Working effectively in the linux shell

1. Introduction

This note outlines a self study course on linux command line, shell scripting and related topics that will help in working effectively in the linux shell. The objective of this course is the following.
1. Get familiar with normally used commands and their arguments
2. Get familiar with regular expression syntax
3. Get familiar with other techniques used in the shell
4. Learn how to write small shell scripts
5. Develop the habit of setting up shortcuts and automate manual tasks as much as possible

2. Resources

1. You can refer to the book "A practical guide to linux - commands, editors and shell programming" fourth edition if available. Not all chapters are required.
2. Following tutorials from tutorials point:https://www.tutorialspoint.com/unix/index.htm.

   Following topics can be read:
   - All topics under "Unix/Linux for beginners". For vi editor topic just get basics. We have another detailed course on editor
   - All topics under "Unix/Linux shell programming". The 'regular expression' topic may be skipped. We will use another resource for regular expression.
   - All topics under Advanced Unix/Linux
3. For regular expressions, refer to the following:
   https://www.linux.com/learn/intro-to-linux-/2017/2/introduction-regular-expressions-new-linux-users
4. Following article for shortcuts on the bash shell
   https://linuxconfig.org/linux-command-line-bash-shell-shortcuts

3. Course outline

1. Study the topics mentioned in numbers 2 and 3 in "Resources" above.
2. Study the article mentioned as #4 in Resources above

4. Exercises

For the following exercise please record the commands you have used and their output, the files you updated/ modified. The commands may be saved in one or multiple files and the input/output files may collected in one directory. These files are required for evaluation.

1. Read the man pages of the following commands, understand what they are used for, understand the arguments and use them in command line in a meaningful way. Record the output. For the commands marked '*' you shall explore the arguments in detail. Others you may just understand the usage.

alias	file	md5sum	head *	
bg	fc	mkdir	tail *	
cat	find *	more	touch	
chmod *	flush	mv	sort	
chown	hexdump	patch	set	
cp	history	printenv	shift	
cut	ifconfig	ps *	test	
dd *	join	pushd	time	
diff *	kill	popd	wc *	
dos2unix	less	pwd	which	
dmesg	locate *	rm	xxd *	
du *	ls *	sudo	free	
echo	lspci	tee	top	
export	lsusb	tree *	vmstat	

2. Create a text file "a.txt" with a few 10 lines with some sentences in each line. Copy this file and create another file "b.txt". Modify a few lines here and there in b.txt. Use diff utility to find the difference between both files. Redirect the output of diff to a file named a.patch. Understand the syntax of a patch file. Now apply the patch file on a.txt. Now see if a.txt and b.txt are same using diff utility.
3. Use the pipe '|' to direct output of one command to another. Use pipe to combine two or more commands in a useful manner
4. Study how to do stdio redirection. Find how to redirect stdout to a file. Try it with some

commands. Find how to append the output to an existing file. Find how to redirect both stdout and stderr to the same file. Try it with some commands. Find how to take input from a file by redirecting stdin. Find how to redirect stdout to a file and see in console at the same time using the tee command.

5. Check if your home directory contains a .bashrc file. This file gets run everytime you start a new shell session. So you can put aliases, shell variables etc that use often in this. If there is no .bashrc file already present, create it. Add some variables using export command for your frequently used variables. Use 'cd' command and the variables to switch to those directories. Add an alias in .bashrc file for grepping the command history. Give a name 'gh'.

6. Try all the shortcuts mentioned in #4 under Resources section above. That is from the following article
https://linuxconfig.org/linux-command-line-bash-shell-shortcuts

7. TODO - add regular expression exercises

8. Write a shell script that takes any number of numbers and prints their sum

9. Write a shell script that find the files in the given directory and prints them with their absolute paths