|  |  |
| --- | --- |
| Branch | Command |
| Basics |  |
|  | * Root directory   + /Users     - /usr    ( usr / local )     - /Applications     - And many others.   + Extract : Tar -xzf filename.tar.gz   + Copy & Move :     - Copy :       * cp file.txt /backup/new\_file.txt       * cp -R Pictures Pictures\_backup         + <https://linuxize.com/post/cp-command-in-linux/>     - Move :       * mv file directory     - DELETE THE FILES IN  a directory       * **rm -R** <FOLDER\_NAME> ( to delete all the files in a folder)       * **rm -iR** <FOLDER\_NAME>( to get the confirmation for any deletion )       * rm MyFile.rtf ( specific file)   + Find files :     - find /Users/tkessler/Desktop/ -name myfile.txt   + Type in: say "anything here"     - Search file :       * Mdfind abc       * This will seach for files , folders , text as well       * Locate filename // search     - 2 different directory in mac       * Usr/local/opt       * /Users/kunip004/     - Home and end page :       * Home : Fn + left arrow       * End : Fn + Right arrow     - Back and Next Page :       * Back : cmd + left arrow       * Next : cmd + Right arrow   + Print screen :     - shift + cmd + 3     - Shift + cmd + 4 for capture portion of the screen.     - Shift + cmd + ctrl + 4 : to copy it to the clipboard.      * + - .bash\_profile       * Will be at User/kunip ( use ls -a ) to see the all files or hidden files.      * + - echo $java\_home // to get the java home       * Which java  // also help to see which one env is referring.      * + - Kill the process       * sudo lsof -i -P         + Get the process id       * kill <PID>         + <https://stackoverflow.com/questions/8428333/maven-eclipse-debug-jdwp-transport-dt-socket-failed-to-initialize-transport-in>       * Kill the ( windows task Mgr)         + Cmd + opt + ESC         + <https://appletree.or.kr/quick_reference_cards/Unix-Linux/CLI-Cheat-Sheet.pdf>      * + - * + <https://www.makeuseof.com/tag/mac-terminal-commands-cheat-sheet/>. (in bookmarks already)      * + - **WSL** : helps to do the linux commands in windows       * **Man** : helps to give the manual for the specific commands.       * Pwd ( print working directory )       * Ls ( list of context )       * Ls <with\_FolderName\_>  ( Ls prajesh/docs)       * Ls -l ( with formatted folder and also the permission details )       * Ls -l -a ( shows the hidden folder as well)       * Cd ( Change directory )       * Cd ..       * Cd /<fullPath>       * Cd / --> to root path       * Cd ~  // to go to the user home directory         + /Users/kunip004       * Touch (to create the file along with the timestamp )       * Rmdir ( to delete the files )       * Rm - v <filename>   // will list of files removed.       * Rm -r <foldername> // will delete the files inside the folder **recursively**         + Rm **-rv** <foldername> // (remove reclusively and let me view ) to delete and list the files which is removed.       * Open : // to open the file       * Open <fileName>       * MV // to move the file       * MV <fileName> <to\_destination\_Folder>       * CP ( Copy )       * CP <FileName> <to\_destination\_Folder> |
| 50 Commands Session | |  |  |  |  | | --- | --- | --- | --- | |  | input | output | notes | | 1 | Whoami //To see who as logged in | kunip004 |  | | 2 | Man whoami //manual something similar to help     hostname  Hostname -I     yum install <software\_name> -y    Sudo su- |  |  | |  | //regarding env variables     * -printenv or env // will show all the env variables * Echo $<particular\_Variable> //example : echo $PATHc, echo $JAVA\_HOME * -- * /usr/local/bin //this is a best place for developer to place the installer , other place we need to run sudo + cmds * ----- * File to store this env variable is * A) bash profile b) ./zshrc  //based on the shell/terminal screen whatever it shows * ----- * //setting the env variables inside the bash/zshrc profile to the **PATH to reflect the path**   + // export K=v ,then get the exported using ${exported\_Key\_Name} and set to the PATH * Examples :   + **Export terraform\_home = /usr/local/bin**     - ${terraform\_home}   + **PATH = ${PATH}: ${terraform\_home}**   + **Export path**     - Now we can do , echo $PATH in our terminal |  |  | | 3 | Clear or ctrl+l or cmd+l  // to clear the screen |  |  | | 4 | Pwd //Prints working directory |  |  | | 5 | Ls   ls –l //-l : will see long format (list permission,DT  ls –la //-a : hidden  **Ls /bin/prajesh. // Navigate to this folder using ls itself**  **instead of cd** |  |  | | 6 | Cd ////Change directory    Cd .. //back to one folder  Cd ../../.. 