT/N, WIDTH/N
UNDO	NUMBER/N
WINDOWTOBACK	
WINDOWTOFRONT	
ZIPWINDOW	

Name: Arguments => Returncodes:

ACTIVATE	
ACTIVATEWINDOW	
APPEND	
ASSIGNCOMMAND	NUMBER/N/K, NAME/K
CHANGEWINDOW	LEFTEDGE/N/K, TOPEDGE/N/K, HEIGHT/N/K, WIDTH/N/K

CLEAR	FORCE/S
COLUMN	DELTA/N/A
COPY	
CURSOR	UP/S, DOWN/S, LEFT/S, RIGHT/S
CUT	
DEACTIVATE	
EXECUTECOMMAND	NUMBER/N/K, NAME/K
FAULT	VAR/K, STEM/K, NUMBER/N/A => DESCRIPTION
FILL	
FIND	PROMPT/S, HEX/S, BACKWARDS/S, QUIET/S, TEXT/F
FINDCHANGE	PROMPT/S, HEX/S, ALL/S, BACKWARDS/S, QUIET/S, FIND, CHANGE
FINDNEXT	BACKWARDS/S, QUIET/S
FONT	NAME/A, SIZE/A/N
GETATTR	OBJECT/A, FIELD, STEM/K, VAR/K
GETBLOCK	VAR/K, STEM/K => BLOCK
GETBYTES	VAR/K, STEM/K, POS/N/A, NUMBER/N/A => HEXSTRING
GOTOBOOKMARK	NUMBER/N
GOTOBYTE	POS/N/A
GOTOCOLUMN	COLUMN/N/A
GOTOLINE	LINE/N/A
GRABMEMORY	START/A/N, END/A/N, FORCE/S
HELP	VAR/K, STEM/K, COMMAND => COMMANDDESC, COMMANDLIST/M
INSERT	
LINE	DELTA/N/A
LOADBLOCK	NAME
LOADBLOCKCLIP	
LOADCOMMANDS	NAME
LOADDISPLAYTYP	NAME
LOADSETTINGS	NAME
MOVEWINDOW	LEFTEDGE/N, TOPEDGE/N
NOP	
OPEN	FILENAME, FORCE/S
PASTE	
POSITION	SOF/S, EOF/S, SOL/S, EOL/S
PRINT	
PRINTBLOCK	
QUIT	FORCE/S
REDO	NUMBER/N
REQUESTFILE	VAR/K, STEM/K, TITLE/K, PATH/K, FILE/K, PATTERN/K => FILENAME
REQUESTNOTIFY	PROMPT
REQUESTRESPONSE	BUTTONS/K, PROMPT/K
REQUESTSTRING	VAR/K, STEM/K, PROMPT/K, DEFAULT/K => STRING
RX	VAR/K, STEM/K, CONSOLE/S, ASYNC/S, COMMAND/F => RC/N, RESULT
SAVE	
SAVEAS	NAME
SAVEBLOCK	NAME
SAVEBLOCKCLIP	
SAVECOMMANDS	
SAVECOMMANDSAS	NAME
SAVESETTINGS	
SAVESETTINGSAS	NAME
SETATTR	OBJECT/A, FIELD, FROMSTEM/K, FROMVAR/K
SETBLOCK	BLOCK/A
SETBLOCKMARK	
SETBOOKMARK	NUMBER/N
SETBYTES	POS/N/A, HEXSTRING/A
SETSCREEN	NAME/K, OWN/S

```
SIZEWINDOW          HEIGHT/N,WIDTH/N
UNDO                 NUMBER/N
WINDOWTOBACK
WINDOWTOFRONT
ZIPWINDOW
```

Command descriptions

\*\*\*\*\*

## 1.4 FileXARexx.guide/ACTIVATE

ACTIVATE

=====

Synopsis:

ACTIVATE

Function:

Reactivates FileX from iconify mode.

Example:

```
DEACTIVATE          /* Equivalent to menu: 'Project/Iconify' */

/* Now an icon will appear on the workbench. FileX 'sleeps'. */

ACTIVATE            /* The window will be opened again */
```

See also:

DEACTIVATE

## 1.5 FileXARexx.guide/ACTIVATEWINDOW

ACTIVATEWINDOW

=====

Synopsis:

ACTIVATEWINDOW

Function:

Activates the FileX window.

## 1.6 FileXARexx.guide/APPEND

APPEND

=====

Synopsis:

---

APPEND

Function:

The current Block will be appended to the file end.

