The **pcANYWHERE Editor Help Index**

Keyboard
The pcANYWHERE Editor Keys
Editor Functions: Alphabetical List
Editor Functions: Functional Groups

Commands
File
Edit
Search
Options
Window

Procedures
Using the pcANYWHERE Editor
Customizing the pcANYWHERE Editor
Customizing Editor Keystrokes
Comparing Two Files
Finding Which Key Performs an Editor Function
Printing All of or Part of a Document
Searching for Files That Contain Specific Text
Searching for Specified Text in an Open Document
Searching with Regular Expressions
Using Editor Macros
Using Multiple Edit Windows

The pcANYWHERE Editor Menu Commands

File menu
 New
 Open
 Close
 Save
 Save As
 Save All
 Insert
 Revert
 Write Block
 Printer Setup
 Page Setup
 Print
 Compare
 Exit
 (list of files)

Edit menu
 Undo
 Cut
 Copy
 Paste
 Delete
 Select All
 Time/Date
 Wrap Paragraph
 Word Wrap
 Record Macro
 Playback Macro

Search menu
 Find
 Find Again
 Replace
 Find Files Containing
 List Found Files
 Goto Line

Options menu
 Document Preferences
 Editor Preferences
 Key Assignments

Tool menu

Reference

Window menu

New Window

Cascade

Tile

Arrange Icons

(list of windows)

See Also

Using the pcANYWHERE Editor
Editor Functions: Alphabetical List
Editor Functions: Functional Groups

The File menu

Commands in this menu let you create, edit, and print script files as well as providing access to other file-related functions.

New	Create a new script file.
Open	Open an existing script file.
Close	Close the active editing window.
Save	Save file in the active editing window.
Save As	Save file in active editing window using different name.
Save All	Save all open files.
Insert	Insert contents of a file at cursor position.
Revert	Undo all changes since last save.
Write Block	Write selected block to a file.
Printer Setup	Select printer and printer options.
Page Setup	Set headers, footers, margins, etc.
Print	Print current script file.
Compare	Compare contents of two files.
Exit	Close the pcANYWHERE Editor windows and exit.
(list of files)	Open or activate a recently used file.

New command (File menu)

Function name: **new_file**
Default keys: **Alt+F,N**

This command opens a new untitled editing window. Use the Save or Save As… command to assign a name to the file.

Open command (File menu)

    Function name: **open_file**
       Default keys: **F3**

This command displays an <u>Open File dialog box</u>. Use it to select a file to edit.

Open File dialog box (Standard Browse)

Use this dialog box to open an existing script file for editing.

File Name: Type or select the filename you want to open. This box lists files with the extension selected in the List Files of Type box.

List Files of Type: Select the type of file you want to open

   Script Files (*.SCR):   Lists all files in the current directory that were saved with an *.SCR extension.

Drives: Select the drive on which the file you want to open is stored.

Directories: Select the directory in which the file you want to open is stored.

Close command (File menu)

Function name: **close_window**

Default keys: **Alt+F,C**

This command closes the active editing window.

If you have modified the file and not yet saved your changes, (*and* it is not open in any other editing window), then the pcANYWHERE Editor prompts you to save your changes before it closes the file.

Save command (File menu)

Function name: **save_file**
Default keys: **F2**

This command immediately saves the file in the active window.

If the file does not yet have a name, a <u>Save As... dialog box</u> appears and prompts you for a path and filename.

Save As dialog box (Standard Browse)

Use this dialog box to specify a path and filename under which to save your script file.

File Name: Type or select the filename under which you want to save your script. This box lists files with the extension selected in the List Files of Type box.

List Files of Type: Select the type of file to save. If you are saving a script file, be sure to specify an extension of .SCR.

   Script Files (*.SCR):   Lists all files in the current directory that were saved with an *.SCR extension.

Drives: Select the drive on which you want to store your script file.

Directories: Select the directory in which you want to store your script file.

Save As command (File menu)

Function name: (none)
Default keys: **Alt+F,A**

This command displays a <u>Save As... dialog box</u>. Use it to select a new path and filename for the file displayed in the active editing window.

You can use this command to save the file with a new name; for example, make a backup of your file by saving it under a different name.

Save All command (File menu)

    Function name: **save_all**
      Default keys: **Alt+F,L**

This command saves all open files.

If any windows are untitled, a <u>Save As... dialog box</u> appears. Use this to specify a path and filename for any open, untitled files.

Insert command (File menu)

Function name: (none)
Default keys: **Alt+F,I**

Use this command to insert the contents of one file into another. The file is inserted at the current cursor location.

An <u>Insert File dialog box</u> appears and prompts you for the name of the file to insert.

Insert File dialog box (Standard Browse)

Use this dialog box to specify the file to be inserted at the current cursor position.

File Name: Type or select the name of the file you want to insert into the active editing window.

List Files of Type: Select the type of files to display. Script source files have an extension of .SCR.

  Script Files (*.SCR): Lists all script files in the current directory.

Drives: Select the drive on which the desired file is stored.

Directories: Select the directory in which the desired file is stored.

Revert command (File menu)

Function name: **revert**
Default keys: **Alt+F,V**

This command undoes all editing changes since the file was last saved. Before reading the file from disk, the pcANYWHERE Editor displays a confirmation box, allowing you to cancel the operation.

Write Block command (File menu)

Function name: (none)
Default keys: **Alt+F,W**

This command copies the currently selected text to a file. It is enabled only when text is selected.

When the <u>Write Block... dialog box</u> appears, specify a path and filename under which to save the text.

Write Block dialog box (Standard Browse)

Use this dialog box to specify the path and filename under which to store the selected text.

File Name: Type or select the name of the file under which you want to save the selected text.

List Files of Type: Select the type of files to display. Script source files have an extension of .SCR.

   Script Files (*.SCR):   Lists all script files in the current directory.

Drives: Select the drive on which to store the new file.

Directories: Select the directory in which to store the new file.

Printer Setup command (File menu)

Function name: (none)
Default keys: **Alt+F,U**

This command displays a <u>Printer Setup dialog box</u> which lets you select a default printer and specify print options.

You can also use the Page Setup… commands to select page header, footer, and margin settings before printing a document.

Use the File Print command to print the document.

Printer Setup dialog box

Use this dialog box to select from the printers you have installed under Windows and to select the various options provided by a printer.

NOTES: Some system-level printer configuration commands, such as installing new printers and selecting printer ports, are available through the Windows Control Panel.

This dialog box sets the selected printer as the Windows default printer; other applications use this printer unless you change it back (usually through the application's own Printer Setup command).

Printer: Use ⬇ and
⬆ to move through the items in the list box. When the desired printer is selected, click OK or press Enter to confirm your choice.

Setup...: Click this button to display a Printer Driver dialog box which displays setup options for the selected printer. For example, on some printers you can specify landscape printing (printing sideways on the page), various printer resolutions and other options. *The Printer Driver dialog box, and any help provided in it, is part of the printer driver, and not a part of pcANYWHERE for Windows.*

Page Setup command (File menu)

    Function name: (none)
      Default keys: **Alt+F,G**

This command displays a <u>Page Setup dialog box</u> which lets you select a font, margins, and header and footer text to be used when files are printed.

You can also use the <u>Printer Setup... command</u> to select a printer and specify print options before printing a document.

Use the File Print command to print the document.

Page Setup dialog box

Use this dialog box to set page formatting options. These settings take effect the next time you use the Print command.

Header and **Footer**:

Enter one line of text to be centered at the top (a header) and/or bottom (a footer) of each page. Or, click the prompt button (or press Alt+↓) to choose from a list of previously-used settings.

You can embed the following special-purpose codes in the text:

%f   Full path and filename of active document
%d   Current date and time
%p   Current page number

To omit a header or footer, leave its text box empty.

Margins:

Set the print margins. The units of measure are the standard units used in your country (inches or centimeters), as set in the Windows Control Panel.

Font:

Click this to display a Printer Font dialog box which lets you choose a typeface and size for the printed output. The entire document, including headers and footers, uses the selected font. The specified settings are used for the current session and all subsequent sessions, until changed.

See Also

Printing a Document
Printing a Block of Text
The File Printer Setup command
The File Print command
Printer Font dialog box

Printer Font dialog box

Use this dialog box to choose a typeface and size for your printed output. The Printer Font dialog box appears when you click the Font... button from the <u>Page Setup dialog box</u>.

Font:

Choose from the list of available fonts. Only fonts available for the current printer are shown (use the Printer Setup command on the File menu to choose a different printer and the Windows Control Panel to install additional fonts).

Size:

Select from the list of available sizes for the currently-selected font.

See Also

<u>Printing a Document</u>
<u>The Page Setup command</u>
<u>The File Printer Setup command</u>
<u>The File Print command</u>

Print command (File menu)

   Function name: **print**
     Default keys: **Alt+P** or **Alt+F,P**

If text is selected in the active window, this command sends the selected text to the default printer.

If no text is selected, the entire file is printed.

Use the Page Setup… and Printer Setup… commands to select a printer and specify print options before using this command.

Compare command (File menu)

Function name: **compare**
Default keys: **Alt+F,E**

This command displays a Compare dialog box which lets you select two files to compare.

If either file is currently open, the pcANYWHERE Editor uses the version which is in memory, rather than reading the file from the disk. If you want to pinpoint recent changes, first save the open file under a different name (use the Save As... command), then compare it to the file on disk.

The pcANYWHERE Editor performs the comparison on a line-by-line basis. Both files are displayed and the first mismatch is highlighted.

Compare dialog box

Use this dialog box to choose two files to compare, their starting line numbers, and how to display the ouput. The Compare dialog box appears when you choose Compare from the File menu.

File 1 and **File 2**:

   Enter the names of the files to compare.

   Or, click the prompt button (or press Alt+↓) to choose from a list of previously-used files.

Or, click Browse... to locate the desired files.

Line:

   To compare all lines in both files, use the default setting of 1. Otherwise, specify a starting line number for each file.

Display:

   Select an option button:

   ⋆ Horizontal: The files are displayed one above the other.

   ⋆ Vertical: The files are displayed side-by-side.

Browse:

   Click this button to display a <u>Compare dialog box</u> which helps you locate a desired file. The selected file is displayed in the File text box which is active.

See Also

<u>Comparing Two Files</u>
<u>The File Compare command</u>

Exit command (File menu)

Function name: **exit**
Default keys: **Alt+F4** or **Alt+F,X**

This command closes all pcANYWHERE Editor windows and exits.

The pcANYWHERE Editor prompts you to save changes you have made to any open file.

List of files (1 to 4) (File menu)

Function name: (none)
Default keys: **Alt+F,1...4**

This command lets you quickly open a recently edited file. The last four files edited appear at the bottom of the File menu. To open one of these files, press Alt+F then 1, 2, 3, or 4.

If the selected file is currently in memory, its window is brought to the front. You can also activate any open or <u>minimized</u> window by selecting it from the Window menu.

The Edit menu

Commands in this menu provide text editing tools, access to the <u>clipboard</u>, and keyboard macro control.

<u>Undo</u>	Undo the most recent text-editing change.
<u>Cut</u>	Copy the current selection to the <u>clipboard</u> and then delete that text.
<u>Copy</u>	Copy selected text to the clipboard.
<u>Paste</u>	Paste clipboard contents to the cursor position.
<u>Delete</u>	Delete current selection.
<u>Select All</u>	Select all text in the active file.
<u>Time/Date</u>	Insert the current date and time at the cursor.
<u>Wrap Paragraph</u>	Reformat the <u>paragraph</u> at the cursor, wrapping at the current margin settings.
<u>Word Wrap</u>	Toggle automatic paragraph wrapping on or off.
<u>Record Macro</u>	(toggles to Stop Recording Macro) Record subsequent keystrokes for automatic recall.
<u>Playback Macro</u>	Replay the keystrokes saved during the most recent Record Macro command.

By default, the pcANYWHERE Editor uses standard keystrokes for text selection and clipboard operations. However, you can assign different (perhaps more convenient) keystrokes to these functions. See <u>Customizing Editor Keystrokes</u> for details.

Undo command (Edit menu)

Function name: **undo**
Default keys: **Alt+BkSp** or **Gray ***

Reverses the effects of a previous editing operation. Each Undo reverses the effects of one editor function or keystroke. The number of operations that can be undone is specified in the Editor Preferences dialog box (from the Options Menu).

