Basic Linux/Unix Commands with Examples and Syntax

File Management becomes easy if you know the right basic command in Linux.

Sometimes, commands are also referred as “programs” since whenever you run a command, it’s the corresponding program code, written for the command, which is being executed.

Let’s learn the must know Linux basic commands with examples:

What You Will Learn

Listing files (ls)

If you want to see the list of files on your UNIX or Linux system, use the ‘**ls’** command.

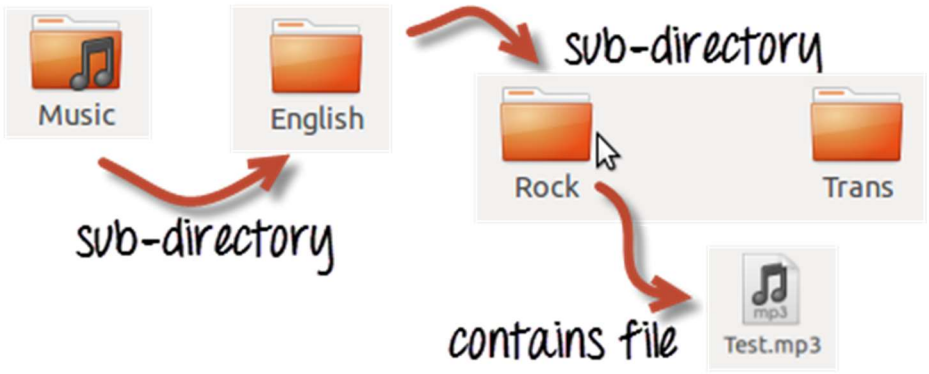
It shows the files /directories in your current directory.

```
guru99@VirtualBox:~$ ls
Desktop    Downloads         Music     Public     Videos
Documents  examples.desktop  Pictures  Templates
guru99@VirtualBox:~$
```

Note:

- Directories are denoted in blue color.
- Files are denoted in white.
- You will find similar color schemes in different flavors of Linux.

Suppose, your “Music” folder has following sub-directories and files.



You can use **‘ls -R’ to shows all the files not only in directories but also subdirectories**

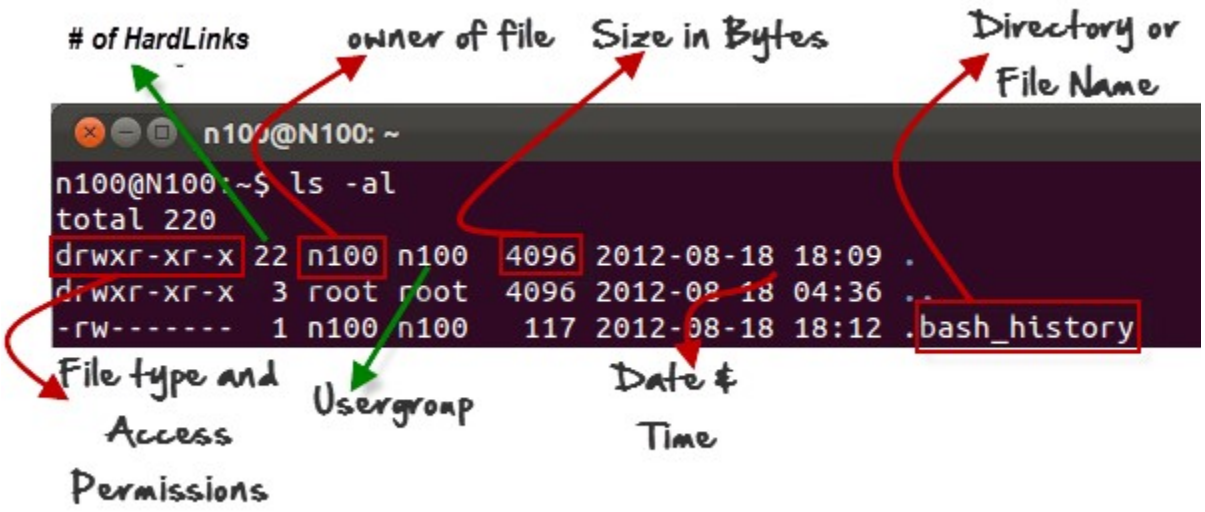
```
guru99@VirtualBox:~$ ls -R
.:
Desktop    Downloads          Music     Public     Videos
Documents  examples.desktop   Pictures  Templates

./Desktop:

./Documents:

./Downloads:

./Music:
English

./Music/English:
Rock  Trans

./Music/English/Rock:
Test.mp3

./Music/English/Trans:

./Pictures:

./Public:

./Templates:

./Videos:
guru99@VirtualBox:~$
```

