

Introduction to vi
Blue Waters Petascale Institute 2017
Week 1, Monday (May 29), 7:00pm-7:30pm
Author: Jennifer Houchins, Shodor

An Introduction to vi

In the Unix world, vi is a powerful text editor that was designed to work on a multitude of terminals. It is possible to use vi even when you have no control keys and no cursor movement keys. Since it is a standard editor for Unix, you can typically count on it being readily available on any Unix system.

There are two main modes of operation in vi: 1) **command** mode and 2) **insert** mode. In command mode, most of the keys on the keyboard represent some editing command. In edit mode, the keys insert text into your document.

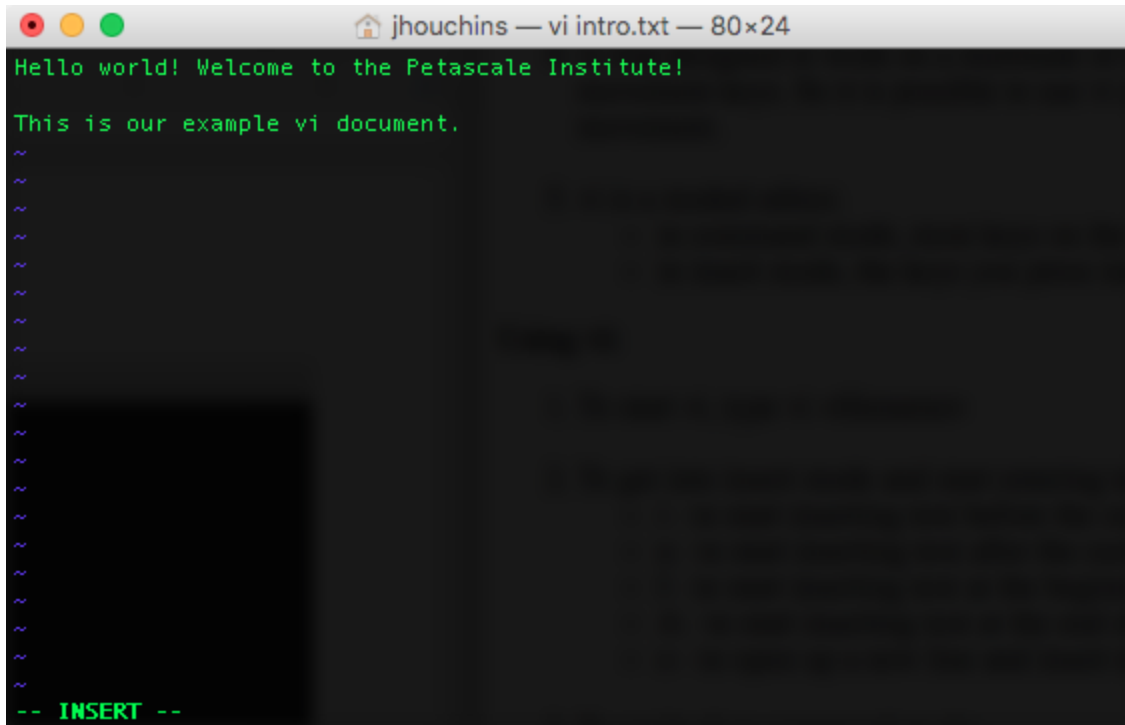
Using vi

Let's get started! First, open up a terminal. (The screen-shots shown in this lesson will be from a Mac OS Terminal, but the commands will be the same if you are using PuTTY on a Windows machine.)

At the command prompt, enter the following command (press enter where it says **<ENTER>**):
vi intro.txt<ENTER>

- ❖ **i** - To start inserting text before the cursor
- ❖ **a** - To start inserting text after the cursor
- ❖ **I** - To start inserting text at the beginning of the current line
- ❖ **A** - To start inserting text at the end of the current line
- ❖ **o** - To open up a new line below and insert text on the new line
- ❖ **O** - To open up a new line above and insert text on the new line

Enter insert mode now, and add some text to your document.

