

NAME

ee - easy editor

SYNOPSIS

ee [-e] [-i] [-h] [+#] [*file* ...]

ree [-e] [-i] [-h] [+#] [*file* ...]

DESCRIPTION

The command *ee* is a simple screen oriented text editor. It is always in text insertion mode unless there is a prompt at the bottom of the terminal, or a menu present (in a box in the middle of the terminal).

The command *ree* is the same as *ee*, but restricted to editing the named file (no file operations, or shell escapes are allowed).

An editor with similar user-friendly qualities but more features is available and is called *aee*.

For *ee* to work properly, the environment variable **TERM** must be set to indicate the type of terminal being used. For example, for an HP 700/92 terminal, the **TERM** variable should be set to "70092".

See your System Administrator if you need more information.

Options

The following options are available from the command line:

- e Turns off expansion of tab character to spaces.
- i Turns off display of information window at top of terminal.
- h Turns off highlighting of borders of windows and menus (improves performance on some terminals).
- +# Moves the cursor to line '#' at startup.

Control keys

To do anything other than insert text, the user must use the control keys (the **Control** key, represented by a "^", pressed in conjunction with an alphabetic key, e.g., ^a) and function keys available on the keyboard (such as **Next Page**, **Prev Page**, arrow keys, etc.).

Since not all terminals have function keys, *ee* has the basic cursor movement functions assigned to control keys as well as more intuitive keys on the keyboard when available. For instance, to move the cursor up, the user can use the up arrow key, or ^u.

^a	Prompt for the decimal value of a character to insert.
^b	Move to the bottom of the text.
^c	Get the prompt for a command.
^d	Move the cursor down.
^e	Prompt for the string to search for.
^f	Undelete the last deleted character.
^g	Move to the beginning of the line.
^h	Backspace.
^i	Tab.
^j	Insert a newline.
^k	Delete the character the cursor is sitting on.
^l	Move the cursor left.
^m	Insert a newline.
^n	Move to the next page.
^o	Move to the end of the line.
^p	Move to the previous page.
^r	Move the cursor to the right.
^t	Move to the top of the text.
^u	Move the cursor up.
^v	Undelete the last deleted word.
^w	Delete the word beginning at the cursor position.
^x	Search.
^y	Delete from the cursor position to the end of line.
^z	Undelete the last deleted line.
^[(ESC)	Pop up menu.

EMACS keys mode

Since many shells provide an Emacs mode (for cursor movement and other editing operations), some bindings that may be more useful for people familiar with those bindings have been provided. These are accessible via the **settings** menu, or via the initialization file (see below). The mappings are as follows:

^a	Move to the beginning of the line.
^b	Back 1 character.
^c	Command prompt.
^d	Delete character the cursor is sitting on.
^e	End of line.
^f	Forward 1 character.
^g	Go back 1 page.
^h	Backspace.

^i	Tab.
^j	Undelete last deleted character.
^k	Delete line.
^l	Undelete last deleted line.
^m	Insert a newline.
^n	Move to the next line.
^o	Prompt for the decimal value of a character to insert.
^p	Previous line.
^r	Restore last deleted word.
^t	Move to the top of the text.
^u	Move to the bottom of the text.
^v	Move to the next page.
^w	Delete the word beginning at the cursor position.
^y	Prompt for the string to search for.
^z	Next word.
^[(ESC)	Pop up menu.

Function Keys

Next Page

Move to the next page.

Prev Page

Move to the previous page.

Delete Char

Delete the character the cursor is on.

Delete Line

Delete from the cursor to the end of line.

Insert line

Insert a newline at the cursor position.

Arrow keys

Move the cursor in the direction indicated.

Commands

Some operations require more information than a single keystroke can provide. For the most basic

operations, there is a menu that can be obtained by pressing the **ESC** key. The same operations, and more can be performed by obtaining the command prompt (^c) and typing in one of the commands below.

!cmd

Execute **cmd** in a shell.

0-9 Move to the line indicated.

case

Make searches case sensitive.

character

Display the ascii value of the character at the cursor.

exit Save the edited text, and leave the editor.

expand

Expand tabs to spaces.

file Print the name of the file.

help

Display help screen.

line Display the current line number.

nocase

Make searches insensitive to case (the default).

noexpand

Don't expand tab to spaces when the TAB key is pressed.

quit

Leave the editor without saving changes.

read *file*

Read the named *file*.

write *file*

Write the text to the named *file*.

