CPSC 1020 Lab 2 – Fall 23

DUE: Monday, September 11, 2023, Midnight

SUBMISSION: Canvas.

Learning Vim using vimtutor

You will need access to a terminal for this assignment. I suggest you use the SoC servers or use virtual.computing.clemson.edu to access the server from your laptop. Your final submission will be a pdf. Since you will be working on the school of computing servers for this lab you can use the LibreOffice document editor. After you have completed the document, you can export it to a PDF. For the OS users the terminal that comes with X Code has access to the vimtutor. Feel free to use your Mac.

Introduction:

Many of you, including myself, use a text editor such as Atom or VS Code when writing your programs. That is perfectly fine. However, there will be times when you need to make changes or create a document on some Linux server, such as one of the SoC servers. Many of you find yourselves making the change on your local computer then using some file transfer program or command to transfer the document to a server. This is a time-consuming process. If you know the basics of vim you could simply make these changes directly to the document on Linux. Many students, after learning vi/vim, realize how powerful it is they choose to make it their primary editor when using Linux or MacOS.
Vim has a tutorial available. Each section of the tutorial is labeled. You will go through all sections of this tutorial. At any point that the tutorial has an exercise for you to do, you are **required** to take a screen shot showing you completed the exercise. Most of these are going to be actual changes you will make in the tutorial. **Also, you are required to explain what change you made and the command you used**. If there are no changes or exercises for you to do, type "no changes or exercises needed". Forcing you to explain the change will "hopefully" help you learn the actual command. I have listed each section below.
To access the tutorial, you will need to log onto one of the SoC servers either by ssh or virtual.computing.clemson.edu. If you use virtual.computing you will need to open a terminal. On the terminal command line type vimtutor (no spaces). You should see a screen that says WECOME TO THE VIM TUTOR – Version 1.7. You are going to work through this tutorial. The documentation said this should take ~25-30 minutes. I hope you take your time exploring the commands you will use in this tutorial.

The link below is a video of someone going through this tutorial making his own comments and suggestions. Either while you are completing the tutorial or later, this video is a good one to watch.
https://www.youtube.com/watch?v=d8XtNXutVto

Each section is worth 2.5 points. Please do not leave a lesson blank. 2.5 points will be deducted from your grade if you leave any lessons blank.

Your screen shots and or comments will be placed under the appropriate sections.

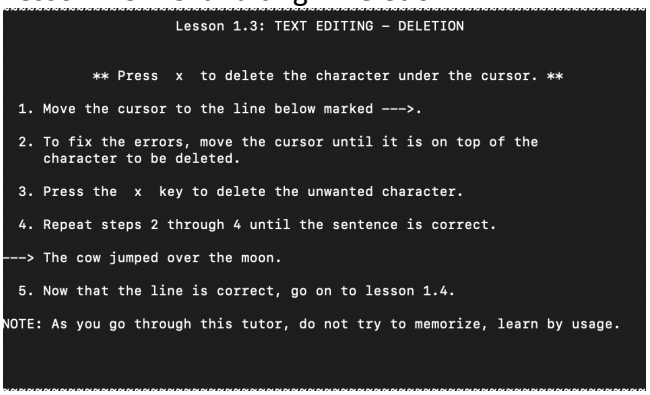
Lesson 1.1 Moving the cursor:

No changes or exercises needed.

Lesson 1.2: Exiting VIM:

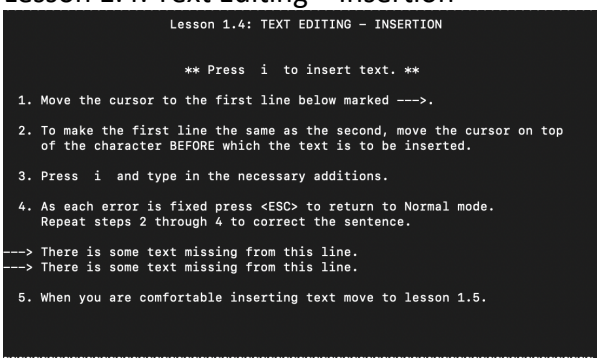
No changes or exercises needed.

