

End Term Exam

REPORT

Academic Year	2020-21
Name of Program	BCA – Data Science
Semester	1
Name of Course	Fundamental of UNIX
Student Name:	Sumit Pandey
URN No:	2020-B-13092002A
Assessment Date:	16 / 12 / 2020

Contents :

Sr. No	Command name	Page NO
1	cal	3
2	cat	3
3	cd	4
4	chmod	4
5	clear	4-5
6	cp	5
7	cut	5-6
8	date	6
9	df	6-7
10	dir	7
11	du	7-8
12	echo	8
13	exit	8
14	man	8-9
15	mkdir	9
16	mv	9
17	passwd	10
18	paste	10
19	pwd	11
20	rmdir	11
21	sort	11-12
22	tail	12
23	touch	12
24	wc	13
25	who	13

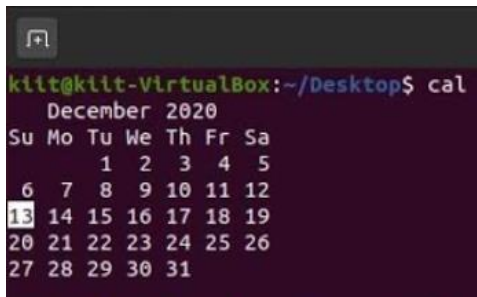
1. cal

Description of the Command: The **cal** command prints a calendar on the standard output.

Syntax: cal

Example: cal

Execution Screenshot:



```
kiit@kiit-VirtualBox:~/Desktop$ cal
December 2020
Su Mo Tu We Th Fr Sa
        1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31
```


2. cat

Description of the Command: The **cat** command allows us to create single or multiple files, view content of file, concatenate files and redirect output in terminal or files.

Syntax: cat [OPTION] [FILE NAME]

Example: *cat textfile.txt* - will display content of textfile.txt
cat > textfile1.txt - will create a text file of name textfile1

Execution Screenshot:



```
kiit@kiit-VirtualBox:~/Desktop$ cat>file1.txt
Z for zebra
Hi my name is SUMIT PANDEY
```

3. cd

Description of the Command: The **cd** command is used to change the directory.

Syntax – cd [directory name]

Example – **cd /** - this will move to root directory, or **cd [directory name]** will change the dir to that particular directory.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ cd /  
kiit@kiit-VirtualBox:/$ cd ~
```

4. chmod

Description of the Command: The **chmod** command allows us to change/update file access permissions.

Syntax - chmod [OPTION] [FILE NAME]

Example - **chmod +x textfile.txt** - will give 'execute' (x) permission to textfile.txt.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ chmod +x file1.txt
```

5. clear

Description of the Command: The **clear** command lets you clear the terminal screen.

Syntax – clear

Execution Screenshot:



```
kiit@kiit-VirtualBox:~$ cd/  
bash: cd/: Is a directory  
kiit@kiit-VirtualBox:~$ clear
```

After executing 'clear' command:



6. cp

Description of the Command: The **cp** command allows us to copy files and directories from one location to another.

Syntax - cp [Guest File] [Host/New File]

Example – cp textfile.txt newfile.txt - contents of textfile is copied to newfile.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ cp file1.txt newfile.txt  
kiit@kiit-VirtualBox:~/Desktop$ cat newfile.txt  
Z for zebra  
Hi my name is SUMIT PANDEY
```

7. cut

Description of the Command: The **cut** command allows us to extract portion of text.

Syntax : cut -[Option] [File_name]

Example : `cut -c 2 newfile.txt` - displays 2nd character from each line of a newfile.txt. after cutting.

Execution Screenshot:



```
kiit@kiit-VirtualBox:~/Desktop$ cut -c 2 newfile.txt
i
```

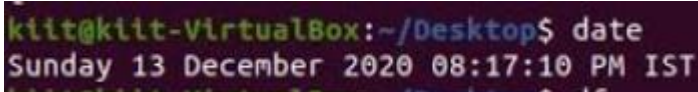
8. date

Description of the Command: The **date** command displays/sets the system date and time.

Syntax - `date` – this will show current date.

Example – `date --set="10 OCT 2020 13:00:00"` - this will set the given date.

Execution Screenshot:



