

**Quill**

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

	<i>TITLE :</i> Quill		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 25, 2024	

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

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Quill</b>	<b>1</b>
1.1	Digital Quill Help . . . . .	1
1.2	About Quill . . . . .	1
1.3	Using the Help . . . . .	1
1.4	Pull-Down Menus . . . . .	2
1.5	Menus: Project - New . . . . .	3
1.6	Menus: Project - Clear . . . . .	3
1.7	Menus: Project - Open . . . . .	4
1.8	Menus: Project - Insert . . . . .	4
1.9	Menus: Project - Save . . . . .	4
1.10	Menus: Project - Save As . . . . .	4
1.11	Menus: Project - Print . . . . .	5
1.12	Menus: Project - Reveal - Next . . . . .	5
1.13	Menus: Project - Reveal - Previous . . . . .	5
1.14	Menus: Project - Reveal - Select . . . . .	5
1.15	Menus: Project - Close . . . . .	6
1.16	Menus: Project - About . . . . .	6
1.17	Menus: Project - Iconify . . . . .	6
1.18	Menus: Project - Quit . . . . .	6
1.19	Menus: Edit - Mark . . . . .	6
1.20	Menus: Edit - Cut . . . . .	6
1.21	Menus: Edit - Copy . . . . .	7
1.22	Menus: Edit - Paste . . . . .	7
1.23	Menus: Edit - Erase . . . . .	7
1.24	Menus: Edit - Kill Line . . . . .	7
1.25	Menus: Edit - Undo . . . . .	7
1.26	Menus: Edit - Redo . . . . .	8
1.27	Menus: Navigation - Find . . . . .	8
1.28	Menus: Navigation - Hunt . . . . .	8
1.29	Menus: Navigation - Find & Replace . . . . .	8

1.30	Menus: Navigation - Set Bookmark . . . . .	9
1.31	Menus: Navigation - Swap Bookmark . . . . .	9
1.32	Menus: Navigation - Go to Bookmark . . . . .	9
1.33	Menus: Navigation - Jump to Line . . . . .	9
1.34	Menus: Navigation - Match Bracket . . . . .	10
1.35	Menus: Format - Convert to Uppercase . . . . .	10
1.36	Menus: Format - Convert to Lowercase . . . . .	10
1.37	Menus: Format - Reformat Paragraph . . . . .	10
1.38	Menus: Format - Centre Line . . . . .	10
1.39	Menus: Macros - Start Learning . . . . .	11
1.40	Menus: Macros - Stop Learning . . . . .	11
1.41	Menus: Macros - Playback . . . . .	11
1.42	Menus: Macros - Save Macro . . . . .	11
1.43	Menus: Macros - Assign Macro . . . . .	11
1.44	Menus: Macros - ExecuteMacro . . . . .	13
1.45	Menus: Macros - Command Shell . . . . .	13
1.46	Menus: Settings - Print/Save Tabs? . . . . .	13
1.47	Menus: Settings - Create Icons? . . . . .	13
1.48	Menus: Settings - Display . . . . .	13
1.49	Menus: Settings - Memory . . . . .	14
1.50	Menus: Settings - Backup Time... . . . .	15
1.51	Menus: Settings - Editing - Insert Mode? . . . . .	15
1.52	Menus: Settings - Editing - Word Wrap? . . . . .	15
1.53	Menus: Settings - Editing - Auto-Indent? . . . . .	16
1.54	Menus: Settings - Editing - False Tabs? . . . . .	16
1.55	Menus: Settings - Editing - Tab Size . . . . .	16
1.56	Menus: Settings - Editing - RightMargin . . . . .	16
1.57	Menus: Settings - Environment - Free Cursor? . . . . .	16
1.58	Menus: Settings - Environment - Drag Marking? . . . . .	16
1.59	Menus: Settings - Environment - Fence Posting? . . . . .	17
1.60	Menus: Settings - Environment - Show Hex? . . . . .	17
1.61	Menus: Settings - Environment - SpeedBar Visible? . . . . .	17
1.62	Menus: Settings - Environment - Large SpeedButtons? . . . . .	17
1.63	Menus: Settings - Load Settings . . . . .	17
1.64	Menus: Settings - Save Settings . . . . .	18
1.65	Menus: Settings - Save Settings As . . . . .	18
1.66	Editing Window . . . . .	18
1.67	Keyboard Controls . . . . .	19
1.68	Command Language . . . . .	20

---

1.69	Introduction to the command language . . . . .	21
1.70	Quill and ARexx . . . . .	21
1.71	Command Reference: ABOUT . . . . .	22
1.72	Command Reference: ACTIVATE . . . . .	22
1.73	Command Reference: ASSIGNMACRO . . . . .	22
1.74	Command Reference: BACKGROUNDPEN . . . . .	22
1.75	Command Reference: BEEPSCREEN . . . . .	23
1.76	Command Reference: CHANGEWINDOW . . . . .	23
1.77	Command Reference: CENTRE . . . . .	23
1.78	Command Reference: CLEAR . . . . .	24
1.79	Command Reference: CLOSE . . . . .	24
1.80	Command Reference: CMDSHELL . . . . .	24
1.81	Command Reference: COLUMN . . . . .	24
1.82	Command Reference: COPY . . . . .	25
1.83	Command Reference: CR . . . . .	25
1.84	Command Reference: CURSOR . . . . .	25
1.85	Command Reference: CUT . . . . .	25
1.86	Command Reference: DEACTIVATE . . . . .	26
1.87	Command Reference: DELETECHAR . . . . .	26
1.88	Command Reference: DISPLAY . . . . .	26
1.89	Command Reference: ERASE . . . . .	26
1.90	Command Reference: ERASECHAR . . . . .	27
1.91	Command Reference: FIND . . . . .	27
1.92	Command Reference: FINDCHANGE . . . . .	28
1.93	Command Reference: FINDNEXT . . . . .	28
1.94	Command Reference: GETATTR . . . . .	28
1.95	Command Reference: GETCHAR . . . . .	30
1.96	Command Reference: GETLINE . . . . .	31
1.97	Command Reference: GETWORD . . . . .	31
1.98	Command Reference: GOTOBOKMARK . . . . .	31
1.99	Command Reference: GOTOCOLUMN . . . . .	31
1.100	Command Reference: GOTOLINE . . . . .	32
1.101	Command Reference: HELP . . . . .	32
1.102	Command Reference: INSERT . . . . .	32
1.103	Command Reference: KILL . . . . .	33
1.104	Command Reference: LEARN . . . . .	33
1.105	Command Reference: LINE . . . . .	33
1.106	Command Reference: LOADSETTINGS . . . . .	34
1.107	Command Reference: LOCKGUI . . . . .	34

---

1.108Command Reference: LOWERCASE . . . . .	34
1.109Command Reference: MARK . . . . .	34
1.110Command Reference: MATCHBRACKET . . . . .	35
1.111Command Reference: MEMORY . . . . .	35
1.112Command Reference: MOVEWINDOW . . . . .	35
1.113Command Reference: NEW . . . . .	36
1.114Command Reference: NEXTWORD . . . . .	36
1.115Command Reference: OPEN . . . . .	36
1.116Command Reference: PASTE . . . . .	36
1.117Command Reference: POSITION . . . . .	37
1.118Command Reference: PREVIOUSWORD . . . . .	37
1.119Command Reference: PRINT . . . . .	38
1.120Command Reference: QUIT . . . . .	38
1.121Command Reference: REDO . . . . .	38
1.122Command Reference: REFORMAT . . . . .	39
1.123Command Reference: REMAKESCREEN . . . . .	39
1.124Command Reference: REQUESTFILE . . . . .	39
1.125Command Reference: REQUESTITEM . . . . .	40
1.126Command Reference: REQUESTNOTIFY . . . . .	40
1.127Command Reference: REQUESTNUMBER . . . . .	40
1.128Command Reference: REQUESTRESPONSE . . . . .	41
1.129Command Reference: REQUESTSTRING . . . . .	41
1.130Command Reference: REVEAL, SELECT . . . . .	41
1.131Command Reference: RX . . . . .	42
1.132Command Reference: RXS . . . . .	42
1.133Command Reference: SAVE . . . . .	43
1.134Command Reference: SAVEAS . . . . .	43
1.135Command Reference: SAVESETTINGS . . . . .	43
1.136Command Reference: SAVESETTINGSAS . . . . .	44
1.137Command Reference: SCREENTOBACK . . . . .	44
1.138Command Reference: SCREENTOFRONT . . . . .	44
1.139Command Reference: SETATTR . . . . .	45
1.140Command Reference: SETBOOKMARK . . . . .	46
1.141Command Reference: SETSTATUSBAR . . . . .	46
1.142Command Reference: SIZEWINDOW . . . . .	46
1.143Command Reference: SWAPBOOKMARK . . . . .	47
1.144Command Reference: SYSTEM . . . . .	47
1.145Command Reference: TEXT . . . . .	47
1.146Command Reference: TEXTPEN . . . . .	48

---

1.147Command Reference: UNDO . . . . .	48
1.148Command Reference: UNLOCKGUI . . . . .	48
1.149Command Reference: UNZOOMWINDOW . . . . .	48
1.150Command Reference: UPPERCASE . . . . .	49
1.151Command Reference: WINDOWTOBACK . . . . .	49
1.152Command Reference: WINDOWTOFRONT . . . . .	49
1.153Command Reference: ZOOMWINDOW . . . . .	50

---

# Chapter 1

## Quill

### 1.1 Digital Quill Help

---

#### Digital Quill Help Table of Contents

---

- About Quill
- Using the Help
- Pull-Down Menus
- Editing Window
- Keyboard Controls
- Command Language (ARexx)

### 1.2 About Quill

Digital Quill is Copyright © 1992-94 by Timothy Aston, all rights reserved, published by Phantom Development.

Digital Quill is a general purpose text editing tool, suitable for a wide variety of tasks, for example: developing programmes, writing ARexx macros or shell scripts, editing your User-Startup file, typing letters, etc.

