Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester: (Spring, Year:2021), B.Sc. in CSE (Day/Eve)

Course Title: Operating System Lab

Course Code: CSE 310 Section: DA

Lab Project Name: Student Database System

Student Details

Name		ID
1.	Jahid Hassan Santo	201002463

Submission Date : 16/05/2022

Course Teacher's Name : Mr. Jahidul Islam

Lab Project Status

Marks: …… **Signature: .....**

Comments: ..... **Date: .....**

Table of Contents

Chapter 1
Introduction

1.1 Introduction

Using this database system, we understand that there should be a database process. In telecommunications, computing, and information architecture, a data bank or databank is a repository of information on one or more topics - a database - organized in such a way as to facilitate the retrieval of local or remote information and to process many continuous queries over a long period of time. In this database, we have tried to use a considerable amount of basic information that a student database system can maintain. Here we create four sections and take some valuable features. We also maintain user safety here.

1.2 Design Goals/Objective

- Organize in a way that facilitates the retrieval of local or remote information and can process many continuous queries over a long period of time.
- To ensure user safety.
- Store student information and database information in a way that is easy to access.
- Keep track of student results and location.

Chapter 2

Design/Development/Algorithms/Implementation of the Project

2.1 Design and Development Software or Kit.

Just like Windows, iOS, and Mac OS, Linux is an operating system. In fact, one of the most popular platforms on the planet, Android, is powered by the Linux operating system. An operating system is software that manages all of the hardware resources associated with your desktop or laptop.

Linux has been around since the mid-1990s and has since reached a user-base that spans the globe. Linux is actually everywhere: It's in your phones, your thermostats, in your cars, refrigerators, Roku devices, and televisions. It also runs most of the Internet, all of the world's top 500 supercomputers, and the world's stock exchanges.

Linux is also distributed under an open-source license. Open source follows these key tenants:

- The freedom to run the program, for any purpose.
- The freedom to study how the program works, and change it to make it do what you wish.
- The freedom to redistribute copies so you can help your neighbor.
- The freedom to distribute copies of your modified versions to others.

These points are crucial to understanding the community that works together to create the Linux platform.

Without a doubt, Linux is an operating system that is “by the people, for the people”. These tenants are also a main factor in why many people choose Linux. It’s about freedom and freedom of use and freedom of choice. You must have

- A PC or Laptop
- Linux Operating System.
- Minimum 1 GB of hard disk space
- 1024×768 or greater screen resolution.

Completing this project we have to install a new package named ‘Whiptail’. Below is instruction for install whiptail in Ubuntu.

Quick Install Instructions of whiptail on Ubuntu Server. It’s Super Easy! simply click on Copy button to copy the command and paste into your command line terminal using built-in APT package manager.

See below for quick step by step instructions of SSH commands, Copy/Paste to avoid miss-spelling or accidently installing a different package.

Quick Install Steps:

- Step 1

  sudo apt-get update -y

- Step 2

  sudo apt-get install -y whiptail

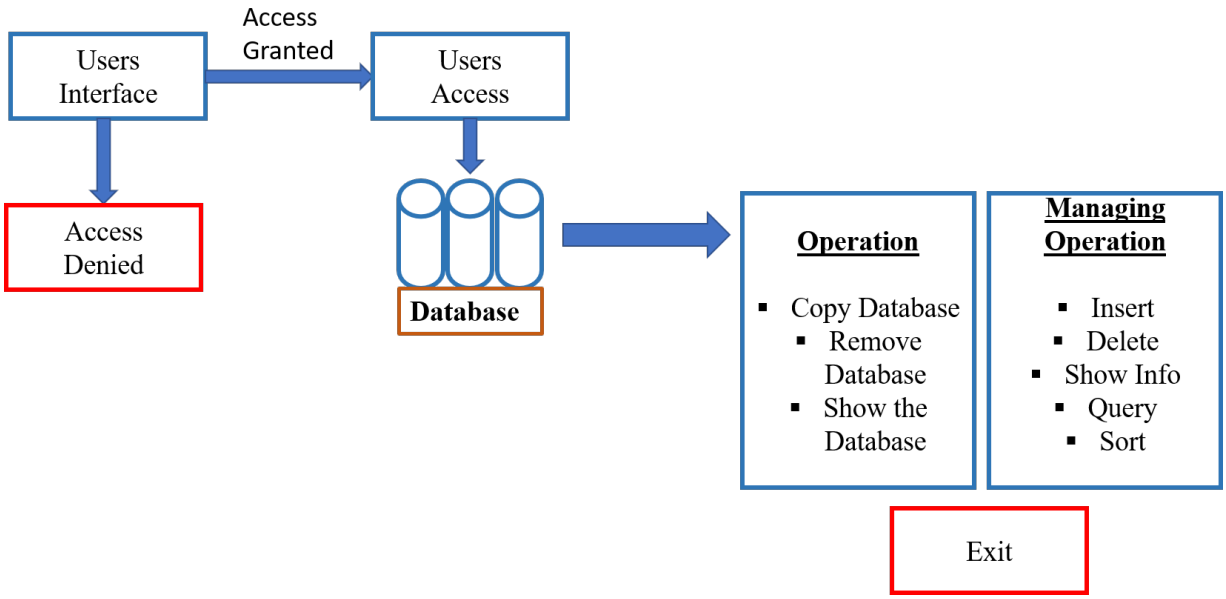
- Step 3

  Check the system logs to confirm that there are no related errors. You can use ZoomAdmin to check the logs, manager servers, host multiple websites and apps on your servers and more. The apps run in docker containers, to learn more

  see ZoomAdmin Features for list of features and demo videos. And you can start with the Free Plan.

Execute the commands above step by step. You can simply hit the copy button to copy the command and paste into the command line interface.

2.2 Project Design/Algorithms.



Chapter 3

Performance Evaluation

2.3 Simulation Environment/Simulation Procedure

This is the user's interface where two option is mention for users. '1' for execute the program and '0' for abort.

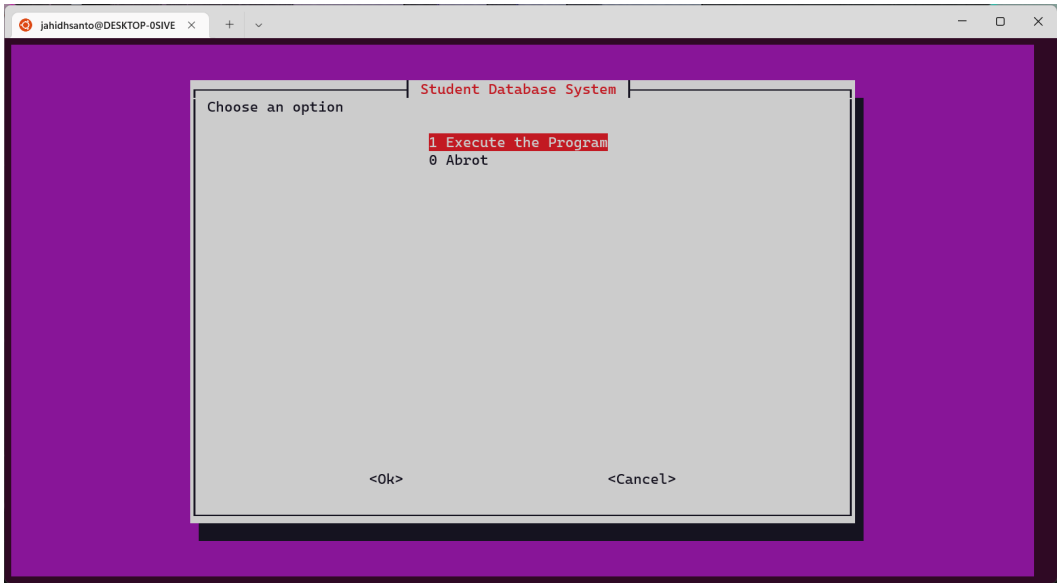


Figure 3.1.1: Users Interface

This is for users log in we have to input the user's id and password.