See also:

Menu: 'Edit/Append'

## 1.7 FileXARexx.guide/ASSIGNCOMMAND

ASSIGNCOMMAND

=====

Synopsis:

ASSIGNCOMMAND NUMBER/N/K,NAME/K

Function:

Sets one of the arexxcommands in the 'ARexx'-menu.

Arguments:

NUMBER/N/K:

A number between 0 and 10 which defines the command's position. A requester will appear if no number had been specified.

NAME/K:

The command's filename. Here, a requester to choose the filename will appear, too, if no name had been specified previously.

Example:

ASSIGNCOMMAND 0 'dh0:MyArexxCommands/FileXcommand'.

See also:

Menu: 'ARexx/Assign command', LOADCOMMANDS, SAVECOMMANDS, SAVECOMMANDSAS

## 1.8 FileXARexx.guide/CHANGEWINDOW

CHANGEWINDOW

=====

Synopsis:

CHANGEWINDOW LEFTEDGE/N/K, TOPEDGE/N/K, HEIGHT/N/K, WIDTH/N/K

Function:

Modifies the size and/or the location of the FileX window.

Arguments:

LEFTEDGE/N:

New left offset of the window to the left border.

TOPEDGE/N:

New upper offset of the window to the upper border.

HEIGHT/N:

New height of the window.

WIDTH/N:

New width of the window.

If no new value is specified, the old value remains unchanged.

Example:

```
/* Resize window to maximum height */  
CHANGWINDOW LEFTEDGE 0 TOPEDGE 0 HEIGHT 900
```

See also:

ZIPWINDOW, MOVEWINDOW, SIZEWINDOW, Attributes: 'WINDOW/#?'

## 1.9 FileXARexx.guide/CLEAR

CLEAR

=====

Synopsis:

CLEAR FORCE/S

Function:

Removes the current file from memory.

Arguments:

FORCE/S:

The requester 'You made x changes' will be suppressed.

Returncodes:

Return code = 5, if the file could not be removed.

See also:

Menu: 'Project/New'

## 1.10 FileXARexx.guide/COLUMN

COLUMN

=====

Synopsis:

COLUMN DELTA/N/A

Function:

---

Moves the cursor to the left or the right.

Arguments:

DELTA/N/A:

Number of moves the cursor shall do. Positive values will move the cursor to the right and negative ones move it to the left.

See also:

GOTOBYTE, GOTOCOLUMN, GOTOLINE, LINE

## 1.11 FileXARexx.guide/COPY

COPY

====

Synopsis:

COPY

Function:

Copies the current marked block to the blockbuffer.

See also:

Menu: 'Edit/Copy', SETBLOCKMARK

## 1.12 FileXARexx.guide/CURSOR

CURSOR

=====

Synopsis:

CURSOR UP/S, DOWN/S, LEFT/S, RIGHT/S

Function:

Moves the cursor.

Arguments:

UP/S:

The cursor will be moved one line up.

DOWN/S:

The cursor will be moved one line down.

LEFT/S:

The cursor will be moved one column left.

RIGHT/S:

The cursor will be moved one column right.

UP and DOWN as well as LEFT and RIGHT exclude each other.

---

See also:

GOTOBYTE, LINE, COLUMN

## 1.13 FileXARexx.guide/CUT

CUT

===

Synopsis:

CUT

Function:

Copies the current marked block to the blockbuffer and cuts it from the file simultaneously.

See also:

Menu: 'Edit/Cut', SETBLOCKMARK

## 1.14 FileXARexx.guide/DEACTIVATE

DEACTIVATE

=====

Synopsis:

DEACTIVATE

Function:

Iconifies FileX. The window will be closed and an icon produced on the workbench.

See also:

ACTIVATE

## 1.15 FileXARexx.guide/EXECUTECOMMAND

EXECUTECOMMAND

=====

Synopsis:

EXECUTECOMMAND NUMBER/N/K, NAME/K

Function:

Executes an arexxscript.

Arguments:

NUMBER/N/K:

A number between 0 and 9 of a command installed by  
ASSIGNCOMMAND.

NAME/K:

Name of an arexxscript.

Both arguments exclude each other.

See also:

ASSIGNCOMMAND

## 1.16 FileXARexx.guide/FAULT

FAULT

=====

Synopsis:

FAULT VAR/K,STEM/K,NUMBER/N/A => DESCRIPTION

Function:

Finds out the faulty text belonging to a dos error number.

Arguments:

NUMBER/N/A:

Dos error number.

Returncodes:

DESCRIPTION:

The belonging text.

