A VIM Editor Crash Course

VIM is a powerful command line interface (CLI) text editor. It was designed to be used solely with a keyboard, so a lot of commands and actions that we may be accustomed to are very different. Because VIM uses the keyboard for commands and text entry, it needs to be able to distinguish between a command being issued, or text being typed into the document.

Opening VIM

Command	Action Taken
vim	Opens VIM
vim *FILENAME*	Opens *FILENAME* in VIM; creates file if it does not exist

When VIM is first opened, it places the user in EDIT mode. This is the mode that allows users to delete, cut, copy, paste, etc. Some useful commands are shown below. **IMPORTANT: All commands are case sensitive! You will not receive any feedback that you are pressing keys until the command executes.** Patterns can be observed in how commands are utilized along with movement commands.

Keyboard Shortcuts for Editing

Key Strokes	Action Taken
G	Move cursor to last line of the file
gg	Move cursor to the first line of the file
n G	Move cursor to line n
$	Move cursor to end of the current line
0 (zero)	Move cursor to the beginning of the current line
x	Delete single character under the cursor. Can be pressed many times.
dd	Cut/Delete current line
dG	Cut/Delete from current line to the last line
n dd	Including the current line, cut/delete n lines
yy	Yank (copy) current line. NOTE: the variations shown for deletion, also apply to yanking
p	Put (paste) below current line
P	Put (paste) on current line

These commands should demonstrate most of the tasks needed for everyday use. VIM is capable of doing much more than what is being shown in this handout, and an Internet search can show the many capabilities and resources available for learning VIM.

When it comes to actually inputting text, VIM offers many different ways to enter INSERT mode. Only a couple are shown below. You will know if you are in EDIT or INSERT mode by looking in the bottom left corner of the terminal window. If it is empty, you are in EDIT mode. If it shows – INSERT – , then you are in INSERT mode. *If you ever get lost, hit the Escape key twice. This will always ensure you are in EDIT mode.*

Entering INSERT Mode

Key Strokes	Action Taken
i	Insert cursor before current position. This is likely the only method you'll need
o	Insert a new line below the cursor and move cursor there
O	Insert a new line above the cursor and move cursor there

When you are in INSERT mode, you are able to type and use the arrow keys to move around. The `Backspace` key works in INSERT mode, but not EDIT mode. After some typing, if you find that you need to make some edits, press `Escape` to enter EDIT mode, make your edits, and then return to INSERT mode.

To issue specific commands to VIM such as save, quit, search, the user needs to type : followed by the command. This can **only** be done from EDIT mode. The : puts the user into COMMAND mode.

File Commands

Key Strokes	Action Taken
`:w` *FILENAME*	Write/Save to *FILENAME*. This is essentially "Save as…"
`:w`	Write/Save file. This is how you save your work in progress
`:q`	Quit VIM. If you have modified the file in any way, this will not work
`:q!`	Force quit. This will quite without saving any changes
`:wq`	Write/Save file and quit