Only changes made since a file was last saved can be undone.

Cut command (Edit menu)

Function name: **cut**
Default keys: **Shift+Del** or **Gray -**

Deletes the selected text from the active window and moves it to the <u>clipboard</u>.

If no text is selected, and if the Cut/Copy Current Line if No Text is Selected check box in the <u>Editor Preferences dialog box</u> is checked, the current line is cut.

Use the Copy command (on the Edit menu) if you want to copy the selection to the clipboard without deleting it. Use Delete if you want to discard the line without changing the contents of the clipboard.

Copy command (Edit menu)

Function name: **copy**
Default keys: **Ctrl+Insert** or **Gray +**

Copies the selected text from the active window into the <u>clipboard</u>.

If no text is selected, and if the Cut/Copy Current Line if No Text is Selected check box in the <u>Editor Preferences dialog box</u> is checked, the current line is copied.

Paste command (Edit menu)

Function name: **paste**

Default key: **Shift+Insert**

Inserts the contents of the <u>clipboard</u> at the current cursor position.

Delete command (Edit menu)

Function name: **delete**

Default key: **Del**

Deletes the currently selected text. If no text is selected, deletes the character to the right of the cursor.

Use the Cut command (on the Edit menu) if you want to copy the selection to the <u>clipboard</u> and simultaneously delete it from your file.

Select All command (Edit menu)

Function name: **select_all**
Default key: **Alt+E,A**

Selects all text in the active window. The cursor is moved to the end of the file.

This is the same as pressing Ctrl+Home and then Shift+Ctrl+End.

Time/Date command (Edit menu)

Function name: **stamp**

Default key: **Shift+F2**

Inserts the date and time at the cursor location.

Word Wrap command (Edit menu)

Function name: **toggle_wordwrap**
Default key: **Ctrl+W**

Enables or disables word wrap in the active window. Word wrap is used when writing narrative paragraphs; turn this feature off when writing program source code or creating formatted tables.

When word wrap is enabled, typing beyond the right margin causes the current word to be moved down to the next line. The word WRAP appears on the status line.

The right margin setting can be changed from the Document Preferences dialog box dialog box on the Options menu.

Use the Wrap Paragraph command on the Edit menu to reformat a paragraph and wrap all lines at the margin.

NOTE: The pcANYWHERE Editor displays text using a monospaced font, so word wrapping is based on a uniform character width. If you use a proportional-spaced font when you print the document, the right edge appears more ragged in the printout than it does on the screen.

Wrap Paragraph command (Edit menu)

Function name: **wrap_para**
Default key: **F12**

Reformats the current <u>paragraph</u> (or the previous paragraph, if the cursor is between paragraphs) so that all the <u>words</u> fit within the margins. The cursor moves to the end of the paragraph.

This command does nothing if word wrap is turned off. Use the Word Wrap command (on the Edit menu) to enable this feature.

Word wrap is used when writing narrative paragraphs. Don't use this feature when writing program source code or creating formatted tables.

Change the margin settings from the <u>Document Preferences dialog box</u>. To display this dialog box, choose Document Preferences from the Options menu.

Record Macro command (Edit menu)

    Function name: **record_macro**
        Default key: **F7**

Begins or ends a keyboard macro definition. While recording a macro, REC appears on the <u>status line,</u> and the menu command toggles to Stop Recording Macro. All keystrokes and functions are recorded until you choose Stop Recording Macro from the Edit menu.

You can use the Playback Macro command to replay the recorded sequence.

See <u>Using Keyboard Macros</u> for related details.

Play Back Macro command (Edit menu)

    Function name: **play_macro**
       Default key: **F8**

Replays the sequence of keystrokes and functions saved by the Record Macro command.

See <u>Using Keyboard Macros</u> for related details.

The Search Menu

Commands in this menu let you search for text in the current file or in a set of files on disk. They also provide a quick way to move to a line given its line number.

Find	Search for text or a regular expression.
Find Again	Continue search started with the Find command.
Replace	Search for and replace text or regular expression.
Find Files Containing	Generate a list of files which contain the specified text or regular expression.
List Found Files	See a list of files found; choose one to edit.
Goto Line	Move the cursor to a line specified by its line number.

Find command (Search menu)

Function name: **find**
Default key: **Ctrl+S**

This command searches the active document for specified text or a <u>regular expression</u>. It displays the <u>Find dialog box</u> which lets you specify search criteria such as what text to find and in which direction to perform the search.

HINT: If you select text before choosing this command, the selected text appears as the default search string in the dialog box.

The search commences at the current cursor location of the active document. If the search is successful, the matcing text is highlighted. If it fails, the <u>status line</u> displays the message Pattern not found.

NOTE: The pcANYWHERE Editor cannot find text that spans two lines.

Find dialog box

This dialog box appears when you choose the Search Find command. Use it to specify the text you want to find and in which direction to perform the search.

Pattern:

Enter the text you want to find, or click the prompt button (or press Alt+↓) to choose from a list of previously-used search strings.

If you want to use regular expression wildcards in your search string, check the Regular Expression check box.

NOTES: By default, the pcANYWHERE Editor uses any currently selected text as the search string.

The pcANYWHERE Editor cannot find text that spans two lines.

Match Upper/Lowercase:

When this check box is checked, the pcANYWHERE Editor checks the case of the text in the Pattern combination box and returns only exact matches.

When cleared, the editor considers text a match even when the case is not the same. For example, goto matches GOTO and Goto.

Regular Expression:

Check this if you want to enable the powerful and flexible regular expression pattern matching. Leave this unchecked (for speed) if you simply want to locate a string.

Next:

Click this or press Enter to confirm your settings and begin searching forward from the cursor location. If a match is found, the found text is highlighted.

Previous:

Click this to confirm your settings and begin searching backward from the cursor location.

See Also

Searching with Regular Expressions
Finding Files that Contain Specified Text
Search Functions

Find Again command (Search menu)

Function name: **find_again**
Default key: **Ctrl+A**

This command continues the search begun by the Search Find command. The search resumes from the current cursor position in the active document and proceeds in the direction previously specified.

If the search is successful, the matching text is highlighted. If it fails, the status line displays the message Pattern not found.

Replace command (Search menu)

Function name: **replace**
Default key: **Ctrl+R**

This command displays a <u>Replace dialog box</u> which lets you find and replace occurrences of text with different text. You can also perform a search and replace on a <u>regular expression</u>.

HINT: If you select some text before using this command, that text appears as the default search string in the dialog box.

The search commences at the current cursor location of the active document. If the search succeeds (and if you checked the Confirm Changes check box), the matching text is highlighted and you can choose to replace or skip each occurrence. The status line reports how many changes were made.

If the search fails, the <u>status line</u> displays the message Pattern not found.

NOTE: The pcANYWHERE Editor cannot find text that spans two lines.

Replace dialog box

Use this dialog box to specify the text to find and the text which replaces it. The search and replace operation always proceeds forward from the current cursor location in the active window.

Search For:

Enter the text you want to find or click the prompt button (or press Alt+⬇) to choose from a list of previously-used search strings.

If you want to use regular expression wildcards in your search string, check the Regular Expression check box.

NOTES: By default, the pcANYWHERE Editor uses any currently selected text as the search string.

The pcANYWHERE Editor cannot find text that spans two lines.

Replace With:

Enter the text you want to use as the replacement string or click the prompt button (or press Alt+⬇) to choose from a list of previously-used replacement strings. You cannot use a text wildcard in this text box.

If you leave this field blank, each occurrence of the search string is deleted (that is, replaced with nothing).

Match Upper/Lowercase:

When this check box is checked, the pcANYWHERE Editor checks the case of the text in the Pattern combination box and returns only exact matches.

When cleared, the editor considers text a match even when the case is not the same. For example, goto matches GOTO and Goto.

Regular Expression:

Check this if you want to enable the powerful and flexible regular expression pattern matching. Leave this unchecked (for speed) if you simply want to locate a string.

Confirm Changes:

Check this if you want to be prompted before a change is performed. When unchecked, the editor replaces all occurrences of the Search For text, starting at the current cursor location and proceeding to the end of the file, without prompting you.

HINTS: It is a good idea to leave this checked and confirm the first one or two changes. When you're satisfied that an appropriate string has been specified in the Search For text box, you can uncheck this checkbox let the editor quickly finish the job.

Repeated use of the Undo command on the Edit menu can repair the effects of undesired changes.

If Confirm Changes is checked, you are prompted to accept or skip each occurrence of the found text. Otherwise, the editor automatically replaces all occurrences from the cursor location onward.

The number of changes made appears on the status line.

See Also

Searching with Regular Expressions
Finding Files that Contain Specific Text
Search Functions

Find Files Containing command (Search menu)

Function name: **find_files_containing**
Default key: **Ctrl+F**

This command displays the Find Files Containing dialog box, which lets you search the disk for files which contain specified text or a regular expression. Specify the text to search for, the types of files to search and a directory.

If one or more files are found to contain the search string, the List Found Files dialog box appears. You can open any of the files from here and edit it.

HINT: After opening a found file, the first occurrence of the search text is displayed and highlighted. Use Ctrl+A to find other occurrences.

Find Files Containing dialog box

Use this dialog box to choose what type(s) of files to scan and specify the text you want to find. The Find Files Containing dialog box appears when you choose Find Files Containing… from the Search menu.

Pattern:

Enter the text you want to find or click the prompt button (or press Alt+⬇) to choose from a list of previously-used search strings.

If you want to use regular expression wildcards in your search string, check the Regular Expression check box.

NOTE: The pcANYWHERE Editor cannot find text that spans two lines.

Files:

Enter one or more wildcards separated by spaces. Or, click the prompt button (or press Alt+⬇) to choose from a list of previously-used file specifications.

You can include a drive ID and path in any file specification. If omitted, only files in the current default directory are searched.

Examples:

*.c *.h *.asm	find a variable or function name in several types of program source code files
c:\wp\ltrs*.doc c:\wp\old*.*	look at files in two directories

Match Upper/Lowercase:

When this check box is checked, the pcANYWHERE Editor checks the case of the text in the Pattern combination box and returns only exact matches.

When unchecked, the editor considers text a match even when the case is not the same. For example, goto matches GOTO and Goto.

Regular Expression:

Check this if you want to enable the powerful and flexible regular expression pattern matching. Leave this unchecked (for speed) if you simply want to locate a string.

See Also

Finding Files that Contain Specified Text
Searching with Regular Expressions
The List Found Files command
Search Functions

List Found Files command (Search menu)

Function name: **list_files_containing**
Default key: **Ctrl+L**

This command displays a <u>List Found Files dialog box</u> containing the the files located using the Search Find Files Containing command. Use it to examine each file.

As you open each file from this dialog box, it is removed from the list. This makes it easy to process the files one-by-one, examining and/or making changes to each file.

HINT: After opening a found file, the first occurrence of the search text is displayed and highlighted. Use Ctrl+A to find other occurrences.

List Found Files dialog box

This dialog box appears when a search started by the Search Find Files Containing command locates one or more files. Thereafter, you can use the List Found Files command to view the list and choose files to open.

Pattern and **Filter**:

These are displayed to remind you of the search criteria used to obtain the files in this list.

Files:

This lists the files found by the most recent use of the <u>Find Files Containing command</u> (on the Search menu), less any files that have already been opened for editing.

Scroll through the list to select a file. Click Open to open the file and remove it from the list.

Open:

This opens the selected file for editing. On future uses of the List Found Files command, the opened file does not appear in the list.

See Also

<u>Finding Files that Contain Specified Text</u>
<u>Searching with Regular Expressions</u>
<u>The List Found Files command</u>
<u>Search Functions</u>

Goto Line command (Search menu)

Function name: **goto_line**
Default key: **Ctrl+G**

This command displays a Goto Line dialog box which lets you specify the number of a line to jump to.

The active document is scrolled to the requested line. If any text is currently selected, the selection is extended to include that line.

HINT: This command is particularly useful for programmers. When a compiler reports an error, you can quickly move to the line containing the error.

Goto Line dialog box

Use this dialog box to specify the line number at which to position the cursor.

Line Number:

Enter a number between 0 and 32,365.

The file in the active window is scrolled to the requested line. If any text is currently selected, the selection is extended to include that line.

See Also

The Goto Line command
Cursor Motion Functions
Search Functions

The Options menu

Commands in this menu set editor and document preferences.