Example:

FAULT 114

```
/* Result value using localized workbench: */  
/* `: Wrong name-pattern' */
```

## 1.17 FileXARexx.guide/FILL

FILL

=====

Synopsis:

FILL

Function:

Fills the current marked area with the current fillstring.

Example:

```
/* If there is a block already marked, */  
/* then neutralize it. */
```

---

```
GETATTR PROJECT MARK
if result=TRUE then SETBLOCKMARK

    /* Set fillstring to 'Fill fillaround!' */

SETATTR PROJECT FILLSTRING FROMVAR '"Fill fillaround!"'

    /* Set cursor to the fill-area's beginning */
GOTOBYTE 10

    /* Set block mark */
SETBLOCKMARK

    /* Move cursor */
GOTOBYTE 100

    /* Fill the area */
FILL

    /* Neutralize block marking */
SETBLOCKMARK
```

See also:

Menu: 'Edit/Fill', SETBLOCKMARK, SETATTR

## 1.18 FileXARexx.guide/FIND

FIND  
====

Synopsis:

FIND PROMPT/S,HEX/S,BACKWARDS/S,QUIET/S,TEXT/F

Function:

Searches a string or a hex string within the current file. By success, the cursor is being positioned at the string's beginning. The search begins at the current cursor position.

Arguments:

PROMPT/S:

Opens the searchrequester. BACKWARDS and QUIET will be ignored.

HEX/S:

The string to be searched was recognized as a hex string.

BACKWARDS/S:

Searches backwards.

QUIET/S:

No requester will be opened by failure.

TEXT/F:

The string or hex string.

Returncodes:

Return code = 5, if the string could not to be found.

Example:

```
/* Search the string '$VER:' without opening */
/* a requester by failure */
FIND QUIET TEXT '$VER:'
```

See also:

Attributs: 'PROJECT/#?SEARCH', FINDNEXT, FINDCHANGE, Menu:  
'Search&Replace/#?'

## 1.19 FileXARexx.guide/FINDCHANGE

### FINDCHANGE

=====

Synopsis:

FINDCHANGE PROMPT/S, HEX/S, ALL/S, BACKWARDS/S, QUIET/S, FIND, CHANGE

Function:

Searches a string or a hexstring within a file and replaces it through a new one. Maybe a requester appears to verify the replacement. The search will start from the current cursorposition.

Arguments:

PROMPT/S:

Opens a 'search and replace'-requester. ALL, BACKWARDS und QUIET will be ignored.

HEX/S:

The text to search for is a hexstring.

ALL/S:

All found positions will be replaced without requester each time.

BACKWARDS/S:

Search backwards.

QUIET/S:

No requester will be opened by failure.

FIND:

The string or hexstring.

CHANGE:

The replacestring or -hexstring.

Returncodes:

rc = 5, if the string could not be found.

---

Example:

```
/* Replaces all 'Claas' through 'Klaas' */  
FINDCHANGE QUIET ALL 'Claas' 'Klaas'  
FINDCHANGE BACKWARDS QUIET ALL 'Claas' 'Klaas'
```

See also:

FINDNEXT, FIND, Menu: 'Search&Replace/#?'

## 1.20 FileXARexx.guide/FINDNEXT

FINDNEXT

=====

Synopsis:

FINDNEXT BACKWARDS/S, QUIET/S

Function:

Continues the last search option, i.e. either search or search&replace.

Arguments:

BACKWARDS/S:  
Continue search backwards.

QUIET/S:  
No requester.

Returncodes:

rc = 5, if search fails.

See also:

FIND, FINDCHANGE

## 1.21 FileXARexx.guide/FONT

FONT

====

Synopsis:

FONT NAME/A, SIZE/N/A

Function:

Sets an specified Font.

Arguments:

NAME/A:  
The font name.

SIZE/N/A.

---

The font size.

Example:

```
FONT 'horror.font' 8
```

## 1.22 FileXARexx.guide/GETATTR

GETATTR  
=====

Synopsis:

```
GETATTR OBJECT/A, FIELD, STEM/K, VAR/K
```

Function:

Finds out the value of one ore more variables used in the program.

Arguments:

OBJECT/A:

Name of the object.

FIELD:

Name of the field.

If FIELD is not defined, a STEM variable must be specified. There will all the object's value be stored.

Example:

```
GETATTR WINDOW TITLE
```

```
/* Result contains now the titel of the window */  
/* e.g.: 'File: ENV:Sys/overscan.prefs (70 Bytes)' */
```

See also:

Attributes, SETATTR

## 1.23 FileXARexx.guide/GETBLOCK

GETBLOCK  
=====

Synopsis:

```
GETBLOCK          VAR/K, STEM/K => BLOCK
```

Function:

Establishes the content of the blockbuffer.

