**Devops**

**Linux:**

**pwd** - print working directory.

**mkdir**- used to create new folder

**mkdir -v file\_name**- “-v(verbose)” is used to print message.

**mkdir -p file\_name1/filename\_2**

🡪 - p means parent.

🡪 We can use v and p together also.

🡪 It based on your requirement.

**sudo su -** 🡪 It will switch to root user, and it will point to the root user home directory and will load the root user configurations.

**sudo su 🡪** It will switch to root user, and it will not point to the root user home directory and will not load the root user configurations.

**yum install package\_name -y**

🡪 this is the syntax to install any package.

🡪 -y is optional. If you keep -y, then it will install the package directly.

**touch file\_name** 🡪 It is used to create an empty file.

🡪 If the file\_name already exists, then data will not override, updated timestamp will set.

**mkdir -m 700 file\_name 🡪** It will create a file with some permissions.

**ls -lrt**

**🡪** ls means list the directory content

🡪 l means longlisting

🡪 r means reverse order

🡪 t access time

**ls -la** 🡪 It will show all files including hidden files.

**ls -lh** 🡪 human readable.

**Inode**:

Inode is data structure it can store the file/directory information. But it doesn’t contain filename.

**ls -li 🡪** It will show all files with inode number.

🡪 Each file and directory will have a unique inode number.

cd ~ (OR) cd🡪 It will take you to user home directory. User home directory will be **/home/ec2-user**.

cd - 🡪 It will take you to previous directory.

**rmdir** **file\_name** 🡪 It is used to delete an empty directory.

**rm -rf file\_name** 🡪 It will remove non-empty directory. Here r means recursively and f

means forcefully.

**vi file\_name** 🡪 The file will open in command mode. It doesn’t matter whether the file has existed or not. If the file has not existed, then it will create and open in command mode.

**nano ->**

**find 🡪** It is used to search and locate the list of files and directories based on conditions.

**find . -type f -empty 🡪** It will search all the empty files.

**find . -iname java.txt 🡪** It will search the java.txt file in the current directory and it will ignore the case.

**umask dir file**

**root 0022 755 644**

**user 0022 755 644**

Base permission for dir is : 0777-0022=0755

Base permission for file is: 0666-0022=0644

r ----read 4

w -----write 2

x -----execute 1

**umask** 🡪 This command is used to change the base permissions of file and dir.

**chmod ->** This command is used to change the permissions of a particular file or dir.

Syntax : chmod 222 file\_name

chmod -R 222 dir\_name -> it will apply for sub directories too.

chown owner\_name file\_name -> it will change the owner of the file if the user is available.

/ 🡪 root directory

/root 🡪 root user home directory

**chgrp group\_name file\_name 🡪**  It will change the group name of that file.

**cat /etc/group 🡪** It will print all the group names available in the server.

**cat\*.txt ->** it will print all text files

**cat -n file\_name**-> it will print with line numbers

**head file\_name** -> It will print first 10 lines

**head file\_name** **-15 file\_name** -> it will print 15 lines

**tail file\_name**-> it will print last 10 lines of the file

**sed -n “100p” filename** -> it will print 100th line only.

**sed -n “100,105p” filename** -> It will print 100 to 105th line

**more file\_name –** it will print the data up to screen filled after that it will wait until you enter. If you enter it will print next line and if you enter cntl+f it will print next page.

**cp file\_name destination\_path 🡪** use -r for folders

**mv file\_name destination\_path**

**wc file\_name 🡪**  It will print the count in the format of lines words characters.

**wc -w \*.txt** 🡪 It will print count of word from all txt files.

**Ln 🡪 Link**

|  |  |
| --- | --- |
| **Hard Link** | **Soft Link** |
| Of and lf will have same properties | Both will not have same properties |
| **ln file\_name** | **ln -s file\_name** |
| Both will contain same inode number. | Both will contain different inode number. |
| **Hard Link** will create only for files | **SoftLink** will create for both files and dir. |
| Even If we delete the original file, we can access hard link. | we can not access if we delete original file. |