FILE SYSTEM

man

- **man** is the system's manual viewer.
- Used to display manual pages, scroll up and down, search for occurrences of specific text and other useful functions.
- Syntax:
  - **man [command name]**
  - Displays the information about following command.

man

- Man pages are generally written by the developer of the corresponding program.
- The man pages are divided into number of sections.
- Every section has a unique number and contains only a specific type of man pages.
- The following table shows various sections:

Organization of man documentation

SECTIONS	SUBJECT
1	Executable programs or commands
2	System calls (functions provided by the kernel)
3	Library calls (functions provided by the library)
4	Special files
5	File formats and conventions (configuration files)
6	Games
7	Macro packages and conventions
8	System administration commands

man

- man page consists of several sections:

1. Name
2. Synopsis
3. Configuration
4. Description
5. Options
6. Exit status
7. Return value
8. Errors
9. Versions
10. Bugs
11. Examples and so on.

pwd

- stands for Print Working Directory.
- It prints the current directory name with the complete path starting from root (/).
- This command is built in shell command and is available on most of the shell – bash, Bourne shell, ksh,zsh, etc.
- pwd displays the absolute pathname
- **pwd syntax**

  pwd [*OPTION*]...

echo

- Used in two ways:

1. To display a message (like echo hi)
2. To evaluate shell variables (like echo $SHELL)

- Escape sequences are also used by echo commands.
  - Example: \a, \n, \f, \t, \b, etc.

ls : LISTING DIRECTORY CONTENTS

- Gives complete list of filenames in the current directory arranged in ASCII collating sequences (numbers first, uppercase and then lowercase), with one filename in each line.

- Includes directories also.

ls

Commands	Descriptions
ls	To list directory contents
ls –a	To show all the hidden files(current ‘.’ and parent directories “..” are also shown)
ls –1	To display One File Per Line
ls –l	To display total information about Files/Directories
ls –A	To show all the hidden files except current ‘.’ and parent directories “..”
ls –lt	To order files based on last modified time

ls

Command	Description
ls –lrt	To Order Files Based on Last Modified Time in Dec-ending Order
ls –m	To Stream output format; files are listed across the page, separated by commas
ls *	To List all subdirectories
ls –hide=*.txt	To make ls hide particular type of files
ls *.txt	Display the files which have extension txt
ls -lS	To sort files with Size

ls

Commands	Descriptions
ls –F	To Do Visual Classification of Files /→ for directory @ → link file Nothing → normal file
ls –r	To display files/directories in reverse order. (By default ls command display files in alphabetically)
ls –X	To sort ls command output based on file extensions
ls –s –h	To make ls display only filenames and file sizes in output
ls –d */	To show directories only

mkdir: MAKING DIRECTORIES

- Directories are created with mkdir command followed by name of directory.
- Sometimes system refuses to create directory:

  Following are the reasons:

  - Directory name may already exist.
  - There may be ordinary file by that name in the current directory.
  - Permission set for the current directory don’t permit the creation of files and directories by the user.

mkdir

COMMAND	DESCRIPTION
mkdir dirname	Creates a new directory
mkdir –m 777 dirname	Create the directory and set its permissions
mkdir dir1 dir2 dir3	to create multiple directories at one time
mkdir -p home/test/test1/test2/test3/test4	to create several subdirectories at one time:
mkdir --version	To know the version

cd : CHANGING THE CURRENT DIRECTORY

- To move from one file system to another, use cd command.
- When used with an argument, it changes the current directory to the directory specified as argument.
- cd can also be used without an argument.
- When cd invoked without argument, it simply reverts to its home directory.

cd

COMMAND	DESCRIPTION
cd /usr/local	Change from current directory to /usr/local.
cd /usr/local/lib	Change from current directory to /usr/local/lib using absolute path.
cd lib	Change from current working directory to /usr/local/lib using relative path.
cd ..	Change Current directory to parent directory.

cd

COMMAND	DESCRIPTION
cd ../..	Change working directory to parents parent directory or two levels up in the directory structure
cd ../../..	changing three levels up in the directory
cd -	Move one directory back from where you are now.
cd .	Change working directory to present working directory.

rmdir : REMOVING DIRECTORIES

- Command is used to remove directories.
- It can delete more than one directory in one slot.
- Two important rules to be followed:

1. Can’t delete a directory with rmdir unless it is empty.
2. Can’t remove a subdirectory unless one is positioned in its parent directory or hierarchically above the one chosen to remove.

rmdir

COMMAND	DESCRIPTION
rmdir DIRNAME	Delete Empty Directories
rmdir -p dir1/dir2/dir3	Delete Nested Empty Directories
rm -rf DIRNAME	Delete Directory Which has Content
rm -ir DIRNAME	Deleting a directory recursively & interactively
rm -i FILENAME	Deleting a file interactively.

touch

- Touch command is used to:
  - create,
  - change and
  - modify timestamps of a file.

- **Syntax:**

  touch options expression filename(s)

touch

COMMAND	DESCRIPTION
touch filename	Create an Empty File
touch file1 file2 file3	Create Multiple Files
touch {A..Z} or touch {1..20} or touch {1..20}.txt	Create lots and lots of files
touch -c filename	Avoid Creating New File
touch -m filename	Change File Modification Time
touch -am filename	Change access and modification time together

expr : COMPUTATION AND STRING HANDLING

Performs two main functions:

1. Performs arithmetic operations on integers.
2. Manipulates strings.

- expr can handle only integers, division yields only integral part.

expr

COMMAND	DESCRIPTION
expr 8 + 4	Addition of two numbers
expr 8 – 4	Subtraction of two numbers
expr 8 / 4	Division of two numbers
expr 8 % 4	Modulus of a number
expr 8 * 4	Multiplication of two numbers

expr: STRING HANDLING

- For manipulating strings, **expr** uses two expression separated by a colon.

- String to be worked upon is placed on the left of the : and regular expression is placed on its right.

expr: STRING HANDLING

- expr can perform three important string functions:
  - Determine the length of the string
  - Extract a substring.
  - Locate the position of a character in a string.

expr: STRING HANDLING

COMMAND	DESCRIPTION
expr “abcdefgh” : ‘.*’	‘.*’ signifies to expr that it has to print number of characters matching the pattern. It returns length of the string=8
s=2003 expr “$s” : ‘..\(..\)’	Extracts last two characters
s=abcdefghi ; expr “$s” : ‘[^d]*d’	Locating the position of a character. Location of d is 4
expr linux : lin **Ans: 3**	returns the number of characters matched
$ expr linux : linx **Ans: 0**	the condition is string 2 entirely should match in string 1.
expr linux : '.*‘ **Ans: 5**	to match any number of characters