Returncodes:

BLOCK:

Contents a block as hexstring.

See also:  
SETBLOCK

## 1.24 FileXARexx.guide/GETBYTES

GETBYTES  
=====

Synopsis:  
GETBYTES VAR/K,STEM/K,POS/N/A,NUMBER/N/A => HEXSTRING

Function:  
Finds out the value of a number of bytes at one position within a file.

Arguments:  
POS/N/A:  
Position of the bytes to be found out.  
  
NUMBER/N/A:  
Number of Bytes (max:32767).

Returncodes:  
HEXSTRING:  
Hexstring of the bytes.

See also:  
SETBYTES

## 1.25 FileXARexx.guide/GOTOBOOKMARK

GOTOBOOKMARK  
=====

Synopsis:  
GOTOBOOKMARK NUMBER/N/A

Function:  
Sets the cursor to a position previously marked with SETBOOKMARK.

Arguments:  
NUMBER/N/A:  
A number between 0 and 10.

See also:  
SETBOOKMARK

---

## 1.26 FileXARexx.guide/GOTOBYTE

GOTOBYTE  
=====

Synopsis:

GOTOBYTE POS/N/A

Function:

Sets the cursor to a specified position.

Arguments:

POS/N/A:  
Position.

See also:

COLUMN, GOTOCOLUMN, GOTOLINE, LINE

## 1.27 FileXARexx.guide/GOTOCOLUMN

GOTOCOLUMN  
=====

Synopsis:

GOTOCOLUMN COLUMN/N/A

Function:

Sets the cursor to a specified column.

Arguments:

COLUMN/N/A:  
Column between 1 and 16.

See also:

COLUMN, GOTOBYTE, GOTOLINE, LINE

## 1.28 FileXARexx.guide/GOTOLINE

GOTOLINE  
=====

Synopsis:

GOTOLINE LINE/N/A

Function:

Sets the cursor to a specified line.

Arguments:

LINE/N/A:  
Number of the line.

---

See also:

COLUMN, GOTOBYTE, GOTOCOLUMN, LINE

## 1.29 FileXARexx.guide/GRABMEMORY

GRABMEMORY

=====

Synopsis:

GRABMEMORY START/N/A,END/N/A,FORCE/S

Function:

Displays a part of the memory instead of a file.

Arguments:

START/N/A:

Memory start address.

END/N/A:

Memory end adress.

FORCE/S:

The requester 'You made x changes' will be suppressed.

Returncodes:

rc = 5, if the command fails. (requester cancelled).

Example:

GRABMEMORY 200000 280000 FORCE

See also:

Menu: 'Project/Grab memory'

## 1.30 FileXARexx.guide/HELP

HELP

====

Synopsis:

HELP VAR/K,STEM/K,COMMAND => COMMANDDESC,COMMANDLIST/M

Function:

Finds out either an overview of all specified arexxcommands or the construction of an command.

Arguments:

COMMAND:

Commands name, whose construction shall be found out.

If COMMAND is not specified, a command overview will be shown.

Returncodes:

If COMMAND specified:

COMMANDDESC:

Command's name. Otherwise:

COMMANDLIST/M:

Command's number and an overview of all commands.

## 1.31 FileXARexx.guide/INSERT

INSERT

=====

Synopsis:

INSERT

Function:

Inserts a block from the blockbuffer at the current cursor position into the file.

See also:

Menu: 'Edit/Insert', SETBLOCKMARK

## 1.32 FileXARexx.guide/LINE

LINE

=====

Synopsis:

LINE DELTA/N/A

Function:

Moves the cursor a defined number of lines.

Arguments:

LINE/N/A:

Number of lines. Positive values move the cursor downwards.

See also:

COLUMN, GOTOBYTE, GOTOCOLUMN, GOTOLINE

## 1.33 FileXARexx.guide/LOADBLOCK

LOADBLOCK

=====

---

**Synopsis:**

LOADBLOCK NAME

**Function:**

Loads a block into the blockbuffer.

**Arguments:****NAME:**

Filename. If not specified, a requester will be opened.

**Returncodes:**

rc = 5, if loading failed.

**See also:**

SAVEBLOCK

## 1.34 FileXARexx.guide/LOADBLOCKCLIP

LOADBLOCKCLIP

=====

**Synopsis:**

LOADBLOCKCLIP

**Function:**

Kopiert den aktuellen Inhalt des Clipboards in den Blockspeicher.

