Helpful terminal commands

- compiled by Anirudh Ravishankar

Typically, the available options of any terminal command can be read using the option −−help after the command (eg: **ls** −−**help**)

Looking around & navigating folders

1. **pwd** : shows the Present Working Directory

2. **cd [TARGET]** : change directory to move to a different directory [TARGET] (which can be any folder path or just a folder name in the current directory)

   (a) **cd ..** : moves backward by one folder
   (b) **cd ../../..** : moves 3 folders backwards

3. **ls** : lists all the files and folders in the current directory

   (a) **ls -l** : lists files and folders in the current directory in a long, descriptive format
   (b) **ls -l [TARGET]** : lists files and folders in a directory defined by [TARGET]

4. **mkdir [NAME]** : makes a new directory called [NAME]

5. **rm [FILENAME]** : removes a specified file defined by [FILENAME]

   (a) **rm *** : removes/deletes all files in the current directory (of course, use with caution!). Files within folders do NOT get deleted
      i. **rm *.abc** : removes all files in the current directory that have a ".abc" file format
   (b) **rm -r *** : recursively removes all files in the current directory. This means files within folders inside the current directory also get deleted.

6. **rmdir [FOLDER]** : remove/delete the specified folder called [FOLDER] (only if it is empty)

7. **mv [SOURCE] [DESTINATION]** : moves a file called [SOURCE] to a folder/filepath called [DESTINATION]

   (a) **mv sample.txt sample-new.txt** : the mv command can be used to rename a file called sample.txt to sample-new.txt

8. **| more** : append this to the end of a command such as **ls** to display the information in a staggered manner. Navigate through the output using the Enter key, and press Q to go back to the command line.

9. Feel free to explore more options!

Creating and editing a file with vi

1. **vi sample.txt** : create a file named sample.txt in the current directory and enter the vi console (Python/bash scripts can also be created this way). You will first enter a view mode. Within the console,

   (a) Press **i** to enter the edit mode
   (b) Write/modify scripts by typing as on a text editor
   (c) To exit the edit mode, Press **Esc**
   (d) In the view mode, you can navigate across the file and use various shortcuts to achieve different functions
   (e) In the view mode, Type **:wq** to save the file and exit the console to go back to the terminal screen. Your file will be saved in the folder you created it in.
   (f) In the view mode, Type **:q** or **:q!** to exit the vi console without saving

Using hdf5 analysis tools

Here are some basic commands for analysing simulation snapshots & group catalogs (typically hdf5 files) from a terminal using two packages: **h5ls** and **h5dump**

1. **h5ls -r snap.hdf5** *or* **h5dump -n snap.hdf5** : can be used to read a group catalog or simulation snapshot's (here, *snap.hdf5*) list of contents. This lists the various "fields" that are stored in a snapshot, such as /PartType1/Coordinates (Dark matter coordinates) and /PartType1/Velocities (Dark matter velocities).

2. **h5dump -g Header snap.hdf5** *or* **h5dump -H snap.hdf5** : read the header of a group catalog or simulation snapshot called *snap.hdf5*. This displays information such as the snapshot time, scale factor, boxsize, cosmological parameters, and default units of various dimensional quantities.

3. **h5dump -d Field snap.hdf5 | more** : display the dataset corresponding to the *Field* keyword from the *snap.hdf5* in a staggered fashion using the **more** command. Caution, snapshot files are usually quite large!