Lesson 1.3: Text Editing – Deletion



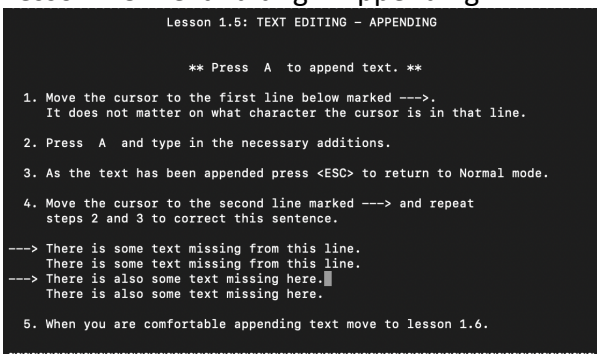
Went through the line with the arrow and deleted several extra letters in the sentence by pressing x while the cursor was over them.

Lesson 1.4: Text Editing – Insertion



Added in the missing words by using the insert command by pressing i while the cursor was before where I wanted to add words.

Lesson 1.5: Text Editing – Appending



Appended the missing letters/words from the sentences by pressing A.

Lesson 1.6: Editing A File

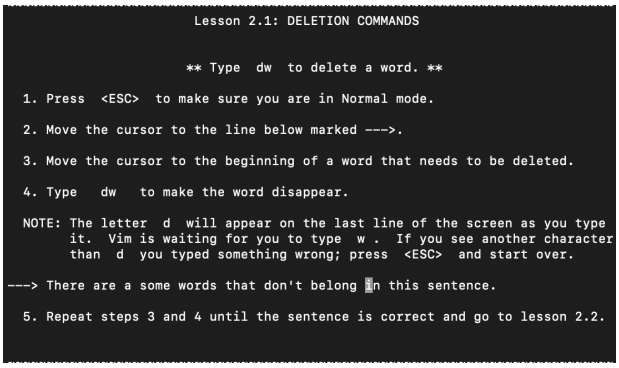
```
tutor
```

Made a file called tutor by calling the vim tutor command and then saved and quit it with :wq.

Lesson 1 Summary:

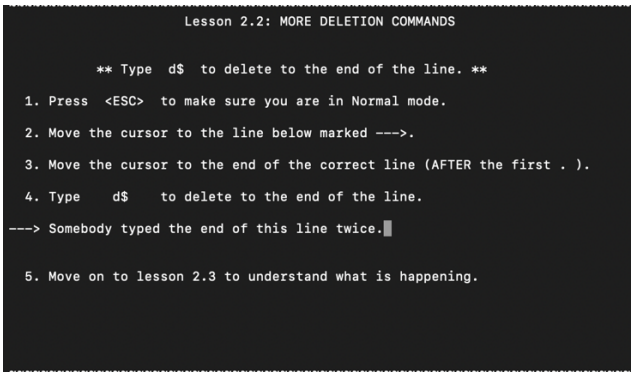
No changes or exercises needed.

Lesson 2.1: Deletion Commands:



I hovered the cursor over the start of the words that did not belong and then entered the command “dw” to delete them.

Lesson 2.2: More Deletion Commands:



Put the cursor before the part of the line that needed to be deleted and the typed the command “d$” to delete the excess.

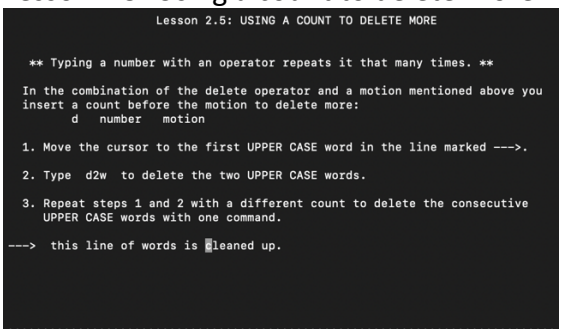
Lesson 2.3: On operators and Motions:

No changes or exercises needed.

Lesson 2.4: Using a count for a Motion:

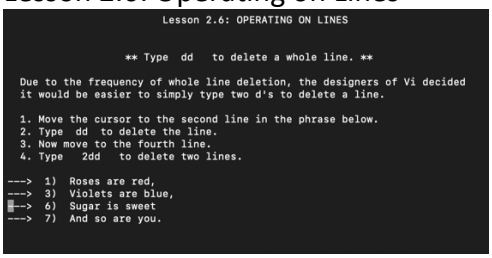
No changes or exercises needed.