## 1.35 FileXARexx.guide/LOADCOMMANDS

LOADCOMMANDS

=====

**Synopsis:**

LOADCOMMANDS NAME

**Function:**

Loads arexxcommands.

**Arguments:****NAME:**

Filename of the commands. If not specified, a requester will be opened

**Returncodes:**

rc = 5, if loading failed.

**See also:**

Menu: 'ARexx/Load commands', SAVECOMMANDS, SAVECOMMANDSAS

---

## 1.36 FileXARexx.guide/LOADDISPLAYTYP

LOADDISPLAYTYP  
=====

Synopsis:

LOADDISPLAYTYP NAME

Function:

Loads a defined displaytype.

Arguments:

NAME:

Filename of the displaytype. If not specified, a requester will be opened.

Returncodes:

rc = 5, if loading failed.

See also:

Menu: 'Settings/Display/#?'

## 1.37 FileXARexx.guide/LOADSETTINGS

LOADSETTINGS  
=====

Synopsis:

LOADSETTINGS NAME

Function:

Loads a settingsfile.

Arguments:

NAME:

Filename of the settings file. If not specified, a requester will be opened.

Returncodes:

rc = 5, if loading failed.

See also:

Menu: 'Settings/Load Settings...'

## 1.38 FileXARexx.guide/MOVEWINDOW

MOVEWINDOW  
=====

Synopsis:

---

```
MOVEWINDOW LEFTEDGE/N, TOPEDGE/N
```

**Function:**

Modifies the FileX window position.

**Arguments:**

LEFTEDGE/N:

Space to the left border.

TOPEDGE/N:

New space to the upper border.

If not specified, the value will remain unchanged.

**See also:**

CHANGEWINDOW, SIZEWINDOW, ZIPWINDOW, Attributes: 'WINDOW/#?'

## 1.39 FileXARexx.guide/NOP

NOP

===

**Synopsis:**

NOP

**Function:**

This command has no, indeed no effect. :-) Very useful for nothing.

## 1.40 FileXARexx.guide/OPEN

OPEN

====

**Synopsis:**

OPEN FILENAME, FORCE/S

**Function:**

Opens a file.

**Arguments:**

FILENAME:

Filename. If not specified, a requester is being opened as usual.

FORCE/S:

The requester 'You made x changes' will be suppressed.

**Returncodes:**

rc = 5, if loading failed.

---

See also:

CLEAR, SAVE, SAVEAS

## 1.41 FileXARexx.guide/PASTE

PASTE

=====

Synopsis:

PASTE

Function:

Writes a block from the blockbuffer into the file beginning with the current cursor position.

See also:

Menu: 'Edit/Paste', SETBLOCKMARK

## 1.42 FileXARexx.guide/POSITION

POSITION

=====

Synopsis:

POSITION SOF/S,EOF/S,SOL/S,EOL/S

Function:

Moves the cursor to a defined position.

Arguments:

SOF/S:

Moves the cursor to the start of the file.

EOF/S:

Moves the cursor to the end of the file.

SOL/S:

Moves the cursor to the start of a line.

EOL/S:

Moves the cursor to the end of a line.

SOF and EOF and also SOL und EOL exclude each other.

## 1.43 FileXARexx.guide/PRINT

PRINT  
=====

Synopsis:  
    PRINT

Function:  
    Prints the file.

## 1.44 FileXARexx.guide/PRINTBLOCK

PRINTBLOCK  
=====

Synopsis:  
    PRINTBLOCK

Function:  
    Prints the block from blockbuffer.

## 1.45 FileXARexx.guide/QUIT

QUIT  
=====

Synopsis:  
    QUIT FORCE/S

Function:  
    Quits FileX.

Arguments:  
    FORCE/S:  
        The requester 'You made x changes' will be suppressed.

## 1.46 FileXARexx.guide/REDO

REDO  
=====

Synopsis:  
    REDO NUMBER/N

Function:  
    A special number of 'Undos' will be cancelled.

---

**Arguments:****NUMBER/N:**

Number of 'Undos' to be cancelled. If not specified, only the last 'Undo' will be cancelled.

## 1.47 FileXARexx.guide/REQUESTFILE

**REQUESTFILE**

=====

**Synopsis:**

REQUESTFILE VAR/K,STEM/K,TITLE/K,PATH/K,FILE/K,PATTERN/K =>  
FILENAME

**Function:**

Opens a filerequester and gives back an chosen filename.