A screenshot of a terminal window with a dark background. The title bar at the top shows a home icon, the username 'jhouchins', the file name '— vi intro.txt', and the window size '80x24'. The terminal content shows two lines of text in green: 'Hello world! Welcome to the Petascale Institute!' and 'This is our example vi document.'. Below these lines are several lines of tilde '~' characters. At the bottom left, the text '-- INSERT --' is displayed in green, indicating the current mode.

To get back to command mode, press **<ESC>**. When in command mode, you can move the cursor around the document in two ways. The most intuitive way is to use the arrow keys. The other way is to use the follow keys:

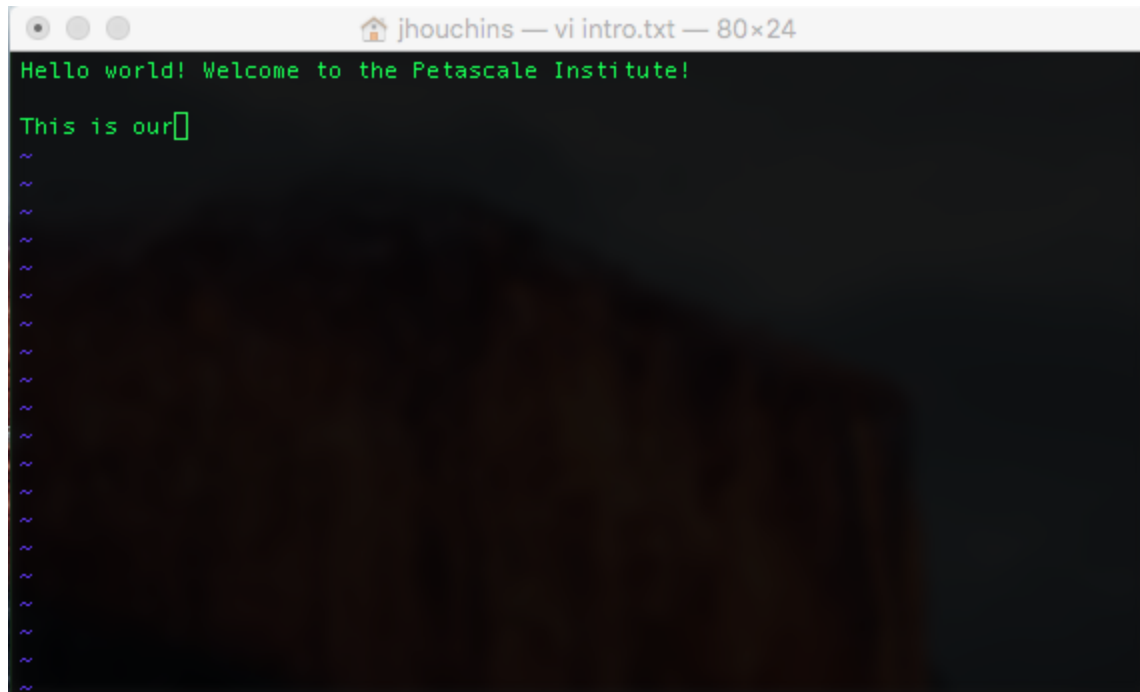
- ❖ Move left using the **h** key
- ❖ Move down using the **j** key
- ❖ Move right using the **l** key
- ❖ Move up using the **k** key

To delete something (in command mode):

- ❖ **x** - deletes the character at the cursor
- ❖ **dd** - deletes the current line

- ❖ **D** - deletes from the current position in the line to the end of the line
- ❖ **dG** - deletes to end of file (includes the current line)

To try out deleting, move your cursor to the start of the word “example” in the last line of our example document and use the **D** command to delete everything to the end of the line. The result should look similar to the following image:



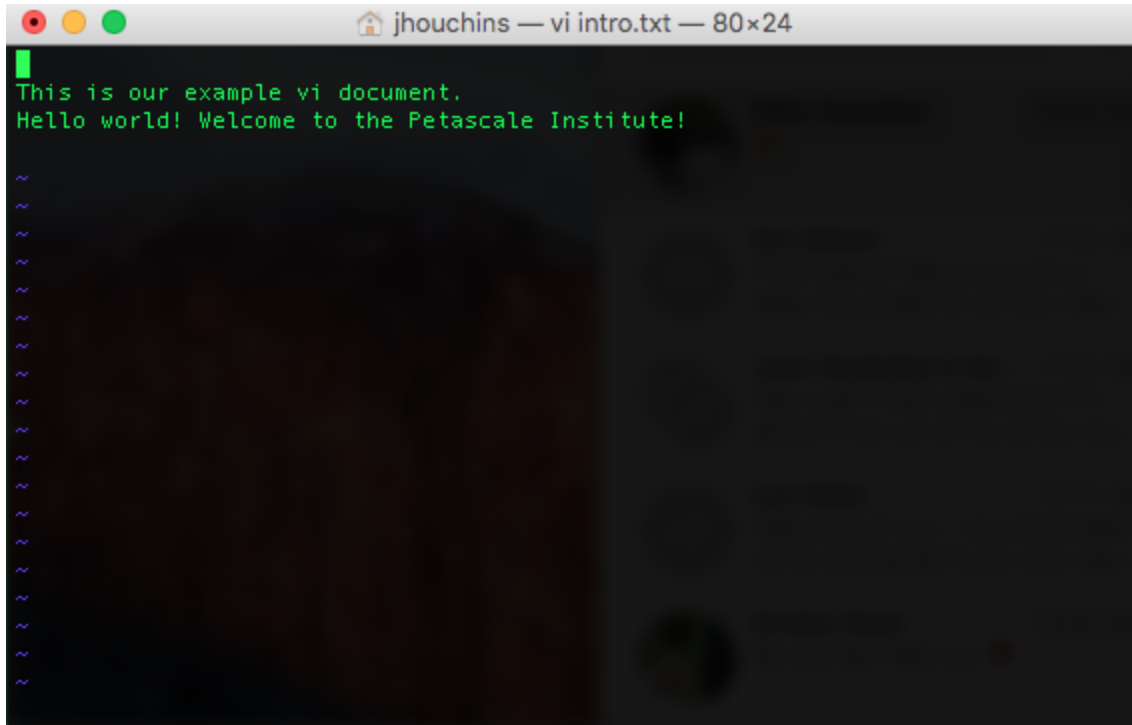
```
jhouchins — vi intro.txt — 80x24
Hello world! Welcome to the Petascale Institute!
This is our
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

What if you didn’t want to delete that after all? Not to worry... just use the **u** key to issue the undo command. This will undo the last command.

Now, we’re going to move things in our document around. To move a line (or a block of lines), you can use the following commands:

- ❖ **dd** - deletes the line from the current position
- ❖ **p** - after moving to the new location; p will put the line(s) after the current line
- ❖ **10dd** - deletes 10 lines starting from the current position (this can be done for any number of lines)

Try moving the last line to the beginning of the document by moving the cursor to that line, pressing **dd**, and then moving the cursor to the beginning of the document and pressing **p**. Remember that the using **p** to move things pastes them after the current line, so you may want to insert a new line at the beginning before you do this.

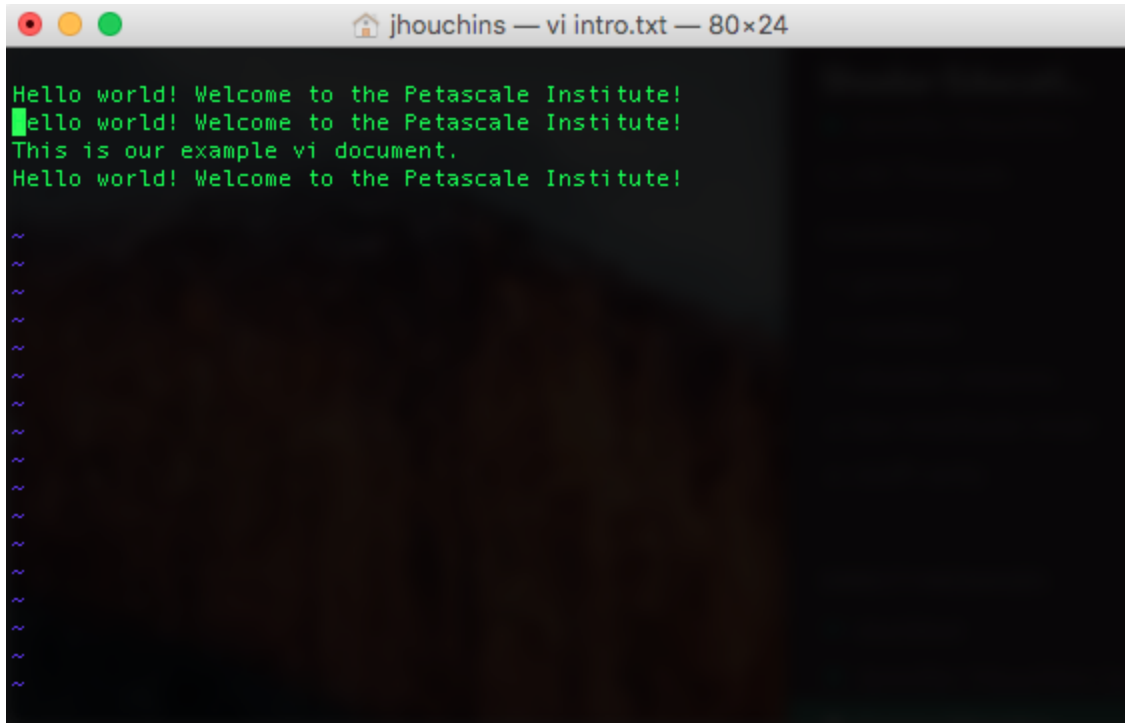