It has been designed with user-friendliness in mind, so that even novice users may take advantage of its power, flexibility and speed. The goal in writing it was to produce a friendly and comfortable text editing environment, rather than one full of confusion and esoteric features.

### 1.3 Using the Help

---



You are in Digital Quill's online help system. This is a context-sensitive hypertext help system, implemented via AmigaGuide®. You have probably used AmigaGuide® before when reading online documentation of other programmes. If, however, you are not familiar with using it, simply press the "Help" gadget above.

Online help is not as a replacement to the user's manual, and is not designed to teach you how to use the programme. It is designed to provide you with easy access to information about using the programme while you are using it. It contains descriptions of all the keystrokes, requesters, menu items, macro commands, etc.

## 1.4 Pull-Down Menus

```
Project
  New
  Clear...
  Open...
  Insert...
  Save
  Save As...
  Print...
  Reveal      »
    Next
    Previous
    Select...
  Close...
  About...
  Iconify
  Quill Digital Quill...
```

```
Edit
  Mark
  Cut
  Copy
  Paste
  Erase
  Kill Line
  Undo
  Redo
```

```
Navigation
  Find...
  Hunt
  Find & Replace...
  Set Bookmark
  Swap Bookmark
  Go To Bookmark
  Jump to Line...
  Match Bracket
```

```
Format
  Convert to Uppercase
  Convert to Lowercase
```

---

- Reformat Paragraph
- Centre Line

#### Macros

- Start Learning
- Stop Learning
- Playback
- Save
- Assign Macro...
- Execute Macro...
- Command Shell...

#### Settings

- Save/Print Tabs?
- Create Icons?
- Display...
- Memory...
- Backup Time...
- Editing »
  - Insert Mode?
  - Word Wrap?
  - Auto-Indent?
  - False Tabs?
  - Tab Size...
  - Right Margin...
- Environment »
  - Free Cursor?
  - Drag Marking?
  - Fence Posting?
  - Show Hex?
  - SpeedBar Visible?
  - Large SpeedButtons?
- Load Settings...
- Save Settings
- Save Settings As...

## 1.5 Menus: Project - New

Opens up a new empty editing window in front of the currently active window. This allows you to edit multiple files at once. You can move from one editing window to another via the system depth gadgets, or using the Quill commands `Next`, `Previous` or `Reveal`.

There is currently a limit of 10 editor windows that can be open at any one time.

Command reference: `NEW`

## 1.6 Menus: Project - Clear

Clears from memory the project in the current editor window, but will not touch the disk file. If you have not saved that file since you last made a

---

change to it, you will be given a chance to save it before the file is cleared.

Command reference: CLEAR

## 1.7 Menus: Project - Open

Allows you to load a new text file into the current editor window. If a file was already loaded into the current window, it will be replaced. If you have not saved that file since you last made a change to it, you will be given a chance to save it before the new file is opened.

Command reference: OPEN

## 1.8 Menus: Project - Insert

Lets you insert a file into the current project at the present cursor position.

Command reference: INSERT

## 1.9 Menus: Project - Save

Saves the file in the current editor window to disk, using the current project name. If no name has been specified yet, you'll be able to select a name using the file requester.

If a block is current being marked, then only the marked area will be saved.

Be careful about overwriting an existing file, you will not be warned about this.

If you wish to save the file under a name other than the current one, use the Save As command.

Command reference: SAVE

## 1.10 Menus: Project - Save As

Presents you with the file requester in order to select the name to save the file in the current window under.

If a block is current being marked, then only the marked area will be saved.

Command reference: SAVEAS

---

## 1.11 Menus: Project - Print

Sends the file in the current editor window to the print "as-is". Be sure that you have your printer ready and setup as you want it to be via the Workbench Printer preferences.

If a block is current being marked, then only the marked area will be printed.

Command reference: `PRINT`

## 1.12 Menus: Project - Reveal - Next

Takes you to the next open editor window, if you have more than one window open. This works on a circular basic, if you do a next from the last editor window, you'll get the first window.

This command is the opposite of the `Previous` command.

See also the `Reveal` command which allows you to select any of the open editor windows.

Command reference: `SELECT NEXT`

## 1.13 Menus: Project - Reveal - Previous

Takes you to the previous editor window, if you have more than one window open. This works on a circular basic, if you do a previous from the first editor window, you'll get the last window.

This command is the opposite of the `Next` command.

See also the `Reveal` command which allows you to select any of the open editor windows.

Command reference: `SELECT PREVIOUS`

## 1.14 Menus: Project - Reveal - Select

Presents you with a requester that allows you to select from any of the currently open editor windows. Also displayed in this requester will the project's icon, as well as its length, etc.

You can also use the `window depth` gadgets to move from window to window, or the `Next` and `Previous` commands.

Command reference: `SELECT PROMPT`

---

## 1.15 Menus: Project - Close

Closes the current editor window. If no other windows are currently open then a requester will appear prompting you to either quit or start a new untitled project. If you have not saved that file since you last made a change to it, you will be given a chance to save it before the file is closed.

Command reference: CLOSE

## 1.16 Menus: Project - About

Presents the copyright information about Digital Quill .

Command reference: ABOUT

## 1.17 Menus: Project - Iconify

Closes down all editor windows and reduces to a simple icon on the Workbench. Simply double-click on this icon and all windows will re-open into the same state they were before you selected Iconify.

Command reference: DEACTIVATE

## 1.18 Menus: Project - Quit

Exits Digital Quill entirely. If the project in any of the editor windows has not been saved since it was last modified, you will be given a chance to save them before quitting.

Command reference: QUIT

## 1.19 Menus: Edit - Mark

Puts the cursor into mark mode. As you move the cursor, you will be highlighting text for such operations as cut and erase . You can duplicate the action of this command by double-clicking the mouse, or simply dragging the mouse over the text, if drag-marking is on.

Command reference: MARK

## 1.20 Menus: Edit - Cut

---

Cuts the current marked area. The marked area is copied to the clipboard and then erased from the window. The standard system clipboard is used, so that the data cut can be pasted into other any editor window, as well as any other applications that are using the system clipboard.

Command reference: CUT

## 1.21 Menus: Edit - Copy

Copies the marked area to the clipboard. The standard system clipboard is used, so that the data cut can be pasted into other any editor window, as well as any other applications that are using the system clipboard.

Command reference: COPY

## 1.22 Menus: Edit - Paste

Takes the text from the standard system clipboard and inserts it into the current editor project at the current cursor position. The data in the clipboard may have been cut or copied from any Quill editor window or any other application using the standard system clipboard.

Command reference: PASTE

## 1.23 Menus: Edit - Erase

Erases the currently marked area. If drag-marking is on, the same action can be done by pressing either the Erase or Delete.

Command reference: ERASE

## 1.24 Menus: Edit - Kill Line

Removes the entire contents of the current line in the current editor project including the newline.

Command reference: KILL LINE

## 1.25 Menus: Edit - Undo

Reverses the last edit made. Anything that changes the current project, entering text, deleting text, doing a find and replace, pasting from the clipboard, erasing a block, etc. can all be undone. You can undo for numerous levels, the limits of which are defined in the Memory requester.

Command reference: UNDO

---

## 1.26 Menus: Edit - Redo

If you have undone an edit and decide that you did not in fact want to undo it, you can redo that edit with this function.

Command reference: REDO

## 1.27 Menus: Navigation - Find

Tries to find a string of text in the current editor project. A requester will appear prompting you to enter the string of text you wish to find. You also have the choice of setting the search as case sensitive, or using AmigaDOS pattern matching in the search. The search will proceed from the current cursor position until the string has been found or the end of the project has been reached. The search can be repeated by simply using the Hunt command.

By default the search is case in-sensitive, that is, when looking for the string, upper and lowercase letters will be considered the same. With the case sensitive option on, the string must match exactly.

Optionally, pattern-matching can be used in the finds. Pattern matching is powerful facility that will search for a string that matches a pattern description, rather than doing a simple equivalence check. If you've used AmigaDOS pattern matching before (for example, "dir devs:#!.device", which the command to list all the files in devs: that end with the .device suffix) then you should be able to easily understand Quill's pattern matching searches. It does the same thing, supporting all the pattern matching symbols that AmigaDOS support (such as: #, ?, (), [], ~, ', etc.), searching from the current cursor position for text that matches the pattern you enter. If you are not familiar with AmigaDOS pattern matching, consult your system manuals for a thorough description.

Command reference: FIND

## 1.28 Menus: Navigation - Hunt

Repeats that last find. If you haven't yet done a find, this command works exactly like the Find command, prompting you with the Find requester.

Command reference: FINDNEXT

## 1.29 Menus: Navigation - Find & Replace

Similar to the Find command, except that it allows you to subsequently replace occurrences of the string you're trying to find.

The Find & Replace requester is very similar to the Find requester, except that it has some addition gadgets. After entering the string to find, the

---

next occurrence of it (if any) will be searched for (starting from the current cursor position). You can then type in a string to possibly replace the string found with. If you wish to replace it, click on the "Replace" gadget. The old string will be deleted and the new one inserted, and the next occurrence of the find string will be searched for. If you don't want to replace that particular occurrence, simply click on "Find Next" and no replace will be done, and the next occurrence will be searched for.

You also have the choice of automatically replacing every single occurrence of the find string from the current cursor position on by clicking on the "Replace All" gadget. Warning: this operation cannot be aborted, it should be used with care.

Like with Find, you have the option of case sensitivity, and pattern matching. Pattern matching generally is not suited to doing replaces, since the string actually found will vary and Quill cannot know what exactly must be deleted before the replace string is inserted. Only use pattern matching with replace if you really know what you're doing.

Command reference: FINDCHANGE PROMPT

### 1.30 Menus: Navigation - Set Bookmark

Places a bookmark at the current cursor position. This means that the position of the bookmark will be remembered so that it can quickly be jumped to later with either Goto Bookmark or Swap Bookmark .

