File & Directory Management in Linux

In the Linux file system, as with its predecessor **UNIX**, everything is a file: data files, binary files, executable programs, even input, and output devices.

These files are placed in a series of directories that act like file folders.

A directory is nothing more than a special type of file that contains a list of other files/directories.

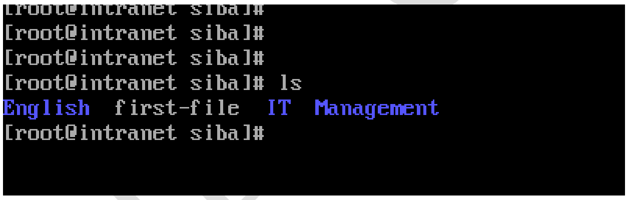
These files and directories are used to create a hierarchical structure that enables logical placement of specific types of files.

Let's learn how to navigate and interact with Linux the file system.

Listing the contents of a Directory:

The ***ls*** command lists the content of the current directory. A number of options can be used with ***ls***, gives more information.

[root@intranet siba]# ls – display all the content inside the SIBA directory.

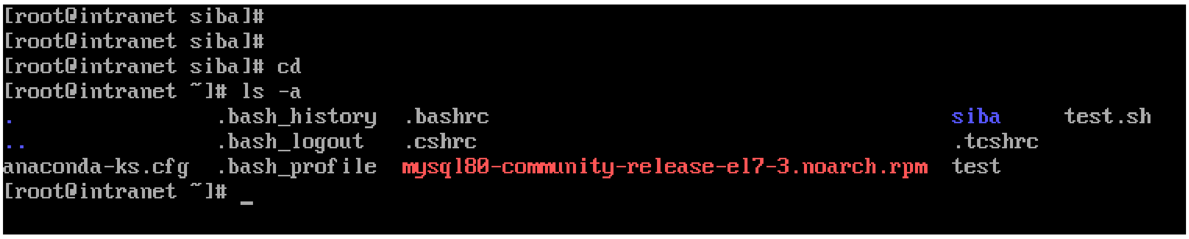


```
[root@intranet siba]#
[root@intranet siba]#
[root@intranet siba]#
[root@intranet siba]# ls
English  first-file  IT  Management
[root@intranet siba]#
```

However, using ***ls*** has some limitations.

First, it does not show hidden files. Hidden files use filenames that start with a period (.) as the first character. They are often used for configuration of specific programs and are not accessed frequently. For this reason, they are not included in a basic directory listing. You can see all the hidden files by adding a **switch** to the command like this:

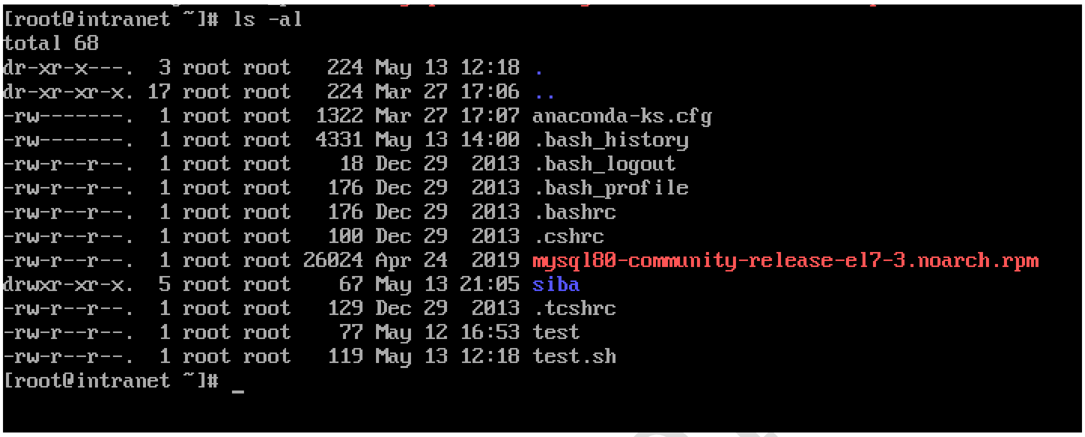
[root@intranet ~]# ls -a



```
[root@intranet siba]#
[root@intranet siba]#
[root@intranet siba]# cd
[root@intranet ~]# ls -a
.                .bash_history  .bashrc                                     siba     test.sh
..               .bash_logout   .cshrc                                      .tcshrc
anaconda-ks.cfg  .bash_profile  mysql80-community-release-el7-3.noarch.rpm  test
[root@intranet ~]# _
```

