UNIX Commands

`tar`

tar is short for (magnetic) tape archiving. More commonly, it is used to combine a few files into a single file for easy storage and distribution

Usage

`tar -cvf file.tar inputfile1 inputfile2`

Options

`-c -f -v -x -r`

If you don't use the `f` option, `tar` assumes you really do want to create a tape archive instead of joining up a number of files.

The `v` option tells `tar` to be verbose, which reports all files as they are added.

To separate an archive created by tar into separate files, at the shell prompt, enter:

`tar -xvf file.tar`

Combining a few files into a single file

```
RKs-MacBook-Pro:unix_commands Deepika$ pwd
/Users/Deepika/Documents/ttn/unix_commands
RKs-MacBook-Pro:unix_commands Deepika$ ls -l
total 24
-rw-r--r--  1 Deepika  staff  14 Jul  8 18:42 deepika.txt
-rw-r--r--  1 Deepika  staff  47 Jul  8 18:39 goodbye.txt
-rw-r--r--  1 Deepika  staff  45 Jul  8 18:38 hello.txt
RKs-MacBook-Pro:unix_commands Deepika$ tar -cvf deep.tar *
a deepika.txt
a goodbye.txt
a hello.txt
RKs-MacBook-Pro:unix_commands Deepika$ ls -l
total 32
-rw-r--r--  1 Deepika  staff  4096 Jul  8 18:48 deep.tar
-rw-r--r--  1 Deepika  staff    14 Jul  8 18:42 deepika.txt
-rw-r--r--  1 Deepika  staff    47 Jul  8 18:39 goodbye.txt
-rw-r--r--  1 Deepika  staff    45 Jul  8 18:38 hello.txt
RKs-MacBook-Pro:unix_commands Deepika$
```

Separating a single tar file into its constituent files

```
RKs-MacBook-Pro:unix_commands Deepika$ pwd
/Users/Deepika/Documents/ttn/unix_commands
RKs-MacBook-Pro:unix_commands Deepika$ ls -l
total 32
-rw-r--r--  1 Deepika  staff  4096 Jul  8 18:48 deep.tar
-rw-r--r--  1 Deepika  staff    14 Jul  8 18:42 deepika.txt
-rw-r--r--  1 Deepika  staff    47 Jul  8 18:39 goodbye.txt
-rw-r--r--  1 Deepika  staff    45 Jul  8 18:38 hello.txt
RKs-MacBook-Pro:unix_commands Deepika$ tar -xvf deep.tar
x deepika.txt
x goodbye.txt
x hello.txt
RKs-MacBook-Pro:unix_commands Deepika$
```

find

find is a command-line utility that searches one or more directory trees of a file system, locates files based on some user-specified criteria and applies a user-specified action on each matched file. The possible search criteria include a pattern to match against the filename or a time range to match against the modification time or access time of the file. By default, find returns a list of all files below the current working directory.

Options

`-name`

`-perm`

```
RKs-MacBook-Pro:unix_commands Deepika$ find . -name hey.txt
./hey.txt
RKs-MacBook-Pro:unix_commands Deepika$ find . -perm 400
./hello.txt
./hey.txt
RKs-MacBook-Pro:unix_commands Deepika$
```

diff

diff stands for difference. This command is used to display the differences in files by comparing the files line by line. It tells us which lines in one file have to be changed to make the two files identical.
diff uses certain special symbols and instructions that are required to make two files identical. It tells you the instructions on how to change the first file to make it match the second file.
a : add
c : change
d : delete

Usage

```
diff [options] file1 file2
```

Options

`-c` (context) : To view differences in context mode, use the -c option.
The first file is indicated by `***`, and the second file is indicated by –.
The line with `**********` is just a separator.

(a) + : It indicates a line in the second file that needs to be added to the first file to make them identical.
(b) – : It indicates a line in the first file that needs to be deleted to make them identical.
Like in our case, it is needed to delete mv and comm from first file and add diff and comm to the first file to make both of them identical.