Lesson 2.5: Using a count to delete More:



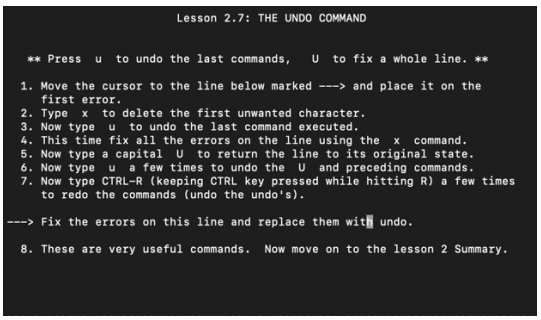
Used the “dw” command but with numbers after the “d” operator so that it deleted multiple words at once.

Lesson 2.6: Operating on Lines



Used the “dd” command to get rid of several unneeded lines and included a number of times for one of the deletions.

Lesson 2.7: The Undo Command

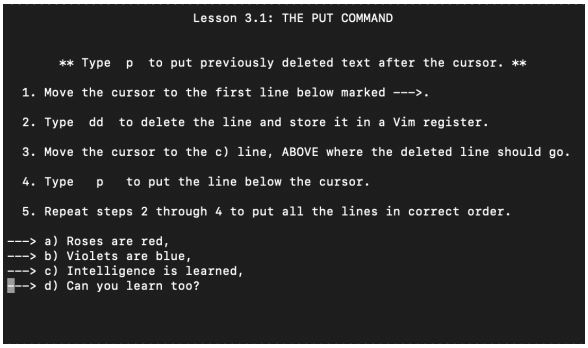


Fixed the errors in the line with the “x” command. Then undid the fixes with the “u”(undo one action) and “U” (undo changes to the entire line). Then used “CTRL R” to undo the undo’s.

Lesson 2: Summary:

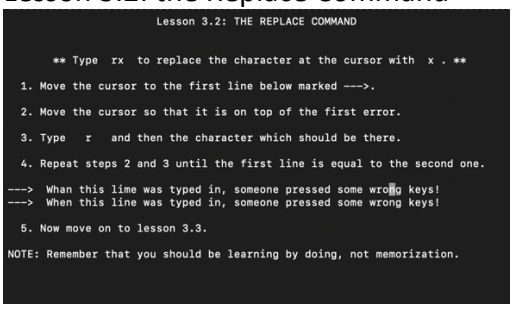
No changes or exercises needed.

Lesson 3.1: the Put Command:



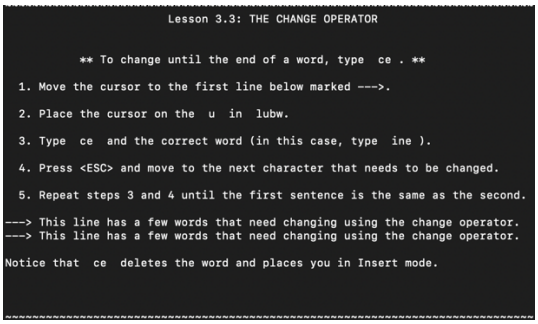
Used a mixture of the “dd” and “p” commands to move the text around so that the poem was in the correct order. Basically a “copy and paste” command but a little different.

Lesson 3.2: the Replace Command



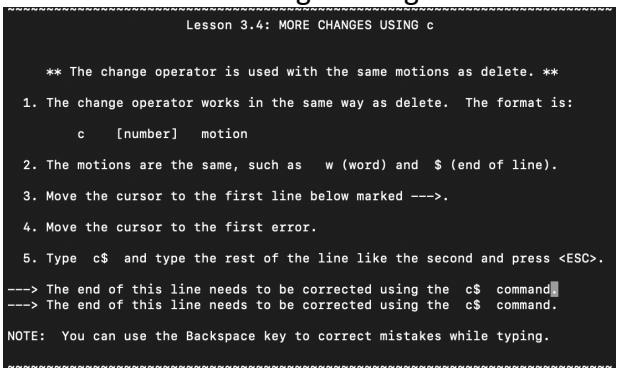
Used the “r” command to replace the incorrect letters in the sentence.

