



JOVE Commands Index

1. INTRODUCTION	3
2. COMMANDS LISTED BY SUBJECT	3
2.1 NAVIGATING THE TEXT	3
2.2 EDITING THE TEXT.....	3
2.3 SETTING A NUMERIC ARGUMENT.....	3
2.4 MANIPULATING BLOCKS OF TEXT	4
2.5 SAVING AND LOADING TEXT	4
2.6 USING MULTIPLE BUFFERS/WINDOWS.....	4
2.7 CREATING AND USING MACROS	4
2.8 BINDING MACROS, COMMANDS, WORD ABBREVIATIONS, ETC	4
2.9 PARAGRAPH FORMAT	5
2.10 SEARCHING THROUGH THE TEXT.....	5
2.11 DISPLAYING INFORMATION	5
2.12 GETTING HELP.....	5
2.13 ADJUSTING SCREEN DISPLAY	5
2.14 CASE.....	5
2.15 NAVIGATING DIRECTORIES	6
2.16 MODES.....	6
2.17 C MODE	6
2.18 LISP MODE	6
2.19 INTERACTIVE PROCESSES	6
2.20 LEAVING JOVE	6
2.21 KEYBOARD FACILITIES.....	6
2.22 INTERACTIVE SHELLS WITHIN JOVE	6
2.23 INSERTING TEXT	6
2.24 MISCELLANEOUS.....	7
3. INDEX OF COMMANDS	8
4. VARIABLES	25

1. Introduction

This document is intended to complement the *Getting Started with JOVE* manual. All the available JOVE commands are listed in this document. In the second section, commands are listed by subject. This section is provided so that a user can swiftly find the names of all commands connected with a specific facet of the editor. Some commands may be listed more than once, under different subject headings.

In the third section of this document, the commands are listed in alphabetical order. Each entry gives a brief description of the command, details of any existing shortcuts, and a list of other relevant commands or variables. In the fourth section, the JOVE variables are listed, with brief descriptions of each, and references to other associated commands or variables.

A more detailed description of each command and variable may be acquired from the JOVE help facility, using "describe-command <command>" or "describe-variable <variable>" (See the section "2.12 Getting Help," below.) This document, however, is intended for quick reference.

For example, a user wishing to find a command with which to save the contents of a buffer would be able to refer to the list of commands under the heading "Saving and Loading" in the second section. Each of these commands could then be hunted out alphabetically in the third section until the user found one that matched his or her requirements.

2. Commands listed by Subject

2.1 Navigating the Text

backward-character	goto-line
backward-paragraph	next-line
backward-word	next-page
beginning-of-file	scroll-lines (variable)
beginning-of-line	scroll-down
beginning-of-window	scroll-left
end-of-file	scroll-right
end-of-line	scroll-step (variable)
end-of-window	scroll-up
exchange-point-and-mark	scroll-width (variable)
first-non-blank	

2.2 Editing the Text

append-region <filename>	kill-s-expression
copy-region	kill-to-beginning-of-sentence
delete-blank-lines	kill-to-end-of-line
delete-next-character	kill-to-end-of-sentence
delete-previous-character	over-write-mode
delete-white-space	recursive-edit
erase-buffer <buffer>	transpose-characters
kill-next-word	transpose-lines
kill-previous-word	yank
kill-region	yank-pop

2.3 Setting a Numeric Argument

digit-n	gather-numeric-argument
---------	-------------------------

2.4 Manipulating Blocks of Text

Where reference is made to a 'selected region,' this should be taken as referring to the area between the 'mark' set by the user, and the point.

append-region <filename>	mark-threshold (variable)
case-region-lower	pop-mark
case-region-upper	replace-in-region <search string>
copy-region	<replacement string>
exchange-point-and-mark	set-mark
fill-region	shift-region-left
filter-region	shift-region-right
highlight-mark (variable)	write-region
kill-region	

2.5 Saving and Loading Text

insert-file <filename>	visit-file <filename>
make-backup-files (variable)	write-file <filename>
make-buffer-unmodified	write-macros-to-file <filename>
save-file	write-modified-files
sync-frequency (variable)	write-region <filename>
tmp-file-pathname (variable)	write-word-abbrev-file <filename>

2.6 Using Multiple Buffers/Windows

delete-buffer <buffer>	page-next-window
delete-current-window	previous-window
delete-other-windows	rename-buffer <new buffer name>
goto-window-with-buffer	select-buffer <buffer name or number>
grow-window	shrink-window
kill-some-buffers	split-current-window
next-window	window-find

2.7 Creating and Using Macros

auto-execute-macro <macro>	execute-macro
<pattern>	local-bind-macro-to-key <macro>
begin-kbd-macro	<key sequence>
bind-macro-to-key <macro>	make-macro-interactive
<key sequence>	name-kbd-macro <macro name>
bind-macro-to-word-abbrev	process-bind-macro-to-key <macro>
<macro> <word abbreviation>	<key sequence>
define-macro	start-remembering
end-kbd-macro	stop-remembering
execute-kbd-macro	write-macros-to-file <filename>

2.8 Binding Macros, Commands, Word Abbreviations, etc

bind-keymap-to-key <keymap> <key sequence>
bind-macro-to-key <macro> <key sequence>
bind-macro-to-word-abbrev <macro> <word abbreviation>
bind-to-key <macro> <key sequence>
define-global-word-abbrev <abbreviation><Return><expansion>
define-mode-word-abbrev <abbreviation><Return><expansion>
describe-bindings
edit-word-abbrevs
local-bind-keymap-to-key <keymap> <key sequence>
local-bind-macro-to-key <macro> <key sequence>

local-bind-to-key *<macro> <key sequence>*
process-bind-keymap-to-key *<keymap> <key sequence>*
process-bind-macro-to-key *<macro> <key sequence>*
process-bind-to-key *<macro> <key sequence>*
unbound

2.9 Paragraph Format

auto-fill-mode
auto-indent-mode
fill-paragraph
left-margin (variable)
left-margin-here
newline
newline-and-backup
newline-and-indent
paragraph-delimiter-pattern (variable)
process-newline
right-margin-here
right-margin (variable)
space-sentence-2 (variable)
tab-width (variable)

2.10 Searching through the Text

backward-list
backward-s-expression
backward-up-list
down-list
i-search-forward *<search string>*
i-search-reverse *<search string>*
match-regular-expressions
(variable)
query-replace-string
<search string> <replacement string>
replace-in-region *<search string>*
<replacement string>
replace-string *<search string>*
<replacement string>
search-exit-char (variable)
search-forward *<search string>*
search-forward-nd *<search string>*
search-reverse *<search string>*
search-reverse-nd *<search string>*
wrap-search (variable)

2.11 Displaying information

buffer-position
current-error
date
dirs
list-buffers
list-processes
next-error
number-lines-in-window
print *<variable>*
pwd
read-word-abbrev-file *<filename>*
string-length

2.12 Getting Help

apropos *<topic>*
describe-bindings
describe-command *<command>*
describe-key *<key sequence>*
describe-variable *<variable>*
spell-buffer