Command reference: SETBOOKMARK

### 1.31 Menus: Navigation - Swap Bookmark

Jumps to the bookmark , making the old cursor position the new bookmark. This can be used to quickly toggle back and forth between two spots in a project.

Command reference: SWAPBOOKMARK

### 1.32 Menus: Navigation - Go to Bookmark

Jumps to the bookmark .

Command reference: GOTOBOKMARK

### 1.33 Menus: Navigation - Jump to Line

Presents you with a requester asking for a line number to jump to.

Command reference: GOTOLINE

---



### 1.34 Menus: Navigation - Match Bracket

If the cursor is currently on a bracket, parenthesis, brace, or inequality sign, the matching symbol will be found and jumped to. This facility fully handles nested brackets.

Command reference: `MatchBracket`

### 1.35 Menus: Format - Convert to Uppercase

Converts from the cursor position to the end of the word to all uppercase letters, regardless of what case they were before. Or, if an area is being marked, that entire area will be converted to all uppercase.

Command reference: `UPPERCASE`

### 1.36 Menus: Format - Convert to Lowercase

Converts from the cursor position to the end of the word to all lowercase letters, regardless of what case they were before. Or, if an area is being marked, that entire area will be converted to all lowercase.

Command reference: `LOWERCASE`

### 1.37 Menus: Format - Reformat Paragraph

Formats the current paragraph into word-wrapped lines that fit within the right margin setting. Quill has no hard definition of a paragraph, only of a line, so it will attempt to detect the end of the paragraph by looking for the following: two consecutive newlines, a newline followed by a space or tab, or the end of text. This should be kept in mind when using this function or you might end up with unexpected results. Reformat Paragraph generally only works well with words written in simple paragraph form with no extra spacing and indenting for formatting.

Command reference: `REFORMAT`

### 1.38 Menus: Format - Centre Line

Centres the current line within the bounds of the right margin.

Command reference: `CENTRE`

---

### 1.39 Menus: Macros - Start Learning

Starts learning a macro. All actions between when you issue this command and Stop Learning will be remembered, and can be later played-back. The macro you learned can be subsequently saved to disk using the Save Macro function. The macro on disk can then be used via the Execute or Assign Macro commands.

Command reference: LEARN

### 1.40 Menus: Macros - Stop Learning

Stops learning the macro you started with Start Learning. Now the macro can be played back or saved to disk.

Command reference: LEARN STOP

### 1.41 Menus: Macros - Playback

If a macro has been previously learned then it can be played back with this command. This allows you to replay all actions between the start and end of learning.

Command reference: LEARN PLAY

### 1.42 Menus: Macros - Save Macro

If a macro has been previously learned then it can be saved to disk with this command. Then this macro can be later used via the Execute or Assign Macro commands.

Command reference: LEARN SAVE

### 1.43 Menus: Macros - Assign Macro

This very powerful requester allows you to define macros of various types for use throughout Quill. Macros are actions, or groups of actions. Macros in Quill allow you to automate just about every function in Quill. Currently, the types of macros supported are:

- Keystroke macros - Macros that are activated by pressing a key combination that you describe.
  - Menu Item macros - These are menu items that you define for the User pull-down menu that activate macros.
  - Speed Button - These are buttons in the SpeedBar (a line of buttons just
-

below the window title bar) that invoke macros.

You select what type of macro by pressing one of the buttons along the top of the Assign Macro requester.

All types are defined in a similar way. The gadgets for working on a macro are as follows:

- Macros - Located at the left of the requester, this is a list of all the macros you've defined (if any). If you want to view or edit the definition of a macro, just click on one of the items in this list.
- New Macro - Creates a new macro. Clicking this button creates a new blank macro for you to fill in the appropriate information.
- Delete Macro - Deletes the macro that is currently selected.
- Move Up - Moves the selected macro up one in the list.
- Move Down - Moves the selected macro down one in the list.
- Command for Macro - This is the most important part of each macro, it is the command that the macro dispatches. There are several ways to enter the command: Select from a list of all the Quill commands, select a disk macro (i.e. an ARexx macro), or select an AmigaDOS command to run. Each of these will enter the appropriate command into the string gadget. You can also type in the command itself into the string gadget, which will be the normal means of entering a command of any complexity. Pressing the "?" button brings up help on the command you have chosen.
- Key Combination (only displayed when defining Keystroke macros) - Keystroke macros need a key combination that invokes them, and you define that in here. The keystroke descriptions are the same as Commodities hotkey descriptions, except that you are limited strictly to input that can be entered via the keyboard. See your Amiga user manual for more information.
- ~Menu Item Text (only displayed when defining Menu Item macros) - This is the text that will appear under the "User" pull-down menu for your macro.
- Images (only displayed when defining SpeedButtons) - This is the list of images that you can choose for your SpeedButton. Many images have default commands associated with them which will automatically be set when you select an image, if no command has already been defined. This helps when setting up commonly used commands (like cut and paste).
- Starts a New Group (only displayed when defining Menu Item macros and SpeedButtons) - For visual cleanliness, it is possible to group Menu Item macros and SpeedButtons by adding space between them (this wouldn't make any sense for Keystroke macros since you can't see them). For Menu Item macros, this takes the form of a separator bar in the menu, and for SpeedButtons it takes the form of an extra gap between buttons.

Along the bottom are the standard buttons that appear in most requesters:

---

OK, Help and Cancel.

Command reference: ASSIGNMACRO

## 1.44 Menus: Macros - ExecuteMacro

Executes a macro from disk. Macros in Quill are ARexx macros, thus are generally found in the REXX: drawer and end in the .quill suffix.

Command reference: RX

## 1.45 Menus: Macros - Command Shell

Opens up the command shell window. This allows you to type any Quill command and have it processed immediately. Commands typed here actually get processed by ARexx before they go through Quill's command interpreter, thus you can use any ARexx command, construct or function in the Command Shell.

Command reference: CMDSHELL

## 1.46 Menus: Settings - Print/Save Tabs?

When this option is on, true tab characters (ASCII 9) will be preserved when printing and saving text. Otherwise, tabs will be expanded into spaces.

Command reference: SETATTR PROJECT SAVETABS

## 1.47 Menus: Settings - Create Icons?

If on, this option will cause insure that any file saved has an icon to go with it. Quill will look in the ENV:Quill directory and in the Icons directory within the directory Quill resides for different file types when the file being saved does not already have an icons.

Command reference: SETATTR PROJECT CREATEICONS

## 1.48 Menus: Settings - Display

This requester is used to setup the Quill display. It allows you to select what screen or screen mode Quill will use, the font for text, the screen's palette and the pens used for text. The gadgets for controlling these items are as follows:

---

- Screen Type - There are three screen types that Quill can be run on: the Workbench screen, another application's public screen, or a custom public screen of Quill's own.
- Select Screen - If you have chosen to run Quill on a public screen, this button brings up a list of all available public screens for you to select one. If you are going to run Quill on a custom screen, this button presents you with a list of available screen modes to choose from. This button is disabled if you have chosen to run Quill on the Workbench.
- Select Text Font - This allows you to select which font to use for the text in Quill's edit windows. This font must be a monospaced (fixed-width) font, such as the Topaz font.
- Screen Palette - These gadgets allow you to adjust the palette if Quill is running on a custom screen.
- ~Text - The pen you wish to use for text in the Quill editing windows.
- Background - The pen you wish to use for the background behind the text in Quill's edit windows.

Along the bottom are the standard buttons that appear in most requesters: OK, Help and Cancel.

Command reference: DISPLAY

## 1.49 Menus: Settings - Memory

» FOR ADVANCED USERS ONLY

The Memory Settings requester allows you to modify how Quill manages memory. This is intended strictly for advanced users, and in most cases you will not need to make any alterations. If you do wish to make some changes, however, the gadgets available to you are as follows:

- ~Initial Buffer Size - When a new project is created, this is the amount of space initially given to that project. This space will grow as needed and as memory permits.
  - Buffer Growth Increment - When the current amount of space allocated for the text buffer is exceeded, a new buffer will be allocated which will be the of kilobytes you select larger than the now full buffer.
  - ~Work Buffer Size - Specifies the size of the work buffer. This is used for many things, amongst them, holding a temporary copy of the current line being edited. This means that no line can be longer than the size of the work buffer.
  - ~Clipboard Buffer Size - Clipboard operations use a buffer. If you do large clips and are finding that they are being truncated, simply increase the amount of memory allocated for the clipboard buffer.
-

- Clipboard Unit - Though the clipboard is global, there are actually 255 different units of it, so that certain applications can share different clipboards and be excluded from others. The "primary clipboard" is unit 0, which most Amiga applications make use of.
- ~Undo Levels - This is the maximum amount of undo levels Quill will keep. You will not be able to go back farther than this number of steps.
- ~Undo Memory - Quill will not allow the memory used to keep track of undo steps to exceed this amount. Global find & replaces take up the most memory to undo, so if you need to undo them often, you may wish to increase this.

Command reference: MEMORY

## 1.50 Menus: Settings - Backup Time...

Allows you to configure the time between automatic timed backups. During a backup, all open projects will get saved as <filename>.backup. This is to protect against accidental loss of data so that you will have a recent version of your file to revert so.

Backup time is in minutes. To disable timed backups altogether, enter a time of 0.

Command reference: SETTATTR PROJECT BACKUPTIME

## 1.51 Menus: Settings - Editing - Insert Mode?

When Insert Mode is on, typing in characters causes all characters under the cursor and beyond to be moved to the right to make a space for the new character. When off, entering characters simply overwrites the characters that were there.

Command reference: SETTATTR PROJECT EDITING INSERT

## 1.52 Menus: Settings - Editing - Word Wrap?

The Word Wrap option gives simple word wrapping capabilities when entering text. Once the end of a word is reached, if the cursor is beyond the right margin, that word will be moved to a new line automatically. This is the only time any checking for word wrapping will be done, paragraphs are not kept dynamically wrapped, you have to use Reformat Paragraph to make a paragraph properly word-wrapped.