NOTE: These Linux basics commands are case-sensitive. If you enter, "**ls – r**" you will get an error.

'ls -al' gives detailed information of the files. The command provides information in a columnar format. The columns contain the following information:

1st Column	File type and access permissions
2nd Column	# of HardLinks to the File
3rd Column	Owner and the creator of the file
4th Column	Group of the owner
5th Column	File size in Bytes
6th Column	Date and Time
7th Column	Directory or File name

Let's see an example –



Listing Hidden Files

Hidden items in UNIX/Linux begin with – • "period" symbol at the start, of the file or directory.

Any Directory/file starting with a '.' will not be seen unless you request for it. To view hidden files, use the command.

```
ls -a
```

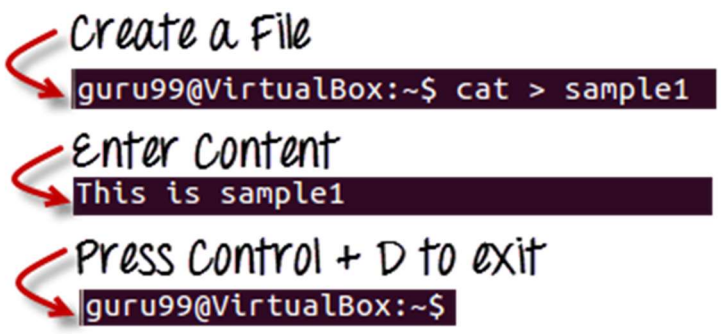
```
guru99@VirtualBox:~$ ls -a
.                .dmrc              .ICEauthority     sample
..               Documents          .local            sample1
.bash_history    Downloads          .mission-control  sample2
.bash_logout     examples.desktop   Music             Templates
.bashrc          .gconf             Pictures          .thumbnails
.cache           .gnome2            .profile          Videos
.config          .gstreamer-0.10    Public            .Xauthority
.dbus            .gtk-bookmarks     .pulse            .xsession-erro
Desktop          .gvfs              .pulse-cookie
guru99@VirtualBox:~$
```

Creating & Viewing Files

The 'cat' server command is used to display text files. It can also be used for copying, combining and creating new text files. Let's see how it works.

To create a new file, use the command

1. cat > filename
2. Add content
3. Press 'ctrl + d' to return to command prompt.



How to create and view files in Linux/Unix

To view a file, use the command –

```
cat filename
```

Let's see the file we just created –

```
guru99@VirtualBox:~$ cat sample1
This is sample1
```

Let's see another file sample2

```
guru99@VirtualBox:~$ cat > sample2
This is sample2
```

The syntax to combine 2 files is –

```
cat file1 file2 > newfilename
```

Let's combine sample 1 and sample 2.

```
guru99@VirtualBox:~$ cat sample1 sample2 > sample
```

As soon as you insert this command and hit enter, the files are concatenated, but you do not see a result. This is because **Bash Shell (Terminal) is silent type**. Shell Commands will never give you a confirmation message like "OK" or "Command Successfully Executed". It will only show a message when something goes wrong or when an error has occurred.