2.13 Adjusting Screen Display

clear-and-redraw
comment-format (variable)
display-filenames-with-bad
-extensions (variable)
highlight-mark (variable)
lc-ctype (variable)
mode-line (variable)
mode-line-should-standout
(variable)
number-lines-in-window
redraw-display
scroll-bar (variable)
visible-bell (variable)
visible-spaces-in-window

2.14 Case

auto-case-abbrev
case-character-capitalize
case-region-upper
case-word-capitalize

case-ignore-search (variable)
case-region-lower

case-word-lower
case-word-upper

2.15 Navigating directories

cd <new directory>
dirs
popd

pushd <directory name>
pushlibd <directory name>

2.16 Modes

auto-fill-mode
auto-indent-mode
file-creation-mode (variable)
fundamental-mode
lisp-mode

over-write-mode
read-only-mode
text-mode
word-abbrev-mode

2.17 C Mode

c-mode
c-argument-indentation (variable)
c-indentation-increment (variable)

fill-comment
handle-tab

2.18 Lisp Mode

add-lisp-special
grind-s-expr

handle-tab
lisp-mode

2.19 Interactive Processes

i-shell-command
interrupt-process
kill-process
 <buffer name or number>
list-processes
process-bind-keymap-to-key
 <keymap> <key sequence>
process-bind-macro-to-key
 <macro> <key sequence>

process-bind-to-key <command>
 <key sequence>
process-dbx-output
process-newline
process-prompt (variable)
 <prompt string>
process-send-data-no-return
quit-process
wrap-process-lines (variable)

2.20 Leaving JOVE

exit-jove

2.21 Keyboard Facilities

meta-key (variable)

2.22 Interactive Shells within JOVE

i-shell-command
shell
shell (variable)
shell-command

shell-command-no-buffer
shell-command-to-buffer
shell-command-with-typeout
shell-flags (variable)

2.23 Inserting Text

character-to-octal-insert
quoted-insert

self-insert

2.24 Miscellaneous

abort-char (variable)
auto-execute-command *<command>*
 <pattern>
compile-it
dbx-format-string
error-format-string (variable)
error-window-size (variable)
expand-environment-variables
 (variable)
files-should-end-with-newline
 (variable)
interrupt-character
 (variable)
one-key-confirmation (variable)
paren-flash
send-typeout-to-buffer (variable)
set *<variable>* *<new value>*
source
tag-file (variable)
update-time-frequency (variable)
write-files-on-make

3. Index of Commands

The first column of the following table contains the command name. To invoke the command using this name, the user should first type <Esc> x to send the pointer to the command line (see the entry for "execute-named-command" in the list below). The shortcuts listed in the second column can be entered without moving to the command line first.

Command	Shortcut	Description	See also
add-lisp-special	None	In lisp (List Processing Language) mode, certain lisp functions such as "defun" and "let" will be specially indented by the editor. This command allows the user to specify other functions that should be indented in the same way.	grind-s-expr handle-tab lisp-mode
append-region <filename>	None	Appends a selected region in the buffer to the specified file. If no file by the specified name exists, one is created, and the text inserted. The user is advised not to use this command, since the results are often unpredictable.	set-mark copy-region write-region
apropos <topic>	None	Provides the user with a list of commands, variables and macros whose names include the specified string. The 'shortcuts' for relevant macros and commands are also displayed, as are the current values of variables.	describe-bindings describe-command describe-key describe-variable
auto-execute-command <command> <pattern>	None	Causes the specified command to execute automatically when a file, whose name matches the given pattern, is loaded.	auto-execute-macro execute-named-command
auto-execute-macro <macro> <pattern>	None	Causes the specified macro to execute automatically when a file, whose name matches the given pattern, is loaded.	auto-execute-command execute-macro execute-kbd-macro
auto-fill-mode	None	Turns "auto-fill mode" on or off in the current buffer. As the default, the editor will operate in "auto-fill mode." In this mode, a new line is started when the last is 'filled,' and a predetermined right margin has been reached. When this mode is turned off, the point does not start a new line until <Enter> is pressed.	auto-indent-mode fill-comment fill-paragraph fill-region right-margin (variable) right-margin-here
auto-indent-mode	None	This command can be used to turn "auto indent mode" on or off in the current buffer. In "auto indent mode," JOVE automatically gives any new line the same indentation as the preceding line.	auto-fill-mode left-margin (variable) left-margin-here
backward-character	<Ctrl B>	Moves point backward one character.	forward-character
backward-list	<Esc>	Searches backward from the point,	forward-list

JOVE Commands Index

	<Ctrl P>	and then moves to the opening bracket of the first bracketed 'list' that is encountered.	
backward-paragraph	<Esc> [Moves point backward to the start of the current paragraph (or the previous paragraph if the point is not initially placed within a paragraph).	forward-paragraph
backward-s-expression	None	Moves the point backward over a Lisp atom or a C identifier, ignoring punctuation or whitespace. The point will halt at the start of this expression. If the point is originally placed directly after a closing bracket - "]", "}" or ")" - then this command acts like "backward-list".	backward-list forward-s-expression
backward-sentence	<Esc> a	Moves point to beginning of current or previous sentence.	forward-sentence
backward-up-list	<Esc> <Ctrl u>	This resembles "backward-list", except that the editor expects the point to be placed within the 'list.' The command searches backward, therefore, for the opening bracket of the list. If such is not found, the message "[Unbalanced Parenthesis]" is displayed.	backward-list
backward-word	<Esc> b	Moves point to the beginning of the current or previous word.	forward-word
begin-kbd-macro	<Ctrl x> (Tells the editor that the ensuing keystrokes are to be 'remembered' and bound into a particular macro. The editor ceases to record the keystrokes in this way when the shortcut for "end-kbd-macro" is typed. (It is necessary to bind shortcut to this command beforehand, since this command cannot be typed directly into the command line.	end-kbd-macro define-macro name-kbd-macro start-remembering For details on how to bind keystrokes to commands, see: bind-to-key
beginning-of-file	<Esc> <	Moves point backward to the start of the buffer. A mark is sometimes set at the point's original position.	end-of-file
beginning-of-line	<Ctrl a>	Moves point to the start of the current line.	end-of-line
beginning-of-window	<Esc> ,	Moves point to the start of the active window.	end-of-window
bind-keymap-to-key <i><keymap> <key sequence></i>	None	Binds a key sequence to a named keymap. This is usually only used to bind another key to the "Esc-map" where the keyboard makes the <Escape> key hard to reach.	bind-macro-to-key bind-macro-to-word-abbrev bind-to-key describe-bindings unbound
bind-macro-to-key <i><macro> <key sequence></i>	None	Binds a key sequence to a named macro.	describe-bindings local-bind-macro-to-key process-bind-macro-to-key unbound