3 parent folder  **Cd / --> root**  **Cd ~ -> to go to my user path (** /Users/kunip004) |  |  | | 7 | Mkdir // Make directory    mkdir summer winter. // many folder at a time  mkdir summer/seeds  // folders inside folder  **mkdir -p friuts/seeds/what**// -p  "parents" helps even the folder is   not there in the parent go and create it. |  |  | |  | rmdir zoo// to delete only empty directory  rm -r  zoo// to delete the directory with files in it ( recursively using -r)  **rm -ri  zoo//** to delete the file ( Confirm me using -i )    rm carrot cat dog //to delete 3 files silently  **rm -v carrot cat dog**// verify when we delete --> removed 'carrot' |  |  | | Move/  Copy | mv //will move or rename the file or folder.    mv <FileName/folderName> <toNewPath/Sampepath for rename>  mv -v cake cookie pie stuff/ --> moving all the three files inside stuff folder    Cp <fileName.txt> <newFileName.txt> //Copy just file  Cp -r <foldernmae> <to new folderName> //Copy folder with -r else will complain |  |  | | Touch and open | Touch <filename>  // will create/update the file  touch sample1.pdf sample2.png  -------------  Open //to open any file which is not empty in mac   Open .   open sample1.pdf |  |  | | Date   > / >> | Date // to represent the current date and has lot of functions to play with date    > // to write/overwrite the file   >> // to append the file. |  |  | | Head / tail            Cat              less            wc            Sort                          uniq count ( group by ) | Head <fileNamewithPath>  <100>    tail <fileNamewithPath>     **tail <fileNamewithPath>   -f    // will not only show the tail portion of the file but we can see**  **any action on that machine ( like admin audit)**      Cat filename  **Cat –n filename** // view the file content with numbers  Cat <fileName1> <CatFileName2> //to view both the files in a screen , we can apply sort   on top of this.      **less <filename>** // to open the file in a editor   /<searchKeyword> // to search for a string using backslash  to go end of the file : G and to go start of the file : g      //Summary of the files  **word count ( no of lines , no of words , no of bytes ) in a file**  **wc <filename> -> 1757  22222  8787 <fileName>**   WC with pipe --> see next column for example      sort <fileName>.   // this will not change the orginal file , just sort the view only.  sort –n  <fileName>// sort integers   sort –U  <fileName>// avoid duplicate  Sort –R  <fileName>//reverse or desc sort  //sort with other function  **Cat <fileName1> <fileName2>  | sort  //both the cat filename will pipe and pass to sort function**  **sort –nu <fileName> | wc -l**    **//scenario , read the file , sort it and write it.**        **uniq actually take the duplicates from the adjacent line** not the whole file , so we need to use   along with sort , then unique to get the whole files remove duplicate     also there was an option –U ( see above with sort ) in sort but uniq allows more options to us     uniq –d  // to remove record which is duplicate , will not return unique record.  uniq –U  // to get only unique  Uniq – c // get the count of unique --> see example next column | **kunip004@C02Z81L4LVDT User360 % ls -l**  **total 136**  **drwxr-xr-x  4 kunip004  staff    128 Apr 25 08:24 output**  **-rw-r--r--@ 1 kunip004  staff  59768 Apr 25 07:06 plugin\_datasources.json**  **drwxr-xr-x  3 kunip004  staff     96 Apr 25 07:37 user data models**  **-rw-r--r--@ 1 kunip004  staff   6234 Apr 25 08:24 user360v2.txt**  **kunip004@C02Z81L4LVDT User360 % ls -l | wc**  **5      40     272**  **-----------------------------------**  **//Sort and Uniq**      kunip004@C02Z81L4LVDT User360 % sort icecreamsurvey.txt  chocolate  chocolate  strawberry  strawberry  strawberry  strawberry  vanila  vanila  vanila  vanila  vanila  vanila  kunip004@C02Z81L4LVDT User360 % sort icecreamsurvey.txt **| uniq -c**     2 chocolate     4 strawberry     6 vanila  kunip004@C02Z81L4LVDT User360 % sort icecreamsurvey.txt **| uniq -c | sort -nr**     6 vanila     4 strawberry     2 chocolate |  | | echo | // helps to print something , also helps to echo something before running the actual linux commands  //echo ~  -- **echo the value of the key** , here ~ returns /USER/Kunip004  //echo $PATH or echo $ User – to run the env variables    //**echo with symbols**  //\*.txt   -- echo ls -l \*.txt  //\*.??  -- ls -l \*.??on    /**/echo to try  and see before running the actual commands**  //echo {a,b,c}.txt -- this will  return a.txt b.txt c.txt files , then we can run touch {a,b,c}.txt | j |  | | diff | Diff helps to compare 2 files  //diff <fileName1> <fileName2>     //-y we can compare side by side   //diff -y <fileName1> <fileName2> | //-y  kunip004@C02Z81L4LVDT User360 % diff -y sentSort.txt sortSentence.txt  this is a test for the sentence       | hi how are you doing        > hi how are you doing        > hi how are you doing        > hi how are you doing        > hi how are you doing        > hi how are you doing |  | | **Find Files/Folder:**                                              **Search Content**  grep | **================ search file/folder=================**  //to find the folder or file in a specificed location   //we can also search the folder/file either             type  Name   Size     //search both the folder and file using **find name:**  find <location> -name '\*u\*' -- to search for  the folder and file name "\*u\*" in the current location     //Specific file = f , directory=d search using **find type and size**  //find . **-type f**name "\*.txt" --size +100k  -- to search for the type f which has the name "\*.txt" in the current location      //we can also**execute the command using fine**  **//find . -type  f –exec cat{} \; // under the current location , any file view it using cat command.**      **================ search Content=================**  // to search the file content where as find is to search for file or folder     grep <recursively> {string\_to\_search} {location}  grep –rn "user\_id" . -- when the search has to display with the line number using "n" |  |  | | **Du ( disk utility)  vs Df (disk free )** | DU // disk utility    Du .    Du –h  -- human readable form | kunip004@C02Z81L4LVDT User360 % du -h .  1.4M ./user data models  12M ./output/final  12M ./output  13M .    kunip004@C02Z81L4LVDT User360 % df -h .  Filesystem     Size   Used  Avail Capacity iused      ifree %iused  Mounted on  /dev/disk1s1  466Gi  204Gi  247Gi    46% 2142174 4880310706    0%   /System/Volumes/Data |  | | **history** | //view my history commands    kunip004@C02Z81L4LVDT User360 % **history**  1188  ls -l  1192  grep "user\_id" .  1193  ls -l  1194  grep -rn "user\_id" .  1203  df -h .    //view my history commands in less editor so I can search for it using "\"  kunip004@C02Z81L4LVDT User360  history | less  //else use grep command to search for  kunip004@C02Z81L4LVDT User360 % **history | grep 'df'**  1201  df .  1202  df -h.  1203  df -h .    //once query is identified run the query using the id  kunip004@C02Z81L4LVDT User360 **% !1193** |  |  | | **PROCESS:**  PS                            TOP                    KILL                  JOBS | Ps (process status )  //to view  the process status  **kunip004@C02Z81L4LVDT User360 % ps**    PID TTY           TIME CMD  49129 ttys000    0:00.64 -zsh  1671 ttys001    0:00.06 /bin/zsh --login -i  1497 ttys005    0:00.07 /bin/zsh --login -i   // ps axww – to view all the process including the system process       Top  wiil show the max no of CPU / Memory which process is using  PID    COMMAND      %CPU TIME     #TH    #WQ  #PORT MEM    PURG   CMPRS  269    WindowServer 22.4 70:38.77 10     4    2845  581M   2020K+ 72M  36460  Google Chrom 12.9 48:21.85 22     1    342   412M-  0B     50M  1285   Google Chrom 10.9 39:45.90 18     1    621   548M   0B     123M    //vew the kill options the process   kill –l | less // to see the diff kill operations    // kill by processID  kill <PID>  kill –9 <PID> //to kill the process very brutely (-9)    // killall by processName  Killall –9 <proceeName>   -- killall –9 node      jobs   fg   bg |  |  | | Gzip        gunZip        tar | //to zip the single file    Gzip –k <fileName to zip>    //to unzip the single file  Gunzip –k <fileName to zip>      //to extract the .tz file  Tar –xzf <.tarfileName>    -C : create   -F : fileName  -X : xtract  -Z : compress and archive   **-t : view**  **-k : keep** |  |  | | Nano vs VIM | Nano editor helps to create and modify the file .    