Name :- Amit.Y.Zope
Roll No. :- 223067
Gr No. :- 21810714
Division :- C
Batch :- C3
Subject :- Operating System

Assignment No 2

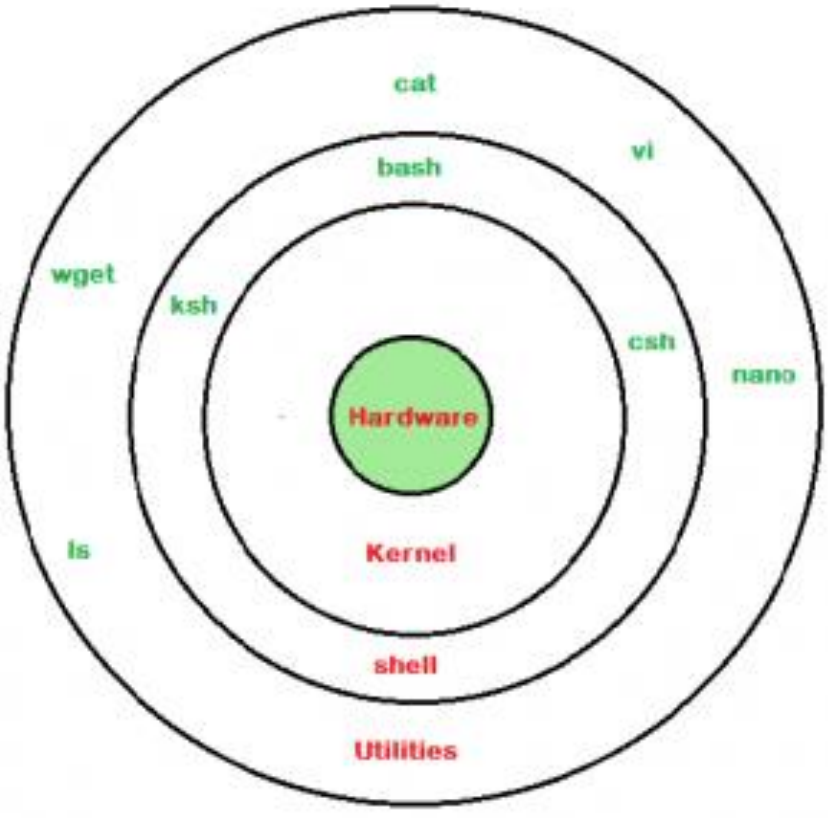
Aim: Creating Address Book Using Shell Script

Objective: To study the frequently used Linux commands

Theory:

What is Shell ?

A shell is special user program which provide an interface to user to use operating system services. Shell accept human readable commands from user and convert them into something which kernel can understand. It is a command language interpreter that execute commands read from input devices such as keyboards or from files. The shell gets started when the user logs in or start the terminal.



Shell is broadly classified into two categories –>

- **Command Line Shell.**
- **Graphical shell.**

Command Line Shell

Shell can be accessed by user using a command line interface. A special program called **Terminal** in linux/macOS or **Command Prompt** in Windows OS is provided to type in the human readable commands such as “cat”, “ls” etc. and then it is being execute. The result is then displayed on the terminal to the user.

Graphical Shells

Graphical shells provide means for manipulating programs based on graphical user interface (GUI), by allowing for operations such as opening, closing, moving and resizing windows, as well as switching focus between windows. Window OS or Ubuntu OS can be considered as good example which provide GUI to user for interacting with program. User do not need to type in command for every actions.

There are several shells are available for Linux systems like : -

1. **BASH (Bourne Again SHell)** – It is most widely used shell in Linux systems. It is used as default login shell in Linux systems and in macOS. It can also be installed on Windows OS.
2. **CSH (C SHell)** – The C shell’s syntax and usage are very similar to the C programming language.
3. **KSH (Korn SHell)** – The Korn Shell also was the base for the POSIX Shell standard specifications etc.

Each shell does the same job but understand different commands and provide different built in functions.

