1. **Connect to the Linux Server and understand the basic directory structure of Linux**

>>> ssh dhruv@192.168.1.5

```
Welcome to Ubuntu 20.04 LTS
Last login: Fri Oct 27 10:21:14 2024 from 192.168.1.3
```

```
onworks@onworks:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap
Templates  Videos
onworks@onworks:~$ pwd
/home/onworks
onworks@onworks:~$ cd /
onworks@onworks:/$ ls
bin    dev   lib    libx32      mnt   root  snap        sys  var
boot   etc   lib32  lost+found  opt   run   srv         tmp
cdrom  home  lib64  media       proc  sbin  swapfile    usr
onworks@onworks:/$ cd home
onworks@onworks:/home$ cd onworks
onworks@onworks:~$ cd Desktop
onworks@onworks:~/Desktop$ ls
onworks@onworks:~/Desktop$ pwd
/home/onworks/Desktop
onworks@onworks:~/Desktop$
```

2. **To understand help commands like-man, info, help, whatis, apropos**

>>> man ls

```
LS(1)                User Commands               LS(1)
NAME
    ls - list directory contents
SYNOPSIS
    ls [OPTION]... [FILE]...
```

>>> info ls

```
File: coreutils.info,  Node: ls invocation,  Next: dir invocation,  Prev: Common c

11.1 'ls': List directory contents
==================================

The 'ls' program lists information about files (of any type, including
directories).  Options and file arguments can be intermixed
arbitrarily, as usual.

    ls [OPTION]... [FILE]...
```

>>> help cd

```
cd: cd [-L|[-P [-e]] [-@]] [dir]
    Change the shell working directory.

    Change the current directory to DIR.  The default DIR is the value of the
    HOME shell variable.
```

>>> whatis ls

```
ls (1)               - list directory contents
```

>>> apropos directory

```
ls (1)               - list directory contents
cd (1)               - change the shell working directory
mkdir (1)            - make directories
rmdir (1)            - remove empty directories
pwd (1)              - print name of current/working directory
```

3. **To understand basic directory navigation commands like cat, cd, my, cp, rm, mkdir, rmdir, file, pwd command**

>>> mkdir test_dir

>>> cd test_dir

>>> pwd

```
/home/test_dir
```

>>> touch file.txt

>>> echo "Hello" > file.txt

>>> cat file.txt

```
Hello
```

>>> cp file.txt /home/user/Desktop/

>>> mv file.txt /home/user/Documents/

>>> mkdir newfolder

>>> rmdir newfolder

4. **To understand basic commands**
   like:-date, cal, echo, bc, ls, who, whoami, hostname, uname, tty, aliase

>>> date

```
Sun Oct 27 15:32:17 IST 2024
```

>>> cal

```
    October 2024
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
```

>>> echo "Hello, World!"

```
Hello, World!
```

>>> echo "5 + 3" | bc

```
8
```

>>> ls

```
file1.txt  folder1  script.sh
```

>>> who

```
user    tty7         2024-10-27 14:12 (:0)
anotheruser  pts/0   2024-10-27 15:45 (192.168.1.5)
```

