

# vi cheat sheet

## ***Cursor movements (items below are sometimes called objects)***

- h - left one character
- l - right one character
- j - down one line
- k - up one line
- w - right one word
- b - back one word
- \$ - to the end of line
- 0 - to the beginning of the line
- ) - right one sentence
- ( - left one sentence
- } - right one paragraph
- { - left one paragraph
- Ctrl-F - forward one page
- Ctrl-B - back one page
- G - go to (without arguments, go to end of file)

## ***Deleting***

- d - delete
  - then add one of the cursor movement symbols to show what should be deleted, i.e.:
  - d\$ - delete to end of line
  - d0 - delete to the beginning of the line
  - d} - delete to the end of paragraph
- dd - delete delete (delete the whole line)
- x - delete character cursor is on

## ***Other basic commands***

- r - replace one character
- ZZ - save and exit (hold down shift and press "z" twice)
- y - yank (copy into temporary buffer)
  - then add cursor movement symbol to show what should be copied, for example: y) - copy to the end of sentence
- Y - yank line cursor is on
- p - paste below cursor line (deleted or copied text)
- P - paste above cursor line
- u - undo last editing command
- /sometext - search for "sometext"

## ***Any command can take numeric argument before the name of "object", i.e.***

- 5dd - delete 5 lines beginning with cursor line (or) d5d - same
- 2dw - delete two words (or) d2w - delete two words
- c3w - change 3 words
- 3Ctrl-B - move up three pages
- 1G - go to the first line

## ***External commands can be performed on the selected text (in lines)***

- if command is started with "!", i.e.:
  - !}fmt - reformat paragraph to 72 columns

## **Command line (sometimes called "ex mode")**

:

**From the command line a "set" command can be executed to customize editing environment, i.e.**

```
:set all - will show the state of all options
:set number - will show on the screen numbers of all lines
:set autoindent // obvious
```

**From the command line operations can be performed on the range of lines, i.e.**

```
:18,24 del - delete from line 18 to line 24
:23,48 copy 17 - block from line 23 to 48 copy to line 17
:2,17 move 92 - block from line 2 to 17 move to line 92
```

**From the command line any external UNIX command can be performed on the range of lines if line range is superseded by "!"**

```
:11,16! sed -e "s/^/\*\//" -e "s/$/\*\//"
      (the command above wraps the block of text with
      "C" style comments - /* text */. It can be done
      easier, but this is an example)
:14,19! sort -r +3
      (sort the table in reverse order by fourth column)
```

## **File manipulation from the command line**

```
:r somefile - read in "somefile"
:x - save and exit (if file is "Read Only", this command will
      exit without saving)
:wq - write and quit (same as above)
:w - write (save) if the file permissions allow it
:w! - save file even if it is read-only as long as we own it
:w somefile - save this file as "somefile"
:q - quit without saving
:q! - quit without saving if changes were made
```

## **Text input commands (all require "Esc" to terminate)**

```
i - insert text before the character cursor is on
I - insert text at the beginning of the line
a - append (insert text after the character cursor is on)
A - append text to the end of the line
c - change (replace previous text with new one)
      takes arguments just like the delete command - it is
      a fast and powerful way of changing original text -
      much more so than typical "overwrite"
R - start overwriting text
o - start entering text at the beginning of the new line
      below the cursor
O - start entering text at the beginning of the new line
      above the cursor
```

**If in doubt, press "Esc"**