Code

Main code :

```
#!/bin/bash
echo -e "\n"
echo "||----------   STUDENT   ADDRESS  BOOK  ------------||"
echo ""
cond=1
while [ $cond -eq 1 ]  #loop will forever until exit 0 condition comes
do
echo -e "\n"
echo "||******************* MENU **********************||"
```

```
echo ""
echo "1. Add a record " #echo to print on stdout
echo "2. Search a record "
echo "3. Delete a record "
echo "4. Display all record"
echo "5. Update a record"
echo "6. Exit"             #OPTIONS FOR USER TO PERFORM LISTED OPERATIONS
echo ""
echo "Enter an option : "
read choice                #TAKING INPUT FROM USER
#switch case to link or to execute the files
case "$choice" in
     1)  ./add.sh          #WILL EXECUTE THE add.sh file
     ;;
     2)  ./search.sh       #WILL EXECUTE THE search.sh file
     ;;
     3)  ./delete.sh       #WILL EXECUTE THE delete.sh file
     ;;
     4)  ./display.sh      #WILL EXECUTE THE display.sh file
     ;;
     5)  ./update.sh       #WILL EXECUTE THE update.sh file
     ;;
     6)  exit 0            #EXITING FROM PROGRAM
      ;;
esac
done
```

Code for Adding Record

```
#!/bin/bash

echo "|********************STUDENT FORM********************|"
echo ""
echo "Enter the First Name : "
read fname                              #TAKING FIRST NAME AS INPUT
echo "Enter the Last Name  : "
read lname                              #TAKING LAST NAME AS INPUT
echo "Enter the Contact Number   : "
read phno                               #TAKING CONTACTno. AS INPUT
echo "Enter the 6 Digit Unique ID     : "
read rno                                #TAKING 6 DIGIT ID AS INPUT
if grep -w -q $rno data.txt
then

      while grep -w -q $rno data.txt       #VALIDATING CHECKING IF ID IS 6 DIGIT
      do                                   #OR NOT
           echo "ID ALREADY PRESENT"
           echo "Enter the Correct 6 Digit Unique ID   : "
           read rno

     done
fi
echo "Enter the Date of Birth    : "
read dob                                #TAKING DATE OF BIRTH AS INPUT
echo "Enter Blood Group : "
read blg                                #TAKING BLOOD GROUP AS INPUT
```

```
cond=1
while [ $cond==1 ]
do
     if [[ $blg =~ (A|B|a|b|O|o|AB|ab)[+|-] ]] #pattern match
                                   #regex
     then
          break
     else

          echo "Invalid Blood Group"
          echo "Enter Valid Blood Group : "
          read blg
     fi
done
echo "Enter the Email Address    : "
read email
echo "Enter the Current Residence City : "
read city

echo $fname $lname $phno $rno $blg $dob $email $city >> data.txt

#WRITING DATA INTO FILES
```

Code for Searching Record

```
#!/bin/bash

echo "::::::::::::Search A Record::::::::::"
echo ""
echo "Enter the StudentID  : "
read id                                  #TAKING INPUT STUDENT_ID
printf "\n\n"
echo "+------------STUDENT DETAILS ARE DISPLAYED------------+"
printf "\n"
echo "-------------------------------------------------------------------------
-------------------------"
echo "|FirstName LastName        PhoneNo.        ID    BloodGroup  DateofBirth
Email         City  |"
echo "-------------------------------------------------------------------------
-------------------------"
if grep -w -i $id data.txt | awk {'printf "%-10s %-15s %-12s %8s %6s %15s %15s
%11s  \n",$1,$2,$3,$4,$5,$6,$7,$8'}
               #FORMATTED OUPUT USING AWK COMMAND & SEARCHING USING GREP COMMAND
then
       echo
else
     echo "RECORD NOT FOUND"
fi
```

Code for deleting record

```
#!/bin/bash

echo "||**********DELETING RECORD**********||"
```

```
echo ""
echo "Enter the StudentId  : "
read ID                          #TAKING INPUT STUDENT_ID

echo -e "||**********RECORD DELETED IS SHOWN BELOW**********||"
echo -e "\n"
echo "-------------------------------------------------------------------------
------------------------"
echo "|FirstName LastName        PhoneNo.        ID    BloodGroup  DateofBirth
Email         City  |"
echo "-------------------------------------------------------------------------
------------------------"
if grep -w $ID data.txt | awk {'printf "%-10s %-15s %-12s %8s %6s %15s %15s %11s
\n",$1,$2,$3,$4,$5,$6,$7,$8'}
          #use to print the data in formatted form

then
     sed -i '/'$ID'/d' data.txt #sed function use to replace or
          #delete from file
else
     echo "Record Does not exist"
fi
```