Lesson 3.3: The Change Operator



Used the “ce” command in order to change the words that we not spelled correctly to their correct spelling.

Lesson 3.4: More changes Using C



Changed the end of the first line to be like the second using the “c$” to delete the end of the line that was wrong and then typed in the correct end of line.

Lesson 3: Summary

No changes or exercises required.

Lesson 4.1: Cursor Location and File Status

I moved through the file using CRTL-G and then using commands like “gg” and “G” to move around the file.

Lesson 4.2: The Search Command

```
--->  "errroor" is not the way to spell error;  errroor is an error.
NOTE: When the search reaches the end of the file it will continue at the
      start, unless the 'wrapscan' option has been reset.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                  Lesson 4.3: MATCHING PARENTHESES SEARCH


            ** Type  %  to find a matching ),], or } . **

  1. Place the cursor on any (, [, or { in the line below marked --->.

  2. Now type the  %  character.

  3. The cursor will move to the matching parenthesis or bracket.

  4. Type  %  to move the cursor to the other matching bracket.

  5. Move the cursor to another (,),[,],{ or } and see what  %  does.

---> This ( is a test line with ('s, ['s ] and {'s } in it. ))

NOTE: This is very useful in debugging a program with unmatched parentheses!


~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                    Lesson 4.4: THE SUBSTITUTE COMMAND


       ** Type  :s/old/new/g  to substitute 'new' for 'old'. **

/errroor
```

Used the “/” command to search for the word “errroor” in the document.

Lesson 4.3: Matching Parentheses Search

```
                  Lesson 4.3: MATCHING PARENTHESES SEARCH


             ** Type  %  to find a matching ),], or } . **

  1. Place the cursor on any (, [, or { in the line below marked --->.

  2. Now type the  %  character.

  3. The cursor will move to the matching parenthesis or bracket.

  4. Type  %  to move the cursor to the other matching bracket.

  5. Move the cursor to another (,),[,],{ or } and see what  %  does.

---> This ( is a test line with ('s, ['s ] and {'s } in it. ))

NOTE: This is very useful in debugging a program with unmatched parentheses!
```

Used the “%” command to move the cursor to the matching),], or } in the file.