To view the new combo file "sample" use the command

```
cat sample
```

```
guru99@VirtualBox:~$ cat sample
This is sample1
This is sample2
```

Note: Only text files can be displayed and combined using this command.

Deleting Files

The 'rm' command removes files from the system without confirmation.

To remove a file use syntax –

```
rm filename
```

How to delete files using Linux/Unix Commands

Moving and Re-naming files

To move a file, use the command.

```
mv filename new_file_location
```

Suppose we want to move the file “sample2” to location /home/guru99/Documents. Executing the command

mv sample2 /home/guru99/Documents

mv command needs super user permission. Currently, we are executing the command as a standard user. Hence we get the above error. To overcome the error use command.

```
sudo command_you_want_to_execute
```

Sudo program allows regular users to run programs with the security privileges of the superuser or root.

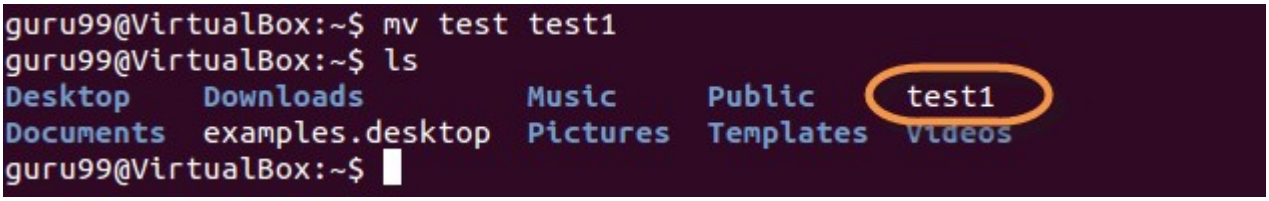
Sudo command will ask for password authentication. Though, you do not need to know the root password. You can supply your own password. After authentication, the system will invoke the requested command.

Sudo maintains a log of each command run. System administrators can trackback the person responsible for undesirable changes in the system.

```
guru99@VirtualBox:~$ sudo mv sample2 /home/quru99/Documents
[sudo] password for guru99: ****
guru99@VirtualBox:~$
```

For renaming file:

```
mv filename newfilename
```



NOTE: By default, the password you entered for sudo is retained for 15 minutes per terminal. This eliminates the need of entering the password time and again.

You only need root/sudo privileges, only if the command involves files or directories not owned by the user or group running the commands

Directory Manipulations

Directory Manipulation in Linux/Unix

Enough with File manipulations! Let's learn some directory manipulation Linux commands with examples and syntax.

Creating Directories

Directories can be created on a Linux operating system using the following command

```
mkdir directoryname
```

This command will create a subdirectory in your present working directory, which is usually your "Home Directory".

For example,

```
mkdir mydirectory
```

```
home@VirtualBox:~$ mkdir mydirectory
home@VirtualBox:~$ ls
Desktop    Downloads         Music        Pictures  Templates
Documents  examples.desktop  mydirectory  Public    Videos
home@VirtualBox:~$
```

If you want to create a directory in a different location other than 'Home directory', you could use the following command –

```
mkdir
```

For example:

```
mkdir /tmp/MUSIC
```

will create a directory 'Music' under '/tmp' directory

```
home@VirtualBox:~$ mkdir /tmp/MUSIC
home@VirtualBox:~$ ls /tmp
keyring-yCD2no  pulse-Ob9vyJcXyHZz  ssh-SSSsjczv1036      virtual-home.HaC7Mw
MUSIC           pulse-PKdhtXMmr18n  unity_support_test.1
home@VirtualBox:~$
```

You can also create more than one directory at a time.

```
home@VirtualBox:~$ mkdir dir1 dir2 dir3
home@VirtualBox:~$ ls
Desktop  dir2  Documents  examples.desktop  Pictures  Templates
dir1     dir3  Downloads  Music             Public    Videos
home@VirtualBox:~$
```

