Folders & Files

`$ cp -a source/. /dest/`	Copy Directory
`$ rm -r mydir`	Remove Directory
`$ sudo du -hs /` `$ sudo du -hs /etc/` `# to restrict to specified directory`	Usage by directory

Misc

`$ dpkg --list`	List installed packages add > `**.txt` to dump list to file
`.tar.gz or tgz` `$ tar xvzf` `***.tar.gz`	Unzip tar - v is for verbose
`$ > .gz` `(.gzip)` `$ gunzip` `***.gz`	
`$ >.tar.bz2 or` `.tbz` `$ tar xvjf` `***.tar.tbz`	
`$ dtrx` `***.tar.gz` `$ dtrx ***.tar.bz2`	

Change Permissions

`$ sudo chmod -R ugo+rw /DATA/-SHARE` [...]	

The breakdown of the above command looks like:

Change Permissions (cont)

• sudo – this is used to gain admin rights for the command on any system that makes use of sudo (otherwise you'd have to 'su' to root and run the above command without 'sudo')
• chmod – the command to modify permissions
•-R – this modifies the permission of the parent folder and the child objects within
•ugo+rw – this gives User, Group, and Other read and write access.

As you can probably surmise, this command, as shown above, opens wide the `SHARE` folder such that anyone on the system can have access to that folder. A more secure method would be to use groups `g` or current user `u`.

The breakdown of user permissions looks like this:

•u – user
•g – group
•o – other
The 'other' entry is the dangerous one, as it effectively gives everyone permission for the folder/file.

The permissions you can give to a file or folder are:

•r – read
•w – write
•x – execute

Using the `-R` switch is important. If you have a number of sub-folders and files within the `SHARE` directory, and you want the permissions to apply from the parent object (the containing folder) to the child objects (the sub-folders and files), you must use the `-R` (recursive) switch so the same permissions are applied all the way to the deepest folder, contained within the parent. [...]