**Arguments:****TITLE/K:**

Title of the filerequester.

**PATH/K:**

Pre-set pathname.

**FILE/K:**

Pre-set filename.

**PATTERN/K:**

Pattern.

**Returncodes:**

If rc = 5, the filerequester was cancelled.

**Otherwise:****FILENAME:**

Chosen filename and its path.

**Example:**

REQUESTFILE TITLE 'Please choose a script:' PATTERN '#?.filex'

## 1.48 FileXARexx.guide/REQUESTNOTIFY

**REQUESTNOTIFY**

=====

**Synopsis:**

REQUESTNOTIFY PROMPT

**Function:**

Opens a simple Messagerequester with an gadget.

---

Arguments:  
PROMPT:  
Title text.

## 1.49 FileXARexx.guide/REQUESTRESPONSE

REQUESTRESPONSE  
=====

Synopsis:  
REQUESTRESPONSE BUTTONS/K,PROMPT/K

Function:  
Opens a requester with several gadget to choose and returns the number of the chosen gadget.

Arguments:  
BUTTONS/K:  
A string which defines the construction of the gadgets (see example).

PROMPT/K:  
Titletext.

Returncodes:  
rc contains the number of the chosen gadget. The gadget at the right has got the number 0. Alle others are enumerated from left to right beginning with 1.

Example:  
/\* Opens requester with 4 gadgets \*/  
REQUESTRESPONSE BUTTONS '"Gadget 1|Gadget2|Gadget 3|Gadget0"'  
  
say 'You have chosen gadget 'rc'.'

## 1.50 FileXARexx.guide/REQUESTSTRING

REQUESTSTRING  
=====

Synopsis:  
REQUESTSTRING VAR/K,STEM/K,PROMPT/K,DEFAULT/K => STRING

Function:  
Opens a requester to enter a string.

Arguments:  
PROMPT/K:  
Titletext.

---

```
DEFAULT/K:
    Pre-set string.
```

```
Returncodes:
    rc = 5, if the requester was cancelled.
```

```
Else:
STRING
    Entered string.
```

## 1.51 FileXARexx.guide/RX

```
RX
==
```

```
Synopsis:
    RX VAR/K, STEM/K, CONSOLE/S, ASYNC/S, COMMAND/F => RC/N, RESULT
```

```
Function:
    Starts an arexxscript.
```

```
Arguments:
    CONSOLE/S:
        If specified, a Window for in- and output will be opened

    ASYNC/S:
        The script shall be executed asynchronous.

    COMMAND/F:
        Name of the command.
```

```
Returncodes:
    If executed synchronous:
    RC/N, RESULT:
        Returncodes of the script.
```

## 1.52 FileXARexx.guide/SAVE

```
SAVE
====
```

```
Synopsis:
    SAVE
```

```
Function:
    Saves a file under its current name. If the file does not have a
    name, a filerequester will appear.
```

```
Returncodes:
```

---

rc = 5, if the file could not be saved.

### 1.53 FileXARexx.guide/SAVEAS

SAVEAS

=====

Synopsis:

SAVEAS NAME

Function:

Saves the file under a new name.

Arguments:

NAME:

Filename, the file shall be saved with. If not specified, a filerequester will appear.

Returncodes:

rc = 5, if the file could not be saved.

### 1.54 FileXARexx.guide/SAVEBLOCK

SAVEBLOCK

=====

Synopsis:

SAVEBLOCK NAME

Function:

Saves the block from the blockbuffer.

Arguments:

NAME:

Name, the block shall be saved with. If not specified, a filerequester will appear.

Returncodes:

rc = 5, if the block could not be saved.

### 1.55 FileXARexx.guide/SAVEBLOCKCLIP

SAVEBLOCKCLIP

=====

Synopsis:

SAVEBLOCKCLIP

---

**Function:**

Copies the block from blockbuffer into the clipboard.

**See also:**

LOADBLOCKCLIP

## 1.56 FileXARexx.guide/SAVECOMMANDS

**SAVECOMMANDS**

=====

**Synopsis:**

SAVECOMMANDS

**Function:**

Saved the installed arexxcommands under the current name.

**Returncodes:**

rc = 5, if the commands could not be saved.

## 1.57 FileXARexx.guide/SAVECOMMANDSAS

**SAVECOMMANDSAS**

=====

**Synopsis:**

SAVECOMMANDSAS NAME

**Function:**

Saves the installed arexxcommands under a new name.

**Arguments:****NAME:**

Filename of the commands. If not specified, a filerequester will appear.

