Vi sh1 – create a
sh sh1

show the permissions of a certain file using ls
ls -l sh1

changing permissions
chmod 700 filenme

$ PATH=$PATH:. → without the . In the path, file wont run as executable

ls -a → shows the hidden files too
ls -i → gives the inode numbers too of the files

BASH PROFILE → configuration env the moment we log in
linux uses BASH RC

see contents of bash profile → **cat .bash_profile**
cat .bashrc

CURRENT DIRECTORY
. means current directory → cat ./file.c → searches for the file in the current directory
pwd gives current directory

cd desktop
env

PWD is a variable in the env
pwd is a command which checks the PWD from the env

ROOT DIRECTORY
go to root → cd /
bin has the executables in it
cd bin
ls

cd ..
cd dev
ls -l → long listing , gives permission info too

when we mount a new device it is saved under dev

lost+found –. when system crashes the files are stored here, recovery is from here. No permission to see
temp → files which will be erased when system starts again

ABSOLUTE AND RELATIVE PATH
Absolute path : always starts from the root
Relative : starts from where we are relatively

change directory to root
give an absolute path to go to etc directory
cd /etc

cd ../dev → absolute

go to home –> cd
show the primary variable ps1
PS1="\w hiSani>";

ERRORS
can come in 3 places
kernel, shell, command

Kernel gives the error → No such file or directory
only the kernel can do the i/o

SECONDARY PROMPT >
say we have a multiline command

echo "hii --enter
hello
bye"

it waits for the " to close, hence the secondary prompt comes >

WORD COUNT WC
filter, taking inout from stdinput and giving output from stdoutput

$ echo "this is
> a three line
> message" | wc
  OUTPUT → 3 6 29

3 6 29 gives word cont information

unix filesystem layout
boot block –. bootstrap code to into the machine to boot the os
super block → state of the file system, how large, how many it can store, free space etc
inode list →
data blocks →

directory stores only the name of the file and the inode, then the os finds the inode in the system and then contents of that file using the inode

7 TYPES OF FILES (ordinary, device, ?)
- stands for regular file
d means directory
c – means character special → physical device that reads one char at a time
b – block special →
l - symbolic link
p – FIFO named pipe for IPC
s – socket

ls -l /* |grep^l → gives all files which have symbolic link

command is looked in the path, argument in the current directory

root /
home ~
parent ..
working directory .

*Dev/*null → is a block hole
Ls -z note 2>/dev/null