Document Preferences Set margin, tabs, indent, etc. for current file only or defaults for all files.

Editor Preferences   Set screen font, cursor, backup, autosave.

Key Assignments   Customize keystrokes for fast access to editor functions.

Document Preferences command (Options menu)

Function name: **document_preferences**
Default key: **F4**

This command displays a Document Preferences dialog box which lets you select options for the currently active document.

NOTE: Document preference settings are used for the current document for the current editing session. They are not saved when you save your script file. However, you can specify default settings, to be used in all editing sessions, from the Editor Preferences dialog box.

See Also

Customizing the pcANYWHERE Editor
Document Preferences dialog box
The Editor Preferences command
The Key Assignments command

Document Preferences dialog box

Use this dialog box to customize settings for the current active document. This dialog box appears when you choose Document Preferences from the Options menu.

NOTE: These settings are used for the current document during the current editing session. They are not saved with the document. However, you can set default preferences by choosing Editor Preferences from the Options menu.

Tab Spacing:

Sets your desired tab-stops for the current document. When you press the Tab key, the cursor advances this many character-widths.

Right Margin:

Sets the wrapping-point to be used when word wrap is in effect. Enter a number of columns. The right margin is ignored unless the Word Wrap checkbox is checked (see below).

NOTE: To set the margins for printing, choose Page Setup from the File menu.

Word Wrap:

Check this to enable word-wrap in the current document for the current editing session. You can toggle this setting by using the Word Wrap command on the Edit menu.

*   When checked, typing beyond the right margin causes the current word to be moved down to the next lower line. Also, this enables the Wrap Paragraph command of the Edit menu.
*   When cleared, the right margin setting is ignored.

Auto Indent:

When checked, pressing Enter positions the cursor directly below the first non-blank character in the previous line. This is handy for programmers.

Expand Tabs with Spaces:

When checked, the editor replaces TAB characters (0x09) with a series of spaces (0x20) as it reads a file. It uses the tab settings set above. When it saves the file, all TAB characters are gone.

Save as Default Settings:

When checked, the above settings are used as the defaults for files opened in the future. Otherwise, the settings are used only for the current file during the current session.

See Also

Customizing the pcANYWHERE Editor
The Editor Preferences command
The Document Preferences command
The Key Assignments command

Editor Preferences command (Options menu)

  Function name: (none)
     Default key: **Alt+O,E**

This command displays an <u>Editor Preferences dialog box</u> which lets you select options to be used by the editor for all documents. These include the display font, cursor style, backup, autosave, and undo level preferences.

NOTE: Editor preference settings are saved and used in the current editing session and remembered for use in future sessions.

See Also

<u>Customizing the pcANYWHERE Editor</u>
<u>Editor Preferences dialog box</u>
<u>The Page Setup command</u>
<u>The Editor Preferences command</u>
<u>The Key Assignments command</u>

Editor Preferences dialog box

Use this dialog box to customize some of the pcANYWHERE Editor features. This dialog box appears when you choose Editor Preferences from the Options menu.

Changes made here are saved automatically and used each time the editor starts.

Font:

Select one of the option buttons. The font and size that is displayed depends on settings in your SYSTEM.INI file in the [boot] section:

* System Fixed Font   depends on **fixedfon.fon=**
* ANSI Fixed Font   depends on **fonts.fon=**
* OEM Fixed Font   depends on **oemfonts.fon=**

The OEM font contains the IBM text-mode box-and-line drawing characters.

Cursor:

Select the option button corresponding to how you want the text cursor to be displayed. Check the Blinking check box to enable a blinking cursor.

* Block   Covers the current character.
* Underline   Underlines the current character.
* Vertical Bar   Standard "insertion point" caret, displayed between characters.

* Blinking   The cursor blinks when this is checked. You can set the blink rate from the Windows Control Panel.

Autosave:

If you want the editor to automatically save open files, put non-zero values in either or both of the text boxes.

Undo Levels:

Sets how many edit changes are remembered for the underlined undo function. A large number tends to use up memory. Setting this to 0 disables the undo function.

Restore Session:

When checked, each file that was open last time you exited the editor is opened automatically next time you start the editor.

Typing Replaces Selection:

When checked, the editor follows the Windows standard convention of replacing any selected text with the first character typed or pasted. When unchecked, typing or pasting inserts the text to the left of the current selection.

Make Backup Files:

When checked, each time you save a file, the editor makes a copy of the original, giving it the same name, but with an extension of .BAK.

File Locking:

When checked, the editor locks each file when you open it and unlocks it only when you close its window. When a file is locked, no other program can delete or modify it. It only has an effect if a network is active and/or the program SHARE.EXE has been executed.

Cut/Copy Current Line if No Text is Selected:

When checked, you can quickly cut or copy the current line using the standard cut or copy keystrokes (without having to first select the line). When unchecked, you can cut or copy a line by using the keys assigned to the cut_line or copy_line functions.

Remove Trailing Spaces:

When checked, the editor discards spaces and tabs characters that are at the end of each line.

See Also

Customizing the pcANYWHERE Editor
The Editor Preferences command
The Document Preferences command
The Key Assignments command

Key Assignments command (Options menu)

Function name: (none)
Default key: **Alt+O,K**

This command displays a Key Assignments dialog box which lets you customize the way the editor interprets and acts on keystrokes. You can assign your own favorite shortcut keys to any editor function.

You can also use this command to see what keystrokes are currently mapped to the editor functions.

See   Editor Functions: Alphabetical List
or   Editor Functions: Functional Groups
for summaries of available editor functions.

NOTE: Be sure to use the Save button in the dialog box if you want your key assignments to be
               remembered for future editing sessions.

See Also

Customizing the pcANYWHERE Editor
Key Assignments dialog box
The Document Preferences command
The Editor Preferences command

Key Assignments dialog box

Use this dialog box to associate custom keystrokes with editor functions. You can also use it to look up the currently-assigned keystroke for an editor function. This dialog box appears when you choose Key Assignments from the Options menu.

NOTE: In order to make a keystroke assignment permanent, you must select a function and a key, click Assign, then click Save.

Function:
Select a function from this list box. As you move the cursor through the items in the list box, the current keystroke for that function is displayed in the Current Keys box below.

See   Editor Functions: Alphabetical List
or   Editor Functions: Functional Groups
for summaries.

Key:
Select a keystroke from the list box. Only the listed keystrokes can be used.

Assign and **Unassign**:
When a name is in both the Function and Key text boxes, you can click these buttons to associate a keystroke with a function or dissolve the association.

HINT: You can assign more than one keystroke to a function. For instance, you can add Alt+C to the keys for the copy function while retaining the standard Ctrl+Insert for compatibility.

Enable Menu Accelerators:
When this is checked, the standard menu accelerator keys such as Alt+F (to drop-down the File menu) take precedence over individual keystroke assignments. In general, this is the normal situation in Windows applications.

If you prefer to be able to use the normally preempted keystrokes for editing functions (for instance, if you want Alt+F to immediately save a file), you can clear this checkbox. You can still access the menus with the mouse, or by pressing and releasing Alt.

Keyboard Configuration File:
Load and **Save**:
When the pcANYWHERE Editor starts, it reads a file named DEFAULT.KEY from its home directory to map keystrokes to functions. If you want to maintain more than one set of keystroke assignments, you can enter a new name in the Keyboard Configuration File text box and click Save. Then, to load that file at a later date, enter its name in this text box and click Load.

NOTE: Changes you make are in effect only for the current session unless you use the Save button to save the assignments.

See Also

Customizing the pcANYWHERE Editor
The Key Assignments command
The Editor Preferences command
The Document Preferences command
Editor Functions: Alphabetical List

Key Assignments dialog box (Standard Browse)

Use this dialog box to locate an existing keyboard configuration file.

File Name: Type or select the desired filename. This box lists files with the extension selected in the List Files of Type box.

List Files of Type: Select the type of files to display in the File Name list box.

   Keyboard Configuration Files (*.KEY):   Lists all files in the current directory that were saved with a .KEY extension.

Drives: Select the drive on which the desired file is stored.

Directories: Select the directory in which the desired file is stored.

The Tool menu

This menu lets you access the script language quick reference dialog box. You can also access the Session Dial dialog box, used in writing scripts for unattended sessions.

Reference   Displays the Reference dialog box.

Reference command (Tool menu)

Function name: **batch_help**

Default key: **Alt+R**

This command displays the pcANYWHERE Editor <u>Reference dialog box</u>, which provides an alphabetically ordered list of script language commands and a description, syntax summary, and example of each command.

You can keep the Reference dialog box open as you create or edit your script file. Use Alt+F6 to toggle back and forth between the Reference dialog box and the editing window.

Reference Dialog Box

Use this dialog box as a quick reference for the 150 commands in the pcANYWHERE script language. This dialog box appears when you choose Reference from the Tools menu.

Commands

   * Use ⬇ and

⬆ to scroll though this list.

*   Or, press the first letter of any command to scroll quickly to that part of the list.

Description

As you scroll through the Commands list box, the Description box displays a short description of the command, its syntax and example of its use.

Close

Click this button to close the Reference dialog box.

Add

This inserts the currently highlighted statement from the Commands list into the text-editing screen at the current cursor position. This handy feature will help you avoid misspelling command names.

Assist

This button is enabled only when the Session Dial command is selected. Use the Assist button for help when you want to write a script to be used in an unattended session. The Norton pcANYWHERE Editor helps you create this file by taking the information you enter in the Assist dialog box and generating the basic commands required for the script. Use this dialog box to ensure that the proper parameters are set.

Dial Commands Dialog Box

Use when you want to write a script for commands to be used in an unattended session. The Norton pcANYWHERE Editor helps you create this file by taking the information you enter in the Assist dialog box and generating the basic commands required for the script. Use this dialog box to ensure that the proper parameters are set.

For specific information about this and other Script Language commands, see Chapter 4 of the Creating Norton pcANYWHERE Scripts manual.

Load Commands Dialog Box

Use when you want to write a script for commands to be used in an unattended session. The Norton pcANYWHERE Editor helps you create this file by taking the information you enter in the Assist dialog box and generating the basic commands required for the script. Use this dialog box to ensure that the proper parameters are set.

For specific information about this and other Script Language commands, see Chapter 4 of the Creating Norton pcANYWHERE Scripts manual.

Receive Commands Dialog Box

Use when you want to write a script for commands to be used in an unattended session. The Norton pcANYWHERE Editor helps you create this file by taking the information you enter in the Assist dialog box and generating the basic commands required for the script. Use this dialog box to ensure that the proper parameters are set.

For specific information about this and other Script Language commands, see Chapter 4 of the Creating Norton pcANYWHERE Scripts manual.

Send Commands Dialog Box

Use when you want to write a script for commands to be used in an unattended session. The Norton pcANYWHERE Editor helps you create this file by taking the information you enter in the Assist dialog box and generating the basic commands required for the script. Use this dialog box to ensure that the proper parameters are set.

For specific information about this and other Script Language commands, see Chapter 4 of the Creating Norton pcANYWHERE Scripts manual.

Wait Commands Dialog Box

Use when you want to write a script for commands to be used in an unattended session. The Norton pcANYWHERE Editor helps you create this file by taking the information you enter in the Assist dialog box and generating the basic commands required for the script. Use this dialog box to ensure that the proper parameters are set.

For specific information about this and other Script Language commands, see Chapter 4 of the Creating Norton pcANYWHERE Scripts manual.

Session Dial dialog box

Use this dialog box to write a script to be used in an unattended session. The Norton pcANYWHERE uses the information you enter here to generate correctly formatted Session commands. These commands contain the parameters used during the unattended session.

Once you have the session outlined, complete the script by specifying the action to occur after the connection has been made. These commands should be inserted into your script at the line `COMMENT: ** insert SESSOPR here **`. If you are using your script to launch an application on the host computer, be sure to use the correct SessOpr commands. The corresponding "regular" commands (such as Send File) in the script language are unavailable for unattended sessions and return an error when used.

Host to Call: Click the prompt button to select the name of the host you want to connect to during the session. The drop-down list box contains the same set of names as those in the Host PC Directory list box in the Call Host PC main window. For example, choosing Carol's Office PC from the list box generates the command:

    SESSION DIAL "Carol's Office PC"

Cancel Host After Session: Check this check box to disable the host TSR when your session is complete. If this box is checked, the following command is generated:

    SESSION HOST EXITMODE CANCEL

Call Immediately: Check this check box if you want the call to be made as soon as the script is executed. You can also specify a date and time at which the call is to be made. If this box is unchecked, then the Time to Call and Date to Call text boxes are enabled.