Code for displaying records

```
#!/bin/bash
echo "||---------DISPLAYING ALL RECORDS OF DATABASE---------||"
echo ""
echo "-------------------------------------------------------------------------
------------------------"
echo "|FirstName LastName        PhoneNo.        ID    BloodGroup  DateofBirth
Email         City  |"
echo "-------------------------------------------------------------------------
------------------------"
cat data.txt | awk {'printf "%-10s %-15s %-12s %8s %6s %15s %15s %11s
\n",$1,$2,$3,$4,$5,$6,$7,$8'}
     #display the data from file to standard output
```

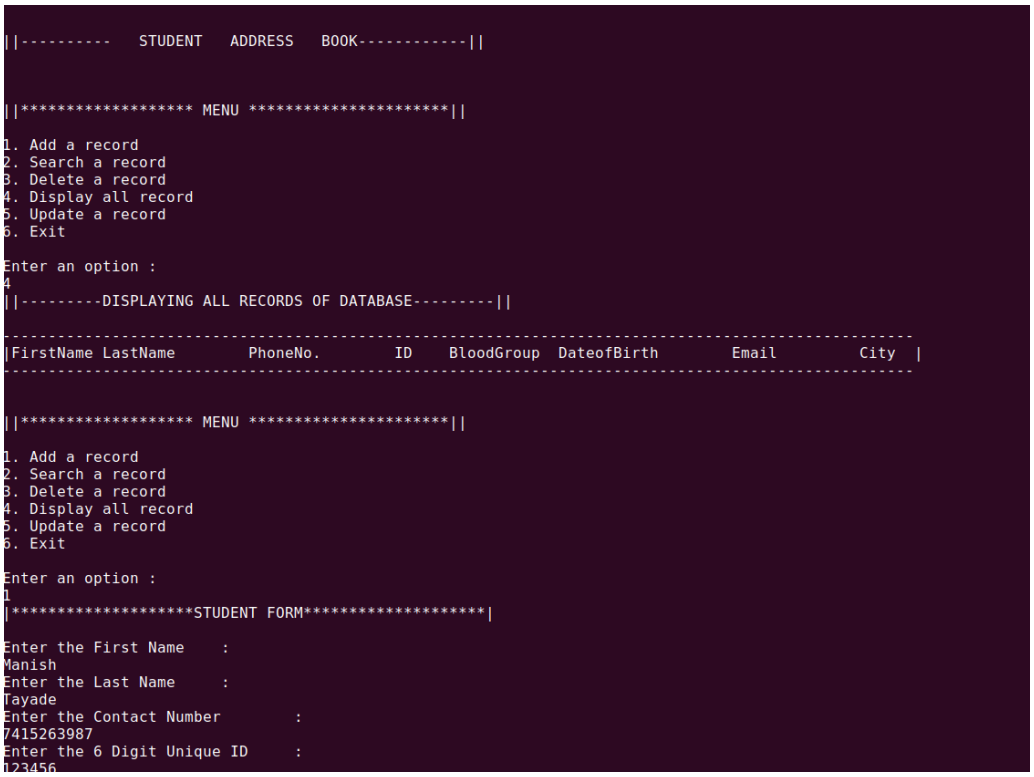
Code for updating records

```
#!/bin/bash
echo "||************UPDATING RECORD************||"
echo
echo "Enter the 6 Digit UniqueID : "
read UID1                        #taking input from user
echo ""
echo "|*|*|****RECORD TO BE MODIFIED****|*|*|"
if grep -w $UID1 data.txt #pattern matcher grep function finds a
#particular record in file
then
     sed -i '/'$UID1'/d' data.txt
     echo "Enter the First Name  : "
     read fname                      #TAKING FIRST NAME AS INPUT
     echo "Enter the Last Name   : "
```

```
      read lname                          #TAKING LAST NAME AS INPUT
      echo "Enter the Contact Number   : "
      read phno
      echo "Enter the Blood Group : "
      read bgl                            #TAKING BLOOGGROUP AS INPUT
      echo "Enter the Date of Birth    : "
      read dob                            #TAKING DATEOFBIRTH AS INPUT

      echo "Enter the Email Address    : "
      read email                          #TAKING EMAIL AS INPUT
      echo "Enter the current residence city :  "
      read city                           #TAKING  CITY AS INPUT

      echo $fname $lname $phno $UID1 $bgl $dob $email $city >> data.txt #input to
file writing data to file
else
      echo " Record Does not Exist "
fi
```

