|  |  |
| --- | --- |
|  | **Shell Basics**  The shell can be used to copy, move, and combine multiple files  Use the command **pwd** to find out what directory you are on your computer  Use the command **ls** to list directory contents  Use flags **-l** and **-lh** to guide the output of the **ls** command  Use the command **cd** to move around your computer  Use the **man** command to look up the manual page for a command, e.g. man ls – this is like asking for help  Use the command **mv** to rename and move files.  Use the command **cp** to create a backup version of an existing file.  Use the command**cat** to output (type) a file to the screen or to combine more than one file of the same file type.  Use the wildcards \* (to stand in for multiple characters) and **?** (for a single character)  Use the command **echo** to output text to the screen as commands are run.  Use the **rm** command to delete a file.  The command **rmdir** will delete a directory.  The command **rmdir -r** will delete a directory, even if it is not empty.  The command **rmdir -r-i** will delete a directory, even if it is not empty, but will ask you to confirm each deletion.  wc stands for word count  Use the wc command with the flags -w and -l and -c respectively to count the words, lines and characters in a file or a series of files.  Use the redirect > character and pathname/filename  > subdirectory/filename to save results to a file into a subdirectory  Use the grep command to search for instances of a string inside files  Use grep with the -c flag to count instances of a string, the -i flag to return a case insensitive search for a string, the -v flag to exclude a string from the results, and -w to return a whole word only search  Use --file=list.txt to use the file list.txt as the source of strings used in a query  Combine these commands and flags to build complex queries in a way that suggests the potential for using the Unix shell to count and mine your research data and research projects.  **Shell: Basics**  **pwd** print working directory  **ls** list directory  **ls -l**: list file information  **ls -lh**: list human readable file information  **cd** change directory  **Shell: Interacting with Files**  **mkdir** make directory  **cat** send file or files to output (in most cases, print to shell)  **head** output first 10 lines of a file or files.  **tail** output last 10 lines of a file or files  **mv** rename or move a file or files. Syntax for renaming a file: mv FILENAME NEWFILENAME  **cp** copy a file or files. Syntax: cp FILENAME NEWFILENAME  **>** redirect output. Syntax with cat: cat FILENAME1 FILENAME2 > NEWFILENAME  **rm** remove a file or files. NB: *USE WITH EXTREME CAUTION!!!*  **Shell: Wildcards**  **?** a placeholder for one character or number  **\*** a placeholder for zero or more characters or numbers  **[]** defines a class of characters  *Examples*  foobar? - matches seven character strings that start with foobar and end with one character or number  foobar\* - matches strings starting with foobar that end with zero or more further characters or numbers  foobar\*txt - matches strings that start with foobar and end with txt  [1-9]foobar? - matches 8-character strings that start with a number, have foobar after the number, and end with any character or number.  **Shell: Counting and Mining**  **wc** word count  wc -w: count words  wc -l: count lines  wc -c: count characters  **sort** sort input  **grep** (stands for global regular expression print)  **grep** -c: displays counts of matches for each file  **grep** -i: match with case insensitivity  **grep** -w: match whole words  **grep** -v: exclude match  **grep** --file=FILENAME.txt: use the file FILENAME.txt as the source of strings used with the **grep** command  | (vertical bar character aka the **pipe**) - make the output from one command the input to the next command |
|  |  |
|  |  |