`-u` (unified) : To view differences in unified mode, use the -u option. It is similar to context mode but it doesn't display any redundant information or it shows the information in concise form.

```
RKs-MacBook-Pro:unix_commands Deepika$ diff hello.txt goodbye.txt
1c1
< Hello, world! I'm doing my UNIX assignment.
---
> Tell me why you say goodbye when I say hello!
RKs-MacBook-Pro:unix_commands Deepika$ diff -c hello.txt goodbye.txt
*** hello.txt   Sun Jul  8 18:38:46 2018
--- goodbye.txt Sun Jul  8 18:39:37 2018
***************
*** 1,2 ****
! Hello, world! I'm doing my UNIX assignment.

--- 1,2 ----
! Tell me why you say goodbye when I say hello!

RKs-MacBook-Pro:unix_commands Deepika$ diff -u hello.txt goodbye.txt
--- hello.txt   2018-07-08 18:38:46.000000000 +0530
+++ goodbye.txt 2018-07-08 18:39:37.000000000 +0530
@@ -1,2 +1,2 @@
-Hello, world! I'm doing my UNIX assignment.
+Tell me why you say goodbye when I say hello!

RKs-MacBook-Pro:unix_commands Deepika$ diff -cu hello.txt goodbye.txt
diff: conflicting output style options
diff: Try `diff --help' for more information.
RKs-MacBook-Pro:unix_commands Deepika$
```

`tail`

It is the complementary of head command. The tail command prints the last n units of data of the given input. By default, it prints the last 10 lines of the specified files. If more than one file name is provided then data from each file is preceded by its file name.

Usage

```
tail [options] [file_names]
```

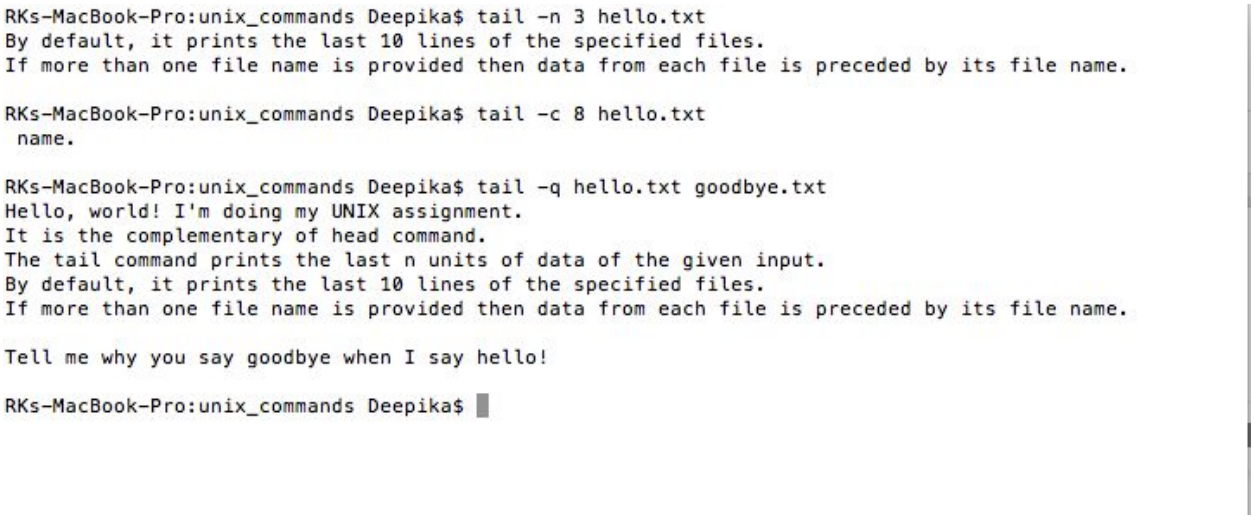
Options

`-n num`: Prints the last 'num' of lines instead of last 10 lines. num is mandatory to be specified in command otherwise it displays an error.

`-c num`: Prints the last 'num' of bytes from the file specified. Newline count as a single character, so if tail prints out a newline, it will count it as a byte. In this option it is mandatory to write -c followed by positive or negative num depends upon the requirement.

`-q`: It is used if more than 1 file is given. Because of this command, data from each file is not preceded by its file name.