Command reference: SETTATTR PROJECT EDITING WORDWRAP

---

### 1.53 Menus: Settings - Editing - Auto-Indent?

Auto-Indent is an option useful to programmers mostly. When you press Return, your new line will be given the same indentation as the previous line. The indentation is the number of tabs or spaces before the first character or the cursor position (whichever comes first).

Command reference: `SETTATTR PROJECT EDITING AUTOINDENT`

### 1.54 Menus: Settings - Editing - False Tabs?

When this option is on, pressing the Tab key inserts the appropriate number of spaces, rather than a true tab character (ASCII 9).

Command reference: `SETTATTR PROJECT EDITING FALSETABS`

### 1.55 Menus: Settings - Editing - Tab Size

Specifies the tab size, in characters. Valid sizes are from 1 to 10 characters, with the default being 4.

Command reference: `SETTATTR PROJECT EDITING TABSIZE`

### 1.56 Menus: Settings - Editing - RightMargin

Specifies the right margin, to be used when word wrapping , reformatting and centring .

Command reference: `SETTATTR PROJECT EDITING MARGIN`

### 1.57 Menus: Settings - Environment - Free Cursor?

Free Cursor mode allows the cursor to be positioned anywhere on a line, not bound by the line length.

Command reference: `SETTATTR PROJECT ENVIRONMENT FREECURSOR`

### 1.58 Menus: Settings - Environment - Drag Marking?

Drag Marking means to be able to mark text simply by dragging the mouse, and not first having to select mark mode via the menus or via double-clicking.

Command reference: `SETTATTR PROJECT ENVIRONMENT DRAGMARKING`

---

## 1.59 Menus: Settings - Environment - Fence Posting?

Toggles fence posting mode. With fence posting on, whenever you enter a closing bracket, Quill will jump to the opening bracket and flash the cursor over it for a moment, before restoring the cursor position. This can be especially useful for ensuring complex bracketed expressions are entered correctly.

Command reference: `SETTATTR PROJECT ENVIRONMENT FENCEPOSTING`

## 1.60 Menus: Settings - Environment - Show Hex?

Toggles show hex mode. With show hex on, the hexadecimal value of the character that the cursor is over will be shown in the window title bar.

Command reference: `SETTATTR PROJECT ENVIRONMENT SHOWHEX`

## 1.61 Menus: Settings - Environment - SpeedBar Visible?

If you have configured a SpeedBar, you can make them visible or invisible. If you have configured SpeedButtons and are finding that your SpeedBar is not showing up, check to make sure this option is on.

Command reference: `SETTATTR PROJECT ENVIRONMENT SPEEDBAR`

## 1.62 Menus: Settings - Environment - Large SpeedButtons?

If you have configured a SpeedBar, you can toggle them between the normal size and the larger size. For the larger size to work, the file "SpeedButtons.8" must be in Quill's directory.

Command reference: `SETTATTR PROJECT ENVIRONMENT LARGEBUTTONS`

## 1.63 Menus: Settings - Load Settings

Allows you to load a settings file from disk using the file requester. Some settings affect all projects (text font, colours) while others only affect the current project.

The settings consist of all parameters set via the Settings pull-down menu, and in addition: the current window position and size, the current directory, the current file requester pattern, etc.

Command reference: `LOADSETTINGS`

---



## 1.64 Menus: Settings - Save Settings

Saves the Digital Quill settings to disk.

The settings consist of all parameters set via the Settings pull-down menu, and in addition: the current window position and size, the current directory, the current file requester pattern, etc.

Command reference: SAVESETTINGS

## 1.65 Menus: Settings - Save Settings As

Presents you with the file requester to select a file to save the Digital Quill settings to.

The settings consist of all parameters set via the Settings pull-down menu, and in addition: the current window position and size, the current directory, the current file requester pattern, etc.

Command reference: SAVESETTINGSAS

## 1.66 Editing Window

Digital Quill loads text into standard Amiga Intuition windows that you will be very familiar with. Around the border of it, you see the standard close gadget on the top left, which is synonymous to the CLOSE command. Over to the right is the zip gadget and the depth gadget, and down the right hand-side of the windows you'll see a scrollbar and arrows, with the standard sizing gadget just below that.

Also in the window border is some status information. Next to the close gadget on the top left is given the number of the project window you've got open, as well as the filename of the project being worked on. Over to the right is the modes of the window, this looks something like:

I·w·A·t·C·D·f·S

Each letter represents a mode item, and it is capitalized when the item is on and in lower case when it is off. Each letter corresponds to the following:

- I - Insert mode
- W - Word wrap
- A - Auto-indent
- T - False tabs
- C - Free Cursor
- D - Drag marking
- F - Fence posting
- S - Save/print tabs

In the above example, that modes would be insert mode on, word wrap off, auto-indent on, false tabs off (use real tabs), free cursor on, drag

---

marking on, fence posting on, and save/print tabs on.

If you have the `Show Hex` option on, then just to the right of the modes will be the hexadecimal ASCII value of the character you're over. And just to the right of that you are shown the current Line and Column that the cursor is on in your project.

## 1.67 Keyboard Controls

Most keystrokes you enter will go directly into the text of the currently active editor window. Quill recognizes your current keymap settings, and will allow you to enter high-bit characters, "dead-key" sequences, etc.

In addition, Quill has several special keystrokes that it recognizes:

`Return`

Splits the current line into two at the current cursor position. If `Auto-Indent` is on, the new line will be indented in the same way as the line you were just on.

`Shift-Return`

Same as above, except the the compliment of `Auto-Indent` is looked at. i.e. if `Auto-Indent` is on, the line will not be indented, but will be if it is off.

`Enter`

Same as `Shift-Return`.

`Shift-Enter`

Same as `Return`.

`CursorUp` / `CursorDown`

Moves the cursor up or down one line. If the cursor is at the upper or lower extreme of the screen, the text will scroll, unless you are at the appropriate extreme of the project.

`Shift-CursorUp` / `Shift-CursorDown`

Moves the cursor to the upper or lower extreme of the screen. If the cursor is already at the appropriate extreme of the screen, the text will scroll up or down one screenful of text.

`Ctrl-CursorUp` / `Ctrl-CursorDown`

Moves the cursor to the top or bottom of the current project.

`Alt-CursorUp` / `Alt-CursorDown`

Same as above.

`CursorLeft` / `CursorRight`

Moves the cursor left or right one character. If the cursor is at the start of the line and you try to move left, it will go up to the end of the previous line.

`Shift-CursorLeft` / `Shift-CursorRight`

Moves the cursor to the left or right extreme of the screen. If the cursor is already at this extreme, the screen will be scrolled left or right one screenful, until you're at the start or finish of the line.

`Ctrl-CursorLeft` / `Ctrl-CursorRight`

Moves the cursor to the extreme left or right of the current line.

`Alt-CursorLeft` / `Alt-CursorRight`

Moves the cursor to the start/end of the previous or next word.

`Backspace`

Deletes the character to the immediate left of the cursor. If the cursor is on the first character on a line, then the current line will be joined with the previous line. If `drag-marking` is on, it will

erase the currently marked area if you were marking.

#### Delete

Deletes the character currently under the cursor (if there is one). If drag-marking is on, it will erase the currently marked area if you were marking.

#### Shift-Backspace

Deletes all characters from the cursor position to the beginning of the current line.

#### Shift-Delete

Deletes all characters from the cursor position to the end of the current line.

#### Ctrl-Backspace

Deletes all characters from the cursor position to the beginning of the current word.

#### Ctrl-Delete

Deletes all characters from the cursor position to the end of the current word.

#### Help

Takes you into Quill's interactive hypertext AmigaGuide® help system (which is what you're using right now). Help can be accessed while in any editing window or requester and will provide context-sensitive aid. You can also press the Help key while the mouse is over a pull-down menu item for information about that.

Do not forget that additional keystrokes can be configured to do whatever you want via the Assign Macro requester.

Quill's requesters also provide extensive keyboard controls. You will notice that the labels for most gadget labels have a letter underlined. This is the key you use to access that gadget. In the case of cycle gadgets or listview gadgets, you can hold down Shift and press the key equivalent to move backwards through the list, while unshifted moves you forwards through the list. In all requesters (with the exception of the system requesters for selecting files, fonts and screenmodes) you can press Return as an equivalent to selecting "OK" and Escape as an equivalent to selecting "Cancel".

## 1.68 Command Language

- Introduction to the command language
- Quill and ARexx

#### Command Reference:

ABOUT	GOTOCOLUMN	REQUESTRESPONSE
ACTIVATE	GOTOLINE	REQUESTSTRING
ASSIGNMACRO	HELP	REVEAL
BACKGROUNDPEN	INSERT	RX
BEEPSCREEN	KILL	RXS
CENTRE	LEARN	SAVE
CHANGEWINDOW	LINE	SAVEAS
CLEAR	LOADSETTINGS	SAVESETTINGS
CLOSE	LOCKGUI	SAVESETTINGSAS
CMDSHELL	LOWERCASE	SCREENTOBACK
COLUMN	MARK	SCREENTOFRONT

COPY	MATCHBRACKET	SELECT
CR	MEMORY	SETATTR
CURSOR	MOVEWINDOW	SETBOOKMARK
CUT	NEW	SETSTATUSBAR
DEACTIVATE	NEXTWORD	SIZEWINDOW
DELETECHAR	OPEN	SWAPBOOKMARK
DISPLAY	PASTE	SYSTEM
ERASE	POSITION	TEXT
ERASECHAR	PREVIOUSWORD	TEXTPEN
FIND	PRINT	QUIT
FINDCHANGE	REDO	UNDO
FINDNEXT	REFORMAT	UNLOCKGUI
GETATTR	REMAKESCREEN	UNZOOMWINDOW
GETCHAR	REQUESTITEM	UPPERCASE
GETLINE	REQUESTFILE	WINDOWTOBACK
GETWORD	REQUESTNOTIFY	WINDOWTOFRONT
GOTOBOOKMARK	REQUESTNUMBER	ZOOMWINDOW

## 1.69 Introduction to the command language

The inner workings of Digital Quill are based around its internal command language. Most of the time, this language is hidden from you, but every gadget click, menu-selection, and even macros you record, go through this language. Quill is thus known as a "command-driven" application.