Lesson 4.4: The substitute Command

```
                    Lesson 4.4: THE SUBSTITUTE COMMAND


       ** Type  :s/old/new/g  to substitute 'new' for 'old'. **

  1. Move the cursor to the line below marked --->.

  2. Type  :s/thee/the <ENTER>  .  Note that this command only changes the
     first occurrence of "thee" in the line.

  3. Now type  :s/thee/the/g .  Adding the  g  flag means to substitute
     globally in the line, change all occurrences of "thee" in the line.

---> the best time to see the flowers is in the spring.

  4. To change every occurrence of a character string between two lines,
     type   :#,#s/old/new/g    where #,# are the line numbers of the range
                               of lines where the substitution is to be done.
     Type   :%s/old/new/g      to change every occurrence in the whole file.
     Type   :%s/old/new/gc     to find every occurrence in the whole file,
                               with a prompt whether to substitute or not.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Substituted “thee” for “the” in the line with the arrow by using the “:s/new/old/g” command.

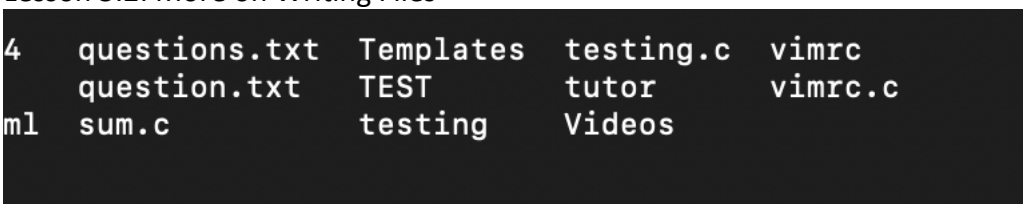
Lesson 4: Summary

No changes or exercises required.

Lesson 5.1: How to Execute an External Command

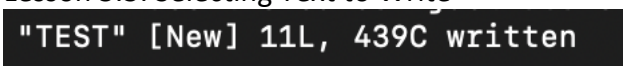
Used the ":! " command in conjunction with the "ls" command in order to see my saved files from the vim file instead of the normal command line in the terminal.

Lesson 5.2: More on Writing Files



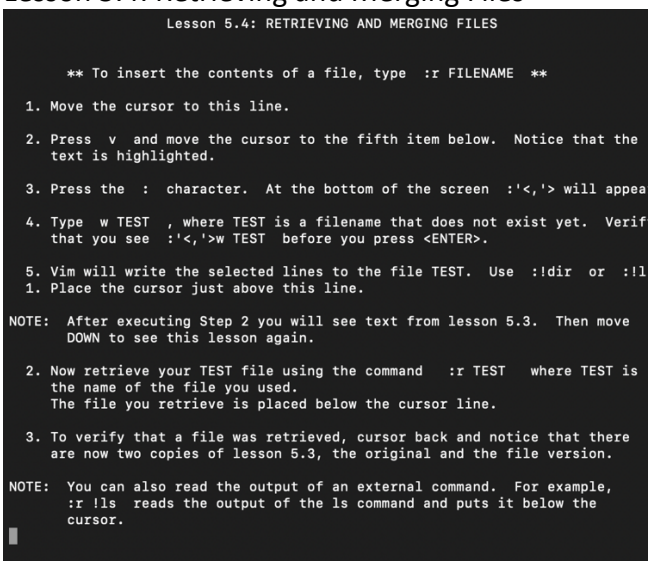
Used the ":!dir" command to see the current directory and then used the ":w" command to create a copy of the current file. The copy is named TEST. Then, I used ":!rm TEST" to delete the copy.

Lesson 5.3: Selecting Text to Write



Used the "v" to select several of the lines and then saved them to a TEST file by using the ":w TEST" command again.

