**Hard Link & Soft Link in Linux with Examples:**

**Hard Links:**

1. Hard Links have the same inodes number.
2. ls -l command shows all the links with the link column showing the number of links.
3. Links have actual file contents
4. Removing any link, just reduces the link count but doesn't affect the other links.
5. You cannot create a Hard Link for a directory.
6. Even if the original file is removed, the link will still show you the contents of the file.

**Example - Hard links**

Lets first create a "Test" directory and inside we create a new file "sample1".

$ mkdir Test

$ cd Test

$ touch sample1

Now, create a hard link to sample1. Name the hard link sample2.

$ ln sample1 sample2

Display inodes for both files using i argument of the ls command.

$ ls -il sample1 sample2

This is what you will get as output

1482256 -rw-r--r-- 2 bruno bruno 21 May 5 15:55 sample1

1482256 -rw-r--r-- 2 bruno bruno 21 May 5 15:55 sample2

From the output, you can notice that both sample1 and sample2 have the same inode number (1482256). Also, both files have the same file permissions and the same size.

Now Remove the original sample1

$ rm sample1

After removing Hard Link just have a look at the content of the "link" sample2.

$ cat sample2

You will still be able to see the contents of the file.

**Soft Links:**

1. Soft Links have different inodes numbers.
2. ls -l command shows all links with second column value 1 and the link points to the original file.
3. Soft Link contains the path for original file and not the contents.
4. Removing soft link doesn't affect anything but when the original file is removed, the link becomes a 'dangling' link that points to nonexistent file.
5. A Soft Link can link to a directory.

**Example - Soft Link (Symbolic links)**

Let's create a soft link for the file 'sample2' using the ln command.

$ ln -s sample2 sample3

Display inodes for both using i argument of ls command.

$ ls -il sample2 sample3

This is what you'll get:

1482256 -rw-r--r-- 1 bruno bruno 21 May 5 15:55 sample2

1482226 lrwxrwxrwx 1 bruno bruno 5 May 5 16:22 sample3 -> sample2

From the output, you can notice that the inodes are different and the symbolic link has an "l" before the rwxrwxrwx. The permissions are different for the link and the original file because it is just a symbolic link.

Now list the contents:

$ cat sample2

$ cat sample3

Now remove the original file

$ rm sample2

And then check the Test directory:

$ ls

It will still display symbolic link sample3 but if you try to list the contents, it will tell you that there is no such file or directory.

$ cat sample3

Now you know about some of the key differences between Hard Links and Soft Links that will make it easier for you to access files and run programs.