JOVE Commands Index

			write-macros-to-file
bind-macro-to-word-abbrev <macro> <abbreviation>	None	Binds a macro to a previously defined word abbreviation. Subsequently, each time the word is typed, the editor will expand the abbreviation, then execute the macro.	describe-bindings define-global-word-abbrev unbound word-abbrev-mode
bind-to-key <command> <key sequence>	<Esc> <Ctrl b>	Binds a key sequence to an internal JOVE command. When this sequence is entered subsequently, the command will be invoked. This replaces any previous binding upon the key sequence.	describe-bindings local-bind-to-key process-bind-to-key unbound
buffer-position	None	Displays on the status line the current file name, current line number, total number of lines, current character number, total number of characters, percentage of the way through the file, and the position of the cursor in the current line.	scroll-bar
c-mode	None	Turns on the C major mode in the current buffer. In this mode, the <Tab>, "}" and ")" behave differently, in order to ensure the system of indentation appropriate to the mode.	c-argument-indentation (variable) c-indentation-increment (variable) fill-comment handle-tab fundamental-mode lisp-mode text-mode
case-character-capitalise	None	Capitalises the character after the point.	case-word-capitalise
case-region-lower	None	Changes all the upper case letters in the region between the point and the mark to the lower case.	case-region-upper
case-region-upper	None	Changes all the lower case letters in the region between the point and the mark to the upper case.	case-region-lower
case-word-capitalise	<Esc> c	This places the current letter in upper case. The ensuing letters of the word are placed in lower case. The cursor then moves to the end of the word. If the point is not originally placed in a word, the first letter of the ensuing word is capitalised.	case-character-capitalise
case-word-lower	<Esc> l	If the point is placed within the word, this command converts the current letter, and those letters of the word that follow it, into lower case. If the point is placed outside a word, the whole of the next word is converted into lower case. Numerical arguments can be used to specify how many words are affected.	case-region-lower case-word-upper
case-word-upper	<Esc> u	This operates in the same fashion as case-word-lower, except that it converts the relevant text into upper case rather than lower case.	case-region-upper case-word-lower

JOVE Commands Index

cd <directory>	None	Changes current directory.	dirs popd pushd pushlibd
character-to-octal-insert	None	After this command has been entered, typing another character causes a backslash and the ASCII value of this character to be inserted at the point.	quoted-insert self-insert
clear-and-redraw	<Esc> <Ctrl l>	Clears the whole screen, redrawing every window from scratch. This should be used if the screen output has become confused, either through an error in JOVE, or garbage characters from another program.	redraw-and-display
compile-it	<Ctrl X> <Ctrl E>	Compiles the user's program by running the UNIX command "make" into a buffer, automatically parsing any error messages.	parse-errors write-files-on-make (variable)
copy-region	<Esc> w	Copies the region between the point and the mark into the kill buffer. This block can be inserted into a specific location by using the "yank" command.	kill-region yank
current-error	None	Moves to the current error in the list of parsed errors.	next-error parse-errors previous-error
date	None	Displays the date on the status line.	
define-global-word-abbrev <abbreviation> <expansion>	None	This defines a global abbreviation. Henceforth, when the abbreviation is typed, JOVE will automatically replace it with the set expansion, providing that the "word abbreviation mode" is activated.	bind-macro-to-word-abbrev define-mode-word-abbrev word-abbrev-mode
define-macro <macro name> <components of macro>	None	Allows the user to record a macro without using "begin-kbd-macro." After entering the macro name, followed by either a space or newline, then the body of the macro.	begin-kbd-macro end-kbd-macro name-kbd-macro write-macros-to-file
define-mode-word-abbrev <abbreviation> <Return> <expansion>	None	Defines a mode-specific word abbreviation, as opposed to a global word abbreviation. A mode-specific abbreviation defined in C mode, for example, will be expanded by JOVE only in buffers in C mode.	define-global-word-abbrev word-abbrev-mode edit-word-abbrev
delete-blank-lines	<Ctrl x> <Ctrl o>	Deletes all blank lines around the point.	delete-white-space
delete-buffer <filename>	<Ctrl x> k	Deletes the current buffer. The memory associated with it is freed.	erase-buffer kill-some-buffers
delete-current-window	<Ctrl x> d	Deletes the active window. The point is moved into one of the remaining. If the user attempts to delete the only window, a message will notify him of the fact, and the attempt will fail.	delete-other-windows split-current-window
delete-next-character	<Ctrl d>	Deletes the character directly after the point. If the point is at the end of a line, the line separator is removed.	delete-previous-character

JOVE Commands Index

delete-other-windows	<Ctrl x> 1	Deletes all windows other than the active one.	delete-current-window
delete-previous-character	<Ctrl h> or <Delete>	Deletes the character just before the point. If the point is at the start of a line, the neighbouring line separator is removed.	delete-next-character
delete-white-space	<Esc> \	Deletes spaces and tabs around the point. This does not affect whitespace separated from the point by text. If the current character is a space, this too is deleted.	delete-blank-lines visible-spaces-in-window
describe-bindings	None	Lists every bound key sequence in alphabetical order, each opposite the command to which it is bound. This list can be placed in a buffer for printing or easy reference by using the "send-type-out-to-buffer" command.	apropos describe-command describe-key describe-variable send-type-out-to-buffer
describe-command <command>	<Esc> ?	Provides a description of the specified JOVE command. Alternatively, "?" can be typed in place of a command name, to view a list of all commands. The <Spacebar> can be used to scroll through this list.	apropos describe-bindings describe-key describe-variable execute-named-command
describe-key <shortcut>	<Ctrl> x ?	Provides the name of the command that is bound to the specified key sequence.	apropos describe-bindings describe-command describe-variable bind-to-key
describe-variable <variable>	None	Provides a description of a specified JOVE variable. Alternatively, "?" can be typed in place of a variable name, to view a list of all variables. The <Spacebar> can be used to scroll through this list.	apropos describe-bindings describe-command describe-key print set
digit- <i>n</i> (where <i>n</i> is a numeric argument)	<Esc> <i>n</i>	This notifies the editor that the ensuing digits are to be considered a numeric argument, <i>n</i> , for another command. JOVE ceases to read digits as part of the argument when another command is typed. The new command then executes using <i>n</i> as a numeric argument. It is sometimes useful to bind "digit-0" to the "0" key on the numeric keypad, "digit-1" to "1" on the keypad, and so forth.	digit-minus gather-numeric-argument
digit-minus	<Esc> -	This operates in the same way as "digit", except that it defines a negative numeric argument.	digit gather-numeric-argument
dirs	None	Prints out the directory stack.	cd pushd pushlibd popd
down-list	<Esc> <Ctrl d>	Searches forward looking for a 'list.' Such a list comprises the text	backward-list backward-up-list