Nano <fileName> //if file name is not there then it will create it.  Ctrl + w : search   Ctrl + s : save  Ctrl + g : help |  |  | | network | How to find the ipadress of the machine.    **curl ipinfo.io**  **curl ipinfo.io/ip //**to find only ip address |  |  | | **chmod** | //to add the permission (exectute ) to a file , do below     |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | | 7 | rwx |  | 6=rw- ,   5= r-X  3=-wx | | 4 | r |  |  | |  |  |  |  |   chmod +x <path\_to\_fileName>    Chmod 777 <path\_to\_fileName>    Chmod 735 <folder\_Name>  **//rwx-wx-r-x**   D = directory   * = file   L = linked file  N = network file |  |  | | **filetype** | .Zshrc -> to run the zsh shell  .bashrc -> to run the bash shell    To edit : vim / vi -> insert ->:wq  To run : **source**~/.zshrc    Now we can validate the change , like echo $PATH. |  |  | | **Network** | [Network Troubleshooting Commands for Linux | Linux Network Commands](https://www.youtube.com/watch?v=t3gXEk3789s)          IfConfig : to gwt the ipadress of the machine and what are all the network interface we have in our machine like ethernet  **curl ipinfo.io**  **curl ipinfo.io/ip //**to find only ip address    Curl  :// hit the request for the server ( test the rest api)  Curl  –X <Operation\_Name>  -d param1=value or—data "k=v@k2=v2"  -url <http://www.google.com>    Traceroute <ipaddress> : // to understand whats happening with the specific ipadress    Ping : to see the response from the website/server     // to know ipadress , nsname ,responsetime etc  //Netstat  //Dig <www.ggogle.com>.  //Nslookup <www.ggogle.com> |  |  | |
| Create Vs View |  |
|  | * Vi  or vim : create/edit * **Touch** : create * Cat : view * Nano : |
| java | //to find where java or any installation to know where it is installed   whereis <app\_name>  which <app\_name>    whereis java  Which java |
| Env |  |
|  | * To get the ip address of the machine   + curl ipinfo.io (<https://opensource.com/article/18/5/how-find-ip-address-linux)> |
| Other | * Parquet tools query :   + parquet-tools cat --json part-00000-06470c74-bd4b-4671-93a6-6973664ebb38.c000.snappy.parquet |
|  | * History  ( to audit the list of query history )   + Control + R -> will open the new search window , start typing what we want like |
| Short-cuts |  |
|  | * To minimize the screen : cmd + M * Command+shift  + Arrow = to select the entire line * Quit : cmd + Q * Tab : Cmd + tab * For task manager or **activity monitor by typing ("Activity")**   + - Graphical user interface, text, application        Description automatically generated * force quit option  , apple menu -> force quit.   + Graphical user interface, text, application, chat or text message      Description automatically generated * To maximize or minimize the screen : CMD+  or CMD-      * **Note : to see hidden files in Macbook**   + In Finder, open up your Macintosh HD folder   + Press **Command+Shift+Dot**   + Your hidden files will become visible. Repeat step 2 to hide them again!   + <https://setapp.com/how-to/show-hidden-files-on-mac#:~:text=See%20hidden%20files%20on%20Mac%20via%20Finder&text=In%20Finder%2C%20open%20up%20your,hidden%20files%20will%20become%20visible>.      * **Note :  Create SSH Key in Mac :**   Cmd to use : ssh-keygen  Then just enter , enter , till last setp.    Copy the rsa\_pub file : pbcopy < ~/.ssh/id\_rsa.pub   Paste in the GIT ssh key option     * **Note : for java related commands**   + $JAVA\_HOME and $PATH for env variables   + Java -version or javac -version or     - which java       * /usr/local/opt/openjdk@8/bin/java   + (For detail ) java -XshowSettings:properties -version   + My local bash file details ( bash\_profile ) |