IT 1100 : Introduction to Operating Systems - Redirection and Compression

Command	Sample Usage		Description
cat	cat foo.tar		display the contents of a tarball
	cat the blue dog does tricks > cat-output.txt echo "the blue dog does tricks" > cat-output.txt		concatenate the input on the screen and redirect it to a file.
	cat file1 file2 > file3		combine two files into a third file
echo	echo "the blue dog does tricks" > echo-save.txt		echo text and redirect it to a file.
	echo "the blue dog does new tricks" >> echo-save.txt		echo text and append it to the end of a file.
less	less myfile.txt cat myfile.txt | less		display the contents of a file with the ability to navigate up and down
sort	sort myfile.txt cat myfile.txt | sort		sort the output into alpha-numeric order based on the first word of the line
uniq	sort myfile.txt | uniq		remove duplicate lines that are adjacent to each other. best when used with the sort command
grep	grep word myfile.txt	search within a file or directory or command output for a search term	
	grep -r word mydirectory/	search recursively within a directory	
	grep -i word myfile.txt	search case insensitive	
	grep -e ".*boo." myfile.txt egrep ".*boo.*" myfile.txt	search using wildcards	
	ls | grep myfile	search within a command output for a search term	
wc	wc myfile.txt	display the line count, word count and byte count information of a file	
	wc -l myfile.txt	display the line count of a file	
	wc -w myfile.txt	display the word count of a file	

	ls | wc -l	display the line count of the output of the ls command
head	head myfile.txt	display the first 10 lines of a file starting at the top
	head -n 20 myfile.txt cat myfile.txt | head -n 20	display the specified number of lines of a file starting at the top
tail	tail myfile.txt	display the last 10 lines of a file starting at the bottom
	tail -n 20 myfile.txt cat myfile.txt | tail -n 20	display the specified number of lines of a file starting at the bottom
	tail -f /var/log/messages	view changes to the file as they occur
tee	ls | tee -a file.txt	redirect standard output to both the terminal and append it to a file
ln	ln -s mydirectory/myfile name-of-symlink ln -s /full/path/to/original /full/path/to/new/link	create a symbolic link, like a shortcut to a file or directory
type	type foo	check if an alias name already exists as a command
alias **unalias**	alias	list current aliases
	alias foo="cat myfile.txt | less"	create an alias for a command, like a nickname or shortcut
	alias ls="ls –color"	redefine an existing command
	unalias -a foo	remove an alias
zip **unzip**	zip -rv foo.zip mydirectory	create a zip file - this directory will be both compressed and zipped extension zip
	unzip foo.zip	extract a zip file - this file will be both uncompressed and unzipped
	zip -sf foo.zip unzip -l	display a list of files in the zip file
	unzip -c	display the contents of a zipped file
	bzip2 -v foo.txt	create a bzip2 file - this will replace the original files with a new file extension bz2 using verbose option

bzip2 bunzip2 bzcat	bunzip2 foo.txt.bz2	extract a bzip2 file - this will replace the original files with a new file
	bzcat foo.txt.bz2 less foo.txt.bz2	display the contents of a bzip2 file
gzip gunzip zcat zless	gzip -v foo.txt	create a gzip file - this will replace the original files with a new file extension gz
	gzip -vr mydirectory	create a gzip file of a directory - this will replace the original files with a new file
	gzip -dv foo.txt.gz gunzip foo.txt.gz	extract a gzip file - this will replace the original files with a new file
	ls -l /etc | gzip > foo.txt.gz	pipe the output of ls -l to the gzip command and store that in foo.txt.gz
	zcat foo.txt.gz	display the contents of a gzip file
	zless foo.txt.gz	display the contents of a gzip file
tar	tar -cvvf foo.tar mydirectory	create a tarball of a directory
	tar -cvvf foo.tar file1.txt file2.txt	create a tarball of a list of files
	tar -xvvf foo.tar	extract a tarball
	tar -rvf foo.tar file3.txt	add a file to the tarball
	tar -tf foo.tar	list the files in the tar file
	tar -cvvf just_tar.tar file?.txt	tar a list of files using the wildcard
tar/gzip	tar -cvzf foo.tgz mydirectory	create a tarball and gzip it at the same time z option send it through the gzip compression
	tar -xvzf just.tar.gz tar -xvzf just.tgz	extract a tarball and gunzip at the same time
	tar -cvzf just.tar.gz file?.txt	tar and gzip at the same time a list of files using the wildcard
tar/bzip2	tar -cvjf just.tar.bz2 dirname/	create a tarball and bzip2 it at the same time j option sends it through the bzip2 compression

	tar -xvjf just.tar.bz2	extract and bunzip2 at the same time
> and >>	ls -l /usr/bin > ls-output.txt	redirect standard output to a new file
	ls -l /usr/bin >> ls-output.txt	redirect standard output and append to the end of a file
<	cat < ls-output.txt	redirect standard input from a file
2> and 2>> /dev/null	ls -l /bin/usr 2> /dev/null	redirect standard error to /dev/null
	ls -l /bin/usr 2>> ls-error.txt	redirect standard error to a file
&> and &>>	ls -l /bin/usr &> ls-output.txt ls -l /bin/usr &>> ls-output.txt	redirect standard output and standard error to the same file
find	find /home -name 'fun.txt'	find files by name searching within the /home directory
	Find / -name *fun*	find files using wildcards to match a name searching within the root (/) directory
	find ./ -iname *fun*	find files by name but be case insensitive searching within the current directory (using a dot-slash)
	find ~ -name 'fun.txt'	find files by name in your $HOME directory
	find / -name "*fun.txt*" 2> /dev/null	find files by name and redirect the errors
altogether now	find / -name *elephant* 2> /dev/null >> save.txt	find a file with elephant in the name and redirect standard error to /dev/null and redirect standard output to save.txt
| (pipe)	cat /etc/passwd | sort | less	redirect the standard output of one command to another command
locate	locate zip | grep bin	find files using a database. Searches for anything in the bin directory with the name zip. Example /bin/zip
updatedb	updatedb	update the locate database so it can find recently created files.