

Lab 3: File Editors

Goals

- A. To get comfortable using Vi and Emacs
- B. To create, save, and edit files using an editor
- C. To issue find and replace
- D. To get comfortable with the various commands of each editor
- E. To run a basic script by editing a file and changing its executable bit

Directions

A. Opening Vi

1. Please enter your terminal and type `cd` to go to your home directory and check that you are there with `pwd`.
2. Once inside, make a directory named `scripts`
3. Enter the directory you just created
4. Create a new file named `hello-vi.txt`
5. Type `vi hello-vi.txt` to open the newly created file in the Vi text editor.
6. Congratulations, you've entered a terminal text editor! Now lets begin creating our file. As you may have read or remember from lecture, Vi has 3 modes: insert, command, and ex. Each mode handles a specific set of tasks. In insert mode, you enter text in a way that you are likely familiar from other word processing programs. In command mode, you can perform interesting navigations and edits by composing a series of commands to complete tasks that otherwise might require many many keystrokes in insert mode. In ex mode, you can make large-scale line, file, and editor settings changes. We'll explore each separately below.

B. Insert Mode

1. To enter insert mode, press the i key on the keyboard. You should see the word – INSERT – appear at the bottom left of the screen. In this mode you can type as you please. If you are in Vim (vi Improved) you can use the arrow keys to navigate the document. Otherwise, you can press the escape key to enter the command mode and move around the keys: h, j, k, and l. When finished navigating, to re-enter insert mode, just press i again.

2. Next, type: This is my first sentence typed in Vi!

3. Press Enter to go to the next line. Type: This is my second sentence typed in Vi!

4. Press Enter again and add a long string of the = character: =====

5. Press Enter twice, and then type out the following 3 poems with the proper formatting seen here:
The Sick Rose

O Rose thou art sick.
The invisible worm,
That flies in the night
In the howling storm:

Has found out thy bed
Of crimson joy:
And his dark secret love
Does thy life destroy

By William Blake
=====

Astrophil and Stella 20

Fly, fly, my friends, I have my death wound, fly!
See there that boy, that murd'ring boy, I say,
Who, like a thief, hid in dark bush doth lie
Till bloody bullet get him wrongful prey.
So tyrant he no fitter place could spy,
Nor so fair level in so secret stay,
As that sweet black which veils the heav'nly eye;
There himself with his shot he close doth lay.
Poor passenger, pass now thereby I did,
And stay'd, pleas'd with the prospect of the place,
While that black hue from me the bad guest hid;
But straight I saw motions of lightning grace
And then descried the glist'ring of his dart:
But ere I could fly thence it pierc'd my heart.

By Sir Philip Sidney
=====

In a Station of the Metro

The apparition of these faces in the crowd;
Petals on a wet, black bough.

By Ezra Pound

6. When you are finished entering text. Press the escape key, then type the colon (:), type w, and press Enter. This saves your work.

C. Command Mode

1. In command mode you can issue many special key-press combinations to navigate a text file. You can find a summary of these commands on your review sheet, in your textbook, or in the O'Reilly manual online. To begin, make sure you are open into the file we created in parts A and B and that the file is filled with text. Also press the escape key to make sure you are in command mode.
2. The most important thing to understand about command mode is that all of the commands are composable. Commands can be divided into two kinds for our purposes. There are commands for navigating the document and commands for manipulating text. Each command does one simple task and does it well. Following the Unix philosophy, the strength of the commands isn't in their simplicity and how well they perform their task but rather in how the commands can be chained together to perform complex tasks. A chain of simple commands forms an assembly line for describing a complex action that is composed of simple parts. With only a few basic commands in each category (movement and editing) you can recombine them to form a multitude of novel and powerful commands.

Let's get started.

Movement:

3. Practice moving around the document in the command mode with the keys h, j, k, l. Get used to this motion.
4. You can traverse by words using the keys w and W to go forward and b and B to go backwards. The lowercase commands traverses alphanumeric words (so the command will stop at dashes and apostrophes and other punctuation) while the uppercase commands traverse the document using spaces to delimit where the beginning and end of a word occurs. Practice moving through the document using these keys.
5. To go to the beginning of a line, you can press 0 while in command mode; to go to the end of a line, you can press \$. Practice this now.
6. To traverse the document by paragraph, you can use the keys, { and } which go backwards and forwards through the document by paragraph. Practice.
7. To go to the top of the current screen, type capital H. To go to the middle of the screen type capital M. To go to the end of the current screen, type capital L. Practice moving through the current screen in this way.
8. If you want to travel up and down pages of the document you can press Ctrl+f to move forward through the document by screen or Ctrl+b to move backwards through the document by screen. You may not have enough text to do this right now (you need to have so much text it won't all fit on the screen at the same time). You can practice this command in the next section when you download a novel from the web.
9. To go to the beginning of a document, you can type the number 1 and then press capital G. To go to the end of a document, you can type just capital G alone. Again, the usefulness of this will become more apparent in the larger document in the next section. You can practice now, but will see the benefits much later. You can go to any line of the document by typing any number, say 10 and then the capital G.

Editing

10. In the section B where you were in insert mode and typed out a few poems, you learned how to input text into a file using Vi. In this section you are going to learn how to delete and edit that text via command mode. First, make sure you are in the file and press the escape key to make sure you are in command mode.

11. Before starting, let's make a backup of our document. To temporarily exit Vi press Ctrl+z. From here, type:

```
cp hello-vi.txt hello-vi.txt.bak
```

Remember the cp command copies files. The first parameter is the file to copy and the second is the name of the new file. There is nothing special about the name and you can name it whatever you like so long as you remember what it is called in case you want to use it again. Now that the file is backed up, type fg into the terminal and press enter. This should bring you back to the original file you were editing.