There is still more information available about each item in a directory. To include details such as the **file/directory permissions**, **owner** and **group**, as well as the **size**, and the **date** and **time** it was last modified, enter the below command:

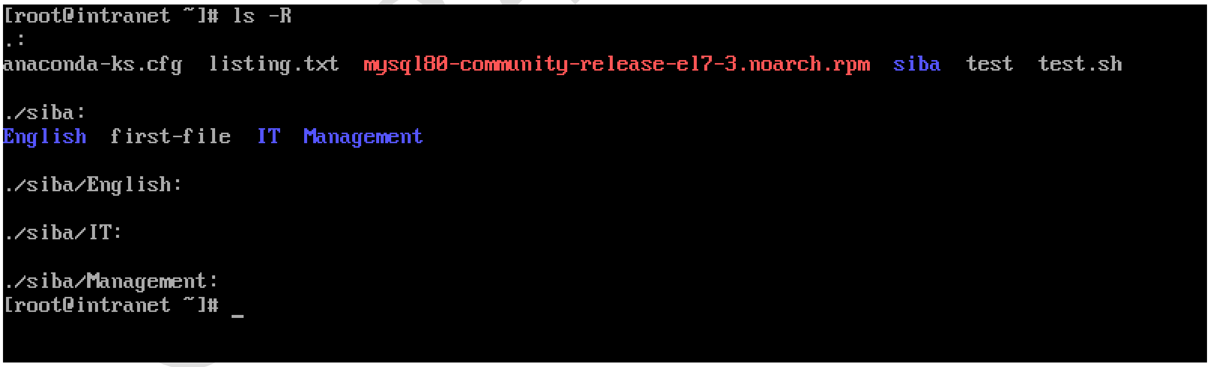
```
[root@intranet ~]# ls -al
```



```
[root@intranet ~]# ls -al
total 68
dr-xr-x---.  3 root root   224 May 13 12:18 .
dr-xr-xr-x. 17 root root   224 Mar 27 17:06 ..
-rw-------.  1 root root  1322 Mar 27 17:07 anaconda-ks.cfg
-rw-------.  1 root root  4331 May 13 14:00 .bash_history
-rw-r--r--.  1 root root    18 Dec 29  2013 .bash_logout
-rw-r--r--.  1 root root   176 Dec 29  2013 .bash_profile
-rw-r--r--.  1 root root   176 Dec 29  2013 .bashrc
-rw-r--r--.  1 root root   100 Dec 29  2013 .cshrc
-rw-r--r--.  1 root root 26024 Apr 24  2019 mysql80-community-release-el7-3.noarch.rpm
drwxr-xr-x.  5 root root    67 May 13 21:05 siba
-rw-r--r--.  1 root root   129 Dec 29  2013 .tcshrc
-rw-r--r--.  1 root root    77 May 12 16:53 test
-rw-r--r--.  1 root root   119 May 13 12:18 test.sh
[root@intranet ~]# _
```

Another useful switch is **-R**:

```
[root@intranet ~]# ls -R
```



```
[root@intranet ~]# ls -R
.:
anaconda-ks.cfg  listing.txt  mysql80-community-release-el7-3.noarch.rpm  siba  test  test.sh

./siba:
English  first-file  IT  Management

./siba/English:

./siba/IT:

./siba/Management:
[root@intranet ~]# _
```

This command scans and lists all the contents of the subdirectories of the current directory. This is likely to be a lot of information, so better output to a text file. Use the below command:

```
[root@intranet ~]# ls -R > filename.txt
```

Changing Directories with *cd*:

Use the ***cd*** command to move one directory to another within the file system. You can remember it as “change directory”.