Menu Operations

Pop-up menus can be obtained by pressing the **escape** key (or **^[** if no **escape** key is present). When in the menu, the escape key can be used to leave the menu without performing any operations. Use the up and down arrow keys, or **^u** for moving up and **^d** for moving down to move to the desired items in the menu, then press **return** to perform the indicated task.

To the left of each menu item is a letter, which if the corresponding letter is pressed on the keyboard selects that menu entry.

The main menu in *ee* is as follows:

leave editor

If changes have been made, the user will get a menu prompting whether or not the changes should be saved.

help

Displays a help screen, with all of the keyboard operations and commands.

file operations

Pops up a menu for selecting whether to read a file, write to a file, or save the current contents of the editor, as well as send the contents of the editor to a print command (see the section **Initializing ee from a file**).

redraw screen

Provides a means to repaint the screen if the screen has been corrupted.

settings

Shows the current values of the operating modes, and right margin. By pressing return when the cursor is on a particular item, the value can be changed. To leave this menu, press the **escape** key. (See **Modes** below.)

search

Pops up a menu in which the user may choose to enter a string to search for, or search for a string already entered.

miscellaneous

Pops up a menu that allows the user to format the current paragraph, execute a shell command, or check the spelling of the text in the editor.

Paragraph Formatting

Paragraphs are defined for *ee* by a block of text bounded by:

- ⊕ Begin or end of file.
- ⊕ Line with no characters, or only spaces and/or tabs.
- ⊕ Line starting with a period ('.') or right angle bracket ('>').

A paragraph may be formatted two ways: explicitly by choosing the **format paragraph** menu item, or by setting *ee* to automatically format paragraphs. The automatic mode may be set via a menu, or via the initialization file.

There are three states for text operation in *ee*: free-form, margins, and automatic formatting.

"Free-form" is best used for things like programming. There are no restrictions on the length of lines, and no formatting takes place.

"Margins" allows the user to type in text without having to worry about going beyond the right margin (the right margin may be set in the **settings** menu, the default is for the margin to be the right edge of the terminal). This is the mode that allows the **format paragraph** menu item to work.

"Automatic formatting" provides word-processor-like behavior. The user may type in text, while *ee* will make sure the entire paragraph fits within the width of the terminal every time the user inserts a space after typing or deleting text. Margin observation must also be enabled in order for automatic formatting to occur.

Modes

Although *ee* is a 'modeless' editor (it is in text insertion mode all the time), there are modes in some of the things it does. These include:

tab expansion

Tabs may be inserted as a single tab character, or replaced with spaces.

case sensitivity

The search operation can be sensitive to whether characters are upper- or lower-case, or ignore case completely.

margins observed

Lines can either be truncated at the right margin, or extend on forever.

auto paragraph formatting

While typing in text, the editor can try to keep it looking reasonably well within the width of the screen.

eightbit characters

Toggles whether eight bit characters are displayed as their value in angle brackets (e.g. "<220>") or as a character.

info window

A window showing the keyboard operations that can be performed can be displayed or not.

emacs keys

Control keys may be given bindings similar to emacs, or not.

16 bit characters

Toggles whether sixteen bit characters are handled as one 16-bit quantity or two 8-bit quantities. This works primarily with the Chinese Big 5 code set.

You may set these modes via the initialization file (see below), or with a menu (see above).

Spell Checking

There are two ways to have the spelling in the text checked from *ee*. One is by the traditional *spell(1)* command, the other is with the optional *ispell(1)* command.

Using *spell*, the words that are not recognized will be placed at the top of the file. For the *ispell* option, the file is written to disk, then *ispell* run on the file, and the file read back in once *ispell* has completed making changes to the file.

Printing the contents of the editor

The user may select a menu item which prints the contents of the editor. *ee* pipes the text in the editor to the command specified by the initialization command **printcommand** (see the section **Initializing ee from a file below**). The default is to send the contents to "lp".

Whatever the user assigns to **printcommand** must take input from standard input. See your system administrator for more details.