Output :

Screen Shots for above code



```
Enter the Date of Birth :
21-06-2000
Enter Blood Group :
A+
Enter the Email Address :
man@gmail.com
Enter the Current Residence City :
Pune


||****************** MENU **********************||

1. Add a record
2. Search a record
3. Delete a record
4. Display all record
5. Update a record
6. Exit

Enter an option :
4
||---------DISPLAYING ALL RECORDS OF DATABASE---------||

-------------------------------------------------------------------------------------------------
|FirstName LastName        PhoneNo.        ID    BloodGroup  DateofBirth        Email         City  |
-------------------------------------------------------------------------------------------------
Manish     Tayade          7415263987     123456     A+      21-06-2000   man@gmail.com        Pune


||****************** MENU **********************||

1. Add a record
2. Search a record
3. Delete a record
4. Display all record
5. Update a record
6. Exit

Enter an option :
2
::::::::::::Search A Record::::::::::

Enter the StudentID     :
123456
```

```
Enter an option :
3
||**********DELETING RECORD*********||

Enter the StudentId  :
123456
||**********RECORD DELETED IS SHOWN BELOW*********||


-------------------------------------------------------------------------------------------------
|FirstName LastName        PhoneNo.        ID    BloodGroup  DateofBirth        Email         City  |
-------------------------------------------------------------------------------------------------
Manish     Tayade          7415263987     123456     A+      21-06-2000  iopw@gmail.com       Delhi


||****************** MENU **********************||

1. Add a record
2. Search a record
3. Delete a record
4. Display all record
5. Update a record
6. Exit

Enter an option :
4
||---------DISPLAYING ALL RECORDS OF DATABASE---------||

-------------------------------------------------------------------------------------------------
|FirstName LastName        PhoneNo.        ID    BloodGroup  DateofBirth        Email         City  |
-------------------------------------------------------------------------------------------------


||****************** MENU **********************||

1. Add a record
2. Search a record
3. Delete a record
4. Display all record
5. Update a record
6. Exit

Enter an option :
s
```

Using Dialog with GUI

Main Code :

```
#!/bin/bash
try=3

while [ $try -ne 0 ]
do
      #login credentials username
      dialog --backtitle "Student Address Book" --title "Admin    Details" --
      inputbox "Username"  10 20 2> temp1.txt

      #login credentials password
      dialog --backtitle "Student Address Book" --title "Admin    Details" --
      passwordbox "Password: "  10 20 2> temp3.txt

      #username stored in temp1.txt is fetched by name variable
      name=$(cat temp1.txt)
      #password stored in temp2.txt is fetched by pass variable
      pass=$(cat temp3.txt)

      #checking if username and password matches
      if [ $name = "Admin" -a $pass = "net" ]
      then
            #options for user to select
            dialog --title " Menu " --backtitle "Logged in as $name" \
            --menu "Hi $name ,Choose an option : " 25 55 6 \
            1 "Add a record " \
            2 "Search a record" \
            3 "Remove a record " \
            4 "Display records " \
            5 "Update a record " \
            6 "Exit" 2> temp2.txt

                  #choose stores the options selected by user
                  choose=$(cat temp2.txt)

                  while [ $choose -ne 6 ]
                  do
                      if [ $choose -eq 1 ] ;then
                         ./add.sh                  #./add.sh file is executed
                      elif [ $choose -eq 2 ];then
                         ./search.sh               #./search.sh file is executed
                         sleep 1
                      elif [ $choose -eq 3 ]; then
                          ./delete.sh              #./delete.sh file is executed
                          sleep 2
                      elif [ $choose -eq 4 ]; then

                          ./display.sh             #./display.sh file is executed
                          sleep 2
                      elif [ $choose -eq 5 ];then
                          ./update.sh              #./update.sh file is executed
                          sleep 2
```

```
                    else [ $choose -eq 6 ]
                      clear
                      exit 0
                    fi
                    dialog --title " Menu " \
                  --menu "Hi $name ,Choose an option : " 25 55 6 \
                  1 "Add a record " \
                  2 "Search a record" \
                  3 "Remove a record " \
                  4 "Display records " \
                  5 "Update a record " \
                  6 "Exit" 2> temp2.txt
                  choose=$(cat temp2.txt)
                 done
      else
            #if username and password does not match displays the following
            #message
            try=`expr $try - 1`
            dialog --msgbox "Bad Credentials You Have Exactly $try Attempts Left"
25 55
               if [ $try -eq 0 ]
               then
                  break
               fi
      fi
      if [ $try -eq 0]
      then
         break

      fi
done
```