11. Deleting text. There are three basic delete commands: d, dd, and x. Pressing x at the current location in the document will delete whatever character was at that location. Pressing dd will delete the entire line on which your cursor rests. Pressing d and any of the commands for movement from the previous section will delete all text from the current position of the cursor to wherever the movement command was issued. So for instance, if you are at the beginning of the line that starts "Fly, fly, my friends" and you type d} you will delete the entire poem! If you were to instead type dW you would delete the first word. Practice this now. When you are finished, press the key u to undo your deletion. Note how powerful this is, you can perform a deletion action that corresponds to any way you are capable of moving through the document. You can for instance, delete the whole document by pressing dG from the start of the document!

12. You can also prefix any deletion command with a number to perform the command however many times you like. For instance, if you want to delete 3 lines, you could press 3dd if you want to delete 5 words you could press 5dW or 5dB. Practice this and then undo your changes. Notice how creative and powerful this can be. With just a few basic commands you can compose a full vocabulary of complex ones.

13. You can replace characters in a document with r and R. r will replace the current character you are on with whatever single character you type next. R will allow you to replace as many characters as you like before returning to command by pressing escape.

14. If you want to repeat the last command you just performed. You can press the . key. If you want to repeat something several times, you can prefix the . with a number. So to do the previous command 3 times, you would type 3. To do it 50 times, you would type 50. Try it out. Perform a command and then repeat it. Be sure to undo any changes.

15. If you need to redo an undo, you can press Ctrl+R

16. You can search for text in Vi by typing / and entering the string you are looking for. So for instance, to find the word fly, type /fly and press enter. To go to each occurrence, you may press the n key.

17. Vi also has a mode for copy and pasting. Pressing the letter p in command mode will paste whatever the last deleted text was from the buffer. If you want to yank some text to the buffer without deleting it, you can enter a visual highlighting mode from command mode by pressing v. From here, using any movement commands you are comfortable with you can navigate the document and all the text you traverse will be highlighted. Highlight the text you wish to copy and once you are satisfied press y. From here, you can move to any part of the document and press p and the previously highlighted text will be inserted into the document at that location.

Try highlighting Ezra Pound's entire poem and pasting a second copy of it at the end of the document. Save your work by typing :w entering

18. There are a few convenient commands for entering insert mode to save typing. To enter insert mode at the end of a line, type A from command mode. To enter insert mode at the beginning of a line type I. To enter insert mode but on a new line after the current line the cursor is on, type o. To enter a new line before the current line the cursor is on, type O.

19. You are now done with the editing section of the lab. Save your work, then type :q to leave Vi.

D. Manipulating a large text file

1. Exit Vi and enter your scripts directory.
2. Type the following command in bash: `wget http://www.gutenberg.org/cache/epub/2265/pg2265.txt`
3. You should have a file named `pg2265.txt`. Verify this.
4. Rename the file to `Hamlet.txt`
5. Open the file `Hamlet.txt` in Vi
6. Practice traversing and editing the document with the commands from the previous sections.

E. Ex Mode

1. Ex Mode is a line editing program inside Vi. Ex is extremely powerful. You can issue commands that will edit lines at a time in a file in a very programmatic way, automating away what could be manual labor. For the purposes of our class we will only be using it for a few, basic tasks.

2. Find and replace. Inside Hamlet.txt, we will find and replace all occurrence of thou with you.

3. Press escape to enter command mode.

2. First we want to verify that there are several occurrences of thou in the document by typing / thou and pressing enter. Recall that to cycle through the occurrences you press the n key.

3. Once you see many of these, press escape

4. From here type : to enter ex mode

5. In ex mode type: %s/ thou / you

Be sure to leave a leading and trailing space at the beginning and end of thou and also at the beginning and end of you.

This command will look for all occurrences of thou and replace it with you.

6. Verify with the string search from step two that thou no longer occurs in the document.

7. Search for you in the document to see that this word occurs in locations where thou once did

8. Now change all occurrences of Hamlet to Macbeth and review the change. Before starting, verify that Hamlet does indeed occur in the document.

F. Creating an Executable shell script

1. We will cover this topic much more thoroughly in subsequent chapters. For now, we just want to get a glimpse of where we're headed and how powerful the terminal in conjunction with a text editor can be. To illustrate this, we will write our first script.

2. Create a new file named `hello-bash-scripting.sh` and enter it using Vi.

3. Type the following lines:

```
#!/bin/bash
```

```
mkdir Top-Secret  
mkdir ./Top-Secret/A  
mkdir ./Top-Secret/B  
mkdir ./Top-Secret/C  
touch ./Top-Secret/A/the-a-file.txt  
touch ./Top-Secret/B/the-b-file.txt  
touch ./Top-Secret/C/the-c-file.txt  
touch ./Top-Secret/C/the-x-file.txt  
echo "A is for apple" >> ./Top-Secret/A/the-a-file.txt  
echo "Aliens are real" >> ./Top-Secret/C/the-x-file.txt
```

4. Save the work

5. Change the permissions on the file to be 754 (`rwxr-x-r`)

6. Run the file by typing `./hello-bash-scripting.sh`

7. Notice that there is now a `Top-Secret` directory on your machine with all of the files commanded by the script you wrote.

8. Move the file `hello-bash-scripting.sh` into another directory that you have write permissions on and run it again

9. Notice that another `Top-Secret` directory with the corresponding files was created in this directory