

Practical I.

Q1] Write a brief description of the following commands, syntax, example & options (any 4).  
 touch, cat, cp, ls, mkdir, man, clear, echo, cd.

→ i) touch

desc - Updates the timestamp of a file or creates an empty file if it doesn't exist.  
 Syntax - touch [options] file...

Eg: touch myfile.txt  
 creates an empty file named myfile.txt or updates its timestamp if it already exists

Options:

- a : Changes only the access time.
- m : changes only the modification time.
- c : Prevents creation of a file if it doesn't exist.
- t : Specifies a timestamp

ii) cat

desc - Concatenates and displays the content of files to the standard output.

Syntax: cat [options] [file...]

Eg. cat file.txt  
 Displays the contents of file.txt

**Options:**

- n: Numbers all output lines
- b: Numbers only non-blank lines
- s: Suppresses repeated empty output lines
- E: Displays a \$ at the end of each line

ii) cp

**desc:** Copies files or directories from a source to a destination.

**Syntax:** cp [options] source destination.

**Example:**

cp[options] source destination.  
cp file.txt to the /backup/ directory.

**Options:**

- r: Copies directories recursively.
- i: Prompts before overwriting files.
- p: Preserves file attributes (e.g. permissions, timestamps).
- v: Displays verbose output, showing files being copied.

iv) ls

**desc:** Lists directory contents  
**Syntax:** ls [optional] [directory ...]

**Eg:** ls -l /home  
Lists the contents of the /home directory in long

format.

**Options:**

- l: Displays detailed listing (long format).
- a: Shows hidden files (starting with .).
- h: Displays file sizes in a human-readable format.
- t: Sorts files by modification time (newest first)

v) mkdir

**desc:** Creates one or more directories.

**Syntax:** mkdir [options] directory ...

**Eg.**

mkdir my\_folder

Creates directory named my\_folder.

**Options:**

- p: Creates parent directories as needed.
- m: Sets the permission mode (e.g. mkdir -m 755 my\_folder)
- v: Prints a message for each created directory.
- context: Sets the permission mode (e.g. mkdir -m SELinux security context (if applicable)).

vi) man

**desc:** Creates one or more Displays the manual page for a command.

**Syntax:** man [options] command

**Eg:** man ls

Shows the manual page for ls command.

EXPERIMENT No \_\_\_\_\_

Page No. \_\_\_\_\_  
Date \_\_\_\_\_

EXPERIMENT No \_\_\_\_\_

Page No. \_\_\_\_\_  
Date \_\_\_\_\_

#### Options:

- f: Displays a more description (equivalent to whatis)
- m: Searches for keywords in manual pages (equivalent to apropos)
- n: Displays all matching manual pages
- v: Shows the location of the manual page file.

#### v) clear

clear: clears the terminal screen.

System: clear [options]

Eg.

clear

clears the terminal display.

Options:

- t: specifies the terminal type (e.g.: clear-ttyw)
- n: clears the screen without clearing the scrollback buffer
- v: Displays the version of the clear command.
- h: Displays help information

#### vii) echo

echo: outputs text or variables to the standard output

System: echo [options] [string...]

Eg. echo "Hello world!"

Prints Hello world into the terminal

Options:

- '\n': Enters the trailing newline.
- e: Enables interpretation of backslash escapes (e.g. \n for newline).
- E: Disables interpretation of backslash escapes (default).
- version: Displays the version of the echo.

#### vi) cd

cd: changes the current working directory

System: cd [directory]

Eg.

cd home/user/docs

Changes the current directory to /home/user/docs

Options:

(No specific options; behaviour depends on shell, e.g. Bash):

- cd -: switches to the previous directory.
- cd ~: changes to the user's home directory.
- cd ..: moves up one directory level.
- cd: without arguments, changes to the home directory.

Teacher's Sign: \_\_\_\_\_

Sundaram

Teacher's Sign: \_\_\_\_\_

EXPERIMENT: No.

Page No.  
Date

EXPERIMENT: No.