JOVE Commands Index

		between a matching pair of "(" and ")", "[" and "]" or "{" and "}". The point is placed just after the next opening bracket.	
edit-word-abbrevs	None	Creates a buffer in which all recorded abbreviations are listed opposite their expansions. The user can edit this list. Any new entries should be arranged in the format "abbreviation:expansion." To leave this buffer, use "exit-jove".	bind-macro-to-word-abbrev define-global-word-abbrev word-abbrev-mode
end-kbd-macro	<Ctrl x>) or another user-defined sequence	Ends the definition of a keyboard macro. The "begin-kbd-macro" tells JOVE that the ensuing keystrokes are to be recorded as a macro. The "end-kbd-macro" notifies the editor that the last of these strokes has been entered. The command itself should not be entered at the status line. Instead, the command should be bound to "<Ctrl x>)" or some other shortcut, which can then be used to invoke it.	begin-kbd-macro define-macro name-kbd-macro stop-remembering
end-of-file	<Esc> >	Moves the point to the end of the file. A mark is sometimes left in the original location of the point.	beginning-of-file
end-of-line	<Ctrl e>	Moves the point to the end of the current line. If the line is longer than the screen is wide, it will be scrolled horizontally so that the end of the line is displayed.	beginning-of-line scroll-width scroll-length
end-of-window	<Esc> .	Moves the point to the last character of current window. If there is a numeric argument <i>n</i> , the point will be placed <i>n</i> lines above the end of the window.	beginning-of-window
erase-buffer <buffer>	None	Erases the contents of the buffer. Unlike "delete-buffer" it does not destroy the buffer itself. If there are unsaved modifications, the editor will ask for confirmation.	delete-buffer
exchange-point-and-mark	<Ctrl x> <Ctrl x>	Swaps the position of the point and the mark. This allows the user to move swiftly between two places in the text.	highlight-mark (variable) set-mark pop-mark
execute-kbd-macro	<Ctrl x> e	Causes the execution of the macro that has been defined using "begin-kbd-macro" and "end-kbd-macro".	begin-kbd-macro define-macro end-kbd-macro execute-macro
execute-named-command <command>	<Esc> x	This command moves the cursor down to the status line, so that the name of a command can be entered.	auto-execute-command execute-macro
execute-macro <macro>	None	Executes the specified macro.	execute-kbd-macro auto-execute-macro
exit-jove	<Ctrl x> <Ctrl c>	Exits JOVE. If there are unsaved modifications in any of the open	

JOVE Commands Index

		buffers, the editor will ask for confirmation. If any interactive processes are running, the user will be asked whether these should be terminated.	
fill-comment	None	This command presents C comments within the text in the format dictated by the variable "comment-format". The commented section should be preceded by "/*".	comment-format
fill-paragraph	<Esc> j	Reformats the current paragraph so that each line but the last is 'filled,' and extends as close to the right margin as possible without passing beyond it.	auto-fill-mode fill-region right-margin-here right-margin (variable)
fill-region	None	This command acts in much the same way as "fill-paragraph" but operates upon a selected region rather than a paragraph.	auto-fill-mode fill-paragraph right-margin-here right-margin (variable)
filter-region	None	Passes the text within the selected region to a specified UNIX command. The output from this command replaces the original contents of the region. Thus, areas of text can be reformatted, sorted, etc, using UNIX commands, then returned to the JOVE buffer.	
find-file <filename>	<Ctrl x> <Ctrl f>	Loads the contents of a specified file into a new buffer. The buffer is then selected. If the file has already been loaded into a buffer, this buffer is merely selected. If the file does not yet exist, JOVE will notify the user of this by printing "(New File)" in the mode line. If possible, the buffer is named after the filename.	visit-file window-find
find-tag <tag>	<Ctrl x> <Ctrl t>	This behaves like "find-file" except that it 'finds' the file that contains the specified tag. The editor searches through the "tags" file stored in the current directory.	find-tag-at-point tag-file (variable) window-find
find-tag-at-point <tag>	<Ctrl]>	This command acts like "find-tag" except that JOVE searches for the tag in which the point is positioned.	find-tag tag-file (variable)
first-non-blank	<Esc> m	Moves the point to the first character of the current line that is not whitespace. This may require the point to move backward or forward.	beginning-of-line end-of-line
forward-character	<Ctrl> f	Moves the point forward one character, or over a single line separator.	backward-character
forward-list	<Esc> <Ctrl n>	This command behaves like "backward-list", except that it searches forward until it finds a bracketed list, and moves the point to position of the closing bracket.	backward-list backward-up-list down-list
forward-	<Esc>]	Moves the point to the end of the	backward-

JOVE Commands Index

paragraph		current paragraph. If the point is not initially positioned within a paragraph, it is moved to the end of the next paragraph.	paragraph
forward-s-expression	<Esc> <Ctrl f>	This operates in the same fashion as "backward-s-expression", except that it moves the point forward over the 's-expression.'	backward-s-expression
forward-sentence	<Esc> e	Moves the point to the end of the current sentence. If the point is not initially positioned within a sentence, it is moved to the end of the next sentence.	backward-sentence
forward-word	<Esc> f	Moves the point forward to the end of the current word. If the point is not initially positioned within a word, it is moved to the end of the next word.	backward-word
fundamental-mode	None	Sets the major mode to 'Fundamental.' This is the mode of the 'Minibuf,' and therefore of all that is typed into the status line.	c-mode lisp-mode text-mode
gather-numeric-argument <i>n</i>	<Ctrl u>	This can be used to specify a numeric argument for an ensuing command, so that it executes <i>n</i> times. If no <i>n</i> is specified, the argument will be set to 4. If "gather-numeric-argument" is entered twice without specifying <i>n</i> , the argument is set to 16, and so forth.	digit- <i>n</i>
goto-line	<Esc> g	Moves the point to the start of the specified line. If the argument is positive, the editor will count down from the top to find the line specified by the number. If the argument is negative, JOVE will count up from the end of the buffer.	goto-window-with-buffer next-line number-of-lines-in-window previous-line
goto-window-with-buffer <buffer>	None	This selects the specified buffer. If a window is already displaying the buffer, it becomes the active window.	goto-line select-buffer window-find
grind-s-expr	None	Re-indent a LISP expression, if the point has been placed over a "(".	add-lisp-mode handle-tab lisp-mode
grow-window	<Ctrl x> ^	This increases the active window by one line, providing there is room for this expansion. This command only works if there is more than one window.	shrink-window split-current-window
handle-tab	<Ctrl I>	Provides the appropriate indentation for C and Lisp mode. In Text mode, the ensuing space or newline character is simply inserted into the text.	c-mode lisp mode
i-search-forward	None	Performs an incremental search. This is similar to a search-forward, except that JOVE performs a search after each character of the string has been typed. When the editor has found a match that satisfies the user,	i-search-reverse search-forward search-exit-char (variable)