`[root@intranet ~]# cd siba` – looks for the directory named '***siba***' and change the prompt as below;

`[root@intranet siba]#`

You can also specify an exact location for a directory, like below:

`[root@intranet ~]# cd /home/indika` – looks for the '***indika'*** directory inside the '***home***' directory.

The ***cd*** command can also be used with several shortcuts. For example, to quickly move up to the ***parent*** directory, the one above the one you are currently in, use the ***cd*** command like this:

`[root@intranet IT]# cd ..`

To return to your home directory from anywhere in the Linux file system, use the ***cd*** command like this:

`[root@intranet siba]# cd`

You can also use the $***HOME*** shell environment variable to accomplish the same thing. Type this command and press Enter to return to your home directory:

`[root@intranet etc]# $HOME`

You can accomplish the same thing by using the ***tilde (~)*** like this:

`[root@intranet IT]# cd ~`

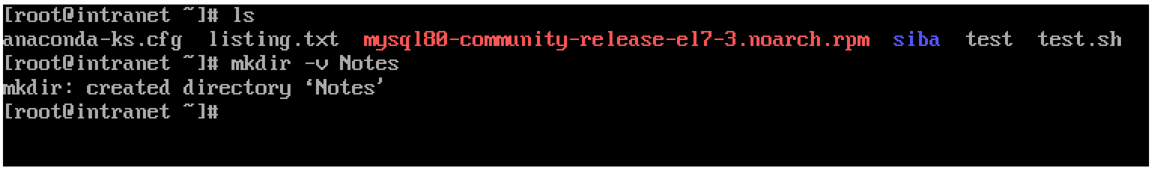
How to create a Directory:

`[root@intranet ~]# mkdir Programming` – will create a Directory named 'Programming'.

Specify the path to create a Directory in a different location as below:

`[root@intranet ~]# mkdir /etc/var/www/intranet` – will create the '***intranet***' Directory inside '***www***' located inside '***var***' which is inside the '***etc***' Directory.

You can also use the **-v** while working with the ***mkdir*** command to print a useful message for each directory you create.



```
[root@intranet ~]# ls
anaconda-ks.cfg  listing.txt  mysql80-community-release-el7-3.noarch.rpm  siba  test  test.sh
[root@intranet ~]# mkdir -v Notes
mkdir: created directory 'Notes'
[root@intranet ~]#
```

In short the above output is very self-explanatory. It tells the user that a new directory has been created with the use of the ***mkdir*** command.

Removing files and directories using *rm* command:

The ***rm*** command is very useful when it comes to removing **files** or an **entire directory** including everything in it such as files and subdirectories.

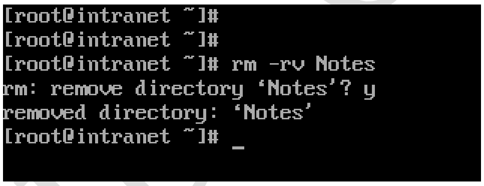
To remove a directory and their contents use the following command.

```
[root@instranet ~]# rm -r newdirectory
```

The **-r** option tells the ***rm*** command to remove the content of a directory recursively.

In order to get some information what is being done while using the ***rm*** command put the **-v** option in use. This helpful option usually stands for verbose.

```
[root@instranet ~]# rm -rv newdirectory
```



```
[root@intranet ~]#
[root@intranet ~]#
[root@intranet ~]# rm -rv Notes
rm: remove directory ‘Notes’? y
removed directory: ‘Notes’
[root@intranet ~]# _
```

By default, ***rm*** does not remove directories. Use **–r** or **–R** options to remove each listed directory, too, along with all of its content.

Another option which is being used with the ***rm*** command is the **-f** option. It forces the remove process and never prompts the user to ask if they are sure what they are doing. ***This option is not recommended when working with very important files.***