**Returncodes:**

rc = 5, if the commands could not be saved.

## 1.58 FileXARexx.guide/SAVESETTINGS

**SAVESETTINGS**

=====

**Synopsis:**

SAVESETTINGS

---

**Function:**

Saves the current settingsfile under the current name.

**Returncodes:**

rc = 5, if the settings could not be saved.

## 1.59 FileXARexx.guide/SAVESETTINGSAS

SAVESETTINGSAS

=====

**Synopsis:**

SAVESETTINGSAS NAME

**Function:**

Saves the current settings under a new name.

**Arguments:****NAME:**

Filename of the settingsfile. If not specified, a filerequester will be opened.

**Returncodes:**

rc = 5, if the settings could not be saved.

## 1.60 FileXARexx.guide/SETATTR

SETATTR

=====

**Synopsis:**

SETATTR OBJECT/A, FIELD, FROMSTEM/K, FROMVAR/K

**Function:**

Sets the value of one or more variables used in the program.

**Arguments:****OBJECT/A:**

Object's name.

**FIELD:**

Field's name.

**FROMSTEM/K:**

Name of a STEM-variable which holds the value to set.

**FROMVAR/K:**

Name of a variable which holds a value to set.

---

If FIELD is not specified, a FROMSTEM-variable must be specified.  
In this variable, several values of the object can be stored.

Example:

```
mystemvariable.STRINGFILL = TRUE
mystemvariable.FILLSTRING = 'Klaas is stupid.'
mystemvariable.HEXFILLSTRING = '08 15'

SETATTR PROJECT FROMSTEM meinestemvariable
```

See also:

Attributes, GETATTR

## 1.61 FileXARexx.guide/SETBLOCK

SETBLOCK  
=====

Synopsis:

SETBLOCK BLOCK/A

Function:

Sets the current block into the blockbuffer.

Arguments:

BLOCK/A:  
Hexstring.

Example:

```
SETBLOCK '"0071 12feee de"'
```

## 1.62 FileXARexx.guide/SETBLOCKMARK

SETBLOCKMARK  
=====

Synopsis:

SETBLOCKMARK

Function:

Switches between block mark mode and normal mode.

See also:

Menu: 'Edit/Mark', Attributs: 'PROJECT/MARK', COPY, CUT, FILL

## 1.63 FileXARexx.guide/SETBOOKMARK

---

SETBOOKMARK  
=====

Synopsis:

SETBOOKMARK NUMBER/N

Function:

Sets a mark at the current cursorposition. By using GOTOBOOKMARK, this mark can be jumped at.

Arguments:

NUMBER/N:

Number between 1 and 10, which defines the mark's number.

See also:

GOTOBOOKMARK

## 1.64 FileXARexx.guide/SETBYTES

SETBYTES  
=====

Synopsis:

SETBYTES POS/N/A, HEXSTRING/A

Function:

Sets a defined number of bytes at a specified position.

Arguments:

POS/N/A:

Position, the bytes shall be set to.

HEXSTRING/A:

A hexstring of bytes.

Example:

SETBYTES 200 '"4b6c 6161 73"'

See also:

GETBYTES

## 1.65 FileXARexx.guide/SETSCREEN

SETSCREEN  
=====

Synopsis:

SETSCREEN NAME/K, OWN/S

---

**Function:**

Definition of the screen FileX shall be opened on. Arguments:

**NAME/K:**

Name of an publicscreen.

**OWN/S:**

An own screen shall be opened. A requester makes it possible to define width, height etc.

## 1.66 FileXARexx.guide/SIZEWINDOW

**SIZEWINDOW**

=====

**Synopsis:**

SIZEWINDOW HEIGHT/N,WIDTH/N

**Function:**

The FileX window height and width may be modified.

**Arguments:****HEIGHT/N:**

New height.

**WIDTH/N:**

New width.

## 1.67 FileXARexx.guide/UNDO

**UNDO**

=====

**Synopsis:**

UNDO NUMBER/N

**Function:**

A defined number of changes made to the file will be cancelled.

**Arguments:****NUMBER/N:**

Number of changes which shall be possible to be cancelled.  
If not specified, only the last change will be cancelled.

## 1.68 FileXARexx.guide/WINDOWTOBACK

WINDOWTOBACK  
=====

Synopsis:  
    WINDOWTOBACK

Function:  
    The window will be moved to the background.

## 1.69 FileXARexx.guide/WINDOWTOFRONT

WINDOWTOFRONT  
=====

Synopsis:  
    WINDOWTOFRONT

Function:  
    The window will be moved to the front.