Page No.  
Date

- 1) touch pencils.txt  
→ touch creates an empty file if it doesn't exist.  
mkclz college
  - 2) touch collegel(a.txt, b.txt, c.txt)
  - 3) printf "%s\n" Ram Shyam Mohan Riya Siya John  
→ since Bob Neha Zara > buckly.
  - 4) cp buckly college /
  - 5) rm college /ram
  - 6) ls college
  - 7) -l college
  - 8) ls -a
  - 9) printf "%s\n" Goldlement Tokel Thenes > buckly
  - 10) touch college /  
{mat, sat, eat, pat, flat, nat, contact, format, orange,  
go, too}
  - a) cd college  
ls ?? ?Deep -F '^a\$'
- 1) a) touch fence.txt  
→ touch creates an empty file if it doesn't exist.
  - 2) b) rm college
  - 3) Display all files ending with at  
→ ls \*at
  - 4) List all filenames one screen at a time  
→ ls less
  - 5) List all filenames with 2 characters, 3 charach.  
→ ls ?? ??
  - 6) List all filenames starting with lowercase letters  
→ ls [aeiou]\*
  - 7) List all filenames with the last character as 'a' or  
'b' or 'c' or 'd'.  
→ ls \*[abcd]
  - 8) List all filenames ends with lowercase letter vowels  
→ ls \*
  - 11) Find the path in which you are currently  
working.  
→ pwd

Teacher's Sign.: \_\_\_\_\_

undaram

Teacher's Sign.: \_\_\_\_\_

EXPERIMENT

No.

Page No.  
Date

EXPERIMENT

No.

Page No. 9  
Date 06/07/25

- 1.) Create a file called dailyroutine with some data.  
 Create another file weekend routine with some data.  
 Combine both files into single file called combinedroutine catfile1>newfile

→ cat > dailyroutine  
 hello

cat > weekendroutine  
 world

cat dailyroutine weekendroutine > combinedroutine  
 cat combinedroutine

→ Hello world.

- 2.) Create main directory projects & inside it, create a Linux, & android subdirectories in a single command

→ mkdir -P assignments [os, linux, android]  
 → mkdir -P projects [ { os, linux, android } ]  
 → ls projects  
 android linux os

- 3.) Display contents of combinedroutine along with line numbers.

→ cat -n combinedroutine  
 1. Hello  
 2. world

Ajyay 17/11

Teacher's Sign

## Practical 2

### Basic Linux Commands

Part I - File System commands : touch, cat, cp, rm, mv.

- a1.) Create 5 files with the name a1, b2, c3, d4, e5 by using  
 → touch a1 b2 c3 d4 e5

- a2.) Create 3 files with the name f6, g7, h8 using  
 cat command with some meaningful information  
 with atleast 5 lines each.

a) cat > f6

[ctrl+d]

b) cat > g7

[ctrl+d]

c) cat > h8

[ctrl+d]

Teacher's Sign

Teacher's Sign

EXPERIMENT: No.

Page No.  
Date

Q3. concurrence the elements of f6 & g7 to a file called new1 (without creating new1)

- a. cat f6 > new1
- b. cat g7 >> new1

Q4. Append the contents of hs to new1  
→ cat hs >> new1

Q5. copy content of f6 to a1,g7 to b2 , hs to c3 by using cp command

- a. cp f6 a1
- cp g7 b2
- cp hs c3

Q6. Create 2 directories with the name dd1 & dd2  
→ mkdirs dd1 dd2

Q7. Copy files a1 & b2 to dd1 & f6 & g7 to dd2.

- a. cp b2 dd1
- b. cp g7 dd2

Q8. Remove files from a1,b2 from dd1  
→ rm a1 b2

Q9. Remove the directory along with its contents  
→ rm -r dd1

EXPERIMENT: No.

Page No.  
Date

Q10. Rename the files f6, g7, hs by newf6, newg7, newhs using mv command

- a. mv f6 newf6
- b. mv g7 newg7
- c. mv hs newhs

Q11. move the files newf6, newg7 to dd2.  
→ mv newf6 newg7 dd2

#### Part II:

ls & ls with options:

- 1) To list all filenames:  
ls
- 2) To list all filenames with on screen at a time.  
ls | more
- 3) To list all filenames with 2,3 characters  
→ ls 9\* and ls 7\*
- 4) To list all filenames with 2,4 characters at the same time.  
ls 11: ??!!
- 5) To list all filenames starting with vowel.  
ls [aeiou]\*

Teacher's Sign.:

Sundaram

Teacher's Sign.:

EXPERIMENT : No.