Time to Call: Enter the time you want Norton pcANYWHERE to dial the host. The format is military time (24-hour clock). If you specify a time, you must also specify a date in the Date to Call text box. For example, specifying a time of 10:00 P.M. on June 30, 1993 generates the command:

    SESSION DIAL "Carol's Office PC" 22:00 93/06/30

Date to Call: Enter the date you want Norton pcANYWHERE to dial the host. The format is YY/MM/DD. If you specify a date, you must also specify a time in the Time to Call text box.

Retries: Enter the number of times you want Norton pcANYWHERE to redial the host. An entry of 3 generates the command:

    SESSION RETRY 3

Time Between Retries: Enter the number of minutes that elapse between retries to dial the host. An entry of 10 generates the command:

    SESSION DELAY 10

If Error Occurs: Click the prompt button to select the action to perform if an error should occur during the unattended session. You can select from:

  END SCRIPT--ends the script and returns control to Norton pcANYWHERE.

  CONTINUE SESSION--ignores the command which caused the error and advances to the next command.

  END CURRENT SESSION--advances to the matching Session End command and continues processing from that point on in the script.

For example, selecting End Script generates the following command:

    SESSION ONERROR END

File Transfer Overwrites: Click the prompt button to select the action to perform when the filename parameter in a SessOpr Host Send or SessOpr Remote Send command is the same as a filename at the destination. You can select from:

ALWAYS--causes the destination file to be overwritten.

NEVER--causes the file sent to be ignored.

OLDER--causes the file at the destination to be overwritten if the file being transferred is more recent (compares date/time stamp).

For example, selecting Older generates the command:

```
SESSION OVERWRITES OLDER
```

Error Time Out: Enter the time (in minutes) of the remote's inactivity timeout for unattended sessions. For example, entering 30 generates the command:

```
SESSION TIMEOUT 30
```

The Window menu

This menu lets you select, create, and manipulate the pcANYWHERE Editor's editing windows.

New Window   Creates an additional window containing a copy of the file in the active window.

Cascade   Rearranges windows in a cascading pattern.

Tile   Rearranges windows in a non-overlapping pattern.

Arrange Icons   Aligns any minimized editing windows.

1 to *n*   Activates one of the editing windows, restoring it to its standard size if it is minimized.

See Also

Managing Multiple Windows
Window Control Functions

New Window command (Window menu)

Function name: **new_window**
Default key: **Alt+W,C**

Creates a new editing window and activates the new window.

If the same file is open in two or more editing windows, editing the file in any of the windows affects the contents of the file. Also, any file-related commands, such as changing the name using the Save As... command, affect all windows. However, the current cursor position, selection, and similar attributes are tracked separately for each window.

See Also

Managing Multiple Windows
Window Control Functions

Cascade command (Window menu)

Function name: **cascade**

Default key: **Alt+W,C**

Reorganizes all open editing windows in a cascading pattern. This usually makes it possible to see the titles of all the open windows, while making one entire window visible.

Any <u>minimized</u> windows remain icons.

See Also

<u>Managing Multiple Windows</u>
<u>Window Control Functions</u>

Tile command (Window menu)

Function name: **tile**
Default key: **Alt+W,T**

Reorganizes all open editing windows in a tiled (non-overlapping) pattern. This makes it possible to see at least a small part of all of the open windows.

Any <u>minimized</u> windows remain icons.

See Also

<u>Managing Multiple Windows</u>
<u>Window Control Functions</u>

Arrange Icons command (Window menu)

Function name: (none)
Default key: **Alt+W,A**

Aligns all <u>minimized</u> editing windows along the bottom of the pcANYWHERE Editor application window.

See Also

<u>Managing Multiple Windows</u>
<u>Window Control Functions</u>

pcANYWHERE Editor Keys

The pcANYWHERE Editor lets you assign your own favorite keystrokes to any of its over 100 editing functions. See <u>Customizing Editor Keystrokes</u> for details.

For information on default keystroke assignments, review the topics below:

<u>Cursor Movement</u>
<u>Selecting Text</u>
<u>Function Keys</u>
<u>Other Keys</u>

See Also

<u>The Key Assignments command</u>
<u>Editor Functions: Alphabetical List</u>
<u>Using Editor Macros</u>

Scripts dialog box

From the Scripts dialog box, you can create a new script, or select a script you have already created and edit, compile or run it.

The Scripts list box contains a list of all the currently defined scripts and a short description of each. Directly beneath the list box is the Script Path text box. If your script files are located in a directory other than the one displayed, you can change directories using the Select… button.

After highlighting a file in the list box, simply click the Edit, Compile or Run button to perform that function on the selected script.
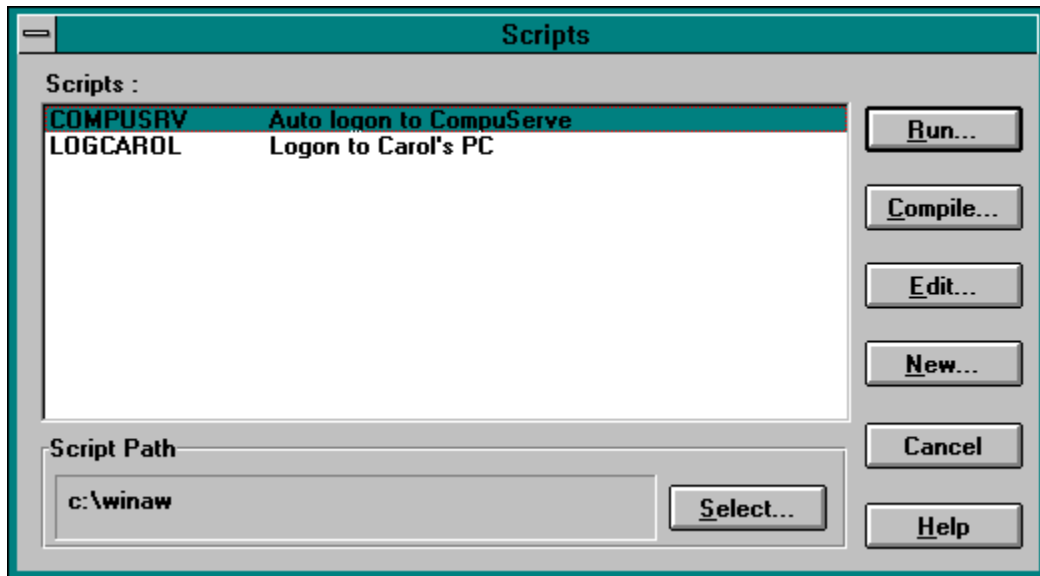
Select Directory dialog box

Use this dialog box to specify the drive and directory where your script files are located.

Directories: Scroll through the list box to locate the directory in which your script files are located.

Drives: Click the prompt button to display a list box containing all available drives.

All script files in the current script path appear here. Select a script and click Run, Compile or Edit.

Displays the currently selected path. If your script files are in a different directory, click Select and select a new drive and directory.

Click here to display the <u>Select Directory dialog box</u> and change your Script Path.

Select a script from the Scripts list box and click here to execute the selected script.

Select a script from the Scripts list box and click here to compile the selected script.

Select a script from the Scripts list box and click here to edit the selected script.

Click here to launch the pcANYWHERE Editor and create a new script.

Click here to close the dialog box and return to the pcANYWHERE main window.

New command (File menu)

Use the New command to create a new document file.

Open command (File menu)

Use the Open command to open an existing document file.

Close command (File menu)

Use the Close command to close active document window.

Save command (File menu)

Use the Save command to save the file in the active window.

Save As command (File menu)

Use the Save As command to save the file in active window under a different name.

Save All command (File menu)

Use the Save All command to save all currently open files.

Insert command (File menu)

Use the Insert command to insert the contents of a file at the cursor position.

Revert command (File menu)

Use the Revert command to undo all changes made since the last save.

Write Block command (File menu)

Use the Write Block command to write the selected text to a file you specify.

Printer Setup command (File menu)

Use the Printer Setup command to select a printer and printer options.

Page Setup command (File menu)

Use the Page Setup command to set headers, footers, margins, etc.

Print command (File menu)

Use the Print command to send the contents of the file in the active document window to the printer.

Compare command (File menu)

Use the Compare command to compare the contents of two documents.

Exit command (File menu)

Use the Exit command to close the pcANYWHERE Editor windows and exit.

(list of files) command (File menu)

Select any number from 1 to 4 to open or activate the recently used file asociated with that number.

Undo command (Edit menu)

Use the Undo command to undo the most recent text-editing change.

Cut command (Edit menu)

Use the Cut command to copy the selected text to the clipboard and then delete the text from the document.

Copy command (Edit menu)

Use the Copy command to copy the selected text to the clipboard.

Paste command (Edit menu)

Use the Paste command to paste the contents of the clipboard to the current cursor position.

Delete command (Edit menu)

Use the Delect command to delete the current selection.

Select All command (Edit menu)

Use the Select All command to select all text in the active file.

Time/Date command (Edit menu)

Use the Time/Date command to insert the current date and time at the current cursor position.

Wrap Paragraph command (Edit menu)

Use the Wrap Paragraph command to reformat the paragraph at the cursor, wrapping at the current margin settings.

Word Wrap command (Edit menu)

Use the Word Wrap command to toggle automatic paragraph wrapping on or off.

Record Macro command (Edit menu)

(toggles to Stop Recording Macro) Use the Record Macro command to record subsequent keystrokes for automatic recall.

Playback Macro command (Edit menu)

Use the Playback Macro command to replay the keystrokes saved during the most recent Record Macro command.

Find command (Search menu)

Use the Find command to search for text or a regular expression.

Find Again command (Search menu)

Use the Find Again command to continue the search started with the Find command.

Replace command (Search menu)

Use the Replace command to search for and replace text or regular expression.

Find Files Containing command (Search menu)

Use the Find Files Containing command to generate a list of files which contain specified text or regular expression.

List Found Files command (Search menu)

Use the List Found Files command to see a list of files found; choose one to edit.

Goto Line command (Search menu)

Use the Goto Line command to move the cursor to a line specified by its line number.

Document Preferences command (Options menu)

Use the Document Preferences command to set the margins, tabs, indent, etc. for current document only or the default for all documents.

Editor Preferences command (Options menu)

Use the Editor Preferences command to set the screen font, cursor, backup, and autosave options.

Key Assignments command (Options menu)

Use the Key Assignments command to customize keystrokes for fast access to editor functions.

Reference command (Tool menu)

Use the Reference command to display the script language Reference dialog box. The Reference dialog box contains the name of each of the script langauge commands, a description, syntax and an example of its use.

New Window command (Window menu)

Use the New Window command to create an additional window on the document in the active window.

Cascade command (Window menu)

Use the Cascase command to rearrange the document windows in a cascading pattern.

Tile command (Window menu)

Use the Tile command to rearrange the document windows in a non-overlapping pattern.

Arrange Icons command (Window menu)

Use the Arrange Icons command to align any minimized document windows.

1 to n command (Window menu)

Select a number from 1 to n to activates one of the document windows, restoring it to its standard size if it is an icon.

Navigation Key Defaults

Key	Default Editor Function
←	cursor_left
Ctrl+←	word_left
→	cursor_right
Ctrl+→	word_right
↑	cursor_up
Ctrl+↑	window_down
↓	cursor_down
Ctrl+↓	window_up
PgUp	page_up
Ctrl+PgUp	top_of_window
PgDn	page_down
Ctrl+PgDn	bottom_of_window
Home	beginning_of_line
Ctrl+Home	beginning_of_buffer
End	end_of_line
Ctrl+End	end_of_buffer
Tab	tab_right
Enter	enter
Ctrl+G	goto_line

Text Selection Key Defaults

Key	Default Editor Function
Shift+↑	select_line_up
Shift+↑	select_line_down
Shift+↑	select_char_left
Shift+Ctrl+↑	select_word_left
Shift+↑	select_char_right
Shift+Ctrl+↑	select_word_right
Shift+PgUp	select_page_up
Shift+PgDn	select_page_down
Shift+Home	select_to_bol
Shift+Ctrl+Home	select_to_top
Shift+End	select_to_eol
Shift+Ctrl+End	select_to_end
Esc	unmark_block

NOTE: The default keys are assigned so that pressing Shift along with a navigation key selects text as the cursor moves.