Code for Deleting a Record :

```
#!/bin/bash

#ID is taken from user in order to delete that record
ID=$(dialog --title "Record Deletion" --inputbox "Enter the Student ID " 25 25 --
output-fd 1)

#if ID taken is matched then it is deleted
if grep -w $ID data.txt
then

      #Displays the record which will be deleted
      dialog --title "Record To be deleted" --msgbox "$(grep "${ID}" data.txt)" 25
      25

      #Confirms to delete the record or not
      dialog --yesno "Are you sure you want to delete the record?" 25 25

            dec=$?
```

```
          if [ $dec -eq 0 ]
          then
               #following command deletes the record
               sed -i '/'$ID'/d' data.txt
               dialog --msgbox "Record Deleted Successfully" 25 25
          else
               dialog --title "Record Deletion" --msgbox "Record Not Deleted "
               25 25
          fi
else
     #if record is not present this will be processed
     dialog --title "Record Deletion" --msgbox "Record Does Not Exist " 25 25

fi
```

Code For Adding Record :

```
#!/bin/bash

#Taking input from user
fname=$(dialog --title "|Enter Details|" --inputbox "Enter First Name " 25 25 -
      output-fd 1 )

lname=$(dialog --title "|Enter Details|" --inputbox "Enter Last Name  " 25 25 -
     output-fd 1)

con=$(dialog --title "|Enter Details|" --inputbox "Enter Contact Number  " 25 25
     --output-fd 1 )

ID=$(dialog --title "|Enter Details|" --inputbox "Enter 6 Digit Unique Id  " 25
     25 --output-fd 1 )

bg=$(dialog --title "|Enter Details|" --inputbox "Enter Blood Group " 25 25 -
     output-fd 1 )

dob=$(dialog --title "|Enter Details|" --inputbox "Enter Date Of Birth " 25 25 --
     output-fd 1 )

email=$(dialog --title "|Enter Details|" --inputbox "Enter Email " 25 25 -
     output-fd 1 )

city=$(dialog --title "|Enter Details|" --inputbox "Enter City" 25 25 --output-fd
     1 )

     #writing into data.txt file
     echo   $fname " " $lname " " $con " " $ID "   " $bg "   "  $dob "   "
     $email "   " $city  >> data.txt

     #displays the record that is added
     dialog --form "Following Record Added " \
     50 50 38 \

     "FirstName:"                    1 1    "$fname"  1 15 50 0 \
     "LastName:"                     2 1   "$lname"   2 15 50 0 \
```

```
"Contact Number:"               3 1   "$con"     3 16 50 0 \
"ID:"                      4 1  "$ID"            4 15 50 0 \
"Blood Group:"                  5 1   "$bg"            5 15 50 0 \
"Date of Birth:"                6 1   "$dob"           6 15 50 0 \
"Email : "                      7 1   "$email"         7 15 50 0 \
"City:"                         8 1   "$city"          8 15 50 0
```

Code For Updating A Record :

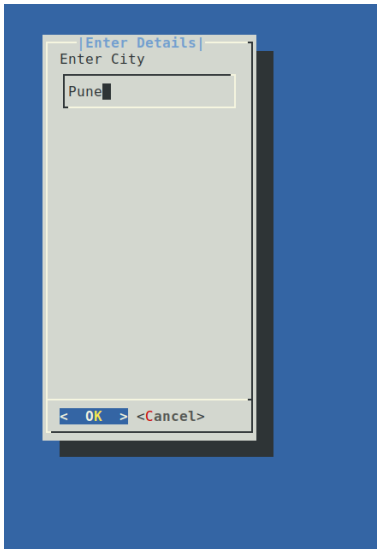
```
#!/bin/bash

#taking id of a particular record to update record
 ID=$(dialog --title "Search Record" --inputbox "Enter the Student ID " 25 25 --
output-fd 1)

#following command in if condition matches the record in data.txt
if grep -w $ID data.txt
then
     #displays the record in message box
     dialog --title "Record To be Modified " --msgbox "$(grep "${ID}" data.txt)"
35 45

     sed -i '/'$ID'/d' data.txt

     #inputs new values in that record
     fname=$(dialog --title "|Enter Details|" --inputbox "Enter First Name " 25
          25 --output-fd 1 )

     lname=$(dialog --title "|Enter Details|" --inputbox "Enter Last Name  " 25
          25 --output-fd 1)

     con=$(dialog --title "|Enter Details|" --inputbox "Enter Contact Number  "
          25 25 --output-fd 1 )

     bg=$(dialog --title "|Enter Details|" --inputbox "Enter Blood Group " 25 25
          --output-fd 1 )

     dob=$(dialog --title "|Enter Details|" --inputbox "Enter Date Of Birth " 25
          25 --output-fd 1 )

     email=$(dialog --title "|Enter Details|" --inputbox "Enter Email " 25 25 -
          output-fd 1 )

     city=$(dialog --title "|Enter Details|" --inputbox "Enter City" 25 25 -
          output-fd 1 )

     #Writing into data.txt file
     echo   $fname " " $lname " " $con " " $ID "   " $bg "   "  $dob "   "
     $email "   " $city  >> data.txt

     dialog --msgbox "Record Modified Successfully" 35 45
```