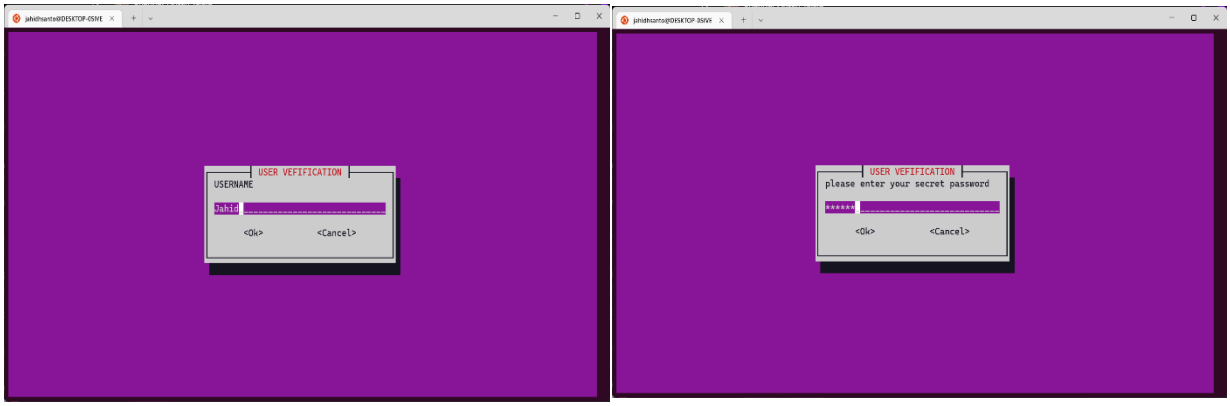


Figure 3.1.2: Users Log In

This is the main page of our project or starting page.

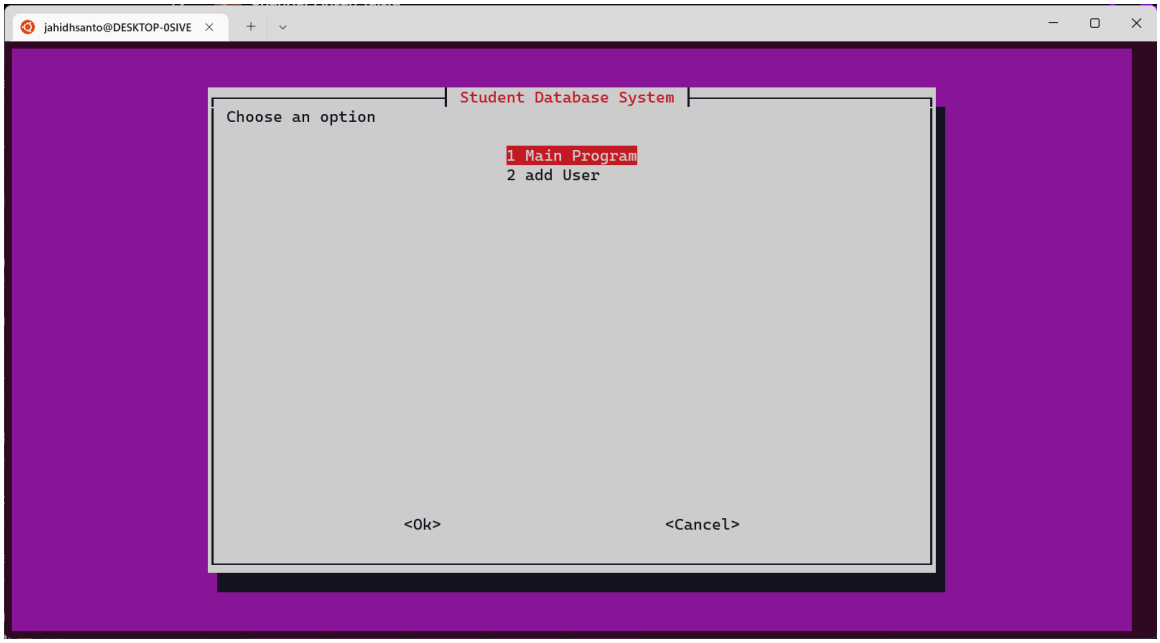


Figure 3.1.3: Entering into the Program

Here 6 option for every option different operation will be execute.