>>> whoami

```
user
```

>>> hostname

```
my-computer
```

>>> uname

```
Linux
```

>>> tty

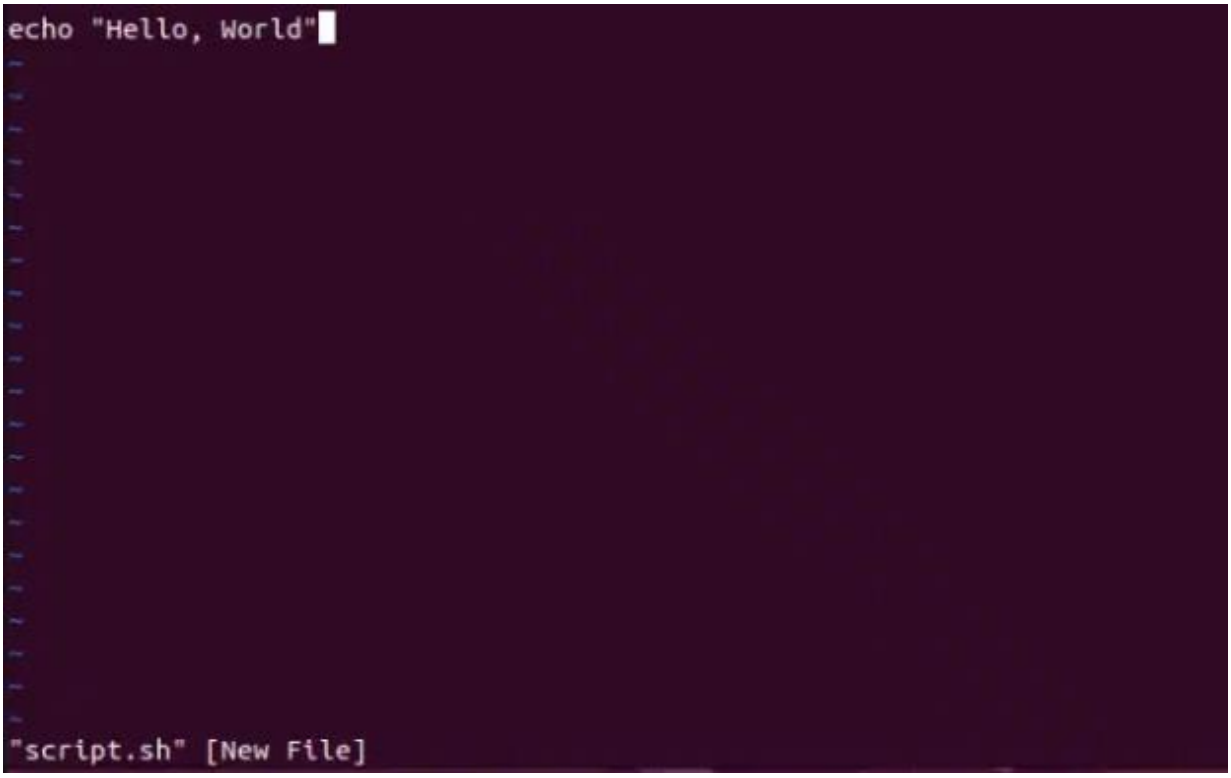
```
/dev/pts/0
```

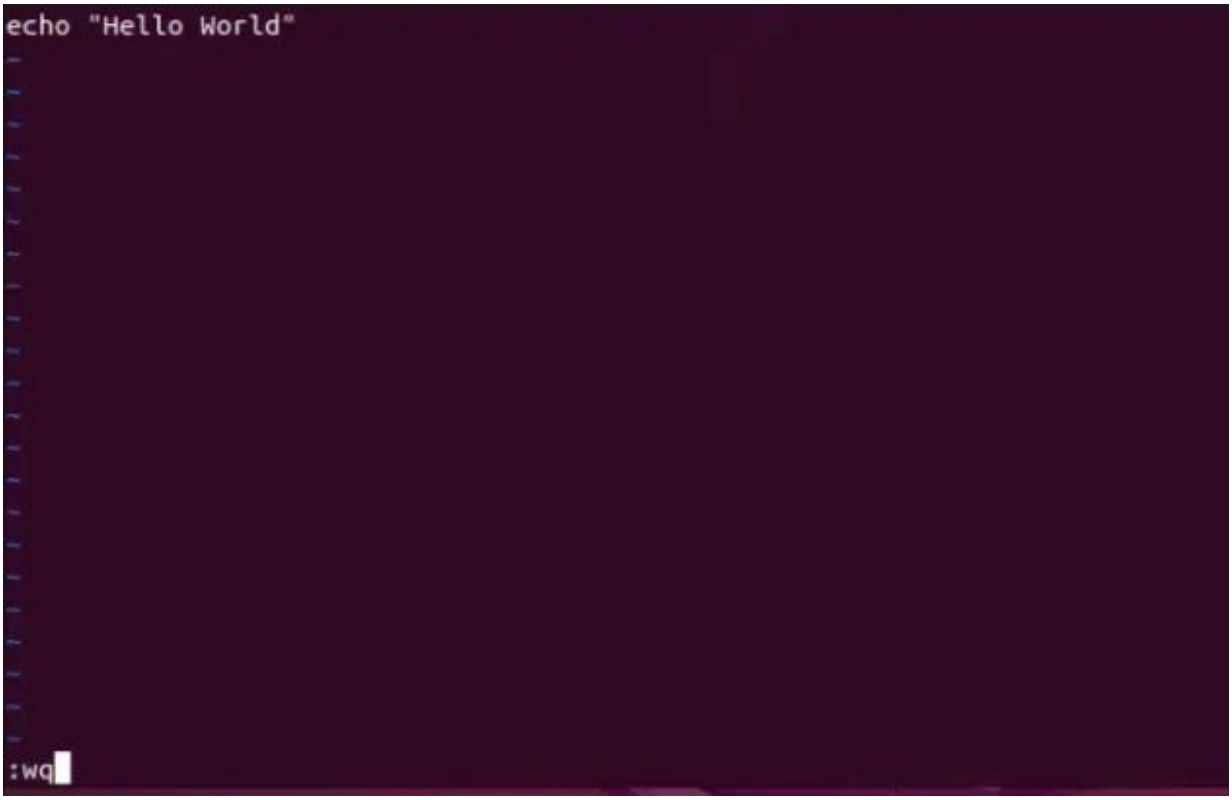
>>> alias

```
alias ll='ls -la'
alias l='ls -CF'
```

5. **To understand vi basics, Three modes of vi Editor, how to write, save, execute a shell script in vi editor**