Shell operations

Shell commands can be executed from within *ee* by selecting the **shell command** item in the

miscellaneous menu, or by placing an exclamation mark ("!") before the command to execute at the **command:** prompt. Additionally, the user may direct the contents of the edit buffer out to a shell operation (via a pipe) by using the left angle bracket (">"), followed by a "!" and the shell command to execute. The output of a shell operation can also be directed into the edit buffer by using a right angle bracket ("<") before the exclamation mark. These can even be used together to send output to a shell operation and read back the results into the editor. So, if the editor contained a list of words to be sorted, they could be sorted by typing the following at the command prompt:

```
><!sort
```

This would send the contents of the editor to be piped into the *sort* utility and the result would be placed into the edit buffer at the current cursor location. The old information would have to be deleted by the user.

Initializing ee from a file

Since different users have different preferences, *ee* allows some slight configurability. There are three possible locations for an initialization file for *ee*: the file */usr/share/misc/init.ee*, the file *.init.ee* in the user's home directory, or the file *.init.ee* in the current directory (if different from the home directory). This allows system administrators to set some preferences for the users on a system-wide basis (for example, the **print** command), and the user to customize settings for particular directories (like one for correspondence, and a different directory for programming).

The file *usr/share/misc/init.ee* is read first, then *\$HOME/.init.ee*, then *.init.ee*, with the settings specified by the most recent file read taking precedence.

The following items may be entered in the initialization file:

case

Sets searches to be case sensitive.

nocase

Sets searches to be insensitive to case (default).

expand

Causes *ee* to expand tabs to spaces (default).

noexpand

Causes *ee* to insert tabs as a single character.

info

A small information window is displayed at the top of the terminal (default).

noinfo

Turns off the display of the information window.

margins

Causes *ee* to truncate lines at the right margin when the cursor passes beyond the right margin as set by the user while text is being inserted (default).

nomargins

Allows lines to extend beyond the right margin.

autoformat

Causes *ee* to automatically try to format the current paragraph while text insertion is occurring.

noautoformat

Turns off automatic paragraph formatting (default).

printcommand

Allows the setting of the print command (default: "lp").

rightmargin

The user can select a value for the right margin (the first column on the screen is zero).

highlight

Turns on highlighting border of information window and menus (default).

nohighlight

Turns off highlighting of border of information window and menus.

eightbit

Turns on display of eight bit characters.

noeightbit

Turns off display of eight bit characters (they are displayed as their decimal value inside angle brackets, e.g., "<220>").

16bit

Turns on handling of 16-bit characters.

no16bit

Turns off handling of 16-bit characters.

emacs

Turns on emacs key bindings.

noemacs

Turns off emacs key bindings.

Save Editor Configuration

When using this entry from the **settings** menu, the user may choose to save the current configuration of the editor (see **Initializing ee from a file** above) to a file named *.init.ee* in the current directory or the user's home directory. If a file named *.init.ee* already exists, it will be renamed *.init.ee.old*.

CAVEATS

THIS MATERIAL IS PROVIDED "AS IS". THERE ARE NO WARRANTIES OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Neither Hewlett-Packard nor Hugh Mahon shall be liable for errors contained herein, nor for incidental or consequential damages in connection with the furnishing, performance or use of this material. Neither Hewlett-Packard nor Hugh Mahon assumes any responsibility for the use or reliability of this software or documentation. This software and documentation is totally UNSUPPORTED. There is no support contract available. Hewlett-Packard has done NO Quality Assurance on ANY of the program or documentation. You may find the quality of the materials inferior to supported materials.

Always make a copy of files that cannot be easily reproduced before editing. Save files early, and save often.

International Code Set Support

ee supports single-byte character code sets (eight-bit clean), or the Chinese Big-5 code set. (Other multi-byte code sets may function, but the reason Big-5 works is that a two-byte character also takes up two columns on the screen.)

WARNINGS

The automatic paragraph formatting operation may be too slow for slower systems.

FILES

/usr/share/misc/init.ee

\$HOME/.init.ee

.init.ee

AUTHOR

The software *ee* was developed by Hugh Mahon.

This software and documentation contains proprietary information which is protected by copyright.
All rights are reserved.

Copyright (c) 1990, 1991, 1992, 1993, 1995, 1996, 2001 Hugh Mahon.

SEE ALSO

termcap(4), terminfo(4), environ(5), spell(1), ispell(1), lp(1), aee(1)