F1...F12 Function Key Defaults

Key	Default Editor Function
F1	editor_help
F2	save_file
F3	open_file
F4	document_preferences
Ctrl+F4	close_window
F5	(none)
Ctrl+F5	restore_window
F6	(none)
Ctrl+F6	next_window
F7	record_macro
F8	play_macro
F9	(none)
F10	(none)
Ctrl+F10	zoom_window
F12	wrap_para

Other Key Defaults

Key	Default Editor Function
BkSp	backspace
Alt+BkSp	undo
Ctrl+BkSp	delete_word_left
Del	delete
Ctrl+Del	delete_word_right
Shift+Del	cut
Ins	toggle_insert
Ctrl+Ins	copy
Shift+Ins	paste
Enter	enter
numpad *	undo
numpad +	copy
numpad -	cut
Tab	tab_right
Alt+ -	close_window
Alt+D	delete_line
Alt+K	delete_to_eol
Alt+N	next_window
Alt+P	print
Alt+R	batch_help
Alt+V	about
Alt+X	save_all_exit

pcANYWHERE Editor Procedures

Using the pcANYWHERE Editor

The pcANYWHERE Editor is the text editor supplied with pcANYWHERE for Windows. It is the default editor used when creating and editing scrips.

Features of the pcANYWHERE Editor include:
* A customizable keyboard--you can assign keystrokes to over 100 editing functions.
* MDI windows--you can open many edit windows, then tile, cascade, and minimize them. You can also open the same script file in more than one window.
* Uses as much memory as it needs for fast editing of large files.
* Flexible find and replace capabilities, including the use of regular expression text wildcards. The editor also remembers recently-used search strings.
* Ability to search for files containing a specific word or phrase and easily open them for inspection or editing.
* Customizable auto-save and automatic backup options.
* Ability to compare two files on a line-by-line basis.
* Auto-indent (handy for programmers).
* Supports word wrapping and paragraph reformatting for simple word processing tasks.
* Remembers recently-edited files--you can reopen the exact same set of files you used in your last editing session.

Opening files when the pcANYWHERE Editor is running:
* Choose the editor's File Open command.
* Or, to open a file that you recently edited, choose it from the list at the bottom of the editor's File menu.
* Or, after using the editor's Find Files Containing command, use the List Found Files command to open any of the found files.

To automatically reload a set of files next time you start:
1 Use the Options Editor Preferences command and check the Restore Session check box.
2 Open the files you want to load the next time you start the editor.
3 Use the File Exit command to quit the editor (Note: don't close the files individually).
4 Next time you start the editor, it automatically opens the same files, arranging the windows in a cascading pattern.

Using Editor Macros

The pcANYWHERE Editor allows you to record and play back a single keyboard macro. You can use this to automate simple, repetititive tasks.

To record an editor macro:
1 Press F7 (Alt+E,R) to start recording. The <u>status line</u> displays the word REC to indicate it is recording.
2 Type text and use menu commands and keystroke functions to accomplish the desired task.
3 Press F7 again (Alt+E,S) to stop recording. The <u>status line</u> displays the message Keyboard macro defined.

To play back the editor macro:
1 Make sure the desired window is active and the cursor is positioned correctly.
2 Press F8 (Alt+E,P) to begin replaying the macro.

NOTE:: Events that take place in dialog boxes are NOT recorded.
Up to 256 editing events can be recorded.
The current macro is not saved between editing sessions.

Using Multiple Edit Windows

The pcANYWHERE Editor conforms to the Multiple Document Interface (MDI) standard to display and interact with document files. This interface lets you see more than one document at a time and provides standard ways to access the documents.

To activate a window (bring it to the front):
Click any part of it, select it from the window list under the Window menu, or press Ctrl+F6 until it moves to the front.

To enlarge a window to see more text:
First <u>maximize</u> the application window (click the maximize button or press Alt+spacebar,X), then maximize the individual document window (click the maximize button or press Alt+hyphen, X).

To see two parts of a document simultaneously:
Activate the desired window. Use the Open Window command of the Window menu. You can scroll the windows independently.

To see as much as possible of two documents:
First <u>maximize</u> the application window, then choose the Window Tile command (Alt+W,T).

To see the title bars of all windows:
Choose the Window Cascade command (Alt+W,C).

To iconize a window:
<u>Minimize</u> the window into an icon (click the minimize button or press Alt+hyphen,N). It is now displayed at the bottom of the application window.

To restore an icon to a normal-size window:
Double-click the icon, or press Ctrl+F6 until it is highlighted, then press Alt+Hyphen,X to maximize it. You can also select it from the Window menu.

To tidy up window icons:
Use the Window Arrange Icons command (Alt+W,A).

Searching with Regular Expressions

The Find, Replace, and Find Files Containing commands of the Search menu provide a powerful <u>text wildcard</u> search capability called <u>regular expression</u> searching.

Although not the full-blown "grep" utility known to UNIX programmers, this feature can be useful in many situations. For instance, it can be used for locating:

* Text in formatted layouts, such as phone numbers, zip codes, text in parentheses, quoted strings, etc.
* Programming elements and idioms; for instance, locating the start of any procedure.
* Proper names or the start of a sentence (i.e., first letter capitalized).
* Blank lines.
* A sequence of digits.
* Lines beginning or ending with some specified text.
* Alternate spellings of a word or name (e.g., Kathy or Cathy).
* Leading or trailing blanks in a line.

One of the powers of this feature over simple text-matching is that you can use it to select entire words or lines that contain a match. For instance,

```
goto        selects just that single word when found
<*goto*>    selects the entire line that contains the word
```

You can use this feature in <u>editor macros</u> to automate some operations. For instance, you can easily copy the matching line or word to the clipboard and paste it to a second file.

Examples

```
[KC]athy                Kathy or Cathy
[KC]athy [A-Z][~ ]@     Kathy or Cathy and her last name
[A-Z][a-z]@ [A-Z][~ ]@      any two-word proper name
[2-9]??-[0-9]???        any phone number
<*[2-9]??-[0-9]???*>        a line that contains any phone number
<* [A-Za-z]@(*){>       a line that defines a C function; e.g., int DoIt(x,y){
<{                      finds "{" when at the start of a line
<[ \t][ \t]>            any line containing only blanks or tabs
\\[nrt]                 finds "\n" or "\r" or "\t"
```

Printing

The Desktop Editor provides printing capabilities that will more than satisfy your needs when you are working with text files, including font selection, and generation of page headers and footers. For more complex formatting, you may need to open the file using a full-featured word processor.

To print an entire document:
  1 Activate the document you want to print by clicking its window title.
  2 Deselect any selected text by pressing Esc.
  3 Choose the File Printer Setup command to specify a printer.
  4 Choose the File Page Setup command to select printing options. You can type text to be displayed at the bottom and/or top of each page, as well as select a font for the printout.
  5 Choose the File Print command to send the file to the printer.

To print a partial document
  1 Select the portion of the document you want to print.
  2 Follow the steps above. You can modify the header or footer to indicate that this is a partial printout. All lines having any text selected are printed.

Comparing Two Files

The pcANYWHERE Editor provides a handy file-comparison feature. This is especially useful for programmers who want to check for differences between versions of program source code.

To compare two files:

* (optional) <u>Maximize</u> the editor window and minimize all open documents. This provides screen space to show larger document windows when displayed.

1 Choose Compare from the File menu to display the <u>Compare dialog box</u>.

2 Select the two files you want to compare.

3 Leave the Line text box set at 1 for both files.

4 Choose the Horizontal option if you want the files to be displayed one above the other, or choose Vertical for a side-by-side display.

5 Click OK. The editor displays both files, highlighting the first difference it finds. If no differences are found, it displays a message on the <u>status line</u>.

6 The editor displays a small dialog box. Click Find Next... to advance through the files, or click Cancel when satisfied.

Finding Text in an Open File

The pcANYWHERE Editor can search for simple text strings or for special text wildcards. See <u>Searching with Regular Expressions</u> for a description of the latter capability. See <u>Searching for Files</u> for a way to locate all files which contain a specified text string.

To search an entire file:
1 Open and activate the file you want to search.
2 Press Ctrl+Home to get to the top of the document (the search always starts at the current cursor location).
3 Press Ctrl+S or use the Find command of the Search menu to bring up the <u>Find dialog box</u>.
4 Enter the text you want to find into the Pattern text box.

   Or, press Alt+{bmp symDnAro.Bmp} to choose from a list of recently-used search strings.

   Or, you can select text before opening the dialog and the selected text is placed into the text box automatically.
5 Uncheck the Match Upper/Lowercase check box to provide the best chance of finding a similar match. Leave the Regular Expressions check box unchecked to speed up the search.
6 Click Next to start searching. If any match is found, it is displayed and highlighted. Otherwise, the message Pattern not found is displayed in the <u>status line</u>.

To continue a search:
* Press Ctrl+A or use the Find Again command on the Search menu. If another match is found, it is displayed and highlighted. Otherwise, the message Pattern not found is displayed in the <u>status line</u>.

To search backward through a document:
1 Position the cursor where you want the search to start.
2 Press Ctrl+S, and follow the steps described above, except use the Previous button in the dialog box to begin the search.
3 Pressing Ctrl+A continues the search backward through the document until it hits the top of the file.

Finding Files that Contain Specified Text

The pcANYWHERE Editor can search the disk for files containing a specified string of text. This is particularly useful for programmers wishing to inspect or modify all source code files that use a particular function or variable.

To find files containing a search string:
1 Use the Find Files Containing command on the Search menu to bring up the <u>Find Files Containing dialog box</u>.
2 In the Pattern text box, enter the search string (it can contain <u>regular expression wildcard characters</u>). Or, select from the list of previously-used search strings.
3 In the Files text box, enter one or more <u>wildcard file specifications</u> separated by spaces.
   Or, select from the list of previously-used filespecs.
   Or, click Directory... and select the directory from the tree display.
4 Clear the Match Upper/Lowercase check box, unless you want the search to be case-sensitive.
5 Check the Regular Expression check box only if you have used text wildcards in the Pattern text box.
6 Click OK to start the search.
7 If any files containing the specified string are found, a list of filenames appears. Just double-click a filename to open it. The first occurrence of the found text is displayed.

To inspect or edit other files that contain the found string:
1 Use the List Found Files command on the Search menu to display the <u>List Found Files dialog box</u>.
2 Select a file from the scrolling list and click Open.
3 As you open files from this dialog, they are removed from the list.

Customizing Editor Keystrokes

The pcANYWHERE Editor lets you customize the keyboard; you can assign any keystroke to any of over 100 editor functions.

Most users are satisfied with the default key assignments. However, when you are accustomed to another editor, you may find it convenient to change some of the pcANYWHERE Editor keystrokes to match your old habits (so your fingers can do the thinking...).

To add a keystroke to a function:

1 Browse the Editor Functions: Functional Groups help topic to familiarize yourself with the available functions.

2 Use the Key Assignments command on the Options menu to display the Key Assignments dialog box.

3 Press ⬆ and scroll through the Functions list box until you highlight the function you want to access; its name appears in the box on top.

   The Current Keys box at the bottom displays what key or keys are currently assigned to the function. You are about to add another keystroke to those assigned to the function.

4 Press Tab and scroll through the Keys list box until you find the keystroke you want to use to access that function.

   The Current Function box at the bottom of the dialog box displays what function (if any) is currently assigned to that keystroke. If you proceed, you lose that assignment, since a key can be assigned to only one function.

5 Click Assign.

6 Click Save to save the changes to the file, DEFAULT.KEY.

7 Click OK.

To change a keystroke assignment:

1 Use the Key Assignments command on the Options menu to display the Key Assignments dialog box.

2 Press Tab and ⬆ and scroll through the Keys list box until you find the keystroke you want to change.

   The Current Function box at the bottom displays what function (if any) is currently assigned to that keystroke. If you proceed, you lose that assignment, since a key can be assigned to only one function.

3 Press Shift+Tab and ⬆ and scroll through the Functions list box until you find the editor function you want to map to the selected keystroke.