Because the processing of its commands is fundamental to the operation of Quill, understanding its command set will help you to better use the programme. As a user, you access the command set directly when using macros in Quill, its command shell, as well as via its ARexx interface. And, as mentioned above, you access it indirectly when doing just about anything else.

In the command reference section, all commands are described in three ways, usage, template, and a general description.

- The usage shows what you would type to use the command. Parts that are optional are enclosed in brackets []. Parts where you must enter some kind of information are enclosed in inequality signs <>. Finally, to indicate a choice the items to choose from will be separated by a pipe |.
- The template is a standard AmigaDOS template and this is the actual argument template used to parse the command arguments. See your Amiga system manual for a description of AmigaDOS templates.
- The general description provides details on using the command and on what the various options do. Note that many commands return values. If you are using ARexx, the values will be returned in the variable RESULT. If there was an error with the command, the variable RC will contain the value 5.

## 1.70 Quill and ARexx

ARexx is the Amiga's macro processing scripting language, it was included with your system. ARexx is very simple yet extremely powerful, allowing to very easily and quickly create programmes. What makes it unique and extraordinarily powerful is its ability to communicate with any application that has an ARexx port.

Quill makes good use of this power. Via its ARexx port, ARexx macros can access any of Quill's commands. This allows you to automate virtually any kind of edit action, and allows you to extend Quill virtually as far as your ability and imagination will let you.

Macros learned within Quill are actually ARexx macros and are played back by running them through the ARexx interpreter. When you execute a macro you are starting an ARexx macro. Quill's ARexx port is a direct link for any outside applications to Quill's extensive command set.

## 1.71 Command Reference: ABOUT

Usage: ABOUT  
Template: ,

Presents the copyright information about Digital Quill in a requester.

## 1.72 Command Reference: ACTIVATE

Usage: ACTIVATE  
Template: ,

Brings the Quill screen to the front and makes the current project window (the one that was last active) the currently active window. If Quill was iconified then it will be de-iconified first.

See Also:  
DEACTIVATE

## 1.73 Command Reference: ASSIGNMACRO

Usage: ASSIGNMACRO  
Template: ,

Presents the Assign Macro requester for configuring macros in Quill.

## 1.74 Command Reference: BACKGROUNDPEN

---

Usage: BACKGROUNDPEN <pen number>  
Template: /N

Changes the colour of the background pen in all project windows to the given pen number.

See Also:  
    TEXTPEN

## 1.75 Command Reference: BEEPSCREEN

Usage: BEEPSCREEN  
Template: ,

Flashes and/or beeps the Quill screen (depending on the user's system settings).

## 1.76 Command Reference: CHANGEWINDOW

Usage: CHANGEWINDOW [LEFTEDGE] <left edge> [TOPEDGE] <top edge>  
                    [WIDTH] <width> [HEIGHT] <height>  
Template: LEFTEDGE/N, TOPEDGE/N, WIDTH/N, HEIGHT/N

Moves and sizes the current project window in one operation.

- LEFTEDGE is the new left edge you want.
- TOPEDGE is the new top edge you want.
- WIDTH is the new width you want.
- HEIGHT is the new height you want.

See Also:  
    MOVEWINDOW  
    SIZEWINDOW  
    ZOOMWINDOW  
    UNZOOMWINDOW  
    WINDOWTOFRONT  
    WINDOWTOBACK  
    SCREENTOFRONT  
    SCREENTOBACK

## 1.77 Command Reference: CENTRE

Usage: CENTRE  
Template: ,

Centres the current line within the bounds of the right margin .

---

## 1.78 Command Reference: CLEAR

Usage: CLEAR [FORCE]

Template: FORCE/S

Clears from memory the project in the current editor window, but will not touch the disk file. If you have not saved that file since you last made a change to it, you will be given a chance to save it before the project is cleared

- FORCE suppresses the Modified Project requester.

## 1.79 Command Reference: CLOSE

Usage: CLOSE [FORCE]

Template: FORCE/S

Closes the current editor window. If that was the only open window, Quill will prompt you to either exit completely or start a new untitled project. Otherwise, the next window will become the current one.

- FORCE suppresses the Modified Project requester.

## 1.80 Command Reference: CMDSHELL

Usage: CMDSHELL

Template: ,

Opens up the interactive ARexx command shell.

## 1.81 Command Reference: COLUMN

Usage: COLUMN <displacement>

Template: /N/A

Moves the cursor left or right relative to the current cursor position. Passing a negative value moves the cursor left, a positive value moves the cursor right. The screen will automatically scroll as appropriate.

See Also:

- GOTOLINE
- GOTOCOLUMN
- CURSOR
- LINE
- NEXTWORD
- PREVIOUSWORD
- POSITION

## 1.82 Command Reference: COPY

Usage: COPY

Template: ,

Copies the marked area to the clipboard. The standard system clipboard is used, so that the data cut can be pasted into other any editor window, as well as any other applications that are using the system clipboard.

See Also:

MARK  
CUT  
PASTE  
ERASE

## 1.83 Command Reference: CR

Usage: CR [INDENT]

Template: INDENT/S

Splits the current line into two at the current cursor position.

- INDENT means that the line should be indented to match the indent of the previous line.

## 1.84 Command Reference: CURSOR

Usage: CURSOR UP | DOWN | LEFT | RIGHT

Template: UP/S,DOWN/S,LEFT/S,RIGHT/S

Moves the cursor up, down, left or right a single line or column.

See Also:

GOTOLINE  
GOTOCOLUMN  
LINE  
COLUMN  
NEXTWORD  
PREVIOUSWORD  
POSITION

## 1.85 Command Reference: CUT

Usage: CUT

Template: ,

Cuts the current marked area. The marked area is copied to the clipboard and then erased. The standard system clipboard is used, so that the data cut can be pasted into other any editor window, as well as any

---



other applications that are using the system clipboard.

See Also:

MARK  
COPY  
PASTE  
ERASE

## 1.86 Command Reference: DEACTIVATE

Usage: DEACTIVATE

Template: ,

Closes down all editor windows and reduces Quill to a simple icon on the Workbench. Simply double-click on this icon or use the `ACTIVATE` command and all windows will re-open into the same state they were before you used the `DEACTIVATE` command.

See Also:

ACTIVATE

## 1.87 Command Reference: DELETECHAR

Usage: DELETECHAR

Template: ,

Deletes the character currently under the cursor (if there is one). If drag-marking is on, it will erase the currently marked area if you were marking.

See Also:

ERASECHAR

## 1.88 Command Reference: DISPLAY

Usage: DISPLAY

Template: ,

Presents the `Display` requester for configuring the Quill display.

## 1.89 Command Reference: ERASE

Usage: ERASE [FORCE]

Template: FORCE/S

Erases the currently marked area.

---

- FORCE supresses the "Are you sure?" requester.

See Also:

MARK  
CUT  
COPY  
PASTE

## 1.90 Command Reference: ERASECHAR

Usage: ERASECHAR

Template: ,

Deletes the character to the immediate left of the cursor. If the cursor is on the first character on a line, then the current line will be joined with the previous line. If drag-marking is on, it will erase the currently marked area if you were marking.

See Also:

DELETECHAR

## 1.91 Command Reference: FIND

Usage: FIND [[TEXT] <text>]

Template: TEXT/F

Tries to find a string of text in the current editor project. If the TEXT parameter is not specified, a requester will appear prompting you to enter the string of text to find. You also have the choice of setting the search as case sensitive, or using AmigaDOS pattern matching in the search. The search will proceed from the current cursor position until the string has been found or the end of the project has been reached. The search can be repeated without having to use the Find requester again by using the FINDNEXT command.

By default the search is case in-sensitive, that is, when looking for the string, upper and lowercase letters will be considered the same. With the case sensitive option on, the string must match exactly.

Optionally, pattern-matching can be used in the finds. Pattern matching is powerful facility that will search for a string that matches a pattern description, rather than doing a simple equivalence check. If you've used AmigaDOS pattern matching before (for example, "dir devs:#?.device", which the command to list all the files in devs: that end with the .device suffix) then you should be able to easily understand Quill's pattern matching searches. It does the same thing, supporting all the pattern matching symbols that AmigaDOS support (such as: #, ?, (), [], ~, \, etc.), searching from the current cursor position for text that matches the pattern you enter. If you are not familiar with AmigaDOS pattern matching, consult your system manuals for a thorough description.

See Also:

---

FINDNEXT  
FINDCHANGE

## 1.92 Command Reference: FINDCHANGE

Usage: FINDCHANGE [ALL] [PROMPT] [FIND <string1>] [CHANGE <string2>]  
Template: ALL/S,PROMPT/S,FIND/K,CHANGE/K

Presents the Find & Replace requester.

- FIND specifies the string to find.
- CHANGE specifies the string to replace.

See Also:  
FIND  
FINDNEXT

## 1.93 Command Reference: FINDNEXT

Usage: FINDNEXT  
Template: ,

Moves the cursor to the next occurrence of the current find string. If there currently is no find string, the Find requester will be displayed.

See Also:  
FIND  
FINDCHANGE

## 1.94 Command Reference: GETATTR

Usage: GETATTR [OBJECT] <object> [[FIELD1] <field1>] [[FIELD2] <field2>]  
[[STEM] <stem>] [[VAR] <var>]  
Template: OBJECT/A,FIELD1,FIELD2,STEM/K,VAR/K

This command allows you to query information from Quill. Attributes are hierarchical, you can fetch one specific attribute into RESULT or a variable given by the VAR argument, or you get a group into a stem variable given by the STEM argument.