Lesson 5.4: Retrieving and Merging Files

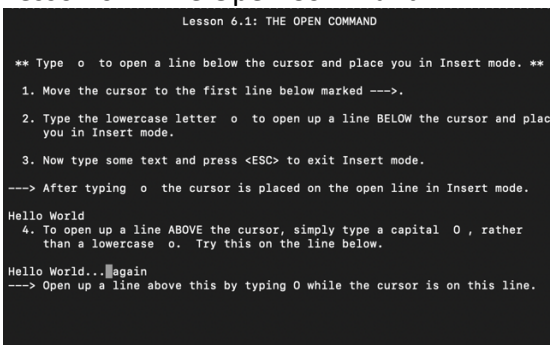


Used the ":r FILENAME" command to open up the TEST file from the 5.3 exercise and opened it in the 5.4 exercise section.

Lesson 5: Summary

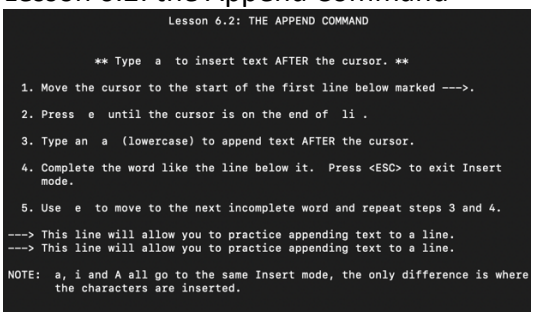
No changes or exercises needed.

Lesson 6.1: The Open Command:



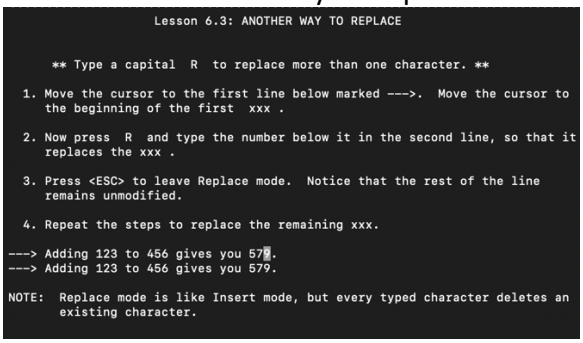
Used the "o" and "O" commands to open up insert mode while also creating a new line to type on.

Lesson 6.2: the Append Command



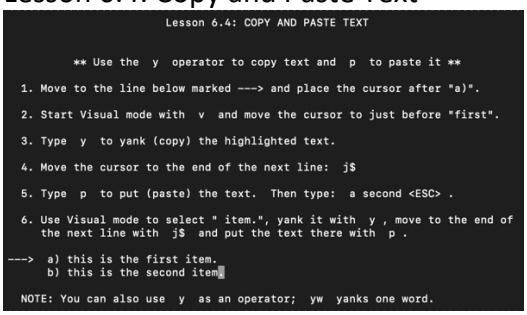
Used the "a" command to append letters to the end of the cursor instead of the end of the line.

Lesson 6.3: Another Way to Replace



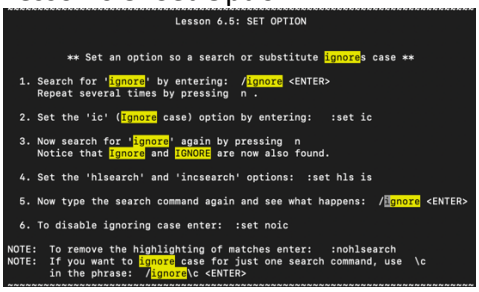
Used the "R" command to replace several characters by going into replace mode.

Lesson 6.4: Copy and Paste Text



Used the “y” and “p” commands to copy and paste selected text (using the “v” command to select it).

Lesson 6.5: Set Option

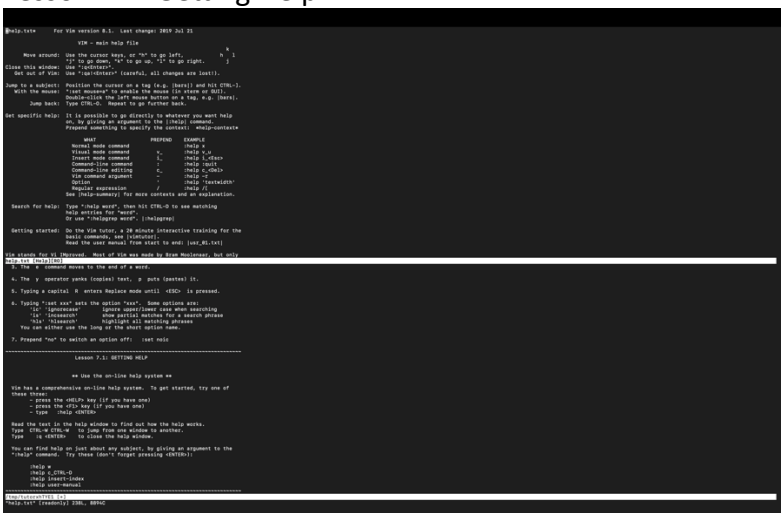


Used the “ic” command to have all instances of “ignore” (without checking for capitalization) and then “hlsearch” and “incsearch” commands to highlight all instances of the word being searched for.

Lesson 6: Summary

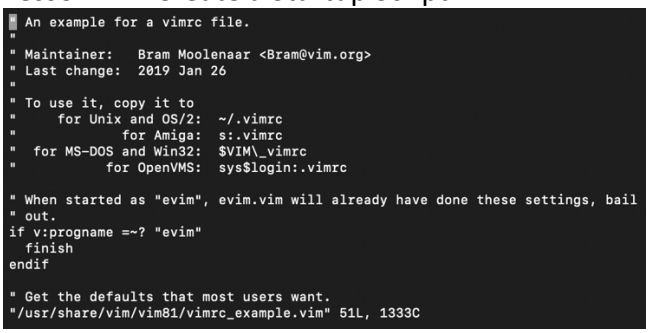
No changes or exercises needed.

Lesson 7.1: Getting Help



Used the “:help” command in order to see the documentation that is used to explain what things in VIM do/are used for.

Lesson 7.2: Create a Startup Script



Used the “:e ~/vimrc” command to start editing the vimrc file and then “:w” to start writing to it. This file is used to set preferences for Vim.

Lesson 7.3 Completion

Used the “CTRL-D” and “<TAB>” commands to look at all the possible commands/file names (“CTRL-D”) and fill in the rest of the command/file name that I started typing (“<TAB>”).

Lesson 7: Summary

No changes or exercises needed.