Index of commands

Basic commands

- `man [command]`
  - `command --help`
  - An interface to the on-line reference manuals.
- `pwd`
  - Print name of current/working directory.
- `cd`
  - Changes the shell working directory.
  - `cd` => brings you back to /home
  - `cd /to/this/path` => Change the working directory to PATH
  - `cd ../` => return in the parental directory
- `ls`
  - List working directory contents.
    - `ls [/to/a/path]` => list PATH directories contents
    - option -a do not ignore entries starting with (hidden files)
    - option -h print sizes of files
    - option -l use a long listing format
    - `ls -hla`
- `echo [string]`
  - Display a line of text.

Handling files and directories

- `mkdir [directory]`
  - Make directories.
- `touch [file]`
  - Change file timestamps. Update the access and modification times of each FILE to the current time. If the FILE does not exist, it will be created empty.
- `cp [/path/to/file.txt] [/where/to/copy/]`

    - Copy files and directories.
    - `cp file.txt .` => copy FILE in the working directory

- `mv [file] [new name]`
    - Move (rename) files.

- `rm [file]`
    - Remove files.
    - `rm -r [directory]` => remove directories and their contents recursively

- `head [file]`
    - Output the first par of files. By default, output the first 10 lines.
    - `head -n50` => output the first 50 lines

- `tail [file]`
    - Output the last part of files. By default, output the last 10 lines.
    - `head -n100` => output the last 100 lines

- `more [file]`
    - Filter for paging through test one screenful at a time.

- `less [file]`
    - Opposite of more. Allows backward movement in the file. No wrap-up line.

- `cat [file] [file]`
    - Concatenate files and print on the standard output.

- `wc [file]`
    - Print newline, word, and byte counts for each file.
    - `wc -l [file]:` line count
    - `wc -w [file]:` word count

Manipulating files

- `gzip [file]`
    - Compress files.
    - `gzip -d [file.gz]` => extract file

- `gunzip [file.gz]`
    - Expand files.
    - `gunzip -c [file.gz]` => extract file in STDIN. The actual file stays compressed.

- `zcat [file.gz]`
    - Extract files.

- `cut [file]`

    - Remove sections from each line. Print selected parts of lines from each file to standard output.
    - `cut -f2 -d”,”` => Print element #2 (column #2) of the file. The delimiter is “,”. By default, the delimiter is “\t”.

- `grep [pattern] [file]`
    - Print lines matching a pattern.
    - `grep -v [pattern] [file]` => Invert the sense of matching, to select non-matching lines.

- `sort [file]`
    - Sort lines of text files.

- `uniq [file]`
    - Report only unique lines (omit repeated lines).
    - `uniq -c [file]:` prefix lines by the number of occurrences
    - `uniq -d [file]:` only print duplicated lines

- `comm [file1] [file2]`
    - Compare two sorted files line by line. Output 3 columns: 1/ Unique lines in the first file, 2/ Unique line of the second file, 3/ Lines in common.
    - `comm -1 -2 [file1] [file2]` => print only the lines in common
    - `comm -2-3 [file1] [file2]` => print only the lines uniquely found in file 1
    - `comm -1 -3 [file1] [file2]` => print only the lines uniquely found in file 2

- `tr [file]`
    - Translate or delete characters.
    - `cat [file] | tr “,” “ “` => replace every comma into a space

- `sed [file]`
    - Stream editor for filtering and transforming text.
    - `sed 1d [file]` => delete the first line
    - `sed 's/day/night/' [file]` => replace the first occurrence of “day” by “night”
    - `sed 's/[a-z]/[A-Z]/g' [file]` => replace every lower case by an upper case.

- `awk '{}' [file]`
    - Pattern scanning and processing language.
    - `awk '{print $1}' [file]` => print first element of each line
    - `awk '{if($3 < $2) print $0}' [file]` => print the line if the value of the third element is inferior to the second element

Regular Expressions

|	OR	gray|grey
()	Grouping	gr(a|e)y
?	Zero of one occurrence of the preceding item	Colou?r => color, colour
*	Zero or more occurrences	ab*c => ac, abc, abbc, abbbc...
+	One or more occurrences	ab+c => ab, abbc, abbbc NO ac
{n}	The preceding item is matched n times	ab{2}c => abbc
{min,}	The preceding item is matched at least min times	ab{2,}c => abbc, abbbc...
{min,match}	The preceding item is matched between min and max times	ab{1,2}c => abc, abbc only
.	Matches any character	a.b => axb, arb, apb..

\t	Tabulation
\s	Whitespace character
\n	Zero of one occurrence of the preceding item
[a-z]	Range of character in lowercase
[A-Z]	Range of character in uppercase
[0-9]	Range of numbers
[a-zA-Z]	Range of characters