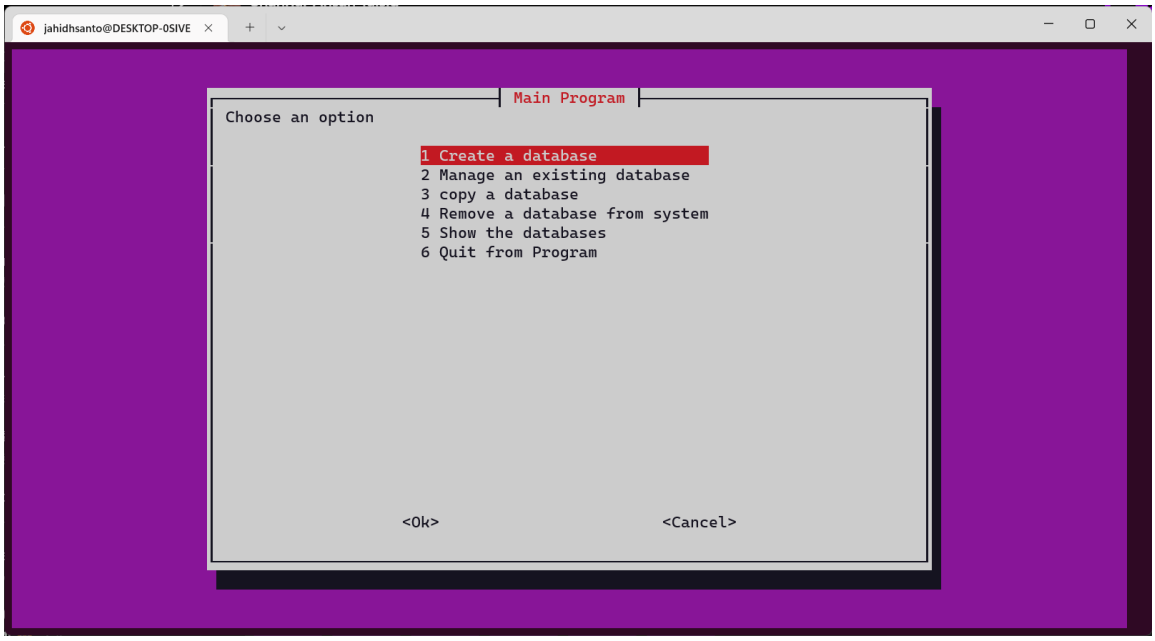
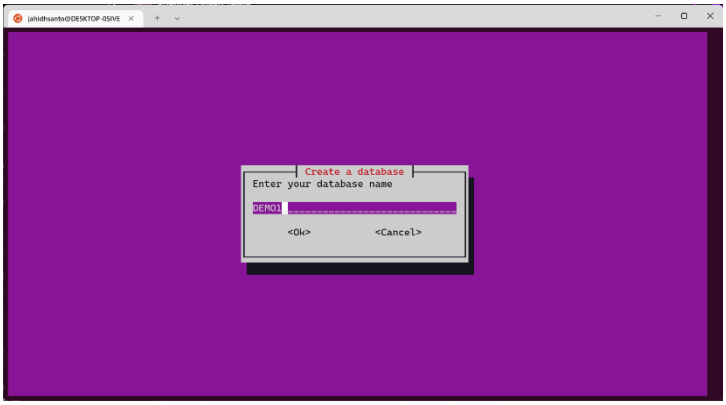


Figure3.1.4: Operation

Here we create a database.