JOVE Commands Index

		<Return> can be typed to end the search.	
i-search-reverse	None	Performs an incremental search in the same fashion as "i-search-forward", except that it searches backwards through the text.	i-search-forward search-reverse search-exit-char (variable)
i-shell-command	None	Behaves in a similar manner to "shell-command", but also launches an interactive process so that the user can continue editing while the command executes.	shell-command wrap-process- lines (variable)
insert-file <filename>	<Ctrl x> <Ctrl I>	Inserts a specified file into the selected buffer at the position of the point. The point is placed at the start of the inserted material.	append-region visit-file
interrupt- process	None	Sends the signal SIGINT to the interactive process in the selected buffer.	
kill-next-word	<Esc> d	'Kills' the text between the point and the end of the current or next word. Text deleted by this and other 'kill' commands is stored on the 'kill ring,' from which the most recently killed text can be retrieved using the "yank" command.	kill-previous- word yank
kill-previous- word	<Esc> <Delete>	Kills the text between the start of the current or previous word, and the location of the point.	kill-next-word yank
kill-process <buffer>	None	Sends the signal SIGKILL to the process in the specified buffer.	
kill-region	<Ctrl w>	Kills the selected region.	kill-next-word yank
kill-s- expression	<Esc> <Ctrl k>	Kills the text between the position of the point to the end of the "s-expression".	kill-next-word yank
kill-some- buffers	None	Causes the editor to give the user the option of killing each of the buffers, in turn.	kill-next-word yank delete-buffer
kill-to- beginning-of- sentence	<Ctrl x> <Delete>	Kills all text between the start of the current or previous sentence, and the position of the point.	kill-next-word yank
kill-to-end-of- line	<Ctrl k>	Kills all text between the point position and the end of the line.	kill-next-word yank
kill-to-end-of- sentence	<Esc> k	Kills all text between the point position and the end of the current or next sentence.	kill-next-word yank
left-margin-here	None	Sets the left margin variable to the position occupied by the point.	auto-indent-mode left-margin (variable) right-margin-here
lisp-mode	None	Turns on Lisp major mode.	add-lisp-mode c-mode fundamental-mode grind-s-expr handle-tab text-mode
list-buffers	<Ctrl x> <Ctrl b>	Lists each buffer currently open, in each case indicating the buffer number, the number of lines within it,	list-processes

JOVE Commands Index

		the type of buffer, the buffer name, the file associated with it, and whether it contains unsaved changes.	
list-processes	None	Lists all active interactive processes.	list-buffers
local-bind-keymap-to-key <keymap> <key sequence>	None	This operates in a fashion similar to "bind-keymap-to-key". However, this binding only applies in the buffer in which the command is issued.	bind-keymap-to-key describe-bindings unbound
local-bind-macro-to-key <macro> <key sequence>	None	This operates in a fashion similar to "bind-macro-to-key". However, this binding only applies in the buffer in which the command is issued.	bind-macro-to-key describe-bindings unbound
local-bind-to-key <command> <key sequence>	None	This operates in a fashion similar to "bind-to-key". However, this binding only applies in the buffer in which the command is issued.	bind-to-key describe-bindings unbound
make-buffer-unmodified	<Esc> ~	Causes the editor to class the current buffer as 'modified,' even when it has been changed. This does not remove the changes.	write-modified-files
make-macro-interactive	<Esc> I	When a keyboard macro is being defined, this command can be used to allow input to be received in the message line while the macro is executing.	begin-kbd-macro end-kbd-macro execute-kbd-macro name-kbd-macro
name-kbd-macro <macro name>	None	Transforms the keyboard macro into an ordinary macro, with an associated name. Subsequently, the 'keyboard macro' is free to use for a different key sequence.	begin-kbd-macro bind-macro-to-key end-kbd-macro execute-kbd-macro rename-buffer
newline	None	Divides the current line at the point. The text after the point is moved down onto the next line, and the point rests at the start of this.	newline-and-backup
newline-and-backup	<Ctrl o>	Divides the current line in the same fashion as "newline". However, the point is left at the end of the former line, not the start of the new.	newline newline-and-indent
newline-and-indent	<Ctrl j>	Divides the current line in the same fashion as "newline", but indents the new line.	newline newline-and-backup
next-error	<Ctrl x> <Ctrl n>	Moves to the next error in the a list of errors parsed with "parse-errors".	current-error parse-errors previous-error
next-line	<Ctrl n>	Moves the point to the corresponding position on the next line, where possible. If the next line is not long enough to permit this, the point is placed at the end of the line.	goto-line previous-line
next-page	<Ctrl v>	Scrolls down, so that the last line of the current display becomes the first line of the new.	page-next-window previous-page
next-window	<Ctrl x> n	Selects the next window in the editor's circular list.	goto-window-with-buffer page-next-window previous-window
number-lines-in-window	None	Indents the text within the window, and places a line number to the left	goto-line

JOVE Commands Index

		of each line. These numbers are not part of the text, and may be removed by using the command again.	
over-write-mode	None	Turns on Over Write mode in the current buffer. When this mode is activated, typed characters replace existing characters, instead of being inserted before them.	
page-next-window	<Esc> <Ctrl v>	Combines the "next-window" and "next-page" commands. This command selects the next window, then scrolls down to the next page. The original window is then selected.	next-page next-window
paren-flash	(,], })	Inserts the characters bound to this command into the text, using C mode curly brace indentation, Lisp mode parenthesis indentation and Show Match mode paren/curly/brac/square-bracket flashing.	c-mode lisp-mode self-insert
parse-errors	None	Parses the list of C compilation errors in the current buffer. The commands "current-error", "next-error" and "previous-error" are used to navigate this list.	current-error next-error parse-spelling-errors-in-buffer previous-error
parse-spelling-errors-in-buffer <buffer>	None	Parses a list of words in the selected buffer, and looks them up in a specified buffer. This list can then be navigated in the same fashion as that produced by "parse-errors".	current-error next-error parse-errors previous-error spell-buffer
pop-mark	None	Jove can 'remember' as many as eight marks, which are stored on a 'ring.' This command allows the user to move backwards through this ring, until the desired mark is found.	exchange-point-and-mark highlight-mark (variable) set-mark
popd	None	Removes one entry from the directory stack.	cd dirs pushd pushlibd
previous-error	<Ctrl x> <Ctrl p>	Behaves in the same way as "next-error" except that it searches backwards for the last error, and not forward for the next.	current-error next-error parse-errors
previous-line	<Ctrl p>	Moves the point up to the corresponding position on the previous line, where possible. If the previous line is not long enough to permit this, the point is placed at the end of the line.	goto-line next-line
previous-page	<Esc> v	Scrolls up, so that the first line of the current display becomes the last line of the new.	next-page
previous-window	<Ctrl x> o or <Ctrl x> p	Selects the previous window in the editor's circular list.	goto-window-with-buffer next-window
print <variable>	None	Displays the value of the specified variable.	describe-variable set