```
jhouchins — vi intro.txt — 80x24
This is our example vi document.
Hello world! Welcome to the Petascale Institute!
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

Copying lines is similar to moving them. The same command is used to paste the copied lines, but to actually copy them you would “yank” the lines you want. That is done with the following command:

- ❖ **yy** - yanks the line from the current cursor position

Try this out by yanking the “Hello World!” line of text and pasting it somewhere else in the document. For example:



When you are done editing and you want to save your work and/or exit vi, the following are ways to do that (note that the colon is part of the commands!):

- ❖ **:w** - save the file without quitting vi
- ❖ **:wq** - save the file and quit vi
- ❖ **:q!** - quit vi without saving
- ❖ **:q** - quit vi (note that if there are unsaved changes, vi won't quit without you saving or adding the exclamation point)

The commands in this lesson are the very basic commands to get you started with vi. You can do most text editing with just these few commands, but there are so many more that could prove useful to you. There are entire books written about vi!

Some other useful commands are provided below, but you can find many more using the built in help. To find the help, open a vi document and type the command **:help<ENTER>** while in command mode. You can even get help for a particular command. For example, open a new vi document (with any name you want), and type **:help zz<ENTER>**. You should see something like this:

```
jhouichins — vi help — 83x25

zz*
zz      Like "z.", but leave the cursor in the same column.
        Careful: If caps-lock is on, this command becomes
        "ZZ": write buffer and exit!  {not in Vi}

*z-*
z-      Redraw, line [count] at bottom of window (default
        cursor line).  Put cursor at first non-blank in the
        line.

*z-*
zb      Like "z-", but leave the cursor in the same column.
        {not in Vi}

=====
4. Scrolling horizontally      *scroll-horizontal*

For the following four commands the cursor follows the screen.  If the
character that the cursor is on is moved off the screen, the cursor is moved
to the closest character that is on the screen.  The value of 'sidescroll' is
scroll.txt [Help] [R0]

~
help
"scroll.txt" [readonly] 332L, 13977C
```

Other frequently used commands:

:set nu : turn on line numbers

gg : jump to the top of a file

G : jump to the bottom of a file

/string<ENTER> : search for a string

n : search for the next occurrence

N : search for the previous occurrence

V : enter **visual** mode, where you can highlight lines and then perform an operation on all of them at once

For more practice you can run **vimtutor**.