Removing Directories

To remove a directory, use the command –

`rmdir directoryname`
Example

`rmdir mydirectory`
will delete the directory mydirectory

```
home@VirtualBox:~$ rmdir mydirectory
home@VirtualBox:~$ ls
Desktop  dir2  Documents  examples.desktop  Pictures  Templates
dir1     dir3  Downloads  Music             Public    Videos
home@VirtualBox:~$
```

Tip: Ensure that there is no file / sub-directory under the directory that you want to delete. Delete the files/sub-directory first before deleting the parent directory.

```
home@VirtualBox:~$ rmdir Documents
rmdir: failed to remove `Documents': Directory not empty
home@VirtualBox:~$
```

Renaming Directory

The ‘mv’ (move) command (covered earlier) can also be used for renaming directories. Use the below-given format:

`mv directoryname newdirectoryname`
Let us try it:

```
home@VirtualBox:~$ mv mydirectory newdirectory
home@VirtualBox:~$ ls
Desktop    Downloads         Music        Pictures  Templates
Documents  examples.desktop  newdirectory  Public    Videos
home@VirtualBox:~$
```

How to rename a directory using Linux/Unix Commands

Other Important Commands

The 'Man' command

Man stands for manual which is a reference book of a Linux operating system. It is similar to HELP file found in popular software.

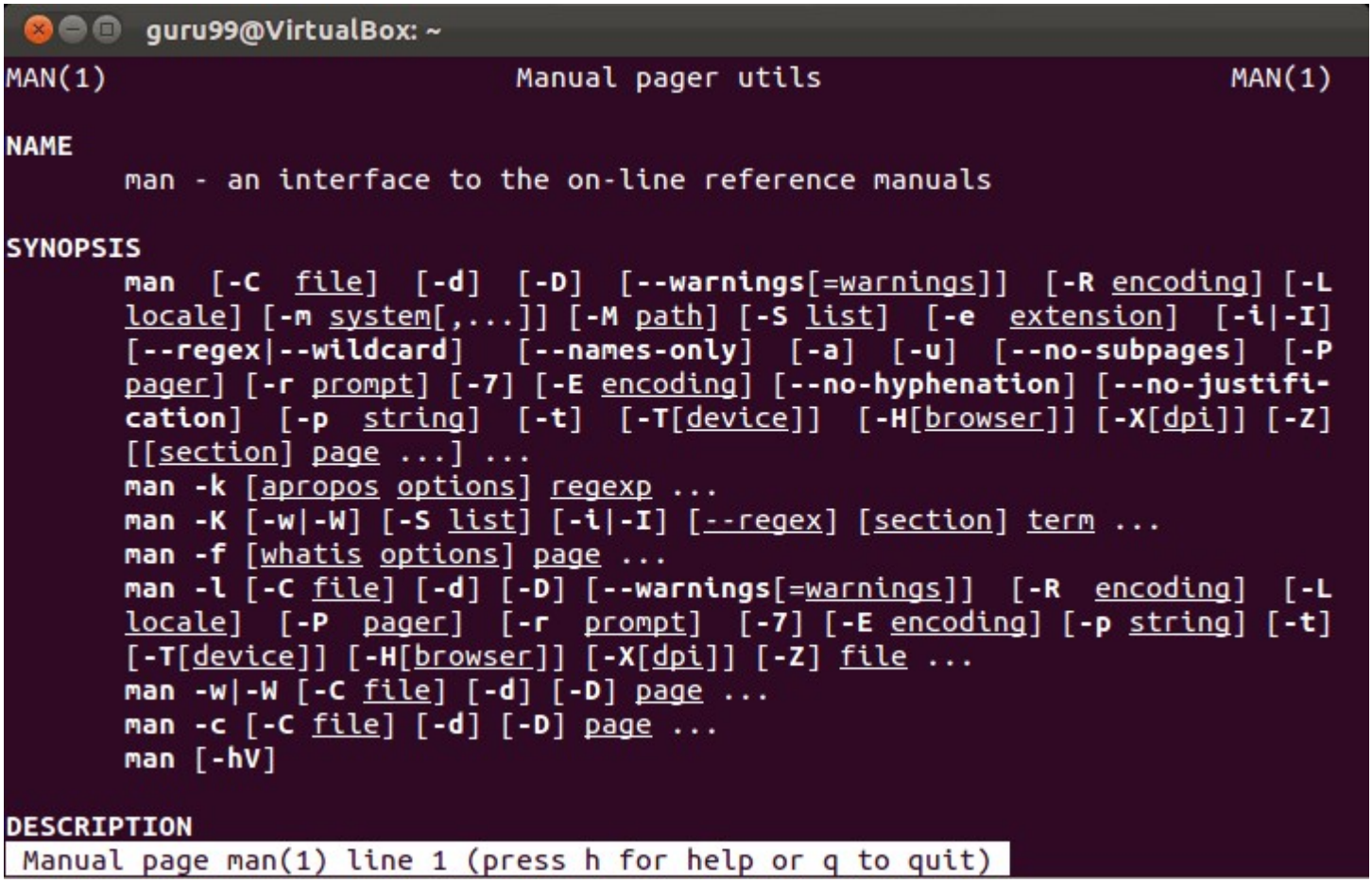
To get help on any command that you do not understand, you can type

```
man
```