- OBJECT can be used to get attributes from any of the following:
  - ~APPLICATION - See fields for the APPLICATION object below.
  - ~PROJECT - See fields for the PROJECT object below.
  - WINDOW - See fields for the WINDOW object below.
  - PROJECTS - See special notes on using the PROJECTS and
  - WINDOWS - WINDOWS objects below.

The PROJECTS and WINDOWS objects are special. They copy the PROJECT and WINDOW attributes of all currently open projects into a stem variable.

---

<stemname>.COUNT will contain the number of projects, <stemname>.<#> will contain the info for each open project, from 0 to n.

- FIELD1 allows you to get a field of information from an object.

With the APPLICATION object, field1 can be used to get the following attributes:

- ~VERSION - Current version of Digital Quill.
- ~SCREEN - The name of the public screen Quill is running on.
- ~ICONIFIED - Whether or not Quill is iconified.

With the PROJECT object, field1 can be used to get the following attributes:

- AREXX - Name of the AREXX port used by the project.
- NUMBER - Number of the project's window.
- FILENAME - Complete filename of the project.
- PATH - Directory (path) the project resides in.
- FILE - Name of project.
- PRIORITY - The task priority that Quill is running at.
- CHANGED - If the current project has been changed since last saved.
- LENGTH - Size of the current project in bytes.
- LINES - Number of lines of the current project.
- POS - Byte offset of the cursor from the start of the current project.
- BOOKMARK - The line of the bookmark in the current project.
- LINE - Line that the cursor is currently on.
- LINELENGTH - Length of the current line.
- MARKINGBLOCK - If a block is currently being marked.
- BLOCKSTARTPOS - Byte position of the start of the block.
- BLOCKSTARTCOL - Column the block starts in.
- BLOCKSTARTLINE - Line the block starts on.
- BLOCKENDPOS - Byte position of the end of the block.
- BLOCKENDCOL - Column the block ends in.
- BLOCKENDLINE - Line the block ends on.
- COLUMN - Column that the cursor is currently in.
- BACKUPTIME - Number of minutes between timed backups .
- SAVETABS - State of the Save Tabs option.
- CREATEICONS - State of the Create Icons option.
- EDITING - See sub-fields for EDITING below.
- ENVIRONMENT - See sub-fields for ENVIRONEMENT below.

With the WINDOW object, field1 can be used to get the following attributes:

- TOP - Position of the top edge of the current project window.
- LEFT - Position of the left edge of the current project window.
- WIDTH - Width of the current project window.
- HEIGHT - Height of the current project window.

- FIELD2 allows you to get a more specific field of information from a field within an object.

From the field1 EDITING in the PROJECT object, the sub-field field2 can be any of:

- ~INSERT - State of the Insert Mode option.
  - ~WORDWRAP - State of the Word Wrap option.
-

- ~AUTOINDENT        - State of the Auto-Indent option.
- ~FALSETABS        - State of the False Tabs option.
- ~TABSIZ        - Current tab size .
- ~RIGHTMARGIN       - Current right margin .

From the field1 ENVIRONMENT in the PROJECT object, the sub-field field2 can be any of:

- ~FREECURSOR       - State of the Free Cursor option.
- ~DRAGMARK       - State of the Drag Marking option.
- ~FENCEPOSTING     - State of the Fence Posting option.
- ~SHOWHEX        - State of the Show Hex option.
- SPEEDBAR        - State of the SpeedBar Visible option.
- LARGEBUTTONS    - State of the Large SpeedButtons option.

#### Examples:

To get the filename of the current project into the ARexx RESULT variable:  
GETATTR PROJECT FILENAME

To get the public screen name of Quill and put it into the pubscreen ARexx variable:

GETATTR APPLICATION SCREEN VAR pubscreen

To get the window information of the current project and place it into the ARexx stem variable window\_attrs:

GETATTR WINDOW STEM window\_attrs

window\_attrs.top will have the window's top edge, window\_attrs.left will have its left edge, etc.

To get the window information on all open projects and place it into the ARexx stem variable windows\_attrs:

GETATTR WINDOW STEM windows\_attrs

In this case, windows\_attrs.0.top will have the top edge of the first project, windows\_attrs.0.top will the top edge of the second project, etc.

To get the project information on all currently open projects and place the info in the ARexx stem variable projects\_info:

GETATTR PROJECTS STEM projects\_info

projects\_info.count will contain the number of open projects.

projects\_info.0.filename will contain the filename of the first project found, projects\_info.1.filename will contain the filename of the second project found, etc.

#### See Also:

SETATTR

## 1.95 Command Reference: GETCHAR

Usage: GETCHAR

Template: ,

Returns the character that the cursor is currently over.

#### See Also:

GETWORD

GETLINE

## 1.96 Command Reference: GETLINE

Usage: GETLINE

Template: ,

Returns the entire contents of the line that the cursor is currently on.

See Also:

GETCHAR

GETWORD

## 1.97 Command Reference: GETWORD

Usage: GETWORD

Template: ,

Returns the word that the cursor is currently over in RESULT, or an error level of 5 in RC if the cursor is not on a word.

See Also:

GETCHAR

GETLINE

## 1.98 Command Reference: GOTOBBookmark

Usage: GOTOBBookmark

Template: ,

Jumps to the bookmark .

See Also:

SETBookmark

SWAPBookmark

## 1.99 Command Reference: GOTOCOLUMN

Usage: GOTOCOLUMN <column>

Template: /N/A

Moves the cursor to a specified column on the current line.

Note that this command does not look at expanded tabs, a tab will always be regarded as one column.

See Also:

---

GOTOLINE  
CURSOR  
LINE  
COLUMN  
NEXTWORD  
PREVIOUSWORD  
POSITION

## 1.100 Command Reference: GOTOLINE

Usage: GOTOLINE [<line>]

Template: /N

Moves the cursor to a specified line, keeping the current column. If no line is specified, the user will be prompted to enter a line number.

See Also:

GOTOCOLUMN  
CURSOR  
LINE  
COLUMN  
NEXTWORD  
PREVIOUSWORD  
POSITION

## 1.101 Command Reference: HELP

Usage: HELP [[COMMAND] <command>] [PROMPT]

Template: COMMAND,PROMPT/S

Provides you with online help information about Quill.

- COMMAND presents you with a list of options available for that command.
- PROMPT activates the graphical hypertext help system.

## 1.102 Command Reference: INSERT

Usage: INSERT [[FILENAME] <filename>]

Template: FILENAME

Opens the specified file and inserts it at the current cursor position in the current editor window. If no filename is passed, the user will be presented with the file requester for selecting a file.

See Also:

OPEN

---

## 1.103 Command Reference: KILL

Usage: KILL LINE | EOL | SOL

Template: LINE/S,EOL/S,SOL/S

Removes text in one of a variety of manners.

- LINE causes the entire line to be removed.
- EOL kills all text from the cursor to the end of line.
- SOL kills all text from the cursor to the start of line.

## 1.104 Command Reference: LEARN

Usage: LEARN FILE <filename> | STOP | PLAY | SAVE

Template: FILE/K,STOP/S,PLAY/S,SAVE/S

Controls Quill's macro learning. With no parameters, causes Quill to start learning a macro. All actions between when you issue this command and LEARN STOP will be remembered, and can be later played-back with LEARN PLAY. The macro you learned can be subsequently saved to disk using LEARN SAVE. The macro on disk can then be used via the RX or Assign Macro commands.

- FILENAME specifies a filename to directly learn the macro to, rather than having to specify LEARN SAVE.

See Also:

RX

## 1.105 Command Reference: LINE

Usage: LINE <displacement>

Template: /N/A

Moves the cursor up or down relative to the current cursor position. Passing a negative value moves the cursor up, a positive value moves the cursor down. The screen will automatically scroll as appropriate.

See Also:

GOTOLINE  
GOTOCOLUMN  
CURSOR  
COLUMN  
NEXTWORD  
PREVIOUSWORD  
POSITION



## 1.106 Command Reference: LOADSETTINGS

Usage: LOADSETTINGS [[FILENAME] <filename>]

Template: FILENAME

Load a Digital Quill settings file. If no file is specified, the user will be presented with the file requester for selecting a file.

The settings consist of all parameters set via the Settings pull-down menu, and in addition: the current window position and size, the current directory, the current file requester pattern, etc.

See Also:

SAVESETTINGS

SAVESETTINGSAS

## 1.107 Command Reference: LOCKGUI

Usage: LOCKGUI

Template: ,

Locks user input to Quill graphic user interface. The pointer will change to the busy pointer and the user will not be able to use the mouse for any operations.

See Also:

UNLOCKGUI

## 1.108 Command Reference: LOWERCASE

Usage: LOWERCASE

Template: CHAR/S,WORD/S,LINE/S

Converts from the cursor position to the end of the word to all lowercase letters, regardless of what case they were before. Or, if an area is being marked , that entire area will be converted to all lowercase.

See Also:

UPPERCASE

## 1.109 Command Reference: MARK

Usage: MARK [OFF]

Template: OFF/S

Puts the cursor into mark mode. As you move the cursor, you will be highlighting text for such operations as cut and erase .

- OFF turns off marking if it was on.

See Also:

CUT  
COPY  
PASTE  
ERASE

## 1.110 Command Reference: MATCHBRACKET

Usage: MATCHBRACKET

Template: ,

If the cursor is currently on a bracket, parenthesis, brace, or inequality sign, the bracket matching it will be found and jumped to. This command fully handles nested brackets.

## 1.111 Command Reference: MEMORY

Usage: MEMORY

Template: ,

» FOR ADVANCED USERS ONLY

Presents the Memory requester for setting memory options in Quill.

## 1.112 Command Reference: MOVEWINDOW

Usage: MOVEWINDOW [LEFTEDGE] <leftedge> [TOPEDGE] <topedge>

Template: LEFTEDGE/N, TOPEDGE/N

Changes the position of the current project window. If the window can't be moved as requested, Quill will do the best job possible to move it as you request.

- LEFTEDGE is the new left edge you want.
- TOPEDGE is the new top edge you want.