`-f`: This option is mainly used to monitor the growth of log files written by Unix programs as they are running.



`less`

less is a terminal pager program used to view (but not change) the contents of a text file one screen at a time. It is similar to more, but has the extended capability of allowing both forward and backward navigation through the file. Unlike most Unix text editors/viewers, less does not need to read the entire file before starting, resulting in faster load times with large files.

Usage

`less [options] [file_name]`

Options

`-M`: Shows more detailed prompt, including file position.
`-N`: Shows line numbers (useful for source code viewing).
`-X`: Leave file contents on screen when less exits.

```
less -M hello.txt
```

```
Hello, world! I'm doing my UNIX assignment.
It is the complementary of head command.
The tail command prints the last n units of data of the given input.
By default, it prints the last 10 lines of the specified files.
If more than one file name is provided then data from each file is preceded by its file name.

hello.txt lines 1-6/6 (END)
```

```
     1 Hello, world! I'm doing my UNIX assignment.
     2 It is the complementary of head command.
     3 The tail command prints the last n units of data of the given input.
     4 By default, it prints the last 10 lines of the specified files.
     5 If more than one file name is provided then data from each file is preceded by its file name.
     6
hello.txt (END)
```

```
RKs-MacBook-Pro:unix_commands Deepika$ less -M hello.txt
RKs-MacBook-Pro:unix_commands Deepika$ less -N hello.txt
RKs-MacBook-Pro:unix_commands Deepika$ less -NX hello.txt
      1 Hello, world! I'm doing my UNIX assignment.
      2 It is the complementary of head command.
      3 The tail command prints the last n units of data of the given input.
      4 By default, it prints the last 10 lines of the specified files.
      5 If more than one file name is provided then data from each file is preceded by its file name.
      6
RKs-MacBook-Pro:unix_commands Deepika$
```

ln

The ln command is a command utility used to create a hard link or a symbolic link (symlink) to an existing file. The use of a hard link allows multiple filenames to be associated with the same file since a hard link points to the inode (file/directory) of a given file, the data of which is stored on disk. On the other hand, symbolic links are special files that refer to other files by name.

Usage

`ln [option] [source_file] [target_file]`

Options

`-b`

`-s`

```
RKs-MacBook-Pro:unix_commands Deepika$ pwd
/Users/Deepika/Documents/ttn/unix_commands
RKs-MacBook-Pro:unix_commands Deepika$ ls -l
total 32
-rw-r--r--  1 Deepika  staff  4096 Jul  8 18:48 deep.tar
-rw-r--r--  1 Deepika  staff    14 Jul  8 18:42 deepika.txt
-rw-r--r--  1 Deepika  staff    47 Jul  8 18:39 goodbye.txt
-rw-r--r--  1 Deepika  staff   313 Jul  8 19:43 hello.txt
RKs-MacBook-Pro:unix_commands Deepika$ cat hello.txt
Hello, world! I'm doing my UNIX assignment.
It is the complementary of head command.
The tail command prints the last n units of data of the given input.
By default, it prints the last 10 lines of the specified files.
If more than one file name is provided then data from each file is preceded by its file name.

RKs-MacBook-Pro:unix_commands Deepika$ ln hello.txt hey.txt
RKs-MacBook-Pro:unix_commands Deepika$ ls -l
total 40
-rw-r--r--  1 Deepika  staff  4096 Jul  8 18:48 deep.tar
-rw-r--r--  1 Deepika  staff    14 Jul  8 18:42 deepika.txt
-rw-r--r--  1 Deepika  staff    47 Jul  8 18:39 goodbye.txt
-rw-r--r--  2 Deepika  staff   313 Jul  8 19:43 hello.txt
-rw-r--r--  2 Deepika  staff   313 Jul  8 19:43 hey.txt
RKs-MacBook-Pro:unix_commands Deepika$ cat hey.txt
Hello, world! I'm doing my UNIX assignment.
It is the complementary of head command.
The tail command prints the last n units of data of the given input.
By default, it prints the last 10 lines of the specified files.
If more than one file name is provided then data from each file is preceded by its file name.

RKs-MacBook-Pro:unix_commands Deepika$
```