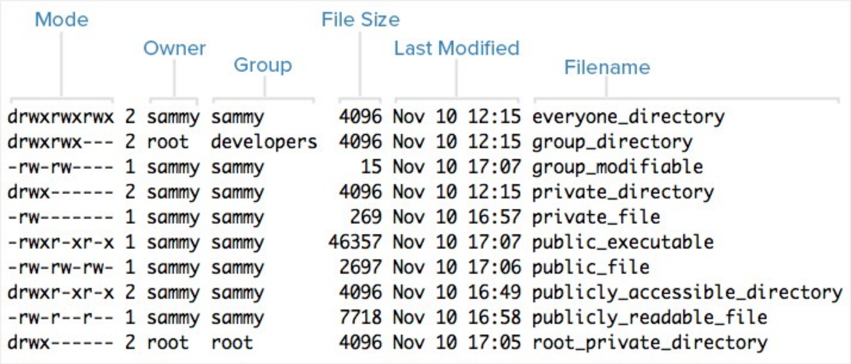
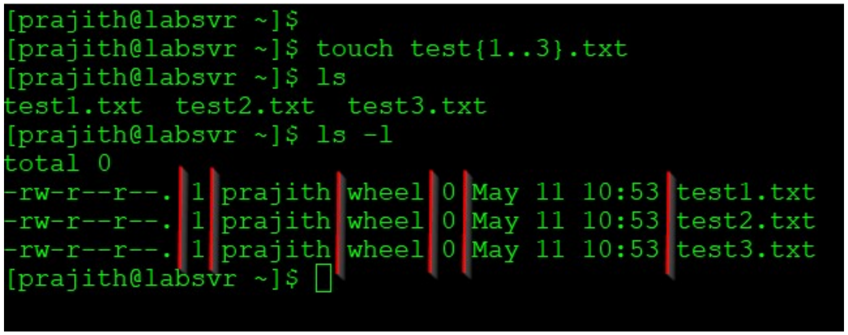
File Permissions

Understanding ls long listing (ls –l)

- **-rw-r--r-** permissions
- **1** : number of linked hard-links
- **sammy**: owner of the file
- **wheel**: which group this file belongs to
- **0**: size
- **May 11 10:53 :** modification/creation date and time
- **test1**: file/directory name



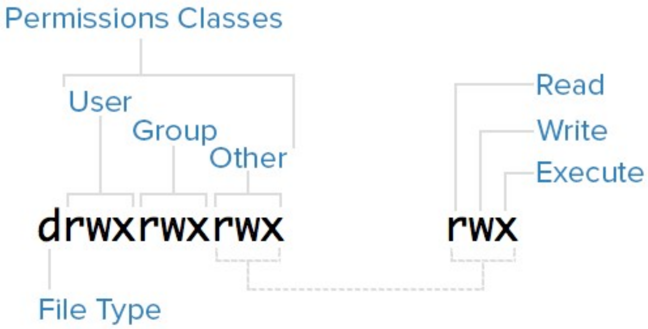
```
drwxrwxrwx 2 sammy sammy       4096 Nov 10 12:15 everyone_directory
drwxrwx--- 2 root  developers  4096 Nov 10 12:15 group_directory
-rw-rw---- 1 sammy sammy         15 Nov 10 17:07 group_modifiable
drwx------ 2 sammy sammy       4096 Nov 10 12:15 private_directory
-rw------- 1 sammy sammy        269 Nov 10 16:57 private_file
-rwxr-xr-x 1 sammy sammy      46357 Nov 10 17:07 public_executable
-rw-rw-rw- 1 sammy sammy       2697 Nov 10 17:06 public_file
drwxr-xr-x 2 sammy sammy       4096 Nov 10 16:49 publicly_accessible_directory
-rw-r--r-- 1 sammy sammy       7718 Nov 10 16:58 publicly_readable_file
drwx------ 2 root  root        4096 Nov 10 17:05 root_private_directory
```