>>> vi script.sh





>>> bash script.sh

```
Hello, World
```

6. To understand process related commands like: -ps, top, pstree, nice, renice in Linux.

>>> ps

```
  PID TTY          TIME CMD
 1012 pts/0    00:00:00 bash
 1057 pts/0    00:00:00 ps
```

>>> top

```
top - 15:35:12 up  1:01,  2 users,  load average: 0.03, 0.02, 0.00
Tasks: 105 total,   1 running, 104 sleeping,   0 stopped,   0 zombie
%Cpu(s):  1.0 us,  0.5 sy,  0.0 ni, 98.5 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   2048.0 total,   1580.2 free,    182.5 used,    285.3 buff/cache
MiB Swap:   1024.0 total,   1024.0 free,      0.0 used.   1765.5 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 1012 user      20   0   10172   2256   1916 S   0.0  0.1   0:00.03 bash
 1057 user      20   0    3924    888    732 R   0.0  0.0   0:00.00 top
```

>>> pstree

```
systemd─┬─sshd─┬─sshd───bash───pstree
        └─2*[sshd───bash]
```

>>> ps -o pid,comm,nice -p 1057

```
  PID COMMAND         NI
 1057 sleep           10
```

>>> renice -n 5 -p 1057

```
1057 (process ID) old priority 10, new priority 5
```

7. **To understand how to examine and change File permissions.**

>>> chmod 644 file.txt

>>> ls -l file.txt

```
-rw-r--r-- 1 user user 1234 Oct 26 14:20 file.txt
```

8. **Set a file to be read-only with the chmod command. Interpret the file permissions displayed by the ls -l command.**

>>> chmod 444 file.txt

>>> ls -l file.txt

```
-r--r--r-- 1 user user 1234 Oct 26 14:20 file.txt
```

9. **Delete one or more directories with the rmdir command. See what happens if the directory is not empty. Experiment (carefully!) with the rm -r command to delete a directory and its content.**

>>> mkdir newfolder

>>> rmdir newfolder

>>> mkdir dir_to_delete

>>> rm -r dir_to_delete

```
rm: remove directory 'dir_to_delete'? y
```

10. **Change your directory to the directory exercises. Create a file in that directory, named the file as example1 using the cat command containing the following text: water, water everywhere and all the boards did shrink; water, water everywhere, no drop to drink.**

>>> cd exercises

>>> cat > example1 <<EOF

```
water, water everywhere
and all the boards did shrink;
water, water everywhere
nor any drop to drink.
EOF
```

11. Write basic shell script to display the table of a number.

```
#!/bin/bash
read -p "Enter a number: " num
for i in {1..10}; do
  echo "$num * $i = $((num * i))"
done
```

>>> bash script.sh

```
Enter a number: 5
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50
```

12. **Write basic shell script to input a character from user and then check whether it is uppercase, lowercase or digit.**

```
#!/bin/bash
read -p "Enter a character: " char
if [[ "$char" =~ [A-Z] ]]; then
  echo "Uppercase"
elif [[ "$char" =~ [a-z] ]]; then
  echo "Lowercase"
elif [[ "$char" =~ [0-9] ]]; then
  echo "Digit"
else
  echo "Special character"
fi
```

>>> bash script.sh

```
Enter a character: g
Lowercase
```

13. Write basic shell script to calculate factorial of a number.

```
#!/bin/bash
read -p "Enter a number: " num
fact=1
for (( i=1; i<=num; i++ )); do
  fact=$((fact * i))
done
echo "Factorial: $fact"
```

>>> bash script.sh

```
Enter a number: 5
Factorial: 120
```

14. Write basic shell script to input the month number and generate corresponding calendar.

```
#!/bin/bash
read -p "Enter month (1-12): " month
read -p "Enter year: " year
cal $month $year
```

>>> bash script.sh

```
Enter month (1-12): 10
Enter year: 2024
    October 2024
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
```

15. **Write basic shell script to list all directories.**

```
#!/bin/bash
ls -d */
```

>>> bash script.sh

```
Documents/  Downloads/  Music/  Pictures/
```

16. Write basic shell script to display greatest of three numbers.

```
#!/bin/bash
read -p "Enter first number: " a
read -p "Enter second number: " b
read -p "Enter third number: " c
if (( a > b && a > c )); then
  echo "Greatest: $a"
elif (( b > a && b > c )); then
  echo "Greatest: $b"
else
  echo "Greatest: $c"
fi
```

>>> bash script.sh

```
Enter first number: 12
Enter second number: 45
Enter third number: 23
Greatest: 45
```

17. Write basic shellscript to check whether the number entered by user is prime or not.

```
#!/bin/bash
read -p "Enter a number: " num
is_prime=1
for (( i=2; i*i<=num; i++ )); do
  if (( num % i == 0 )); then
    is_prime=0
    break
  fi
done
if (( is_prime && num > 1 )); then
  echo "Prime"
else
  echo "Not Prime"
fi
```

>>> bash script.sh

```
Enter a number: 10
Not Prime
```