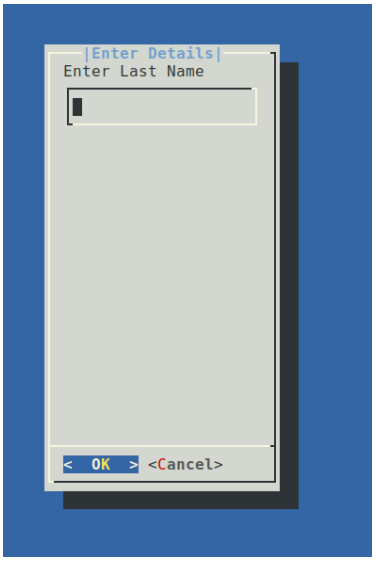
```
else
      echo " Record Does not Exist "
fi
```

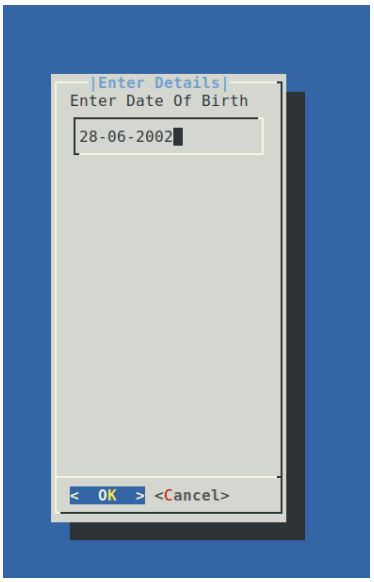
Code for Searching Record :

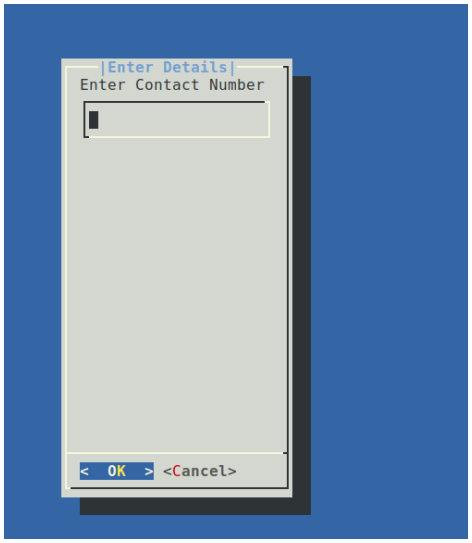
```
#!/bin/bash

#taking id of a particular record to search record

ID=$(dialog --title "Search Record" --inputbox "Enter the Student ID " 25 25 --
output-fd 1)

#if record is found then it is saved in to founddata.txt
if grep -w -i $ID data.txt
then

      grep -w $ID data.txt > founddata.txt

      #fetches the record stored in founddata.txt
      read  fname lname   con   uid     Bg     dob   email   city < founddata.txt

      #Displays the record that is mathced
      dialog --title "Record Found" --backtitle "Logged in as Admin"\
      --form "RECORD DETAILS ARE GIVEN BELOW :  " \
      50 50 38 \

      "FirstName:"                    1 1     "$fname"      1 15 50 0 \
      "LastName:"                     2 1   "$lname"   2 15 50 0 \
      "Contact Number:"               3 1   "$con"     3 15 50 0 \
      "ID:"                     4 1   "$uid"           4 15 50 0 \
      "Blood Group:"                  5 1   "$Bg"            5 15 50 0 \
      "Date of Birth:"                6 1   "$dob"           6 15 50 0 \
      "Email : "                      7 1   "$email"         7 15 50 0 \
      "City:"                         8 1   "$city"          8 15 50 0

else
      #if record is not found the following is processed
      dialog --title "Search Record" --msgbox "Record Not Found " 25 25

fi
```

