UNIX File System & Permissions

* Give the execute permission for the user for a file chap1:  
  bash  
    
  chmod u+x chap1
* Give execute permission for user, group, and others for a file add.c:  
  bash  
    
  chmod a+x add.c
* Remove the execute permission from user, give read permission to group and others for a file aa.c:  
  bash  
    
  chmod u-x,go+r aa.c
* Give execute permission for users for a.c, kk.c, nato, and myfile using a single command:  
  bash  
    
  chmod u+x a.c kk.c nato myfile
* Change the directory to root directory. Check the system directories, like bin, etc, usr, etc.:  
  bash  
    
  cd /

ls bin etc usr

Using Pipes and Filters

* Redirect the content of the help document ls, into a file called as lsdoc:  
  bash  
    
  ls --help > lsdoc
* Display the content of the lsdoc page wise:  
  bash  
    
  less lsdoc
* Display only the first 4 lines of the lsdoc file:  
  bash  
    
  head -n 4 lsdoc
* Display only the last 7 lines of the file lsdoc:  
  bash  
    
  tail -n 7 lsdoc
* Remove the file lsdoc:  
  bash  
    
  rm lsdoc
* There will be B’day celebration from the friends file, find how many B’day parties will be held. If two of the friends have the B’date on the same day, then we will be having one party on that day:  
  bash  
    
  awk '{print $3}' friends | sort | uniq | wc -l
* Display the lines starting with Ma, in the file friends:  
  bash  
    
  grep '^Ma' friends
* Display the lines starting with Ma, ending with i or ending with id, in the file friends:  
  bash  
    
  grep '^Ma.\*\(i\|id\)$' friends
* Print all the files and the directory files from the current directory across all the subdirectories, along with its path:  
  bash

find . -type f

* Print only the Directory files:  
  bash  
    
  find . -type d

* Display the files starting with chap, along with its path:  
  bash  
    
  find . -type f -name 'chap\*'
* Sort the file friends in ascending order of names:  
  bash  
    
  sort friends
* Display the contents of the file friends in uppercase letters:  
  bash  
    
  cat friends | tr '[a-z]' '[A-Z]'
* Store the contents of your home directory in a file called dir:  
  bash  
    
  ls ~ > dir
* From the above file dir, display the file permissions and the name of the file only:  
  bash  
    
  ls -l $(cat dir) | awk '{print $1, $9}'
* From the same dir file, store only the file names in a file called files:  
  bash  
    
  ls -l $(cat dir) | awk '{print $9}' > files
* From the same dir file, store only the permissions of files in a file called perms:  
  bash  
    
  ls -l $(cat dir) | awk '{print $1}' > perms
* From the same dir file, store only the file sizes in a file called sizes:  
  bash  
    
  ls -l $(cat dir) | awk '{print $5}' > sizes
* Display the file names, sizes, and permissions from your directory in that order:  
  bash  
    
  ls -l | awk '{print $9, $5, $1}'
* Display the number of users working on the system:  
  bash  
    
  who | wc -l
* Find out the smallest file in your directory:  
  bash  
    
  ls -lS | tail -n 1
* Display the total number of lines present in the file friends:  
  bash  
    
  wc -l friends
* Write a command sequence that prints out date information in this order: time, day of the week, day number, month, year (e.g., 13:44:42 IST Sun 16 Sept 1994):  
  bash

date "+%T %A %d %b %Y"

* Write a command sequence that prints the names of the files in the current directory in descending order of the number of links:  
  bash  
    
  ls -l | sort -k2 -n -r | awk '{print $9}'
* Write a command sequence that prints only names of files in the current working directory in alphabetical order:  
  bash  
    
  ls -1 | sort
* Write a command sequence to print names and sizes of all the files in the current working directory in order of size:  
  bash  
    
  ls -lS | awk '{print $9, $5}'
* Determine the latest file updated by the user:  
  bash  
    
  ls -lt | head -n 1