Module 2 Cheat Sheet - Introduction to Linux Commands

Getting information

Return your user name:	
whoami	
Return your user and group id:	
id	
Return operating system name, username, and other info:	
uname -a	
Display reference manual for a command:	
man top	
List available man pages, including a brief description for each command:	
man $-k$.	
Get help on any command (for eg: curl):	
curl —help	

about:blank 1/10

This provides a brief overview of the curl command's usage and options.
Return the current date and time:
date
Navigating and working with directories
List files and directories by date, newest to last:
ls -lrt
Find files in directory tree that end in .sh:
find -name "*.sh"
Return path to present working directory:
pwd
Make a new directory:
mkdir new_folder
Change the current directory:
Un one level:

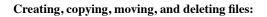
about:blank

cd ../

	To home:
	cd ~` or `cd
	To some other directory: cd path_to_directory
Remov	ve directory verbosely:
	rmdir temp_directory -v
Mon	itoring system performance and status
WIOH	noring system periormance and status
List se	lection of/all running processes and their PIDs:
	ps
	ps -e
Displa	y resource usage:
	top
List m	ounted file systems and usage:

about:blank 3/10

df



Create an empty file or update existing file's timestamp:

touch a_new_file.txt

Copy a file:

cp file.txt new_path/new_name.txt

Change file name or path:

 $\label{eq:mv_this_file.txt} \mbox{ mv this_file.txt that_path/that_file.txt}$

Remove a file verbosely:

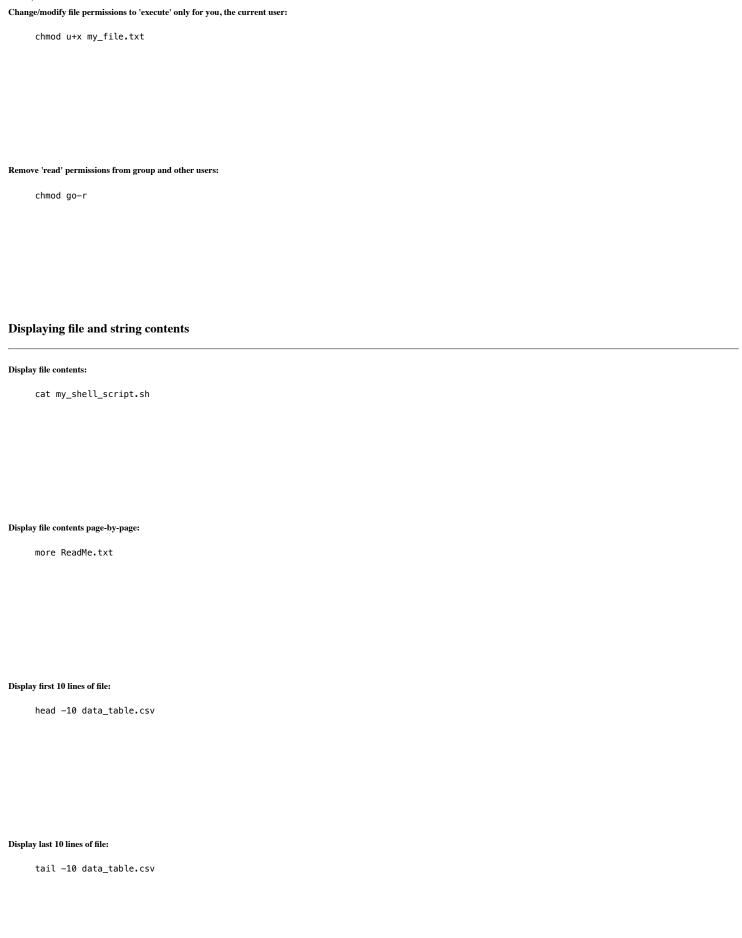
rm this_old_file.txt -v

Working with file permissions

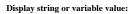
Change/modify file permissions to 'execute' for all users:

chmod +x my_script.sh

about:blank 4/10



about:blank 5/10



echo "I am not a robot" echo "I am \$USERNAME"

Basic text wrangling

Sorting lines and dropping duplicates:

Sort and display lines of file alphanumerically:

sort text_file.txt

In reverse order:

sort -r text_file.txt

Drop consecutive duplicated lines and display result:

uniq list_with_duplicated_lines.txt

Displaying basic stats:

Display the count of lines, words, or characters in a file:

Lines:

wc -l table_of_data.csv

Words:

wc -w my_essay.txt

about:blank 6/10

Characters:

wc -m some_document.txt

Extracting lines of text containing a pattern:

Some frequently used options for grep:

Option	Description
-n	Print line numbers along with matching lines
-с	Get the count of matching lines
-i	Ignore the case of the text while matching
-v	Print all lines which do not contain the pattern
-w	Match only if the pattern matches whole words

Extract lines containing the word "hello", case insensitive and whole words only:

```
grep -iw hello a_bunch_of_hellos.txt
```

Extract lines containing the pattern "hello" from all files in the current directory ending in .txt:

```
grep -l hello *.txt
```

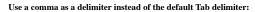
Merge two or more files line-by-line, aligned as columns:

Suppose you have three files containing the first and last names of your customers, plus their phone numbers.

Use paste to align file contents into a Tab-delimited table, one row for each customer:

```
paste first_name.txt last_name.txt phone_number.txt
```

about:blank 7/10



paste -d "," first_name.txt last_name.txt phone_number.txt

Use the cut command to extract a column from a table-like file:

Suppose you have a text file whos rows consist of first and last names of customers, delimited by a comma.

about:blank

Extract first names, line-by-line:

cut -d "," -f 1 names.csv

Extract the second to fifth characters (bytes) from each line of a file:

cut -b 2-5 my_text_file.txt

Extract the characters (bytes) from each line of a file, starting from the 10th byte to the end of the line:

cut -b 10- my_text_file.txt

Compression and archiving

Archive a set of files:

tar -cvf my_archive.tar.gz file1 file2 file3

Compress a set of files:

zip my_zipped_files.zip file1 file2
zip my_zipped_folders.zip directory1 directory2



about:blank 9/10

Authors

Jeff Grossman Sam Propupchuk

Other Contributors

Rav Ahuja



about:blank 10/10