

ls

Action: List files.

Notes: If you don't provide a folder it will look in the current folder.

Often it's useful to see the list in a vertical format. You can do that using the `-l` option. Also sometimes you need to see the hidden files.

Use `a` option.

Examples:

`ls stuff/` : list contents of the `stuff/` folder

`ls -alh` : list contents of the current directory vertically, including hidden ones.

cd

Action: Change into another directory.

Notes: Can go up one level in the directory tree by using `..`.

Examples:

`cd stuff/` : Change to the `stuff/` directory.

`cd ..` : Go up one level.

`cd ../../other` : Go up two levels and change to the `other` directory.

mkdir

Action: Create an empty directory.

Notes: Can use `-p` option to make multiple sub-directories quickly.

Examples:

`mkdir stuff` : Create the `stuff/` directory.

`mkdir -p work/ironhack/assignment1` : Create the `work/` with `ironhack/assignment1` directories inside.

rm

Action: Remove a file or directory.

Notes: Use `-r` option for deleting directories.

Examples:

`rm file.txt` : Delete the `file.txt` file.

`rm -rf work/ironhack/assignment1` : Delete the `work/ironhack/assignment1` directory.

cp

Action: Copy a file or directory.

Notes: Takes two parameters: source and destination. Can use wildcard in source `*` to copy files that match a pattern. Use `-r` option for directories.

Examples:

```
cp program.rb app.rb : Copies program.rb to app.rb .
cp stuff/*.html html-files/ : Copies all files in stuff/ that
end in .html into the html-files/ folder.
cp stuff/ stuff-copy : Copies the stuff/ directory as
the stuff-copy/ directory.
```

mv

Action: Move and/or rename a file or directory.

Notes: Takes two parameters: source and destination. Can use wildcard in source `*` to move files that match a pattern. If the destination is a folder, it will move the files into the folder and leave the file's name unchanged.

Examples:

```
mv program.rb app.rb : Renames program.rb to app.rb .
mv program.rb stuff/ : Moves program.rb into stuff/ .
mv stuff/*.html html-files/ : Moves all files in stuff/ that
end in .html into the html-files/ folder.
```

touch

Action: Create a new empty file.

Notes: Takes one parameter, the name of the file. You can create multiple files at once by putting more than one name

Examples:

```
touch newfile.txt
touch file1.txt file2.txt file3.txt
```

open

Action: Open a file or directory.

Notes: Takes one parameter file or folder to open. If we don't specify an app it will open the file with the default application for that file-type. We can choose the program we want to use adding `-a appName` between open and filename. We can also open

more than one file at a time concatenating files.

Examples:

```
open myFile1 : Opens myFile1
open myDirectory : Opens myDirectory
open myFile1 myFile2 myFile3 : Opens myFile1, myFile2 and
myFile3
open -a appName myFile1 : Opens myFile1 with appName
open . : Opens current directory
```

more

Action: Output the contents of a file.

Notes: Use the spacebar or arrow keys to navigate a long files. Use the q key to quit out of it.

Examples:

```
more program.rb
```

pwd

Action: Show the current directory.

Notes: Short for *print working directory*, not *password* (I know!). Doesn't take any options, just use it as-is to remind yourself where you are.

Examples:

```
pwd
```