See Also:

SIZEWINDOW  
CHANGEWINDOW  
ZOOMWINDOW  
UNZOOMWINDOW  
WINDOWTOFRONT  
WINDOWTOBACK  
SCREENTOFRONT  
SCREENTOBACK

### 1.113 Command Reference: NEW

Usage: NEW  
Template: ,

Opens up a new empty editing window in front of the currently active window. This allows you to edit multiple files at once.

There is currently a limit of 10 editor windows that can be open at any one time.

### 1.114 Command Reference: NEXTWORD

Usage: NEXTWORD  
Template: ,

Moves the cursor to the start of the next word.

See Also:  
GOTOLINE  
GOTOCOLUMN  
CURSOR  
LINE  
COLUMN  
PREVIOUSWORD  
POSITION

### 1.115 Command Reference: OPEN

Usage: OPEN [[FILENAME] <filename>] [FORCE]  
Template: FILENAME, FORCE/S

Opens the specified project into the current editor window, replacing whatever text was already there. If no filename is passed, the user will be presented with the file requester for selecting a file.

- FORCE suppresses the Modified Project requester.

See Also:  
INSERT

### 1.116 Command Reference: PASTE

Usage: PASTE  
Template: ,

Takes the text from the standard system clipboard and inserts it into the current editor project at the current cursor position. The data in the clipboard may have been cut or copied from any Quill editor window or

---

any other application using the standard system clipboard.

See Also:

MARK  
CUT  
COPY  
ERASE

## 1.117 Command Reference: POSITION

Usage: POSITION SOF | EOF | SOL | EOL | SOW | EOW | SOV | EOY | SOS | SOV

Template: SOF/S,EOF/S,SOL/S,EOL/S,SOW/S,EOW/S,SOV/S,EOY/S,SOS/S,EOS/V

Moves the cursor to the new position specified.

- SOF moves it to the beginning of the file.
- EOF moves it to the end of the file.
- SOL moves it to the beginning of the line.
- EOL moves it to the end of the line.
- SOW moves it to the beginning of the word.
- EOW moves it to the end of the word.
- SOV moves it to the beginning of the view, i.e. the far left of the screen. If the cursor is already at the start of the view, the screen will be scrolled left.
- EOY moves it to the end of the view, i.e. the far right of the screen. If the cursor is already at the end of the view, the screen will be scrolled right.
- SOS moves it to the top of the screen. If the cursor is already at the start of the screen, the screen will be scrolled up.
- EOS moves it to the bottom of the screen. If the cursor is already at the bottom of the screen, the screen will be scrolled down.

See Also:

GOTOLINE  
GOTOCOLUMN  
CURSOR  
LINE  
COLUMN  
NEXTWORD  
PREVIOUSWORD

## 1.118 Command Reference: PREVIOUSWORD

Usage: PREVIOUSWORD  
Template: ,

Moves the cursor position to the end of the previous word.

See Also:  
GOTOLINE  
GOTOCOLUMN  
CURSOR  
LINE  
COLUMN  
NEXTWORD  
POSITION

### 1.119 Command Reference: PRINT

Usage: PRINT [FORCE]  
Template: FORCE/S

Sends the file in the current editor window to the print "as-is". Be sure that you have your printer ready and setup as you want it to be via the Workbench Printer preferences.

If a block is current being marked, then only the marked area will be printed.

- FORCE suppresses the confirmation requester.

### 1.120 Command Reference: QUIT

Usage: QUIT [FORCE]  
Template: FORCE/S

Exits Digital Quill entirely. If the project in any of the editor windows have not been saved since they were last modified, you will be given a chance to save them before quitting.

- FORCE suppresses the Modified Project requester.

### 1.121 Command Reference: REDO

Usage: REDO  
Template: ,

If you have undone an edit and decide that you did not in fact want to undo it, you can redo that edit with this function.

See Also:  
UNDO

---

## 1.122 Command Reference: REFORMAT

Usage: REFORMAT

Template: ,

Formats the current paragraph into word-wrapped lines that fit within the right margin setting. Quill has no hard definition of a paragraph, only of a line, so it will attempt to detect the end of the paragraph by looking for the following: two consecutive newlines, a newline followed by a space or tab, or the end of text. This should be kept in mind when using this command or you might end up with unexpected results. REFORMAT generally only works well with words written in simple paragraph form with no extra spacing and indenting for formatting.

## 1.123 Command Reference: REMAKESCREEN

Usage: REMAKESCREEN

Template: ,

Does a complete remake of the Quill display. All windows and, if open on a custom screen, the Quill screen, will be closed down and then re-opened and refreshed.

## 1.124 Command Reference: REQUESTFILE

Usage: REQUESTFILE [TITLE <title>] [PATH <path>] [FILE <file>] [PATH <path>]

Template: TITLE/K,PATH/K,FILE/K,PATTERN/K

Brings up a file requester. You may specify default path and file names.

The full pathname selected will be returned in RESULT, or a warning level of 5 in RC if the user cancels the requester.

- TITLE is what is to appear in the title bar of the file requester.
- ~PATH is the default path to appear in the drawer gadget of the file requester. If you don't provide a path then the current directory will be used.
- FILE is the default filename to appear in the file gadget of the file requester.
- PATTERN is the default pattern for matching files to appear in the pattern gadget. If you don't provide a pattern the #? (match all files) will be used. Note that the file requester will always filter out files ending in ".info".

See Also:

REQUESTITEM  
REQUESTSTRING  
REQUESTNUMBER  
REQUESTRESPONSE

---

REQUESTNOTIFY

## 1.125 Command Reference: REQUESTITEM

Usage: REQUESTFILE [PROMPT] <prompt text> STEM <stem>  
Template: PROMPT,STEM/K/A

Brings up a requester allowing the user to select an item from a list. The list is provide in an ARexx stem variable, with each item in the stem <stemname>.# where # is from 0 to n and <stemname>.COUNT is the total number of items.

The index of the item selected is returned in RESULT, or a warning level of 5 in RC if the user cancels the requester.

See Also:

REQUESTFILE  
REQUESTSTRING  
REQUESTNUMBER  
REQUESTRESPONSE  
REQUESTNOTIFY

## 1.126 Command Reference: REQUESTNOTIFY

Usage: REQUESTNOTIFY [PROMPT] <prompt text>  
Template: PROMPT

Brings up a requester to notify the user of some circumstance. The requester will have a "Continue" gadget for the user to use when he/she is ready to continue.

See Also:

REQUESTFILE  
REQUESTITEM  
REQUESTSTRING  
REQUESTNUMBER  
REQUESTRESPONSE

## 1.127 Command Reference: REQUESTNUMBER

Usage: REQUESTNUMBER [PROMPT] <prompt text> [DEFAULT <default value>]  
Template: PROMPT,DEFAULT/K

This command brings up a number entry requester. You may specify a prompt and a default number to be placed in the string editing gadget. Currently only integer values are accepted, floating point values (i.e. values witha decimal point) are not allowed.

The user may enter a value in hexadecimal form by preceeding it with "0x", eg. "0x20" for 20 base16, 32 base10.

---

The number entered will be returned in RESULT, or a warning level of 5 in RC if the user cancels the requester.

See Also:

- REQUESTFILE
- REQUESTITEM
- REQUESTSTRING
- REQUESTRESPONSE
- REQUESTNOTIFY

## 1.128 Command Reference: REQUESTRESPONSE

Usage: REQUESTRESPONSE [PROMPT] <prompt text>

Template: PROMPT

This command brings up a query requester. You can specify a prompt string for which the user may respond to with either "OK" or "Cancel".

A warning level of 5 will be returned in RC if the user selects cancel.

See Also:

- REQUESTFILE
- REQUESTITEM
- REQUESTNOTIFY
- REQUESTNUMBER
- REQUESTSTRING

## 1.129 Command Reference: REQUESTSTRING

Usage: REQUESTSTRING [PROMPT] <prompt text> [DEFAULT <default value>]

Template: "PROMPT,DEFAULT/K

This command brings up a string entry requester. You may specify a prompt and a default string to be placed in the string editing gadget.

The string entered will be returned in RESULT, or a warning level of 5 in RC if the user cancels the requester.

See Also:

- REQUESTFILE
- REQUESTNUMBER
- REQUESTRESPONSE
- REQUESTNOTIFY

## 1.130 Command Reference: REVEAL, SELECT

Usage: REVEAL [NAME] <name> | [NUMBER] <number> | NEXT | PREV | TOP | BOTTOM | ↔  
PROMPT



```
SELECT [NAME] <name> | [NUMBER] <number> | NEXT | PREV | TOP | BOTTOM | ↵  
PROMPT
```

Template: NAME,NUMBER/N,NEXT/S,PREV=PREVIOUS/S,TOP/S,BOTTOM/S,PROMPT/S

Selects a project to be the currently active one.

- NAME is the name of the project you want to make active. If this is not a project Quill already has open then Quill will try to load it from disk.
- NUMBER specifies a project number of the project you want to make active.
- NEXT selects the next project after the current one to be active, or the first project if the last one is currently active.
- PREV selects the next project before the current one to be active, or the last project if the first one is currently active.
- TOP selects the last project to be active.
- BOTTOM selects the first project to be active.
- PROMPT offers the user a list of all projects for her/him to select from.

## 1.131 Command Reference: RX

Usage: RX [COMMAND] <command>

Template: COMMAND/F

Starts an ARexx macro. All macros are executed asynchronously and will inherit Quill's output stream.

- COMMAND is the name of the macro you want to execute. If not given, the user will be prompted with the file requester.

See Also:

RXS

## 1.132 Command Reference: RXS

Usage: RXS [STRING] <command>

Template: STRING/F

Sends a string command to ARexx. The command will be given an output console if needed automatically.

- STRING is the command you want to send to ARexx.

See Also:

RX

---

### 1.133 Command Reference: SAVE

Usage: SAVE [FORCE]

Template: FORCE/S

Saves the file in the current editor window to disk, using the current project name. If no name has been specified yet, you'll be able to select a name using the file requester.

If a block is current being marked, then only the marked area will be saved.