Page No.  
DatePage No.  
Date

- 7) To list all filenames -with last exactly 2 characters in which second character is a vowel.  
 ls ? [aeiou].  
 ↗ ls -a
- 8) To list all filenames starting with characters a:  
 ↗ ls a\*
- 9) To list all 4 characters filenames starting with characters a and third character is b:  
 ls [a][b]?
- 10) To list all filenames whose first character is any thing other than an alphabet in the range d to m.  
 ls [!d-m]\*
- 11) To list all filenames whose first character is 'm' or 'r' or is in the range e to f or v to z.  
 → ls [mr|efv-z]\*
- 12) Construct a command to display the total no. of files within tree chord.  
 ls -l | wc -l
- 13) To construct a command to display the total no. of files with exactly 2 or 3 or 4 characters.  
 → ls ?? ???.wc -l
- 14) List the contents of the directory.  
 ls  
 ↗ ls -a
- 15) List the contents along with hidden files.  
 ls -a
- 16) List the contents along with the hidden files except . files  
 → ls -A
- 17) List all files with their attributes & permissions  
 → ls -l
- 18) List all files identifying directories & non files.  
 → ls -F
- 19) List all files with size  
 ls -s
- 20) List all files with according to size  
 ls -S
- 21) List all files in reverse order  
 ls -r
- 22) List all files in columns  
 ls -c

Teacher's Sign.:

Sundaram

Teacher's Sign.:

14/7/21

EXPERIMENT :

No.

Page No.	
Date	

Practical 3-

(Q1) Display using "find" command all the filenames under:

a) /usr/lib/bin one screen at a time  
→ find /usr/lib/bin -print | more

b) /usr/lib/bin beginning with lowercase "c".  
→ find /usr/lib/bin -name c\* -print

c) /usr/lib/bin in capital letters beginning with lowercase "c"  
→ find /usr/lib/bin -name C\* -exec 'tr '[a-z]'['A-Z]' "

d) /usr/lib/bin which are over 5k in size in uppercase  
→ find /usr/lib/bin -size 5k -exec 'tr '[a-z]'['A-Z]' "

(Q2) Display parts of file using head or tail command

a) Display and count all the lines in the etc/mime.types  
→ cat | etc/mime.types lwc -1

b) Display first 10 lines of the file.  
→ head -n 10 etc/mime.types

c) Display last 10 lines of the file.  
→ tail -10 etc/mime.types

(Q3) d) Display first 25 lines of the file  
→ head -n 25 etc/mime.types

e) Classify, Count & Compare files.

a) Find odd sized files types you have in the following directories

i) /etc

→ file /etc

ii) /bin

→ file /bin

b) Redirect etc listing to a new file etcfiles.txt  
→ file /etc > etcfiles.txt

c) Append the listing of /usr/bin to etcfiles.txt  
→ file /usr/bin >> etcfiles.txt

d) Construct a command to find out how many files are there in /usr/bin directory.

→ file /usr/bin lwc -1

(Q4) Sorting:

a) Sort the etcfiles.txt into reverse alphabetical order on the first field.

→ sort -t " " -r -k 1 etcfiles.txt

Teacher's Sign. \_\_\_\_\_

Teacher's Sign. \_\_\_\_\_

EXPERIMENT:

No.

Page No.	
Date	

EXPERIMENT:

No.

Page No.	
Date	

- b) Repeat the first sorting ignoring case differences  
→ sort -t " " -k 1,1,4 file1

- c) Sort on second files (file type)  
→ sort -t " " -k 2 etcfiles.txt

- d) Find out how many English txt files are listed in etcfiles.txt  
→ grep 'En-c' 'English' etcfiles.txt

- (Q5) Create a file using vi editor editor with the following concepts.

→ vi file1  
Mahesh Deshpande 234  
Naresh Naik 431  
Allen D'Souza 121  
Hari Kuttan 231  
Rakesh Dubey 231  
Akshay Das 256

- a) Sort on first names only.  
→ sort -t " " -k 1 file1

- b) Sort on last names only.  
sort -t " " -k 1,1,4 file1

- c) Sort on first 4 characters only.

