

Assignment 1

OPERATING SYSTEM

TOPIC: BASIC UNIX COMMANDS & FILE SYSTEM COMMANDS

1. Display the date using the “**date**” command.

```
rudra@DESKTOP-40986GA:~$ date
Wed Jul 24 19:27:59 IST 2024
rudra@DESKTOP-40986GA:~$
```

2. Check who are the users logged in using the “**who**” command.

```
rudra@DESKTOP-40986GA:~$ who
rudra@DESKTOP-40986GA:~$
```

3. Check the running processes using the “**ps**” command.

```
rudra@DESKTOP-40986GA:~$ ps
  PID TTY          TIME CMD
    9 tty1      00:00:00 bash
   74 tty1      00:00:00 ps
rudra@DESKTOP-40986GA:~$
```

4. List the files with “**ls**” command with and without **-l** option.

```
rudra@DESKTOP-40986GA:~$ ls
'*.html.gz'  Rudradeep_50  calculator2  equal  fruit  hellnworld1  loop2.sh  max.sh  string.sh
'='          args.sh       case.sh     equal2  fruit2  hellnworld1.c  loop3  mca  sum
BCA         assignment-6  chosenOne  even    gal.sh  hellnworld2  loops  mult  test
BCA1        avg          code.c     exam    greater  loop  marks.sh  palindrome.sh  while
BCA2020     bca         comp       filecomp.sh  greater.sh  loop  marks.sh  practice  while2
BCA3        break.sh    deb       filegreater.sh  greater2  loop.sh  matrix  rudra  whileloop.sh
Exam        calculator  dvd       filepractice.sh  he  loop2  max  salary.sh
```

```
rudra@DESKTOP-40986GA:~$ ls -l
total 4
-rw-r--r-- 1 rudra rudra 27 Nov 4 2022 '*.html.gz'
-rw-r--r-- 1 rudra rudra 0 Nov 13 2022 '='
-rw-r--r-- 1 rudra rudra 18 Apr 28 2021 BCA
-rw-r--r-- 1 rudra rudra 9 Apr 28 2021 BCA1
drwxr-xr-x 1 rudra rudra 4096 Sep 19 2022 BCA2020
-rw-r--r-- 1 rudra rudra 9 Apr 28 2021 BCA3
drwxr-xr-x 1 rudra rudra 4096 Jun 15 2021 Exam
drwxr-xr-x 1 rudra rudra 4096 Nov 13 2022 Rudradeep_50
-rw-r--r-- 1 rudra rudra 440 Nov 13 2022 args.sh
drwxr-xr-x 1 rudra rudra 4096 Oct 21 2022 assignment-6
drwxr-xr-x 1 rudra rudra 4096 May 5 2021 avg
drwxr-xr-x 1 rudra rudra 4096 Apr 12 2021 bca
-rwxr-xr-x 1 rudra rudra 89 Nov 13 2022 break.sh
drwxr-xr-x 1 rudra rudra 4096 May 12 2021 calculator
drwxr-xr-x 1 rudra rudra 4096 May 15 2021 calculator2
-rwxr-xr-x 1 rudra rudra 156 Nov 13 2022 case.sh
drwxr-xr-x 1 rudra rudra 4096 Nov 13 2022 chosenOne
-rwxr-xr-x 1 rudra rudra 2238 Nov 13 2022 code.c
drwxr-xr-x 1 rudra rudra 4096 Aug 28 2022 comp
drwxr-xr-x 1 rudra rudra 4096 Aug 26 2022 deb
drwxr-xr-x 1 rudra rudra 4096 Aug 26 2022 dvd
drwxr-xr-x 1 rudra rudra 4096 May 12 2021 equal
drwxr-xr-x 1 rudra rudra 4096 May 15 2021 equal2
drwxr-xr-x 1 rudra rudra 4096 Aug 13 2022 even
drwxr-xr-x 1 rudra rudra 4096 Jun 15 2021 exam
-rw-r--r-- 1 rudra rudra 0 Aug 28 2022 filecomp.sh
-rw-r--r-- 1 rudra rudra 166 Aug 13 2022 filegreater.sh
-rwxr-xr-x 1 rudra rudra 221 Jun 14 2021 filepractice.sh
drwxr-xr-x 1 rudra rudra 4096 May 12 2021 fruit
drwxr-xr-x 1 rudra rudra 4096 May 15 2021 fruit2
-rwxr-xr-x 1 rudra rudra 269 Nov 13 2022 gel.sh
```

5. Check the *manual of ls* command.

```
LS(1) User Commands LS(1)
NAME
    ls - list directory contents
SYNOPSIS
    ls [OPTION]... [FILE]...
DESCRIPTION
    List information about the FILES (the current directory by default). Sort entries alphabetically if none of --cfuvSWX nor --sort is specified.
    Mandatory arguments to long options are mandatory for short options too.
    -a, --all
        do not ignore entries starting with .
    -A, --almost-all
        do not list implied . and ..
    --author
        with -l, print the author of each file
    -b, --escape
        print C-style escapes for nongraphic characters
    --block-size=SIZE
        with -l, scale sizes by SIZE when printing them; e.g., '--block-size=M'; see SIZE format below
    -B, --ignore-backups
        do not list implied entries ending with ~
    -c
        with -lr: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name;
    manual page 1 of 1, line 1 (press h for help or q to quit)
```