The terminal would open the manual page for that command.

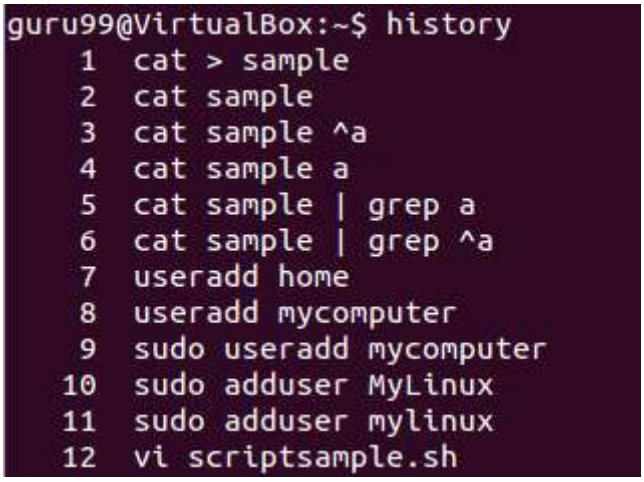
For an example, if we type *man man* and hit enter; terminal would give us information on man command

```
guru99@VirtualBox:~$ man man
```



The History Command

History command shows all the basic commands in Linux that you have used in the past for the current terminal session. This can help you refer to the old commands you have entered and re-used them in your operations again.



The clear command

This command clears all the clutter on the terminal and gives you a clean window to work on, just like when you launch the terminal.

Linux Command List

Below is a Cheat Sheet of Linux/ Unix basic commands with examples that we have learned in this Linux commands tutorial

Command	Description
ls	Lists all files and directories in the present working directory
ls – R	Lists files in sub-directories as well
ls – a	Lists hidden files as well
ls – al	Lists files and directories with detailed information like permissions, size, owner, etc.
cat > filename	Creates a new file
cat filename	Displays the file content
cat file1 file2 > file3	Joins two files (file1, file2) and stores the output in a new file (file3)
mv file "new file path"	Moves the files to the new location
mv filename new_file_name	Renames the file to a new filename
sudo	Allows regular users to run programs with the security privileges of the superuser or root
rm filename	Deletes a file
man	Gives help information on a command

Command	Description
history	Gives a list of all past basic Linux commands list typed in the current terminal session
clear	Clears the terminal
mkdir directoryname	Creates a new directory in the present working directory or a at the specified path
rmdir	Deletes a directory
mv	Renames a directory