→ sort on -t " " -k 1,1,4 file1

- d) Sort on their numbers only  
→ sort -t " " -k 2 file1

- (Q6) Construct and execute the commands to create a file with the name stud <20110> with the following fields separated by a blank space having the below mentioned values:

FORMATS

→ vi stud01

101 Aman Sharma 12-05-2002 489  
102 Rahul Mehta 03-04-2001 523  
103 Priya Sinha 22-08-2003 411  
104 Neha Reddy 11-11-2002 550  
105 Karan Yadav 01-01-2000 475

- a) Sort the data on first names only  
→ sort -t " " -k 2 stud01

- b) Sort the data on marks  
→ sort -t " " -k 5 -n stud01

- d) Prepare a ranked merit list <> with students first & last name & only store in a file merit <20110> & display its content  
→ sort -t " " -k 1 -n stud01 | cut -d " " -f 2 stud01 > merit01

Teacher's Sign. :

Teacher's Sign. :

Teacher's Sign. :

EXPERIMENT:

No.

Page No	
Date	

- Q1) Construct the commands & execute them to
- Create a file fsp01 having the listing of atleast 50 lines  
\$ ls /usr/bin > fsp01
  - Display first 2 lines of the file & convert all the characters into capital letters.  
→ head -n 2 fsp01 | tr '[a-z]' '[A-Z]'
  - Display last 15 lines  
→ tail -15 fsp01
  - Display last 15 lines  
→ tail -rt fsp01
  - Display the lines starting with vowel.  
→ grep -i '^aeiou' fsp01
  - Split fsp01 into files with 20 lines of each  
→ split -l 20 fsp01 fsp
  - Split fsp01 into 3 subparts & display the line count.  
1. ls -1 /usr/bin | wc -l  
2. split -l 60 fsp01 fsp

EXPERIMENT:

No.

Page No	
Date	

- Q2) Brief description, syntax, Examples, Options.
- i) find
- Desc: Search for files & directories in a directory hierarchy.
- Syntax: find [path] [expression]
- Eg: find /home/kali-name \*txt"
- Options:
- name Search by file name (case sensitive)
  - iname Search by file name (case insensitive)
  - type f Search only files
  - type d Search only directories
  - size +1M Files larger than 1mb
  - mtime -7 modified within last 7 days
  - user kali Files owned by user "kali"
  - exec cmd {} ; Execute command on each file found.
- ii) tr
- desc: Translate or delete characters from stdin.
- Syntax: tr [Option] set1 set2
- Eg: echo "Hello" | tr 'a-z' 'A-Z'
- Options:
- d delete characters in SET1
  - s squeeze repeated characters
  - c complement the set
  - t Truncate SET1 to length of SET2.

Teacher's Sign. :

Teacher's Sign. :

EXPERIMENT: NO

Page No.  
Date

### i) head

Description: Displays the first N lines of a file

Syntax: head [option]... [file]...

Eg: head -n 5 sample.txt

Options:

- n N Shows first N lines (default is 10)
- c N Shows first N bytes
- q Quiet, no headers
- v Always show the file name.

### ii) tail

Description: Displays the last N lines of a file

Syntax: tail [option]... [file]...

Eg:

tail -n 5 sample.txt

Options:

- n N Shows last N lines
- c N Shows last N bytes
- f Follow file as it grows (useful for logs)
- q Quiet, suppress headers

### v) wc (word count)

Description: Counts lines, words, characters in a file

Syntax: wc [option]... [file]...

Eg: wc -l sample.txt

Options:

- l Count lines
- w Count words
- c Count bytes
- m Count characters

-L Display length of longest line.

### vi) file

Description: Detects file type based on content, not extension

Syntax: file [option]... file

Eg: file script.sh

Options:

- b Brief output (no filename)
- i Show mime type
- z Try to look inside compressed files
- f FilenamesRead filenames from a file.

Teacher's Sign.:

Teacher's Sign.:

EXPERIMENT:

No.

Page No.	
Date	

EXPERIMENT:

No.

Page No.	
Date	

### vii) sort

Desc: Sort lines of text files

Syntax: sort [options] ... [file] ...

Example: sort names.txt

Common Options:

- r Reverse sort
- n Numeric sort
- k N sort by column N
- u Remove Duplicate lines

### viii) Split

Desc: Splits a file into smaller parts

Syntax: split [option] ... [INPUT [Prefix]]

Eg:

split -15 sample.txt part -

Options:

- N split into chunks of N lines
- bN split into chunks of N bytes
- a use Numeric suffix instead of alphabet
- c size split at size boundaries, but max 2147 bytes

Teacher's Sign.:

Teacher's Sign.: