Notes 7

cat

- **Definition**
- The cat command is used for displaying the contents of a file
- **Usage**
- `cat + option +file(s) to display`
- **Example**
- Display the content of a file located in the pwd
- `cat todo.lst`
- Display the content of a file using absolute path
- `cat ~/Documents/todo.lst`
- Display the content of a file with line numbers
- `cat -n ~/Documents/todo.md`

tac

- **Definition**
- The tac command is used for displaying the content of a file in reverse order
- **Usage**
- `tac + option + file(s) to display`
- **Example**
- Display the content of a file located in the pwd
- `tac todo.md`
- Display the content of the file using absolute path
- `tac ~/Documents/todo.md`
- reverse a files contents line by line
- `tac filename.txt`

head

- **Definition**
- prints the first 10 lines of a file or more
- **Usage**
- `head + option + file(s)`
- **Example**
- Display the first 10 lines of a file
- `head ~/Documents/Books/dracula.txt`
- Display the first 5 lines of a file
- `head -5 ~/Documents/Book/dracula.txt`
- Display first lines of many files using wildcards
- `head -n 1 *.csv *.py`
- Display the given number of lines of the output of a given command
- `ls -1 ~/cis106/ | head -n 2`
- Display the name of a file in the output

- `head -v -n 7 ~/Documents/Books/dracula.txt`
- Display a given number of bytes instead of lines
- `head -c 50 ~/Documents/Books/dracula.txt`

tail

- **Definition**
- Prints the last 10 lines of a file
- **Usage**
- `tail + option + file`
- **Example**
- Display the last 10 lines of a file
- `tail ~/Documents/Book/dracula.txt`
- Display the last 5 lines of a file
- `tail -5 ~/Documents/Book/dracula.txt`
- Display the last lines using wildcard
- `tail -n 1 *.csv *.py`
- Display a given number of lines of the output of a given command
- `ls -1 ~/cis106/ | tail -n 2`
- Display the name of a file in the output
- `tail -v -n 7 ~/Documents/Books/dracula.txt`
- Display a given number of bytes instead of lines
- `tail -c 50 ~/Documents/Books/dracula.txt`

cut

- **Definition**
- used to extract specific section of each line of a file and display it on the screen
- **Usage**
- `cut + option + files(s)`
- **Example**
- Display a list of all the users in your system
- `cut -d ';' -f1 /etc/passwd`
- Display the list of all users in your system with their login shell
- `cut -d ';' -f1 /etc/passwd`
- cut a range of bytes per line
- `cut -b 1-5 usernames.txt`
- cut a file using a delimiter but changing the delimiter in the output
- `cut -d ';' -f1,7 --output-delimiter=' => ' /etc/passwd`
- cut the permissions from the output of ls
- `ls -1 | cut -d ' ' --complement -s -f1`

sort

- **Definition**
- the sort command is used for sorting files
- **Usage**
- `sort + option + file`

- **Example**
- Sort a file
- `sort users.list`
- Sort a file and save the output to a new file
- `sort -o sorted.lst users.lst`
- Sort a file in reverse order
- `sort -r users.txt`
- Sort a file in column number
- `sort --k 2 users.txt`

wc

- **Definition**
- the wc command is used for printing the number of lines, characters and bytes in a file
- **Usage**
- `ec + option + file(s)`
- **Example**
- display the number of characters in a file
- `wc -m users.txt`
- Display the number of lines in a file
- `wc -l users.txt`
- Display the number of words in a file
- `wc -w users.txt`

tr

- **Definition**
- used for translating or deleting characters from the standard output
- **Usage**
- `standard output | tr + option + set + set`
- **Example**
- Translate a period to a comma
- `cat file.txt | tr n'.' ','`
- Translate white space into tabs
- `cat program.py | tr "[:space:]" '/t/`
- Translate labs into space
- `cat program.py | tr -s "[:space:]" ' '`

diff

- **Definition**
- compares files and displays the differences between them
- **Usage**
- `diff + option + file1 + file2`
- **Example**
- Display the difference between two files
- `diff cars.csv cars-backup.csv`
- Display the difference between two files in a column

- `diff -y cars.csv cars-backup.csv`

grep

- **Definition**
- grep is used to search text in given files and works line by line
- **Usage**
- `grep + option + search criteria +file(s)`
- **Example**
- search any line that contains the word dracula
- `grep 'dracula' ~/Documents/dracula.txt`
- search any word regardless of the case
- `grep -i`
- remove this word from search
- `grep -v`
- display line number for every line matched
- `grep -n`
- search and display the word which matches
- `grep -o`

-E

- Used to display $ sign at the end of every line

-d

*specifies the delimiter to use

-f1 + a number

- specifies that for every line in /etc/passwd the first and seventh field should be cut