## 1.70 FileXARexx.guide/ZIPWINDOW

ZIPWINDOW  
=====

Synopsis:  
    ZIPWINDOW

Function:  
    Same function as the zip gadget of the FileX window has got.

## 1.71 FileXARexx.guide/Attributes

Attributes  
\*\*\*\*\*

Some variables of FileX can be read by the option GETATTR. Probably they may also be modified through SETATTR. Both commands require at least the objectname and maybe a fieldname, too.

There are three different fieldtypes: Strings, numbers and boolean values ('TRUE' or 'FALSE').

Following are all objects and their belonging fields. For every field, its name and the type will be displayed, moreover if it can only be read. After that, a content description follows.

---

## APPLICATION-Object:

=====

"SCREEN", String, read only:

Name of the public screen on which FileX has opened its window.

"VERSION", String, read only:

Version number as string.

"REGNUMBER", Number, read only:

Registration number.

"REGNAME", String, read only:

A name to which this version is registered to.

## PROJECT-Object

=====

"AREXX", String, read only:

Name of the arexxport.

"FILENAME", String:

Current filename and its path.

"PATH", String:

Path of the current file.

"FILE", String:

Filename without path.

"CHANGES", Number, read only:

Number of changes in the file.

"PRIORITY", Number:

Task priority of FileX.

"CURSORPOS", Number, read only:

Current cursorposition.

"MARKPOS", Number, read only:

Position of the mark's beginning.

"FILELEN", Number, read only:

Length of the current file.

"EDITABLE", Bool:

TRUE, if the file can be edited.

"EDITASCII", Bool:

TRUE, if the active cursor is in the ascii area.

"MARK", Bool, read only:

TRUE, if a block is being marked at the moment.

"SEARCHSTRING", String:

Search string.

---

"REPLACESTRING", String:  
Replace string.

"HEXSEARCHSTRING", String:  
Search hex string.

"HEXREPLACESTRING", String:  
Replace hex string.

"STRINGSEARCH", Bool:  
TRUE, if last a string was searched.

"CASESENSITIVESEARCH", Bool:  
TRUE, if last a case sensitive string was searched.

"WILDSEARCH", Bool:  
TRUE, if within the last search a pattern was used.

"FILLSTRING", String:  
Fill string.

"HEXFILLSTRING", String:  
Fill hex string.

"STRINGFILL", Bool:  
TRUE, if last a string was filled.

"DISPLAYTYP", Number:  
Current defined display type.

"MAXUNDOMEMSIZE", Number:  
Size of the maximum UNDO memory.

"MAXUNDOLEVEL", Number:  
Size of the maximum UNDOs.

"ALTJUMP", Number:  
Jump width in lines for Alt-CURSORUP/DOWN.

"SCROLLBORDER", Number:  
Distance in lines to the upper/lower limit from where on the window shall be scrolled.

"CLIPBOARDUNIT", Number:  
Number of the clipboard for the exchange of data from blockbuffer into the clipboard.

"COMMANDSHELLWINDOW", String:  
Description of the commandshell window.

"COMMANDWINDOW", String:  
Description of the input and output window for arexxscripts.

"OVERWRITE", Bool,  
TRUE, if no requester before saving an already existing file shall appear.

---

"USEASL", Bool:

TRUE, if 'asl.library' and 'intuition.library' should be used for the requester. FALSE for 'reqtools.library'. gebraucht.

"DISPLAY", Number:

Defines in which form the file shall be displayed: 1 = only hexddisplay, 2 = only Asciiidisplay, 3 = Hex-ASCII mixed display.

"DISPLAYSPACES", Number:

Defines how many Bytes shall be grouped to a block in the hexdisplay: 3 = no blocks, 0 = one byte blocks, 1 = word blocks, 2 = long word blocks.

WINDOW-Object

=====

"LEFT", Number, read only:

Window offset to the left border.

"TOP", Number, read only:

Window offset to the right border.

"WIDTH", Number, read only:

Width of the window.

"HEIGHT", Number, read only:

Height of the window.

"TITLE", String, read only:

Current windowtitel.

"MIN\_WIDTH", Number, read only:

Minimum width of the window.

"MIN\_HEIGHT", Number, read only:

Minimum height of the window.

"MAX\_WIDTH", Number, read only:

Maximum width of the window.

"MAX\_HEIGHT", Number, read only:

Maximum height of the window.

"SCREEN", String, read only:

Name of the public screen on which the window shall be opened on.

---