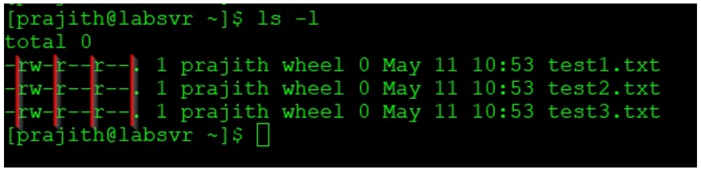


```
[prajith@labsvr ~]$
[prajith@labsvr ~]$ touch test{1..3}.txt
[prajith@labsvr ~]$ ls
test1.txt  test2.txt  test3.txt
[prajith@labsvr ~]$ ls -l
total 0
-rw-r--r--. 1 prajith wheel 0 May 11 10:53 test1.txt
-rw-r--r--. 1 prajith wheel 0 May 11 10:53 test2.txt
-rw-r--r--. 1 prajith wheel 0 May 11 10:53 test3.txt
[prajith@labsvr ~]$
```

File Permissions

- **Read (r)**: The read permission allows the user to open the file and read its contents.
- **Write (w)**: The write permission allows the user to modify or change the contents of the file.
- **Execute (x)**: File execute permission
- **l,d,c** in the first field stands for link, directory and character file respectively





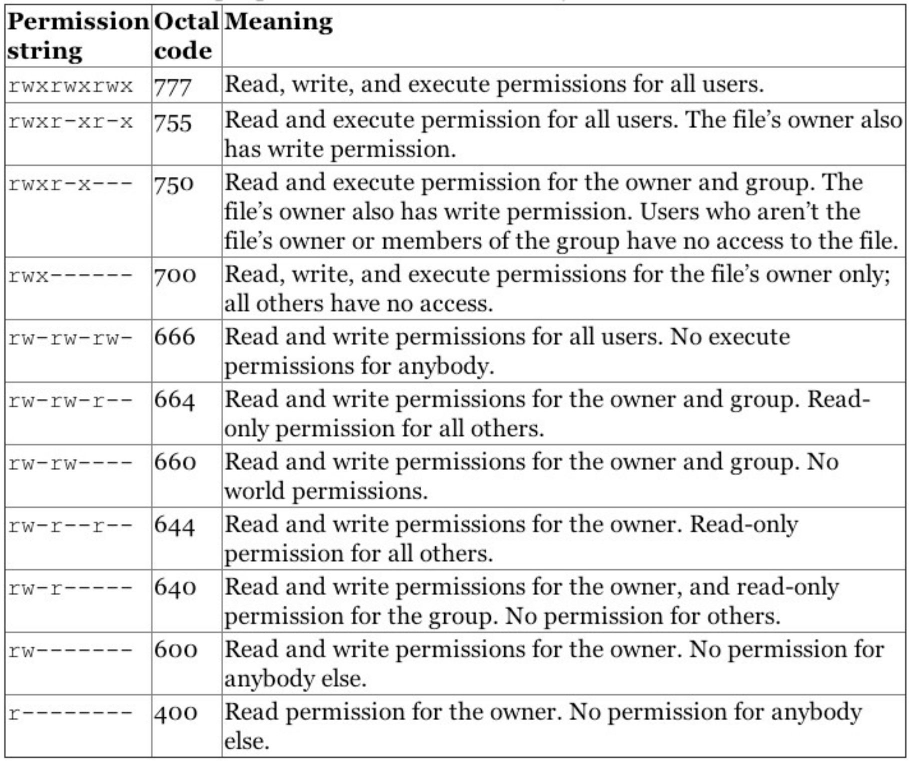
```
[prajith@labsvr ~]$ ls -l
total 0
-rw-r--r--. 1 prajith wheel 0 May 11 10:53 test1.txt
-rw-r--r--. 1 prajith wheel 0 May 11 10:53 test2.txt
-rw-r--r--. 1 prajith wheel 0 May 11 10:53 test3.txt
[prajith@labsvr ~]$ 
```

