Basic Commands:

Cd : Change directory [Open a Directory/Folder in Linux]

Cd .. : Back to previous folder

Ls -a : Display all files and directories

Ls -l : Display all files with information

Ls -al : Display files with permissions

Mkdir [filename] : To create a Directory[Folder]

Su root : switch user to root/administrator

File creation commands:

1. Touch [Filename] : to create a fle instantly [You cannot add content by doing this]
2. Cat > [Filename] : to create a file and add content in file and press CTRL+D to save. To view content of any file use *Cat [Filename]*
3. Vi [Filename]: create a file with Vi editor.

How to perform your worksheet:

- Open your ubuntu.
- Open terminal
- Type *ls -a* to display all files and folder.
- To go to Desktop use *cd Desktop*.
- Now, create a file for bash extension is .sh
- Create with touch command *touch project.sh*
- Now, check its permissions via *ls -al*. It won't have executable permission which is denoted by x, r for read and w for write.
- To give it executable permission, use *chmod +x project.sh*
- Now, open this file by double click on file on desktop.
- First, we have to give path of bash to file given by *#! /bin/bash*
- Echo is like Print of python you can print anything using echo like echo "Shika". It will print Shika.

Code to perform the task:

- Open Linux and open terminal and use `cd Desktop` to go to Desktop.
- Create a file using `touch newfile.sh`
- Use `chmod +x newfile.sh` to grant permission for execution.
- Paste both commands to install dialog in linux.

`sudo apt-get update`

`sudo apt-get install dialog`

- Double click to open your file and paste the code below.
- Switch to terminal and use `./newfile.sh` to execute this file.

```
#Path for bash
#!/bin/bash

#Function to create date and time
datetime()
{
   dialog --title "System date and Time" --infobox "Date is `date`" 3 40
   #Date will display Time and Date of the system
   # 3 - Height of dialog box
   # 4 - Width of dialog box
read
return

}
#Function to display calender
calender()
{
   cal > menuchoice.temp.$$
   dialog --title "My Calender" --infobox "`cat menuchoice.temp.$$`" 9 25
   #Calender will be stored in a new file named menuchoice
   # 9 and 25 are the height and width of dialog
   read
   rm -f menuchoice.temp.$$
   #After displaying calender, Removing menuchoice file
   return

}
delete()
{
 dialog --title "Delete file"\
 --inputbox "Enter directory path (Enter for Current Directory)"\
 10 40  2>/tmp/dirip.$$
```

```
rtval=$?

case $rtval in
    1) rm -f /tmp/dirip.$$ ; return ;;
    255) rm -f /tmp/dirip.$$ ; return ;;
esac

mfile=`cat /tmp/dirip.$$`

if [ -z $mfile ]
then
    mfile=`pwd`/*
else
    grep "*" /tmp/dirip.$$
    if [ $? -eq 1 ]
    then
   mfile=$mfile/*
    fi
fi

for i in $mfile
do
   if [ -f $i ]
   then
   echo "$i Delete?" >> /tmp/finallist.$$
   fi
done


dialog  --title "Select File to Delete"\
--menu "Use [Up][Down] to move, [Enter] to select file"\
20 60 12 `cat /tmp/finallist.$$` 2>/tmp/file2delete.tmp.$$

rtval=$?

file2erase=`cat /tmp/file2delete.tmp.$$`

case $rtval in
    0) dialog --title "Are you sure"\
     --yesno "\n\nDo you want to delete : $file2erase " 10 60

       if [ $? -eq 0 ] ; then
     rm -f  $file2erase
        if [ $? -eq 0 ] ; then
           dialog
```

```
        --title "Information: Delete Command" --infobox "File: $file2erase is
Sucessfully deleted,Press a key" 5 60
        read
       else
        dialog
        --title "Error: Delete Command" --infobox "Error deleting File:
$file2erase, Press a key" 5 60
            read
           fi
    else
      dialog
      --title "Information: Delete Command" --infobox "File: $file2erase is not
deleted, Action is canceled, Press a key" 5 60
      read
    fi
     ;;
    1)  rm -f /tmp/dirip.$$ ; rm -f /tmp/finallist.$$ ;
        rm -f /tmp/file2delete.tmp.$$; return;;
    255) rm -f /tmp/dirip.$$ ;  rm -f /tmp/finallist.$$ ;
         rm -f /tmp/file2delete.tmp.$$; return;;
esac
 rm -f /tmp/dirip.$$
 rm -f /tmp/finallist.$$
 rm -f /tmp/file2delete.tmp.$$
 return

}

# creating a menu
echo "SELECT YOUR CHOICE";
echo "1. Date/Time"
echo "2. Calender"
echo "3. Delete"
echo "4. Exit"
echo -n "Enter your menu choice [1-4]: "

# Running a forever loop using while statement
# This loop will run untill user select the exit option.
while :
do
# reading choice
read choice

case $choice in
  1)  datetime ;;
```

```
  2)  calender;;
  3)  delete;;
  4)  echo "Quitting ..."
      exit;;
# Default Choice
  *) echo "invalid option";;

esac
  echo -n "Enter your menu choice [1-4]: "
done
```

Your code for display all info of system:

To display value of variable in echo use ($[var_name])

CPU information is stored in */proc/cpuinfo* to view this file we use cat [filename]

Memory Information is stored in */proc/meminfo*

Lsblk command used to display all partition of hard disk.

Mounted files stored in */proc/mounts*

```
#!/bin/bash
OSTYPE = `OSTYPE`

echo "SELECT Your Choice";
echo "1. Your operating system type"
echo "2. Computer cpu information"
echo "3. Memory information"
echo "4. Hard disk information"
echo "5. File system (Mounted)"
echo "6. Exit"
echo -n "Enter your menu choice [1-4]: "

while :
do
read choice
case $choice in
  1)  echo "Your Operating system is ($OSTYPE)" ;;
  2)  echo "CPU information:"
      cat /proc/cpuinfo    ;;
  3)  echo "Memory information:"
      cat /proc/meminfo    ;;
  4)  echo "Hard Disk Information:"
      lsblk;;
  5)  echo "Mounted Files Information:"
      cat /proc/mounts;;
  6)  echo "Quitting ..."
      exit;;
  # Default Choice
  *) echo "invalid option";;

esac
  echo -n "Enter your menu choice [1-6]: "
done
```