JOVE Commands Index

process-bind-keymap-to-key <keymap> <key sequence>	None	Behaves in the same fashion as "bind-keymap-to-key". However, the bindings set through this command only take effect in buffers attached to an interactive process.	bind-keymap-to-key describe-bindings local-bind-keymap-to-key unbound
process-bind-to-key <command> <key sequence>	None	This command is similar to "bind-to-key", but the bindings set thus only take effect in buffers attached to an interactive process.	bind-to-key describe-bindings local-bind-to-key unbound
process-dbx-output	None	This command will only have an effect in a buffer in which an interactive shell process is running. If "dbx" is running in a window, this command will cause JOVE to display the file that is being stepped through, in another window.	dbx-format-string
process-newline	<Return> (usually)	This command is used within an interactive process buffer. If it is entered at the last line, the line is terminated and made available as input to the process. If it is elsewhere in the buffer, the line is copied to the end of the buffer, where it can be edited before being sent to the process.	newline process-prompt (variable)
process-send-data-no-return	None	Behaves like "process-newline", except that it simply sends the contents of the line to the process without performing the 'newline.'	process-newline
pushd <directory>	None	Pushes a directory onto the directory stack, and changes to it.	cd dirs popd pushlibd
pushlibd <directory>	None	Behaves like "pushd", except that it pushes the JOVE sharable library onto the directory stack.	cd dirs popd pushd
pwd	None	Displays the pathname of the working directory.	cd dirs
query-replace-string <search string> <Return> <replacement string>	<Esc> q	Searches forward through the text, replacing the first string with the second. Each time it discovers an instance of the first string within the text, the user is given a list of options. Details of these can be found in the Help entry for this command.	match-regular-expressions (variable) replace-in-region replace-string wrap-search (variable)
quit-process	None	Sends the signal SIGQUIT to the interactive process in the selected buffer, so that the process quits.	
quoted-insert	<Ctrl q> or <Ctrl 6>	Allows characters, which would otherwise be bound to JOVE commands, to be inserted into the text. "<Ctrl> x", for example, will be represented on the screen as "^x".	character-to-octal-insert self-insert
read-only-mode	None	Turns on or off the Read-only minor mode. The user is unable to modify	

JOVE Commands Index

		the buffer while Read-only mode is activated.	
read-word-abbrev-file <filename>	None	Reads a specified file containing a list of abbreviations definitions. This command makes the abbreviations available.	word-abbrev-mode
recursive-edit	None	Enters a recursive editing level.	
redraw-display	<Ctrl l>	Scrolls the screen so that the current line is centred vertically within the window. If the line is already in place, the screen is cleared and redrawn.	clear-and-redraw
rename-buffer <new name>	None	Allows the user to rename the selected buffer.	name-kbd-macro
replace-in-region <search string> <Return> <replacement string>	None	Behaves in the same way as "replace-string", except that it only replaces instances of the former string that occur between the point and mark.	match-regular-expressions (variable) query-replace-string replace-string
replace-string <search string> <Return> <replacement string>	<Esc> r	Replaces all occurrences of the first string with the second. The only difference between this and "query-replace-string" is that the latter offers the user options for the treatment of each occurrence of the first string.	match-regular-expressions (variable) query-replace-string replace-in-region wrap-search (variable)
right-margin-here	None	Sets the right margin to the position of the point. The right margin must be further to the right than the left margin.	auto-fill-mode left-margin-here right-margin (variable)
save-file <filename>	<Ctrl x> s, <Ctrl x> <Ctrl s>	Saves the contents of the temporary buffer into the file that is being 'visited.' If the changes are being saved from the MAIN buffer, the user will be prompted for a file name. When the buffer contents have been saved, the editor will display the numbers of lines and characters saved.	make-backup-files (variable) write-file write-modified-files
scroll-down	<Esc> z	Scrolls backward one line.	scroll up
scroll-left	None	Scrolls the active window to the left. If no numeric argument is specified, the window will scroll the number of columns specified by "scroll-width".	scroll-width (variable) scroll-right scroll-all-lines (variable)
scroll-right	None	This resembles "scroll-left", but scrolls the active screen to the right.	scroll-width scroll-left
scroll-up	<Ctrl z>	Scrolls forward one line.	scroll-down
search-forward <string>	<Ctrl s>	Searches forward from the point for a specified string. The point is placed at the end of the first occurrence of this string found.	i-search-forward match-regular-expressions (variable) search-forward-nd search-reverse wrap-search (variable)
search-forward-	None	Acts like the "search-forward"	i-search-forward search-forward

JOVE Commands Index

nd <string>		command, but does not assume or set a default search string. This is useful if the user wishes to leave the existing default search string unchanged.	search-reverse-nd
search-reverse <string>	<Ctrl r>	Searches backward from the point for a specified string. The point is placed at the end of the first occurrence of this string found.	i-search-reverse search-forward search-reverse-nd
search-reverse-nd <string>	None	Acts like the "search-reverse" command, but like "search-forward-nd", does not assume or set a default search string.	i-search-reverse search-reverse search-forward-nd
select-buffer <buffer name or number>	<Ctrl x> b	Selects the specified buffer. Either the name or the number of the buffer should be specified. "select-buffer-1" will select buffer 1, "select-buffer-10" will select buffer 10 if it exists, and so forth.	goto-window-with-buffer window-find
self-insert	Most characters	This inserts into the text the character that invoked the command. Most characters are bound to "self-insert".	character-to-octal-insert paren-flash quoted-insert
set <variable> <new value>	None	Sets the specified variable to a new value.	describe-variable print
set-mark	None	This sets a mark at the current position of the point. The editor can remember as many as eight marks, which are stored on a ring. To change the current mark to the next on the ring, the command "pop-mark" should be used. Commands that affect the 'selected region' affect the text between the point and the current mark.	exchange-point-and-mark highlight-mark (variable) pop-mark
shell	None	This command is meant to start up an interactive shell in a window. If an interactive shell already exists, the relevant buffer is selected. The user is advised not to use this command, since the results are often unpredictable.	wrap-process-lines (variable)
shell-command	<Ctrl x>	Runs a UNIX command, the output of which is placed in a buffer.	wrap-process-lines (variable) shell-command-no-buffer
shell-command-no-buffer	None	Behaves like "shell-command" except that the output is not saved into a buffer. The success of the command will be reported.	shell-command
shell-command-to-buffer <buffer>	None	Behaves like "shell-command" except that it enables the user to specify the buffer into which the output of the shell command will be saved.	shell-command
shell-command-with-typeout	None	Behaves like "shell-command" except that the output from the UNIX command is not saved into a buffer. Instead, it is typed out on the screen,	shell-command