4 Click Assign.

5 Click Save to make the assignment permanent.

6 Click OK.

Finding Which Key Performs an Editor Function

The pcANYWHERE Editor features over one hundred editing functions. You can use this on-line help to learn the default keystroke assignments. Just peruse either of the topics:

Editor Functions: Alphabetical List
Editor Functions: Functional Groups

Since some keystrokes may have been reassigned (or if you want avoid this help system), you can find the current key assignments as follows:

To see what key performs a function:

1 Use the Key Assignments command on the Options menu to bring up the Key Assignments dialog box.

2 Press ↑ and scroll through the Functions list box until you highlight the function you want to access; its name appears in the box on top.

The Current Keys box at the bottom displays what key or keys are currently assigned to the function.

3   Click Cancel to avoid changing any settings.

Customizing the pcANYWHERE Editor

The pcANYWHERE Editor lets you customize many of its features. If you are already familiar with another text editor, you can modify the pcANYWHERE Editor to conform to your editing habits. And, as you become comfortable with the pcANYWHERE Editor, you can make minor adjustments to speed up common operations.

Some customization is automatic. For example, the pcANYWHERE Editor automatically remembers:
* Files that you have recently edited (they are placed on the File menu).
* Search strings you have previously used.
* File types recently used when finding files.
* Files that you have recently compared.
* Page printing settings, including margins, font face and size, headers, and footers.

Here are some other features and the menu commands that let you customize them:

Feature	Menu command
Automatic file backups	Options Editor Preferences
Automatic file saving	Options Editor Preferences
Automatic line indenting	Options Document Preferences
Automatic file loading	Options Editor Preferences
Convert tabs to spaces	Options Document Preferences
Cursor shape	Options Editor Preferences
Features not on menus	Options Key Assignments
File locking	Options Editor Preferences
Header/Footer text	Options Document Preferences
Keyboard macro	Edit Record Macro (see Using Macros)
Keystroke assignments	Options Key Assignments
Line indenting	Options Document Preferences
Printer Font	Options Document Preferences
Printer Margins	Options Document Preferences
Programmer's indenting	Options Document Preferences
Reload recent files	Options Editor Preferences
Remove trailing spaces	Options Editor Preferences
Screen Font	Options Editor Preferences
Screen margin settings	Options Document Preferences
Tab settings	Options Document Preferences
Typing replaces selected text	Options Editor Preferences
Undo levels	Options Editor Preferences
Word wrapping	Options Document Preferences

word

For the purposes of word-wrap:
    An unbroken sequence of non-blank characters.

As defined for cursor motion, selection, and deletion:
    A sequence of characters unbroken by space, tab, or any common programing text <u>delimiters</u> such as
    ; , { } ( ), etc.

paragraph

A sequence of lines of text, containing no blank lines.

When determining how to wrap a paragraph, the editor looks backward from the cursor location until it finds a blank line, then assumes that the paragraph includes all lines after that, down to, but not including, the first blank line.

status line
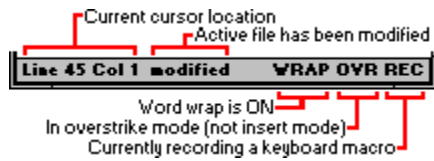
The pcANYWHERE Editor status line, at the bottom of its application window, displays information about the currently active edit window.

Information, including error messages and search-and-replace status is displayed momentarily on the left side of the status line, until the next keystroke or mouse click.

delimiter

When using editor functions to advance or select by "words," the following characters are considered as word-ending delimiters:

tab (0x09)
space (0x20)
end-of-line

_ ( ) " ' , # $ @ ! % ^ & * { } [ ] \ ? / | < > ; . + - = ~ :

regular expression "wildcard" codes

?	any single character or space
*	zero or more characters or spaces
[chars]	any one of the characters between [and]
[~chars]	any one character except those after the ~
[c1-c2]	any one of the range of characters c1 to c2
@	zero or more of the previous character
% or <	the beginning of a line
$ or >	the end of a line
\t	a tab character (0x08)
\f	a formfeed character (0x0c)
\c	the literal character c; e.g. \\ means \

wildcard file specification

You can use wildcard characters to specify a set of files which share similar names. The wildcard characters are:

  * any 1-to-8 character filename (or 0-to-3 character file extension)
  ? any single character

Examples:

.   any filename and any extension
*.bak   any filename with an extension of .BAK
*.?   any file that has a 1-character extension
*.   any file that has no extension
*.WK?   any filename that has an extension that starts with .WK and then any other single character
rpt??jul.* any filename starting with RPT, then any two characters, then JUL, and having any extension

Editor Functions: Alphabetical List

-A-

about

-B-

backspace   batch_help
beginning_of_buffer   beginning_of_line
bottom_of_window

-C-

cascade   change_case
close_window   compare
copy   copy_block
copy_line   cursor_down
cursor_left   cursor_right
cursor_up   cut
cut_block   cut_line

-D-

delete   delete_line
delete_to_eol   delete_word_left
delete_word_right   document_preferences

-E-

editor_help   end_of_buffer
end_of_line   enter
exit   exit_windows

-F-

find   find_again
find_files_containing

-G-

goto_line

-L-

list_files_containing   lowercase

-N-

new_file   new_window
next_window

-O-

open_file

-P-

page_down   page_up
paste   play_macro
prev_window   print

-R-

record_macro   replace
restore_window   revert

-S-

save_all

save_all_exit

save_all_exit_windows

save_file

save_file_close_window

select_all

select_char_left

select_char_right

select_line

select_line_down

select_line_up

select_page_down

select_page_up

select_to_bol

select_to_end

select_to_eol

select_to_top

select_word

select_word_left

select_word_right

split_line

stamp

-T-

tab_right

test_batch

tile

to_bottom

to_center

to_top

toggle_backup

toggle_insert

toggle_wordwrap

top_of_window

-U-

undo

unmark_block

uppercase

-W-

window_down

window_up

word_left

word_right

wrap_para

-Z-

zoom_window

Editor Functions: Functional Groups

Cursor Motion Functions

beginning_of_buffer

beginning_of_line

bottom_of_window

cursor_down

cursor_left

cursor_right

cursor_up

end_of_buffer

end_of_line

enter

goto_line

page_down

page_up

split_line

tab_right

to_bottom

to_center

to_top

top_of_window

window_down

window_up

word_left

word_right

Cut, Copy, and Paste Functions

copy

copy_block

copy_line

cut

cut_block

cut_line

paste

undo

Delete Functions

backspace	delete_to_eol
cut	delete_word_left
cut_line	delete_word_right
delete	replace
delete_line	undo

File Functions

close_window

compare

find_files_containing

list_files_containing

new_file

open_file

revert

save_file

save_file_close_window

save_all

save_all_exit

save_all_exit_windows

toggle_backup

Line Functions

beginning_of_line

copy

copy_line

cursor_down

cursor_up

cut

cut_line

delete_line

delete_to_eol

end_of_line

enter

goto_line

select_line

select_line_down

select_line_up

select_to_bol

select_to_eol

split_line

to_bottom

to_center

to_top

window_down

window_up

Miscellaneous Functions

about

batch_help

change_case

document_preferences

editor_help

enter

exit

exit_windows

lowercase

play_macro

print

record_macro

save_all_exit_windows

split_line

stamp

tab_right

test_batch

toggle_backup

toggle_insert

toggle_wordwrap

undo

uppercase

Paragraph Functions

document_preferences

enter

split_line

toggle_wordwrap

wrap_para

Text Selecting Functions

select_all	select_to_bol
select_char_left	select_to_end
select_char_right	select_to_eol
select_line	select_to_top
select_line_down	select_word
select_line_up	select_word_left
select_page_down	select_word_right
select_page_up	unmark_block

NOTE: The default keys are assigned so that pressing Shift along with a navigation key selects text as the cursor moves.

Search and Replace Functions

find

find_again

find_files_containing

list_files_containing

replace

Window Functions

cascade

close_window

new_window

new_file

next_window

prev_window

restore_window

tile

to_bottom

to_center

to_top

window_down

window_up

zoom_window

See Also

Managing Multiple Windows

about

Displays the editor's version number, copyright notice and miscellaneous system information.

Default keystroke: **Alt+V**

backspace

If text is selected, deletes it; otherwise, deletes the character to the left of the cursor.

If the cursor is at the beginning of the line, appends the current line to the previous line.

Default keystroke: **BkSp** or **Shift+BkSp**

batch_help

Displays help text on each script langauge <u>command</u> and lets you select commands to enter into your current file.

Default keystroke: **Alt+R** or **Alt+T,R**

beginning_of_buffer

Moves the cursor to the beginning of the document in the active window.

Default keystroke: **Ctrl+Home**

beginning_of_line

Moves the cursor to the beginning of the current line.

Default keystroke: **Home**

bottom_of_window

Moves the cursor to the bottom line of the window.

Default keystroke: **Ctrl+PgDn**

cascade

Resizes and rearranges all open windows in an overlapping pattern. Same as the Cascade command on the Window menu.

Default keystroke: **Alt+W,C**

change_case

Within the selected block, changes the case of all letters, making uppercase letters into lowercase, and lowercase letters into uppercase.

Default keystroke: (none)

close_window

Closes the active window. If you have made changes to the file, the pcANYWHERE Editor prompts you to save them if you have not already done so.

Same as the Close command on the File menu. You can also close a window by double-clicking its document control-menu button.

Default keystroke: **Ctrl+F4**, or **Alt+hyphen,C**

compare

Prompts for two filenames, then compares them line-by-line.

Same as the Compare command on the File menu.

Default keystroke: **Alt+F,E**

copy

Copies the selected text to the clipboard.

If no text is selected, but the Editor Preferences dialog option Cut/Copy line if no text is selected is checked, then this copies the entire line to the clipboard.

Same as the Copy command on the Edit menu.

Default keystroke: **Ctrl+Ins**, or **numpad +**

copy_block

Copies the selected text to the clipboard.

This variation of the copy command has no effect if no text is selected.

Default keystroke: (none)

copy_line

Copies the current line to the clipboard.

Default keystroke: (none)

cursor_down

Moves the cursor down one line.

Default keystroke: **DownArrow**

cursor_left

Moves the cursor one character to the left or, if in the leftmost column, to the end of the line above.

Default keystroke: **LeftArrow**

cursor_right

Moves the cursor one character to the right or, if at the end of the line, to the beginning of the line below.

Default keystroke: **RightArrow**

cursor_up

Moves the cursor up one line.

Default keystroke: **UpArrow**

cut

Removes the selected text, placing it in the clipboard.

If no text is selected, but the editor's Preferences dialog option Cut/Copy Line if No Text is Selected is checked, then this cuts the entire line from the document and copies it to the clipboard.

Same as the Cut command on the Edit menu.

Default keystroke: **Shift+Del**, or **numpad -**

cut_block

Removes the selected text, placing it in the clipboard.

This variation of the cut command has no effect if no text is selected.

Default keystroke: (none)

cut_line

Removes the current line, placing it in the clipboard.

Default keystroke: (none)

delete

Deletes the selected text. If no text is selected, deletes the character at the cursor.

The text can be recovered using the <u>undo</u> function.

Default keystroke: **Del**

delete_line

Deletes the current line.

The line is **not** copied to the clipboard, but can be recovered using the <u>undo</u> function.

Default keystroke: **Alt+D**

delete_to_eol

Deletes all text from the cursor to the end of the current line.

The text can be recovered using the <u>undo</u> function.

Default keystroke: **Alt+K**

delete_word_left

If the cursor is in a <u>word</u>, deletes the text from the cursor to the beginning of the word.

If the character to the left of the cursor is a space or tab, deletes the text from the cursor to the previous non-blank character.

If the character to the left of the cursor is a <u>delimiter</u> other than space or tab, deletes that delimiter.

Default keystroke: **Ctrl+BkSp**

delete_word_right

If the cursor is in a <u>word</u>, deletes the text from the cursor to the end of the word.

If the character to the right of the cursor is a space or tab, deletes the text from the cursor to the next non-blank character.

If the character to the right of the cursor is a <u>delimiter</u> other than space or tab, deletes that delimiter.

Default keystroke: **Ctrl+Del**

document_preferences

Displays the <u>Document Preferences dialog box</u>, which lets you select options for the file in the active window.

Same as the Document Preferences command on the Options menu.

Default keystroke: **F4**

editor_help

Accesses the editor's help system.