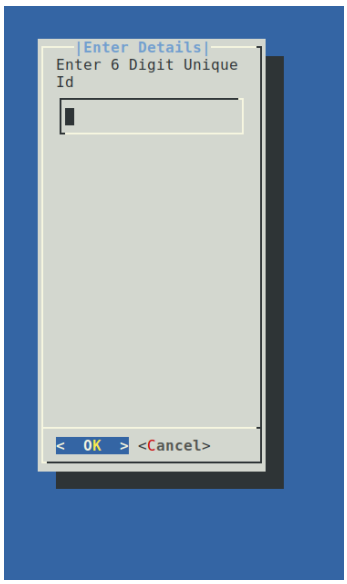
OUTPUT'S USING GUI DIALOG

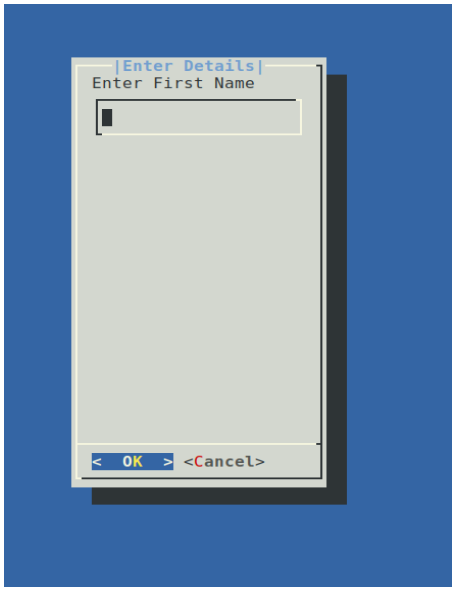




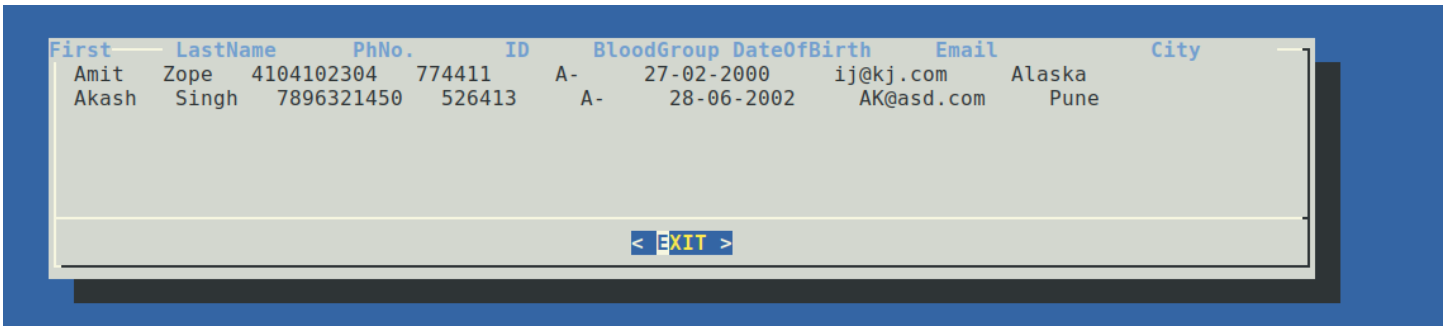


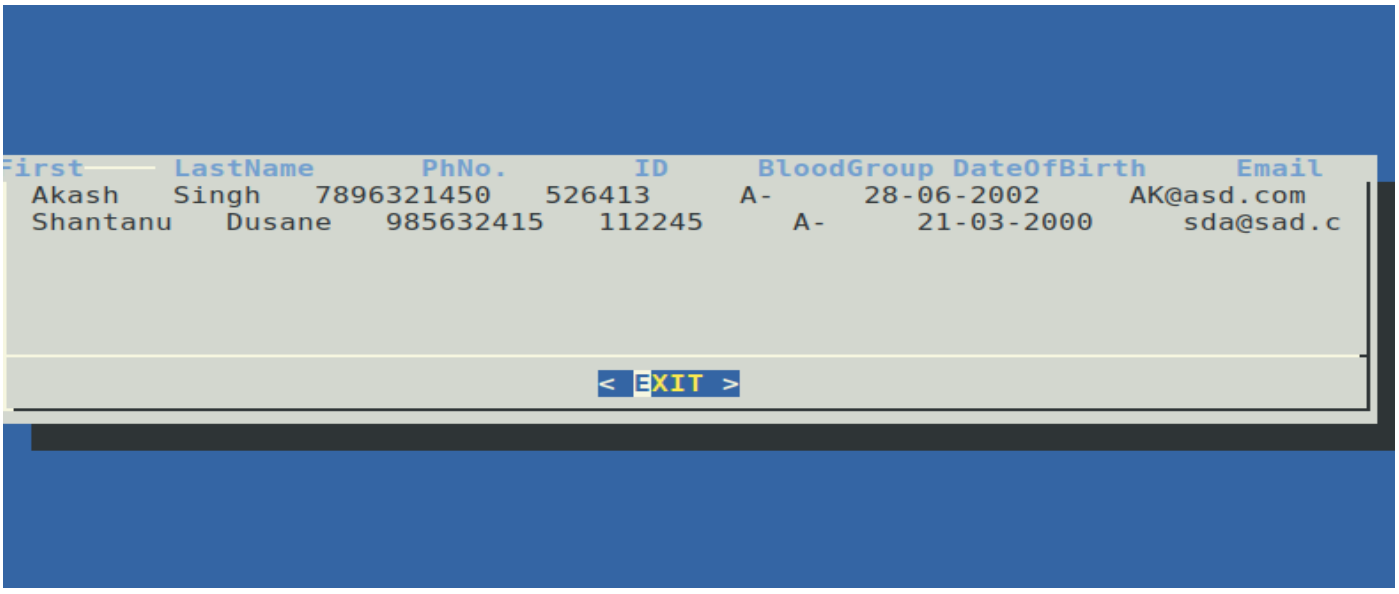




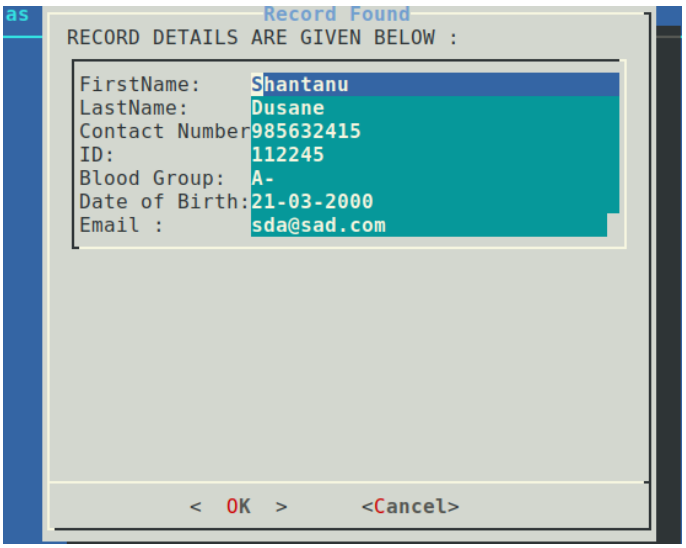

|Enter Details|
Enter First Name
< OK >
<Cancel>


Record To be Modified
Akash Singh 4563217896 526413 A+
14-06-2000 ak@singh.com Pune
< OK >


First LastName PhNo. ID BloodGroup DateOfBirth Email City
Amit Zope 4104102304 774411 A- 27-02-2000 ij@kj.com Alaska
Akash Singh 7896321450 526413 A- 28-06-2002 AK@asd.com Pune
< EXIT >


First LastName PhNo. ID BloodGroup DateOfBirth Email
Akash Singh 7896321450 526413 A- 28-06-2002 AK@asd.com
Shantanu Dusane 985632415 112245 A- 21-03-2000 sda@sad.c
< EXIT >


Record Modified Successfully
< OK >


as
Record Found
RECORD DETAILS ARE GIVEN BELOW :
FirstName: Shantanu
LastName: Dusane
Contact Number 985632415
ID: 112245
Blood Group: A-
Date of Birth: 21-03-2000
Email : sda@sad.com
< OK > <Cancel>