JOVE Commands Index

		in the same fashion as the text displayed after "list-buffers".	
shift-region-left	None	The selected region is moved to the left either by "c-indentation increment", or by the number of columns specified in the numeric argument.	shift-region-right c-indentation-increment (variable)
shift-region-right	None	The selected region is moved to the right either by "c-indentation increment", or by the number of columns specified in the numeric argument.	shift-region-left c-indentation-increment (variable)
shrink-window	None	Reduces the size of the window by one line. Every window must have at least one line of text, and another line as the mode line.	grow-window split-current-window
source <file>	None	This command informs the editor that the contents of a specific file are to be regarded as 'source.' The file should consist of a series of JOVE commands, each placed on a separate line. The "source" command causes the editor to read and execute each of the listed commands.	write-macros-to-file
spell-buffer	None	This command is designed to run the selected buffer through the Unix "spell" program. The output is placed in a buffer named "Spell". The user is advised not to use this command, since the results are often unpredictable.	parse-spelling-errors-in-buffer
split-current-window	<Ctrl x> 2	Divides the current window into the number of windows specified by the numeric argument. If no such argument is supplied, then the window is split into two. The new windows are of equal size.	delete-current-window delete-other-windows
start-remembering	None	This is another name for "begin-kbd-macro".	begin-kbd-macro
stop-remembering	None	This is another name for "end-kbd-macro".	end-kbd-macro
string-length	None	Displays the number of characters in the string in which the point is positioned. A string is bounded by double quotes.	
text-mode	None	Turns the major mode to Text. This influences the way in which JOVE decides what characters form parts of words.	c-mode fundamental-mode lisp-mode
transpose-characters	<Ctrl t>	Reverses the positions of the character before the point with the character after the point.	transpose-lines
transpose-lines	<Ctrl x> <Ctrl t>	Reverses the positions of the current line and the line above it.	transpose-characters
unbound <key sequence>	None	If the user wishes to unbind a key sequence, the sequence should be bound to this command using	bind-keymap-to-key bind-macro-to-key bind-to-key

JOVE Commands Index

		"bind-to-key". Subsequently, if the sequence is typed, the editor will respond as if the sequence is unbound.	describe-bindings
version	None	Displays a number of the version of JOVE.	
visible-spaces-in-window	None	This is a toggle command which changes the way that whitespace is displayed within the current buffer. Spaces are displayed as underscores, and tabs as a 'greater than' sign, followed by spaces.	delete-whitespace
visit-file <file>	<Ctrl x> <Ctrl r>, <Ctrl x> <Ctrl v>	Loads the specified file into the current buffer, replacing any text existing within it. The editor will ask for confirmation before replacing text containing unsaved modifications.	insert-file find-file
window-find	<Ctrl x> 4	Finds and selects another buffer in another window. The way in which this is achieved depends upon the characters that succeed the command, as shown below.	goto-window-with-buffer next-window previous-window
window-find t <tag>	See above	Finds a window by seeking the specified tag. The window is selected.	find-tag
window-find <Ctrl t> <tag>	See above	Finds a window by seeking a specified tag at point. The window is selected.	find-tag
window-find f <file>	See above	This either selects a window displaying the specified file, or loads the file, in the style of "find-file".	find-file
window-find b <buffer>	See above	Selects the specified buffer in the style of "select-buffer".	select-buffer
word-abbrev-mode	None	Turns on or off the Word Abbrev minor mode in the current buffer. The editor maintains a list of word abbreviations, associated with specific expansions. When Word Abbrev mode is activated, JOVE will automatically replace these abbreviations in the text with the appropriate expansions.	
write-file <file>	<Ctrl x> <Ctrl w>	Saves the contents of the selected buffer to the specified file. If the file already exists, JOVE will ask for confirmation.	make-backup-files (variable) save-file write-file
write-macros-to-file <file>	None	Saves the currently defined macros to the specified file. This file is formatted so that its contents may be read back using the "source" command.	source write-word-abbrevs-to-file
write-modified-files	<Ctrl x> <Ctrl m>	Saves all buffers that have been modified. If a numeric argument is supplied, the editor will ask for confirmation before saving each buffer.	make-backup-files (variable) make-buffer-unmodified save-file write-file
write-region <file>	None	Saves the selected region into the specified file. If the file already	

JOVE Commands Index

		exists, JOVE will ask for confirmation before saving the region.	
write-word-abbrev-file <filename>	None	Saves all currently defined abbreviations to the specified file. This can be viewed and edited by using "read-word-abbrev-file".	read-word-abbrev-file word-abbrev-mode write-macros-to-file
yank	<Ctrl y>	This inserts into the buffer at point the 'current' item on the killed ring. Where several kills have been performed in a row, "yank" will recall all at once.	yank-pop
yank-pop	<Esc> y	Rotates the 'kill ring' so that a different item is at the front of the ring. This can be used to "yank" back one of the last sixteen kills. This command can only be used after a "yank" or another "yank-pop" command. At present, the user is advised to use only the shortcut for "yank-pop", since the command may not operate if typed at the command line.	yank

4. Variables

Below is a complete list of JOVE variables. The setting of each of these may be adjusted through the use of a command in the following format:

```
set <variable> <new value>
```

Further information on a variable may be found by typing

```
describe-variable <variable>
```

at the command line. To discover the current value of a variable, the user should type the following:

```
print <variable>
```

Further details of the commands "set", "print" and "describe-variable" may be found in the above list.

Variable	Description	Default setting	See also
abort-char	When the 'abort character' is typed, JOVE abandons its current task, printing the message "[Aborted]" to the status line.	<Ctrl G>	interrupt-character (variable)
allow-^S-and-^Q	On a machine which is controlled via a tty, <Ctrl Q> may be used to start scrolling, and <Ctrl S> to stop scrolling. If this variable is set, <Ctrl S> and <Ctrl Q> are treated as normal characters. If it is not set, the tty driver will continue to use them for flow control.	on	quoted-insert search-forward
allow-bad-characters-in-filenames	When this variable is set, filenames may be created which contain 'bad' characters such as "*", "&", "%", "!", "[", "]", etc.	off	
auto-case-abbrev	If this variable is "on", then the case of word abbreviations is adjusted automatically. If the abbreviation is typed with no uppercase letter, the standard expansion is not altered. If one or more uppercase letters are included in the abbreviation, the first character in the expansion is capitalised, and each letter in the expansion immediately preceded by whitespace or "-" is capitalised. When this variable is "off", the editor will differentiate between abbreviations that share the same letter but differ in case. For example, "JOVE" and "Jove" must be separately defined as representing particular expansions.	on	word-abbrev-mode

JOVE Commands Index

bad-filename-extensions	This variable contains a list of 'bad filename extensions,' separated by spaces. These will not be included in filename completion. As a default, these include ".o"	See help file	display-filenames-with-bad-extensions
c-argument-indentation	The value of this variable indicates the number of characters by which lines that are part of nested expressions in C will be indented.	-1	c-indentation-increment (variable)
c-indentation-increment	This variable can be used to define a series of tabstops independent of the value of the "tab-width" variable. This variable will only be used when in C mode.	8	c-argument-indentation (variable)
case-ignore-search	If this variable is 'on,' then the editor will treat ignore the case of strings while performing searches. For example, the string "Fred" will be treated as matching with "fred", "FRED" and "fReD".	off	research-string search-forward
comment-format	This variable dictates the manner in which JOVE formats 'commented' sections of text when the command "fill-comment".	C format	fill-comment
display-filenames-with-bad-extensions	When this variable is set to 'on,' an "!" will appear before the names of any files that have 'bad extensions.' The variable "bad-filename-extension" defines which extensions are classed as 'bad.'	on	bad-filename-extension
error-format-string	When the command "parse-errors" is used to find the error messages within a buffer, this is achieved by performing a search using a string in the form of a regular expression. This string is stored in the variable "error-format-string."	See help file	parse-errors
error-window-size	The value in this variable gives the percentage of the screen to be occupied by the error-window. When commands such as "compile-it" or "spell-buffer", an error window of this size will be formed.	20%	compile-it spell-buffer grow-window shrink-window
expand-environment-variables	When this variable is on, JOVE will automatically expand strings of the form "\$var" into the value of the environment variable "var".	on	
file-creation-mode	The octal value in this variable indicates the mode that should be used while creating files. This mode is affected by the current umask setting.	0666 or 0644	
files-should-end-with-newline	When this variable is 'on,' then JOVE will automatically place a newline at the end of each file.	on	
highlight-mark	When this variable is 'on,' the location of the current mark is indicated by means of an underscore.	off	exchange-point-and-mark pop-mark set-mark