Default keystroke: **F1**

end_of_buffer

Moves the cursor to the end of the document in the active window.

Default keystroke: **Ctrl+End**

end_of_line

Moves the cursor to the end of the current line.

Default keystroke: **End**

enter

In insert mode, inserts a carriage return and line feed, then moves the cursor to the beginning of the next line (or positions the cursor below the first non-blank character in the previous line if Auto Indent is checked in the Document Preferences dialog box).

In overwrite mode, simply moves the cursor to the beginning of the next line.

Default keystroke: **Enter**

exit

Prompts you to save any modified files, then closes all windows and exits from the editor.

Same as the **Exit** command on the **File** menu.

Default keystroke: **Alt+F4**

exit_windows

Prompts you to save any modified files, then issues an ExitWindows call to end the current Windows session.   The session is not terminated unless all applications agree to terminate.

Default keystroke: (none)

find

Displays the Find dialog box, which lets you set search criteria for the active document and begins searching.

Same as the Find command on the Search menu.

Default keystroke: **Ctrl+S**

find_again

Continues a search you initiated using the <u>find</u> function.

Same as to the Find Again command on the Search menu.

Default keystroke: **Ctrl+A**

find_files_containing

Displays the Find Files Containing dialog box. Use this to set search criteria and begin searching for files. Use the list_files_containing function to see and open the found files.

Same as the Find Files Containing command on the Search menu.

Default keystroke: **Ctrl+F**

goto_line

Prompts for a line number, then moves the cursor to the specified line in the active document.

If any text is selected, the selection is extended to include the requested line.

Same as the Goto Line command on the Search menu.

Default keystroke: **Ctrl+G**

list_files_containing

Use after using the find_files_containing function to display a list of found files. You can then double click a file to open it and remove it from the list.

Same as the List Found Files command on the Search menu.

Default keystroke: **Ctrl+L**

lowercase

Converts all uppercase characters in the marked block to lowercase.

If no block is marked, acts on the character at the cursor.

Default keystroke: (none)

new_file

Opens a blank untitled document window. Use Save or Save As... on the File menu to give it a name and save the contents to disk.

Same as the New command on the File menu.

Default keystroke: **Alt+F,N**

new_window

Opens an additional window on the current active document.

Changes made to either window affect the same document. If you want two different versions, first, use the Save As... command of the File menu to make a duplicate, then open both files.

Same as the New Window command on the Window menu.

Default keystroke: **Alt+W,N**

next_window

Activates the next window in the editor's circular list.

Same as the Next Window command on the document control-menu.

Default keystroke: **Ctrl+F6**, or **Alt+N**

open_file

Prompts for a directory and filename and opens a window on that file.

Same as the Open command on the File menu.

Default keystroke: **F3**

page_down

Moves the cursor down one page; that is, the number of lines visible in the window.

Default keystroke: **PgDn**

page_up

Moves the cursor up one page; that is, the number of lines visible in the window.

Default keystroke: **PgUp**

paste

Inserts the contents of the clipboard at the cursor location.

Same as the Paste command on the Edit menu.

Default keystroke: **Shift+Ins**

play_macro

Replays the keystrokes and functions recorded by the most recent use of the record_macro function.

If no macro has been recorded, that is indicated on the status line.

Same as the Play Back Macro command on the Edit menu.

Default keystroke: **F8**

prev_window

Activates the previous window on the editor's circular list.

Default keystroke: (none)

print

Prints the active document, using settings set from the Page Setup and Printer Setup commands under the File menu.

Same as the Print command on the File menu.

Default keystroke: **Alt+P**

record_macro

Starts recording keystrokes and editor functions.   Recording continues until the next use of this function.

Same as the Record Macro/Stop Recording Macro commands on the Edit menu.

Default keystroke: **F7**

replace

Prompts for search and replace criteria and then replaces specified text in the active document, starting at the cursor location.

Same as the Replace command on the Search menu.

Default keystroke: **Ctrl+R**

restore_window

Resizes the active window to its user-configurable "intermediate" size.

Same as the Restore command on the <u>document control-menu</u>.

Default keystroke: **Ctrl+F5**, or **Alt+hyphen,R**

revert

Prompts for verification, then undoes all changes to the contents of the active document since the file was last saved.

Same as the Revert command on the File menu.

Default keystroke: **Alt+F,V**

save_all

Saves the contents of all modified document to disk. For untitled windows, prompts for filenames.

Same as the Save All command on the File menu.

Default keystroke: **Alt+F,L**

save_all_exit

Saves the contents of all modified document to disk, then ends the editor session.

Default keystroke: **Alt+X**

save_all_exit_windows

Saves the current contents of all modified documents to disk, then ends the current Windows session. The Windows session is not terminated unless all applications agree to terminate.

Default keystroke: (none)

save_file

Saves the document in the current active window to disk. If the window is untitled, you are prompted for a filename.

Same as the Save command on the File menu.

Default keystroke: **F2**

save_file_close_window

Saves the contents of the current active document to disk, then closes the active window. If the window is untitled, the pcANYWHERE Editor prompts you for a filename.

Default keystroke: (none)

select_all

Selects (highlights) the entire contents of the active document.

Same as the Select All command on the Edit menu.

Default keystroke: **Alt+E,A**

select_char_left

Selects the character to the left of the cursor, or cancels the selection if that character is already selected.

If the cursor is at the beginning of a line, it moves to the end of the previous line.

Default keystroke: **Shift+LeftArrow**

select_char_right

Selects the character to the right of the cursor, or cancels the selection if that character is already selected.

If the cursor is at the end of a line, it moves to the beginning of the next line.

Default keystroke: **Shift+RightArrow**

select_line

Selects the current line.

Default keystroke: (none)

select_line_down

Extends the current selection down one line, or cancels the selection if that line is already selected.

Default keystroke: **Shift+DownArrow**

select_line_up

Extends the current selection up one line, or cancels the selection if that line is already selected.

Default keystroke: **Shift+UpArrow**

select_page_down

Extends the current selection down one page and scrolls the window.

Default keystroke: **Shift+PgDn**

select_page_up

Extends the current selection up one page and scrolls the window.

Default keystroke: **Shift+PgUp**

select_to_bol

Selects the text from the cursor to the beginning of the current line, or cancels the selection if the text is already selected.

Default keystroke: **Shift+Home**

select_to_end

Selects the text from the cursor to the end of the document in the active window.

Default keystroke: **Ctrl+Shift+End**

select_to_eol

Selects the text from the cursor to the end of the current line, or cancels the selection if the text is already selected.

Default keystroke: **Shift+End**

select_to_top

Selects the text from the cursor to the beginning of the document in the active window.

Default keystroke: **Ctrl+Shift+Home**

select_word

Selects the word containing the cursor. (You can also select a word by positioning the cursor on the word and double-clicking the primary mouse button.)

Default keystroke: (none)

select_word_left

Extends the current selection to the beginning of the <u>word</u> to the left of the cursor, or cancels the selection if the text is already selected.

Default keystroke: **Ctrl+Shift+LeftArrow**

select_word_right

Extends the current selection to the beginning of the <u>word</u> to the right of the cursor, or cancels the selection if the text is already selected.

Default keystroke: **Ctrl+Shift+RightArrow**

split_line

Inserts a line break at the cursor without advancing the cursor.

Default keystroke: **Ctrl+N**

stamp

Inserts the current date and time at the cursor location.

Same as the Time/Date command on the Edit menu.

Default keystroke: (none)

tab_right

In insert mode, inserts a tab character at the cursor (or inserts spaces, if Expand Tabs with Spaces in the Document Preferences dialog box is checked).

In overwrite mode, moves the cursor to the next tab position, as set in that same dialog.

Default keystroke: **Tab**

test_batch

Executes the script commands in the active editing window.

Default keystroke: (none)

tile

Resizes and rearranges all open windows to fit within the editor main window with no overlap.

Same as the Tile command on the Window menu.

Default keystroke: **Alt+W,T**

to_bottom

Scrolls the active window to place the current line at the bottom of the window.

Default keystroke: **Ctrl+B**

to_center

Scrolls the active window to place the current line to the center of the window.

Default keystroke: **Ctrl+C**

to_top

Scrolls the active window to place the current line to the top of the window.

Default keystroke: **Ctrl+T**

toggle_backup

Toggles the Make Backup Files option in the Editor Preferences dialog box.

When the backup option is on and you save a file, the original file is retained, but with an extension of .BAK.

Default keystroke: (none)

toggle_insert

Toggles between insert mode and overwrite mode.   The current mode is displayed on the <u>status line</u>.

Default keystroke: **Ins**

toggle_wordwrap

Toggles into and out of word wrap mode.   The current mode is displayed on the <u>status line</u>.

When word wrap is on, typing past the right margin causes the current <u>word</u> to be moved to the next lower line.

Same as the Word Wrap option on the Edit menu.

Default keystroke: **Ctrl+W**

top_of_window

Moves the cursor to the top visible line of the window.

Default keystroke: **Ctrl+PgUp**

undo

Reverses the effects of the a previous editing operation. Repeated use undoes up to 300 edit operations.

Same as the Undo command on the Edit menu.

Default keystroke: **Alt+BkSp**, or **numpad ***

unmark_block

Deselects the selected text and returns the cursor to its location before the text was selected.

Default keystroke: **Esc**

uppercase

Converts all lowercase characters in the marked block to uppercase. If no block is marked, converts the character at the cursor.

Default keystroke: (none)

window_down

Moves the cursor up one line and at the same time moves the text in the window down one line.

Default keystroke: **Ctrl+UpArrow**

window_up

Moves the cursor down one line and at the same time moves the text in the window up one line.

Default keystroke: **Ctrl+DownArrow**

word_left

Moves the cursor to the beginning of the previous <u>word</u> or to the end of the previous line, whichever comes first.

Default keystroke: **Ctrl+LeftArrow**

word_right

Moves the cursor to the beginning of the next <u>word</u> or to the end of the line, whichever comes first.

Default keystroke: **Ctrl+RightArrow**

wrap_para

Reformats the current <u>paragraph</u> within the margins of the active document. Margins are set using the Document Preferences command on the Options menu.

Same as the Wrap Paragraph command on the Edit menu.

Default keystroke: **F12**

zoom_window

Resizes the active document window to its maximum possible size within the editor's window.

Same as the Maximize command on the document control-menu. You can also maximize and restore a window by double-clicking its title bar.

Default keystroke: **Ctrl+F1** or **Alt+hyphen,X**

Definitions

-A-

ASCII
assignment operator

-B-

baud rate
binary operator
bitwise And
bitwise Or
bps
break length
bug

-C-

case sensitivity
clipboard
command
comment
communications protocol
compile
compile time
constant
control-menu box
control menu

-D-

data rate
directive

-E-

equality operator
exclusive Or

-F-

file transfer
flow control

-G-

global variable

-H-

hardware configuration
host

-I-

identifier
integer
integer constant

-K-

key word

-L-

ASCII
Standardized codes used to represent characters, commands and other data, thereby enabling different programs, computer systems and hardware devices to exchange information; stands for American Standard Code for Information Interchange.

assignment operator
An operator that assigns the value on the right side of the equal sign to the variable on the left side of the equal sign. The Norton pcANYWHERE assignment operator is the equal sign (=). The assignment operator can be used with integers or strings. Unless otherwise noted, you can also assign values to the script language's reserved variables.

baud rate
The number of times per second a signal changes in a communications channel. The term baud is often erroneously used to describe the speed at which the modem can transfer data. The correct measure for data rate is bits per second (bps).
A baud can vary in the number of bits it can represent. For example, a 300-baud modem that transmits one bit per baud (300 x 1) is also a 300-bps modem. However, what is often called a 1200-baud modem is really a 300-baud modem that transmits four bits per baud (300 x 4) or 1200-bps. *See also* bps and data rate.

binary operator
An operator having two terms. Norton pcANYWHERE can use binary arithmetic operators, relational operators and logical operators. Compare to <u>unary operator</u>.

bitwise And
Usually used in masking, a bit by bit Anding of two binary values.

bitwise Or
Usually used in masking, a bit by bit Oring of two binary values.

bps
Abbreviation for bits per second, a measure of the speed at which a device such as a modem can transfer data. Also known as data rate. *Compare* <u>baud rate</u>.

break length
In terminal emulation, this is the length of a break signal used to interrupt a program running on a mainframe or minicomputer. This is different than the CTRL-C or CTRL-BRK used by PCs. In pcANYWHERE, you can specify the length of the break signal individually for each terminal emulation you are using.

bug
Any problem that exists within a program.

case sensitivity
Differentiation between uppercase letters and lowercase letters. Case sensitivity means that a string such as "Password" does not match "PASSWORD." On the other hand, case insensitivity means that "Dial Number" matches "DIAL NUMBER," "dial number" or any other combination of upper and lowercase letters.

clipboard

A temporary storage area for cut or copied text or graphics. The clipboard can be used as a holding pen for information you want to pass between Windows programs.

