VI Editor Reference Card

Command Line Options	
vi *filename*	Enter vi, reads in *filename*
vi -r *filename*	Recover file after crash
vi -R *filename*	Edit file in readonly mode

Miscellaneous Commands	
ESC	return to command mode, cancel unexecuted vi command
(use ESC to make sure you are in command mode before using other commands)	
^L	redraw screen
^G	display current file name and current line number
:!*command*	execute one shell *command*
:!csh	spawn csh subprocess
:set all	show all vi variables
:set number	display all line numbers, to left of line
:set nonumber	display all line numbers, to left of line
.	repeat last change
~	toggle capitalization of current character

Saving Text / Exiting	
:q	quit vi
:q!	quit vi, do not save changes
:w	write changes to current file
:wq	write changes to current file and quit
:w *filename*	write changes to *filename*
:e *filename*	edit *filename*
:n	edit next file in filelist

Addressing Lines	
(these can be used as modifiers to other commands to specify active ranges)	
.	current line
^	first line in file
$	last line in file
%	all lines in file
+	one line below current line
-	one line above current line
+#	# lines below current line
-#	# lines above current line
.,$	range from current line end of file
x,*y*	range from line *x* to line *y*
.,+#	range from current line to next # lines
.,'*a*	range from current line to mark *a*
'*a*,'*b*	range from mark *a* to mark *b*

Moving Around	
0	beginning of current line
$	end of current line
w	move forward one word
b	move backward one word
e	move to end of current word
(	move to previous sentence
)	move to next sentence
{	move to previous paragraph
}	move to next paragraph
H	move to top of screen
M	move to middle of screen
L	move to bottom of screen
^U	scroll up $\frac{1}{2}$-screen
^D	scroll up $\frac{1}{2}$-screen
j,↓-key	move down one line (same column)
k,↑-key	move up one line (same column)
h,←-key	move left one character
l,space,→-key	move right one character
#|	move to column # (this is the vertical bar)
m*a*	set mark *a*
'*a*	go to mark *a*
:#	go to line #

Inserting Text

i	insert text before the current character
a	insert text after the current character
A	insert at end of current line
r	replace current character (ESC not necessary)
cw	replace current word
#cw	replace # words starting with the current word
S	replace current line
C	replace from cursor to end of current line
R	same as C but does not change text not typed over
o	open new line below current line
O	open new line above current line
p	place most recent yanked/deleted buffers below the cursor
P	place most recent yanked/deleted buffers above the cursor
u	undo last change
U	counteract changes to current line
"*a*p	place text from buffer *a* below the cursor position
"*a*P	place text from buffer *a* after the cursor position
ESC	Ends the text insertion
:r *filename*	read in *filename* at current position
:r !*command*	read output of shell *command* below current line

Deleting Text

x	delete the current character
#x	delete # characters starting with the current character
dw	delete current word
#dw	delete # words starting with the current word
dd	delete current line
#dd	delete # lines starting with the current line
D	delete from cursor to end of line
:*x*,*y*d	deletes lines *x* through *y*
d'*a*	deletes from current line to mark *a*
"*a*dd	place deleted line into buffer *a*

Copying Text

y	yank the current character
#y	yank # characters starting with the current character
yw	yank current word
#yw	yank # words starting with the current word
yy	yank current line
#yy	yank # lines starting with the current line
D	yank from cursor to end of line
:*x*,*y*y	yanks lines *x* through *y*
y'*a*	yanks from current line to mark *a*
"*a*yy	place yanked line into buffer *a*

Moving Text

>>	shift line right one tab stop
<<	shift line left one tab stop
>'*a*	shift lines from mark *a* to current line right one tab stop
J	join current line to line below it
#J	join # lines, starting with the current line

Searching and Replacing Text

/*string*	search forward for next occurance of *string*
?*string*	search backward for next occurance of *string*
n	repeat last search forwards
N	repeat last search backwards
:g/*string*/*command*	*command* affects lines containing *string*
:v/*string*/*command*	*command* affects lines not containing *string*
:set wrapscan	wrap around file during search
:set nowrapscan	do not wrap around file during search
:set ignorecase	case is ignored during search
:set noignorecase	case is distinguished during search
:set noic	case is distinguished during search (shortcut)

:*x*,*y*s/*oldstring*/*newstring*/*flags*

x,*y*	range for substitution (see addressing lines)
s	substitution command
oldstring	string to replace
newstring	string to substitute
flags	[default] replace first occurrence of string in line
	g - replace all occrences of string
	c - wait for confimation