File Permissions

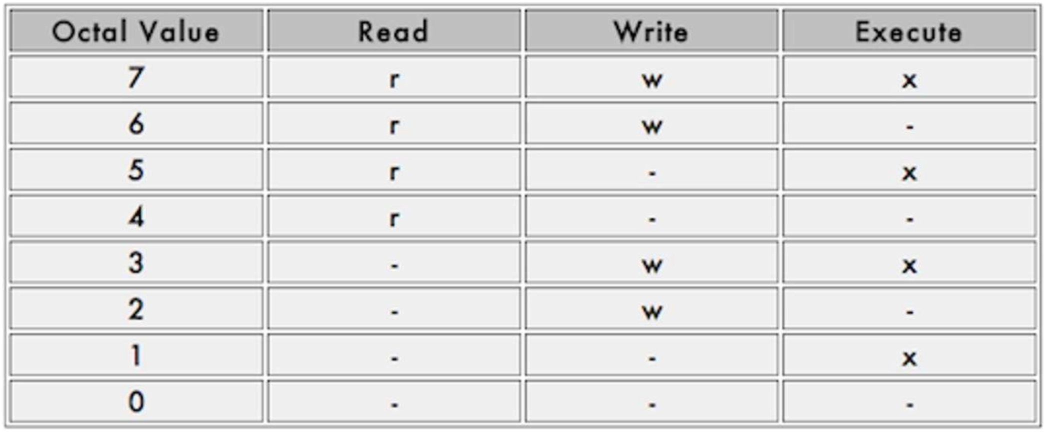
-rw-------:	A file that is only accessible by its owner
-rwxr-xr-x:	A file that is executable by every user on the system. A “world-executable” file
-rw-rw-rw-:	A file that is open to modification by every user on the system. A “world-writable” file
drwxr-xr-x:	A directory that every user on the system can read and access
drwxrwx---:	A directory that is modifiable (including its contents) by its owner and group
drwxr-x---:	A directory that is accessible by its group

File Permissions

- File permission are represented in terms of octal value



Permission string	Octal code	Meaning
rwxrwxrwx	777	Read, write, and execute permissions for all users.
rwxr-xr-x	755	Read and execute permission for all users. The file's owner also has write permission.
rwxr-x---	750	Read and execute permission for the owner and group. The file's owner also has write permission. Users who aren't the file's owner or members of the group have no access to the file.
rwx------	700	Read, write, and execute permissions for the file's owner only; all others have no access.
rw-rw-rw-	666	Read and write permissions for all users. No execute permissions for anybody.
rw-rw-r--	664	Read and write permissions for the owner and group. Read-only permission for all others.
rw-rw----	660	Read and write permissions for the owner and group. No world permissions.
rw-r--r--	644	Read and write permissions for the owner. Read-only permission for all others.
rw-r-----	640	Read and write permissions for the owner, and read-only permission for the group. No permission for others.
rw-------	600	Read and write permissions for the owner. No permission for anybody else.
r--------	400	Read permission for the owner. No permission for anybody else.



Octal Value	Read	Write	Execute
7	r	w	x
6	r	w	-
5	r	-	x
4	r	-	-
3	-	w	x
2	-	w	-
1	-	-	x
0	-	-	-

Additional Permissions

+t mode (sticky bit)

+s mode (setuid bit)

- **Sticky bit or +t mode:**
- When set only the owner (or root) can delete or rename files within that **directory**, regardless of which users have write access to the directory by way of group membership or ownership. (not applicable for file)
- Example:
  - chmod +t /home/sam/testdir/

- **Setuid bit or +s mode:**
- When set on files allows users with permissions to execute a given file the ability to run that file with the permissions of file owner
- Example:
  - chmod g+s /home/sam/testdir1
  - Chmod u+s /home/sam/testdir2

Changing file Ownership - chown

- By default, all files are “owned” by the user who creates them and by that user’s default group
- To change the ownership use chown command
- Example:
  - chown user1:group1 /testdir/testfile1
  - chown –R user1:group1 /testdir/