You can place data in the clipboard by using the Cut or Copy command of the Edit menu. Insert data from the clipboard into a document by using the Edit Paste command.

The Windows application named Clipboard Viewer displays the current contents of this system-wide resource.

command
Commands are instructions to Norton pcANYWHERE. Each command has a name consisting of one or more <u>reserved words</u>, followed by zero or more <u>parameters</u>. Parameters provide additional information about how the instruction is to be executed. Depending on the instruction, a parameter may be an integer or string <u>variable</u> or <u>constant</u>, or an arithmetic expression.

comment
A string which is ignored by the Norton pcANYWHERE compiler and is not executed. Use comments in your script files to provide documentation and to serve as reminders during subsequent modifications to the script. A comment is designated by a semicolon (;) that is not within delimiting quote marks. Norton pcANYWHERE considers all text to the right of the semicolon as a comment. Comments can be entered on a line by themselves, or added to the end of any other command line.

communications protocol
A set of rules designed to allow computers to exchange data with one another with as little error as possible. Some protocols, such as <u>RS-232-C</u>, refer to hardware standards; others, such as XMODEM, refer to file transfer protocols. *See also* <u>flow control</u>.

compile
The process of converting a high-level script into a low-level set of commands which can be executed, or run. Syntax errors are discovered when a script is being compiled.
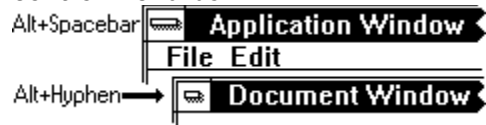
compile time
Refers to the events which occur during the compilation process.

constant
A symbol for a fixed numeric or character value. Norton pcANYWHERE supports integer and string constants.

control-menu box

Click this button to open a menu which provides general control for the active window.

Double-click it to close the window.

Click it once to drop-down the application control-menu which lets you close, resize or move the window.

Keyboard users can press Alt+spacebar to access the control-menu.

For programs which support multiple documents (such as the pcANYWHERE Editor), press Alt+hyphen to drop-down the document control-menu for the active editing window.

control menu

Click the control-menu box (Alt+spacebar) to drop-down the control menu. This menu provides a way for keyboard users to move and resize the window.

data rate
The speed at which information is moved from one location to another; measured in bits per second, or bps. *Compare* <u>baud rate</u>.

directive
A directive is a script file statement which helps Norton pcANYWHERE interpret your intentions.
Directives in the script language are the `integer` and `string` declarations.

equality operator
The symbol testing whether the values on each side of it are equal. Norton pcANYWHERE's equality operator is a double-equal sign (==). *See also* relational operator.

exclusive Or
A logical comparison in which only *one* of the two conditions being compared is true.

file transfer

The process of using communications to move or transmit a file from one computer to another. In communications, a <u>protocol</u> must be agreed upon by sending and receiving computers before file transfer can take place.

flow control

A signal that acknowledges that communication or the transfer of information can take place. When a modem or computer receives data at a faster rate than it can process, data is stored in a   special area of memory called a data buffer. Flow control, also called handshaking, prevents data loss by temporarily halting data transmission when the buffer approaches its capacity.

Handshakes can be controlled by either hardware or software. A hardware handshake, as between a computer and a modem, is an exchange of signals, over specific wires, in which each device signals its readiness to send or receive data. A software handshake, usually exchanged during modem-to-modem types of communication, consists of actual information transmitted   between the sending and receiving devices. A software handshake establishes agreement between devices on the protocols that both will use in communicating. *See also* protocol.

global variable
Global variables are variables whose values are remembered and remain accessible while you are running linked or nested scripts. These variables retain their values for the duration of the script execution. Norton pcANYWHERE contains 127 global variables, named $1 through $127.

hardware configuration
In pcANYWHERE, a set of hardware parameters--such as modem type, port/device, data rate--that is used as a singular named resource in launching a host or remote session.

host
The main computer in a system of computers or terminals connected by communications links. In remote computing, this term refers to the computer that is controlled by the remote computer.

identifier

Identifiers are the names supplied for <u>variables</u>, <u>parameters</u>, <u>labels</u>, and <u>commands</u> in scripts. An identifier is a single letter, or else a sequence of letters and/or digits that begins with a letter. Only the first eight characters of an identifier determine its uniqueness within the script.

integer
Any whole (non-fractional) number, such as 0, 1 or 9275.

integer constant

Any value built from the digits 0 through 9. Unlike string constants, integer constants are not delimited by quote marks. Integer constants in Norton pcANYWHERE can have a value from -2,147,483,648 to 2,147,483,647 and cannot contain any commas or decimal points.

key word

Key words are identifiers that have special meaning to Norton pcANYWHERE and cannot be used as user-defined variable names. Norton pcANYWHERE key words include the titles of all <u>commands,</u> <u>reserved variables</u> and <u>global variables</u>.

label

An identifier that serves as a destination address or marker. In the script language, a label is used with the Goto and Gosub commands. The destination to which the Goto or Gosub command is to transfer control is indicated by a colon following the label name.

LAN
Acronym for local area network; a group of computers and other devices dispersed over a relatively limited area and connected by a communications link that enables any device to interact with any other device on the network.

loop
A portion of a program that gets repeated. The script language uses Goto statements, in conjunction with If...Then statements, to set up loops.

macro

A set of keystrokes and instructions recorded, saved and assigned to a short key code. When the key code is typed, the recorded keystrokes and instructions are executed (played back). Macros can simplify day-to-day operations that otherwise become tedious. For example, a single "macro" keystroke can set up a connection using pcANYWHERE.

macro keys
Key codes assigned to sets of specific instructions. *See also* <u>macro</u>.

minimize and **maximize**

When you click ▼ the Minimize button (Alt+Spacebar,N) located near the top-right corner of a window, that window shrinks into an icon and appears at the bottom of your desktop.
You can double-click the icon to restore it to its previous size.

When you click ▲ the Maximize button (Alt+Spacebar,X) located at the top-right corner of a window, that window instantly expands to fill the screen.

You can click ⬍ the Restore button to restore the window to its previous size.

TIP: You can also maximize or restore a window by double-clicking its title bar.

nest

A nested expression is an expression whose result is used as an operand in another expression. The expressions in the innermost parentheses are evaluated first.

For example, the command

```
IF ((a == b) && (c != d)) THEN GOTO @continue
```

contains two nested expressions: `(a==b)` and `(c!= d)`. These expressions are evaluated first, and their results are used as the operands for the *logical And* (&&) expression.

online service
A computer communications system or service that allows users to dial in for information, messages and files. CompuServe is an example of a commercial online service.

operator

A symbol within a statement that denotes a type of operation. For example, the equal sign (=) is the assignment operator, and the double-equal sign (==) is the equality relational operator.

parameter

Information that is supplied to a command. Parameters follow the command name and are separated by spaces. Some parameters must be specified; others are optional.

parity

In communications, parity refers to an error-checking procedure in which the number of 1's must always be the sameeither even or oddfor each group of bits transmitted without error.

In typical modem-to-modem communications, parity is one of the parameters that must be agreed upon by sending and receiving parties before transmission can take place.

Parity can be computed in any of the following ways:

Even: Adding the data bits and the parity bit yields an even number. If a character with an even number of bits arrives with the parity bit set (to 1), an error must have occurred during transmission.

Odd: Adding the data bits and the parity bit yields an odd number. If a character with an odd number of bits arrives with the parity bit set (to 1), an error must have occurred during transmission.

None: There is no parity bit.

Space: Sometimes a parity bit is used, but it is always set to 0. If a character is received with a parity bit set to 1, an error may have occurred during transmission. Space can also be used to transmit seven-bit characters to a device which is expecting eight-bit characters. Also referred to as *bit trimming*.

Mark: Works the same was as Space, except that the parity bit is always set to 1. Also referred to as *bit forcing*.

password
A security measure used to restrict access to computer systems and sensitive files. A password is a unique string of characters that a user types in as an identification code. The system compares the code against a stored list of authorized passwords and users. If the code is legitimate, the system allows the user access, at whatever security level has been approved for the owner of the password.

port
A hardware location for passing data in and out of a computing device. Computers have ports for connecting peripheral devices, such as the COM (or RS-232-C) ports used to connect modems and printers.

protocol
A set of rules designed to allow computers to exchange data with one another with as little error as possible. *See also* communications protocol.

relational operator
A binary operator for comparing the values on either side of the symbol. For example, in the script language > is the relational operator for "greater than" and == is the equality operator (the relational operator for "equal to"). The script language can use relational operators with either integers or strings.

reserved variable
Reserved variables are variables that have special meaning to Norton pcANYWHERE. They may contain, for example, current parameter settings, results of operations, or the current date and time. You can use reserved variables anywhere that variables of the appropriate type (integer or string) are allowed.

return value
The information returned by a command. Many commands store the return value in the $RESULT reserved variable; others, such as Find First, allow you to specify a user variable to store it.

RS-232-C standard
An industry standard for serial communication connections. Specific lines and signal characteristics are used to control the transmission of serial data between devices.

run time
Refers to the events which occur while the program is actually executing.

script

A standard ASCII test file containing statements of the script language for Norton pcANYWHERE. A script usually consists of instructions expressed using the script language's rules and syntax, combined with simple control structures.

Norton pcANYWHERE source scripts have an extension of .SCR.
Compiled, executable Norton pcANYWHERE scripts have extensions of .SCX.
You can create a Norton pcANYWHERE script using the Norton pcANYWHERE Editor or any other standard ASCII text editor.

scope
The set of rules governing when and how a variable may be accessed. These rules determine when the variable name is valid; that is, when it can be used in a script. Since script files can call other script files to be run, the variables named in the first script may not be accessible to the called script.

statement
A program unit consisting of a command, an assignment operator or a combination of both. A statement is equivalent to a script file line.

string
Any series of characters surrounded by a pair of single or double quotation mark delimiters.

string constant
A value that is delimited by quotation marks.

string variable
A placeholder for text. For example, in the statement `prompt = "Please click Yes, No or Cancel"` the variable named `prompt` is a string variable.

syntax
The set of rules governing the way a command can be used.

syntax error

Syntax errors are detected during <u>compilation</u> and occur when you make a mistake entering a command, such as not enclosing a string in quotes, or specifying the wrong number of parameters. The syntax errors are written to a file with the same source filename and the extension .ERR. You can use the Norton pcANYWHERE Editor to view the .ERR file, make corrections to the script and attempt compilation again.

terminal emulation
The technique of imitating a terminal by using software that conforms to a standard such as the ANSI standard for terminal emulation. pcANYWHERE can make your computer act as if it were a particular type of terminal in order to communicate with another computer, such as a mainframe.

timeout
A predetermined period of time during which a given task must be completed. If the timeout value is reached before or during the execution of the task, the task is canceled. You can configure a pcANYWHERE host to disconnect from a remote computer after a certain amount time has passed without activity.

translation table
A code that allows data to be converted from one form to another.

unary operator
An arithmetic operator having only one term. The script language's unary operator is the minus sign (-). Compare to <u>binary operator</u>.

user variable
User variables are those variables that you define for your own use during the running of a single script. At the end of script execution, the values stored in these variables are forgotten.

variable
Variables identify data which may change during the running of a script or each time a script is run. *See also* global variable, reserved variable and user variable.

wildcard
A symbol that enables multiple matching values to be returned based on a shared feature. The script language has two wildcards: the question mark (?) and the asterisk (*). The question mark stands for any single character, and the asterisk stands for any character string of any length. For example, the file specification *.* would return all files, regardless of their filenames; the file specification *.SC? would return all filenames having a three-character extension beginning with SC (COMPUSRV.SCR, COMPUSRV.SCX, etc.).

wrap
The ability of the program to continue displaying information on a new line or page when the end of that line or page is reached.