JOVE Commands Index

interrupt-character	The character specified by this variable may be used to interrupt non-interactive processes, by sending a SIGINT signal from the tty driver. Non-interactive process will be interrupted. If no such process is running, this character will crash JOVE.	<Ctrl]>	abort-char (variable)
lc-ctype	This variable is a string which specifies how non-ASCII characters are displayed, and which characters the editor should regard as printable, lower-case, and so forth.	Empty string	
left-margin	This variable is set to the number of columns by which lines are indented when Auto Indent mode is turned on.	0	auto-indent-mode left-margin-here right-margin (variable)
make-backup-files	When this variable is 'on,' the editor will preserve the old version of the file as "#<filename>~" each time a modified file is saved,	off	save-file write-file write-modified-files
mark-threshold	When executing a command, the editor may only move through a certain number of lines before a mark is set, to allow quick return. This variable allows the user to specify this number of lines.	22	set-mark
match-regular-expressions	When this variable is 'on,' the editor will match regular expressions in search patterns.	off	wrap-search (variable)
meta-key	If the terminal has a real Meta key which forces the eighth bit of each character, this variable should be set to 'on.'	off	
mode-line	This allows the user to determine the format of the mode line. Further details can be found from JOVE's help facility.	See help file	mode-line-should-stand-out (variable) update-time-frequency
mode-line-should-standout	If this variable is set, and if the terminal can support it, the text of the mode line is displayed in reverse video.	on	mode-line (variable)
one-key-confirmation	JOVE will sometimes ask the user for confirmation of certain commands, e.g. when asked to leave JOVE without saving unmodified files. When this variable is set, the user can respond to these queries with "y" or "n" instead of typing out a prefix of "yes" or "no", followed by a return.	off	
paragraph-delimiter-pattern	This variable specifies the pattern that JOVE will interpret as marking the start of a new paragraph.	See help file	
process-prompt	This variable is the string that appears as a prompt in the shell created by the "shell" and "i-shell-command" processes.	%	i-shell-command shell
right-margin	This variable is set to position of the right hand margin that will restrict lines when Auto Fill mode is used.	78	auto-indent-mode left-margin (variable) right-margin-here

JOVE Commands Index

scroll-all-lines	If this is 'on,' then the editor will apply horizontal scrolling commands to the whole window. If it is 'off' such commands will only affect the current line.	off	
scroll-bar	When this variable is turned on, a section of the mode line will be displayed out of reverse video. In this display, the bar represents the text of the buffer. The section out of reverse video represents the text displayed in the window. Thus the user can see what fraction of the buffer is displayed on the screen, and how far through the buffer this text lies.	off	buffer-position mode-line (variable)
scroll-step	This variable specifies how many lines should be scrolled when the point is moved beyond the bounds of the window by a "previous-line" or "next-line" command.	0	next-command previous-command
scroll-width	When a line is too long to be displayed on the screen in its entirety, JOVE allows the user to scroll horizontally. When the point is moved along the line beyond the bounds of the screen, the text of the line will scroll in the direction, so as to keep the point on the screen. The "scroll-width" variable determines how far the screen scrolls, measured in characters.	10	auto-fill-mode
search-exit-char	This variable is set to a character, which can then be typed at any point to exit incremental search.	<Ctrl j>	i-search-forward i-search-reverse
send-typeout-to-buffer	In response to some commands, such as "describe..." commands, JOVE will output text that temporarily overwrites the text on the screen. If this variable is set, this output is written into a buffer instead.	off	
shell	This specifies the shell to be used with commands such as "shell-command", "shell-command-to-buffer" and so forth. If the SHELL environment variable is set, this is the default value of the variable "shell" Otherwise, the default is "/bin/csh".	/bin/ csh	shell-command shell-command-no-buffer shell-command-to-buffer shell-command-with-typeout
shell-flags	JOVE requires a flag argument to inform the shell when the next argument will be a command to be executed. The variable "shell-flag" specifies this flag.	-c	
space-sentence-2	If a 'fill' command is executed when this variable is 'on,' the text will be formatted so that two spaces are left after the end of each sentence. If the variable is 'off,' one space will be left after each sentence.	on	auto-fill-mode fill-paragraph fill-region

JOVE Commands Index

sync-frequency	This variable specifies how many modifications must be made before temporary files are forced out to disk.	50	tmp-file-pathname
tab-width	When the <Tab> key is pressed, the point is moved to the next multiple of the value in this variable.	8	
tag-file	This variable is set to the name of the file in which the editor needs to search for tag definitions.	tags	find-tag
tmp-file-pathname	This variable specifies where JOVE stores temporary files.	/tmp	sync-frequency
update-time-frequency	This variable specifies the interval in seconds how often the mode line, and in particular the clock, is updated. The interval between updates is defined in seconds by the variable.	30	mode-line (variable)
visible-bell	If the terminal has a visible bell, and the variable is 'on,' JOVE will display the bell instead of beeping.	off	
wrap-process-lines	When this variable is 'on,' process output is formatted so that it wraps just before the number of characters in the line equals the number of columns on the screen.	off	i-shell-command
wrap-search	When this variable is 'on,' searches that reach the end of the buffer will wrap around to the beginning again, and stop only at the place where the search began. When it is 'off,' searches do not proceed beyond the end of the buffer.	off	match-regular-expressions (variable) replace-string search-forward
write-files-on-make	If the "compile-it" command is executed when this variable is 'on,' all modified files are written out before "make" is called.	on	compile-it

© Tao Group Ltd or Tao Systems Ltd. 2000, 2001. All Rights Reserved.

Copyright in the software either belongs to Tao Group Ltd or Tao Systems Ltd. The software may not be used, sold, licensed, transferred, copied or reproduced in whole or in part or in any manner or form other than in accordance with the licence agreement provided with the software or otherwise without the prior written consent of either Tao Group Ltd or Tao Systems Ltd.

No part of this publication may be reproduced in any material form (including photocopying or storing it in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication) without the written permission of the copyright owner.

Elate®, intent® and the Tao logo are registered trademarks of Tao Group Ltd.

Digital Heaven™ is a trademark of Tao Group Ltd.

The rights of third party trademark owners are acknowledged.