- FORCE suppresses any confirmation requesters.

See Also:

SAVEAS

### 1.134 Command Reference: SAVEAS

Usage: SAVEAS [[FILENAME] <filename>] [FORCE]

Template: FILENAME, FORCE/S

Saves the current project to the specified filename. If no filename is passed, the user will be presented with the file requester for selecting a file.

If a block is current being marked, then only the marked area will be saved.

- FILENAME specifies the new name for the project. If no name is given then the file requester will appear so that a name can be given.
- FORCE suppresses any confirmation requesters.

See Also:

SAVE

### 1.135 Command Reference: SAVESETTINGS

Usage: SAVESETTINGS

Template: ,

Saves the Digital Quill settings to disk.

The settings consist of all parameters set via the Settings pull-down menu, and in addition: the current window position and size, the current directory, the current file requester pattern, etc.

See Also:

LOADSETTINGS

SAVESETTINGSAS

---

## 1.136 Command Reference: SAVESETTINGSAS

Usage: SAVESETTINGSAS [[FILENAME] <filename>]

Template: FILENAME

Save the Digital Quill settings to a file. If no file is specified, the user will be presented with the file requester for selecting a file.

The settings consist of all parameters set via the Settings pull-down menu, and in addition: the current window position and size, the current directory, the current file requester pattern, etc.

See Also:

LOADSETTINGS

SAVESETTINGS

## 1.137 Command Reference: SCREENTOBACK

Usage: SCREENTOBACK

Template: ,

Puts the Quill screen behind all other screens.

See Also:

MOVEWINDOW

SIZEWINDOW

CHANGEWINDOW

ZOOMWINDOW

UNZOOMWINDOW

WINDOWTOFRONT

WINDOWTOBACK

SCREENTOFRONT

## 1.138 Command Reference: SCREENTOFRONT

Usage: SCREENTOFRONT

Template: ,

Makes the Quill screen to the frontmost screen

See Also:

MOVEWINDOW

SIZEWINDOW

CHANGEWINDOW

ZOOMWINDOW

UNZOOMWINDOW

WINDOWTOFRONT

WINDOWTOBACK

SCREENTOBACK

## 1.139 Command Reference: SETATTR

Usage: SETATTR [OBJECT] <object> [[FIELD1] <field1>] [[FIELD2] <field2>]  
 [[VALUE] <value>] [[VAR] <var>]

Template: OBJECT/A, FIELD1, FIELD2, VALUE/K, VAR/K

Allows you to set an attribute within Quill. Attributes are hierarchical, and currently you can only change one attribute at a time, thus you must specify the full hierarchy of the attribute.

- OBJECT can be used to set attributes from any of the following:
  - ~PROJECT - See fields for the PROJECT object below.
- FIELD1 allows you to set a field of information from an object.

With the PROJECT object, field1 can be used to get the following attributes:

- SAVETABS - (Boolean)
  - CREATEICONS - (Boolean)
  - BACKUPTIME - (Integer)
  - FILENAME - (String)
  - PATH - (String)
  - FILE - (String)
  - EDITING - See sub-fields for EDITING below.
  - ENVIRONMENT - See sub-fields for ENVIRONEMENT below.
- FIELD2 allows you to get a more specific field of information from a field within an object.

From the field1 EDITING in the PROJECT object, the sub-field field2 can be any of:

- ~INSERT - (Boolean)
- ~WORDWRAP - (Boolean)
- ~AUTOINDENT - (Boolean)
- ~FALSETABS - (Boolean)
- ~TABSIZ - (Integer)
- ~RIGHTMARGIN - (Integer)

From the field1 ENVIRONMENT in the PROJECT object, the sub-field field2 can be any of:

- ~FREECURSOR - (Boolean)
- ~DRAGMARK - (Boolean)
- ~FENCEPOSTING - (Boolean)
- ~SHOWHEX - (Boolean)
- ~SPEEDBAR - (Boolean)
- LARGEBUTTONS - (Boolean)

Boolean attributes you can set to either ON or OFF, or you can specify TOGGLE to toggle their current state.

Integer attributes you can pass a numeric value for, or if you don't pass a value a numeric requester will appear for the user to type a value in that will become the new value for the attribute.

Examples:

To make sure insert mode is on:

```
SETATTR PROJECT EDITING INSERT ON
```

To toggle fence-posting:

```
SETATTR PROJECT ENVIRONMENT FENCEPOSTING TOGGLE
```

To change the tab size to 8:

```
SETATTR PROJECT ENVIRONMENT TABSIZE 8
```

To allow the user to enter a new right margin value:

```
SETATTR PROJECT ENVIRONMENT RIGHTMARGIN
```

See Also:

```
GETATTR
```

## 1.140 Command Reference: SETBOOKMARK

Usage: SETBOOKMARK

Template: ,

Places a bookmark at the current cursor position. This means that the position of the bookmark will be remembered so that it can quickly be jumped to later with either the GOTOBOOKMARK or SWAPBOOKMARK commands.

See Also:

```
GOTOBOOKMARK
```

```
SWAPBOOKMARK
```

## 1.141 Command Reference: SETSTATUSBAR

Usage: SETSTATUSBAR [TEXT] <text>

Template: TEXT/F

Sets a status message. Status messages in Quill appear in the title bar. A status message is only temporary and will be cleared by any subsequent action the user makes.

## 1.142 Command Reference: SIZEWINDOW

Usage: SIZEWINDOW [WIDTH] <width> [HEIGHT] <height>

Template: WIDTH/N,HEIGHT/N

Changes the size of the current project window to the given width and height.

- WIDTH specifies the new width for the window.
- HEIGHT specifies the new height for the window.

See Also:

```
MOVEWINDOW
```

---

```
CHANGEWINDOW  
ZOOMWINDOW  
UNZOOMWINDOW  
WINDOWTOFRONT  
WINDOWTOBACK  
SCREENTOFRONT  
SCREENTOBACK
```

### 1.143 Command Reference: SWAPBOOKMARK

Usage: SWAPBOOKMARK

Template: ,

Jumps to the bookmark , making the old cursor position the new bookmark. This can be used to quickly toggle back and forth between two spots in a project.

See Also:

```
SETBOOKMARK  
GOTOBOOKMARK
```

### 1.144 Command Reference: SYSTEM

Usage: SYSTEM [ASYNC] [PUBSCREEN] [HOSTPORT] [COMMAND] <command>

Template: ASYNC/S,PUBSCREEN/S,HOSTPORT/S,COMMAND/F

Executes an AmigaDOS command.

- ASYNC specifies that the command be run asynchronously and that control be immediately returned to Quill.
- PUBSCREEN causes the name of the screen Quill is running on to be included in the command string with the PUBSCREEN keyword. Use this to get other programmes to open on the same screen as Quill.
- HOSTPORT causes the name of the ARexx port Quill is using to be included in the command string with the HOSTPORT keyword. Use this to pass Quill's ARexx port to another programme for that programme to communicate with Quill.
- COMMAND is the AmigaDOS command to run. Standard redirection with < and > is supported.

### 1.145 Command Reference: TEXT

Usage: TEXT [TEXT] <text>

Template: TEXT/F

Inserts text into the current project window at the current cursor position.

---

## 1.146 Command Reference: TEXTPEN

Usage: TEXTPEN <pen number>

Template: /N

Changes the colour of the text pen in all project windows to the given pen number.

See Also:

BACKGROUNDPEN

## 1.147 Command Reference: UNDO

Usage: UNDO

Template: ,

Reverses the last edit made. Anything that changes the current project, entering text, deleting text, doing a find and replace, pasting from the clipboard, erasing a block, etc. can all be undone. You can undo for numerous levels, the limits of which are defined in the Memory requester.

See Also:

REDO

## 1.148 Command Reference: UNLOCKGUI

Usage: UNLOCKGUI

Template: ,

Unlocks Quill's graphic user interface if it had been previously locked.

See Also:

LOCKGUI

## 1.149 Command Reference: UNZOOMWINDOW

Usage: UNZOOMWINDOW

Template: ,

If the current project window is zoomed this command will unzoom it, equivalent to clicking on the zoom gadget in the project window while the window is in its zoomed state.

See Also:

MOVEWINDOW

SIZEWINDOW

CHANGEWINDOW

ZOOMWINDOW

WINDOWTOFRONT

---

WINDOWTOBACK  
SCREENTOFRONT  
SCREENTOBACK

## 1.150 Command Reference: UPPERCASE

Usage: UPPERCASE  
Template: CHAR/S,WORD/S,LINE/S

Converts from the cursor position to the end of the word to all uppercase letters, regardless of what case they were before. Or, if an area is being marked , that entire area will be converted to all uppercase.

See Also:  
LOWERCASE

## 1.151 Command Reference: WINDOWTOBACK

Usage: WINDOWTOBACK  
Template: ,

Moves the current project window behind all others.

See Also:  
MOVEWINDOW  
SIZEWINDOW  
CHANGEWINDOW  
ZOOMWINDOW  
UNZOOMWINDOW  
WINDOWTOFRONT  
SCREENTOFRONT  
SCREENTOBACK

## 1.152 Command Reference: WINDOWTOFRONT

Usage: WINDOWTOFRONT  
Template: ,

Brings the current project window to the front.

See Also:  
MOVEWINDOW  
SIZEWINDOW  
CHANGEWINDOW  
ZOOMWINDOW  
UNZOOMWINDOW  
WINDOWTOBACK  
SCREENTOFRONT  
SCREENTOBACK

---



## 1.153 Command Reference: ZOOMWINDOW

Usage: ZOOMWINDOW

Template: ,

Zooms the current project window to the alternate zoom size. Equivalent to clicking on the zoom gadget in the project window while the window is in its unzoomed state.

See Also:

- MOVEWINDOW
- SIZEWINDOW
- CHANGEWINDOW
- UNZOOMWINDOW
- WINDOWTOFRONT
- WINDOWTOBACK
- SCREENTOFRONT
- SCREENTOBACK