```
kiit@kiit-VirtualBox:~/Desktop$ date
Sunday 13 December 2020 08:17:10 PM IST
```

9. df

Description of the Command: The **df** command shows disk file system space usage.

Syntax - `df` - [Option]

Example – `df -a` - shows all files of the disk,

`df -h` - shows space statistics in ‘human readable’ format, means it gives the details in mb, gb etc

Execution Screenshot:



```
kiit@kiit-VirtualBox:~/Desktop$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            5578956         0   5578956   0% /dev
tmpfs           1121424      1324   1120100   1% /run
/dev/sda5       19992176 6867772  12085812  37% /
tmpfs           5607116         0   5607116   0% /dev/shm
tmpfs           5120          4       5116   1% /run/lock
tmpfs           5607116         0   5607116   0% /sys/fs/cgroup
/dev/loop0       56320      56320         0 100% /snap/core18/1880
/dev/loop2      223232     223232         0 100% /snap/gnome-3-34-1804/60
/dev/loop3       56704      56704         0 100% /snap/core18/1885
/dev/loop1      261760     261760         0 100% /snap/gnome-3-34-1804/36
/dev/loop4       51072      51072         0 100% /snap/snap-store/467
/dev/loop5       63616      63616         0 100% /snap/gtk-common-themes/1506
/dev/loop6       51968      51968         0 100% /snap/snap-store/481
/dev/loop7       31104      31104         0 100% /snap/snapd/9279
/dev/loop8       31744      31744         0 100% /snap/snapd/9607
/dev/sda1        523248         4   523244   1% /boot/efi
tmpfs           1121420         28   1121392   1% /run/user/1000
kiit@kiit-VirtualBox:~/Desktop$
```

10. dir

Description of the Command: The **dir** command lists the contents of a directory.

Syntax - dir

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ dir
B10  file1.txt  Kumar  newfile.txt  OM
```

11. du

Description of the Command: The **du** command shows disk space usage of files present in a directory & its sub-directories.

Syntax - du - [Option]

Example – du -a - shows all disk usage of files and directories.

Execution Screenshot:



```
kiit@kiit-VirtualBox:~/Desktop$ du
4      ./OM/oks
8      ./OM
4      ./Kumar
44     ./B10
68     .
```

12. echo

Description of the Command: The **echo** command prints a text of line provided to it.

Syntax - echo “Hello” - This will print “Hello”

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ echo "this is an echo command"
this is an echo command
```

13. exit

Description of the Command: The **exit** command is used to exit a shell.

Syntax – exit

14. man

Description of the Command: The **man** command is used to view the on-line reference manual pages for commands/programs.

Syntax – man man – this will show information about man command.

Execution Screenshot:



```
MAN(1) Manual pager utils MAN(1)
NAME
man - an interface to the system reference manuals
SYNOPSIS
man [man options] [[function] page ...] ...
man -k [apropos options] regexp ...
man -k [man options] [section] term ...
man -f [whatis options] page ...
man -l [man options] file ...
man -w [man options] page ...
DESCRIPTION
man is the system's manual pager. Each page argument given to man is normally the name of a program, utility or function. The manual page associated with each of these arguments is then found and displayed. A section, if provided, will direct man to look only in that section of the manual. The default action is to search in all of the available sections following a pre-defined order (see SECTIONS), and to show only the first page found, even if page exists in several sections.
The table below shows the section numbers of the manual followed by the types of pages they contain.
1 Executable programs or shell commands
2 System calls (functions provided by the kernel)
3 Library calls (functions within program libraries)
4 Special files (usually found in /dev)
5 File formats and conventions, e.g. /etc/passwd
6 Games
7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7)
8 System administration commands (usually only for root)
9 Kernel routines (non standard)
A manual page consists of several sections.
Conventional section names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS, and SEE ALSO.
The following conventions apply to the SYNOPSIS section and can be used as a guide in other sections.
bold text type exactly as shown.
italic text replace with appropriate argument.
[-abc] any or all arguments within [ ] are optional.
+abc options delimited by | cannot be used together.
argument ... argument is repeatable.
[expression] ... entire expression within [ ] is repeatable.
Exact rendering may vary depending on the output device. For instance, man will usually not be able to render italics when running in a terminal, and will typically use underlined or
```

15. mkdir

Description of the Command: The **mkdir** command is used to create single or more directories.

Syntax - mkdir - [New Dir name]

Example – mkdir newdir - makes a new directory of name ‘newdir’.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ mkdir newdir
kiit@kiit-VirtualBox:~/Desktop$ dir
B10 file1.txt Kumar newdir newfile.txt OM
```

16. mv

Description of the Command: The **mv** command is used to rename files or directories, move a file or directory to another location.

Syntax - mv [current_name] [new_name]

Example – mv project unix_project - renames ‘project’ as ‘unix_project’.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ mv file1.txt file0.txt
kiit@kiit-VirtualBox:~/Desktop$ cat file0.txt
Z for zebra
Hi my name is SUMIT PANDEY
```

17. passwd

Description of the Command: The **passwd** command is used to create/update passwords for user accounts.

Syntax - passwd [user's name]

Example – passwd user (*user's name is 'user' only.*)

current password -

new password -

Your typed password won't be visible.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ passwd
Changing password for kiit.
Current password: █
```

18. paste

Description of the Command: The **paste** command allows us to merge lines of files.

Syntax - paste - [Option] [File_name]

Example – paste -d, -s newfile.txt - will join all lines and separate them with a comma after applying paste command

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ paste -d, -s newfile.txt  
Z for zebra,Hi my name is SUMIT PANDEY
```

19. pwd

Description of the Command: The **pwd** command displays the name of current/working directory.

Syntax - **pwd** [OPTION]

Example – *echo "\$PWD \$OLDPWD"* - checks the current & previous working directory in one go.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ pwd  
/home/kiit/Desktop
```

20. rmdir

Description of the Command: The **rmdir** command is to delete/remove empty directories.

Syntax - **rmdir** [DIRECTORY NAME]

Example – *rmdir newdir* - deletes 'newdir' directory.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ rmdir newdir  
kiit@kiit-VirtualBox:~/Desktop$ dir  
B10 file0.txt Kumar newfile.txt OM
```

21. sort

Description of the Command: The **sort** command is to sort lines of text in the specified file(s) or from stdin.

Syntax - sort [OPTION] [FILE NAME]

Example – *sort textfile.txt* - this simply sorts ‘textfile’ in alphabetical order (by default).

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ sort file0.txt
Hi my name is SUMIT PANDEY
Z for zebra
```

22. tail

Description of the Command: The **tail** command is used to show last lines (10 lines by default) of the specified file.

Syntax - tail - [Option] [file name]

Example – *tail -5 textfile.txt* - shows last 5 lines of textfile.txt.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ tail -n1 file0.txt
Hi my name is SUMIT PANDEY
```

23. touch

Description of the Command: The **touch** command is used to change file timestamps; it can also be used to create a file.

Syntax - touch - [Option] [FILE NAME]

Example – *touch empty_file* - this simply creates an empty (zero byte) new file called 'empty_file'.

Execution Screenshot: we can see word count is zero for empty_file.

```
kiit@kiit-VirtualBox:~/Desktop$ touch empty_file.txt
kiit@kiit-VirtualBox:~/Desktop$ wc empty_file.txt
0 0 0 empty_file.txt
```

24. wc

Description of the Command: The **wc** command is used to display newline, word, byte counts for each file specified & total for many files.

Syntax - wc - [Option] [file name]

Example – *wc filetext.txt* - this simply shows number of lines, words, bytes along with the file's name.

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ wc file0.txt
2  9 39 file0.txt
```

25. who

Description of the Command: The **who** command is used to show information about users who are currently logged in.

Syntax - who

Execution Screenshot:

```
kiit@kiit-VirtualBox:~/Desktop$ who
kiit      :0                2020-12-13 20:22 (:0)
```