6. Show the commands used to display (i) filenames (ii) processes (iii) users.

```
rudra@DESKTOP-40986GA:~$ ls
*.html.gz  Rudradeep_50  calculator2  equal  fruit  helloworld1  loop2.sh  max.sh  string.sh
*?         args.sh       case.sh     equal2  fruit2  helloworld1.c  loop3  mca  sum
BCA1       assignment-0  chosenOne  even   gal.sh  helloworld2  loop4  mult  test
BCA2       avg         code.c     exam   greater  loop  m.sh  palindrome.sh  while
BCA2020    bca        comp      filecomp.sh  greater.sh  loop  marks.sh  practice  while2
BCA3       break.sh   deb       filegreater.sh  greater2  loop.sh  matrix  rudra  whileloop.sh
Exam       calculator  dvd       filepractice.sh  he        loop2  max  salary.sh
```

```
rudra@DESKTOP-40986GA:~$ ps
PID TTY          TIME CMD
  9 tty1          00:00:00 bash
 74 tty1          00:00:00 ps
rudra@DESKTOP-40986GA:~$
```

```
rudra@DESKTOP-40986GA:~$ who
rudra@DESKTOP-40986GA:~$
```

7. Check and state the difference between man and whatis command by checking **man cp** & **whatis cp**.

```
CP(1) User Commands CP(1)
NAME
    cp - copy files and directories
SYNOPSIS
    cp [OPTIONS]... [-i] SOURCE DEST
    cp [OPTIONS]... SOURCE... DIRECTORY
    cp [OPTIONS]... -t DIRECTORY SOURCE...
DESCRIPTION
    Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
    Mandatory arguments to long options are mandatory for short options too.
    -a, --archive
        same as -dR --preserve=all
    --attributes-only
        don't copy the file data, just the attributes
    --backup[=CONTROL]
        make a backup of each existing destination file
    -b
        like --backup but does not accept an argument
    --copy-contents
        copy contents of special files when recursive
    -d
        same as --no-dereference --preserve=links
    -f, --force
        if an existing destination file cannot be opened, remove it and try again (this option is ignored when the -n option is also used)
    manual page 1 of 1, line 1 (press h for help or q to quit)
```

```
rudra@DESKTOP-40986GA:~$ whatis cp
cp (1) - copy files and directories
rudra@DESKTOP-40986GA:~$
```

8. What is the primary difference between **printf** and **echo** command? Check and print.

echo: Simple and easy to use for printing text to the terminal. It automatically adds a newline at the end.

printf: More powerful and flexible, similar to the C language printf, allowing formatted output and no automatic newline.

```
rudra@DESKTOP-40986GA:~$ echo "Hello, World!"
Hello, World!
rudra@DESKTOP-40986GA:~$
```

```
rudra@DESKTOP-40986GA:~$ printf "Hello, World!"
Hello, World!rudra@DESKTOP-40986GA:~$
```

9. In the home directory, create a directory *MCA2022*. Inside the *MCA2022*, create another directory *<FirstName_Section_ClassRoll>* and get into the directory [*~ /MCA2022/Ankur_A_oo\$*].

```
rudra@DESKTOP-40986GA:~$ mkdir MCA2023
rudra@DESKTOP-40986GA:~$ cd
rudra@DESKTOP-40986GA:~$ cd MCA2023
rudra@DESKTOP-40986GA:~/MCA2023$ mkdir Rudradeep_A_13
rudra@DESKTOP-40986GA:~/MCA2023$ cd Rudradeep_A_13
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13$
```

10. Go to the subdirectory and create another subdirectory “*Unix_File_System*” within it.

```
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13$ mkdir Unix_File_System
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13$ cd Unix_File_System
```

11. Create the subdirectories *TestA*, *TestB*, *TestC* and corresponding sub-subdirectories *TestA-1*, *TestA-2*, *TestB-1*, *TestB-2*, *TestB-3*, *TestC-1*, *TestB-2-i* in a single command.

```
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13/Unix_File_System$ mkdir -p TestA/TestA-1 TestA/TestA-2 TestB/TestB-1 TestB/TestB-2 TestB/TestB-3 TestC/TestC-1 TestB/TestB-2/TestB-2-i
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13/Unix_File_System$
```

12. Show the absolute path of *TestB-2-i*.

```
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13/Unix_File_System$ readlink -f TestB/TestB-2/TestB-2-i
/home/rudra/MCA2023/Rudradeep_A_13/Unix_File_System/TestB/TestB-2/TestB-2-i
rudra@DESKTOP-40986GA:~/MCA2023/Rudradeep_A_13/Unix_File_System$
```