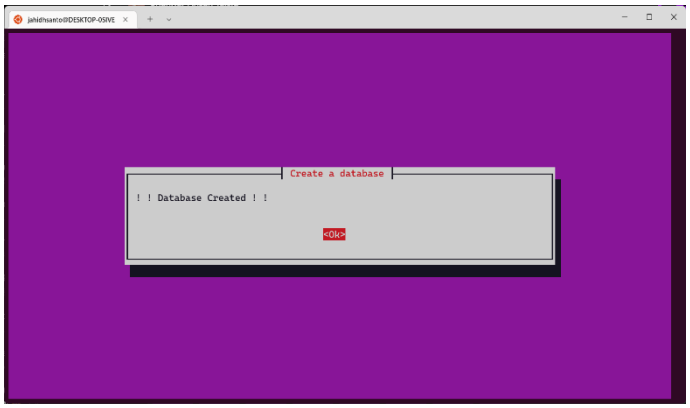


Figure 3.1.5: Create Database

Here we show the database are created

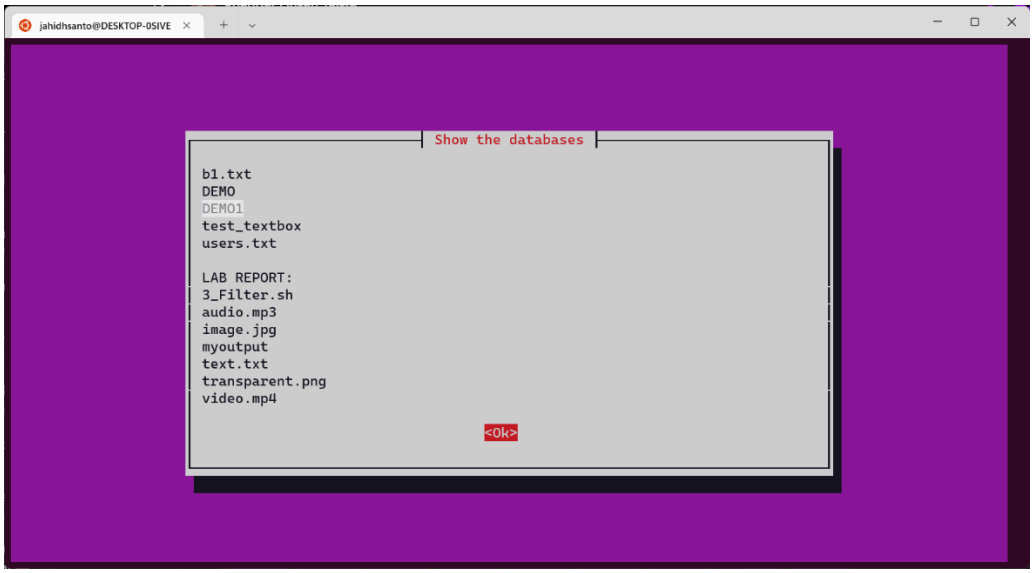
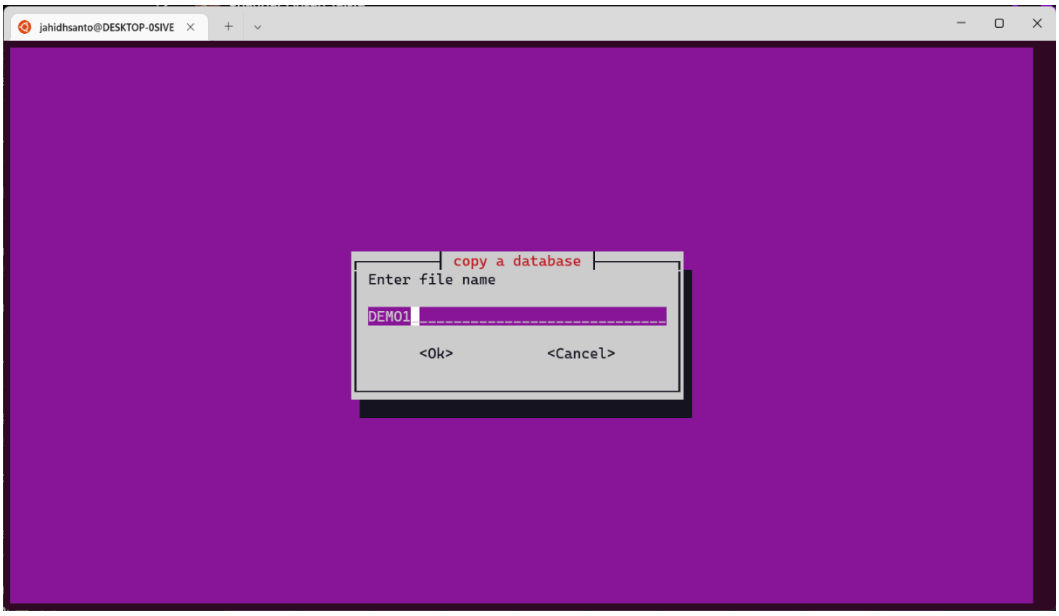


Figure3.1.6: Show the Database



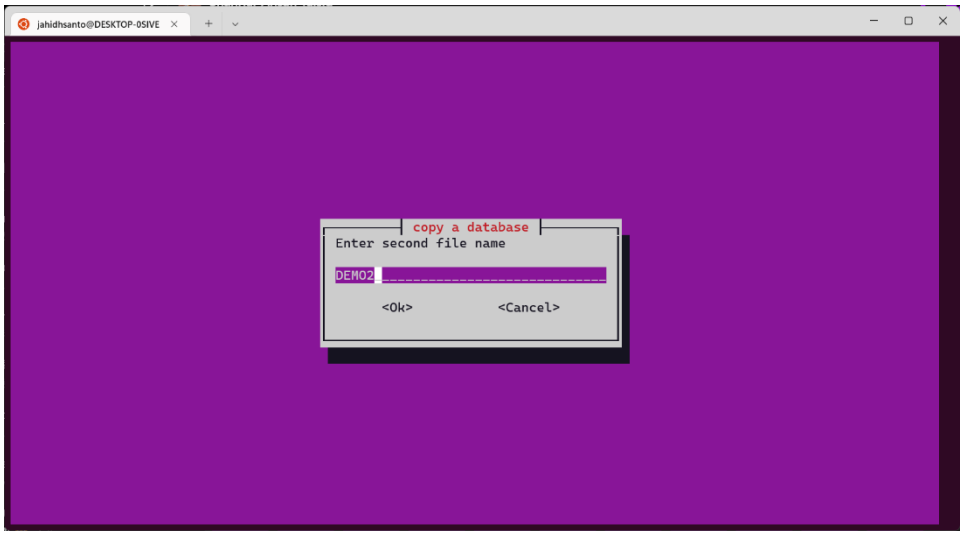


Figure 3.1.7: Copy a Database

Here we remove a database.

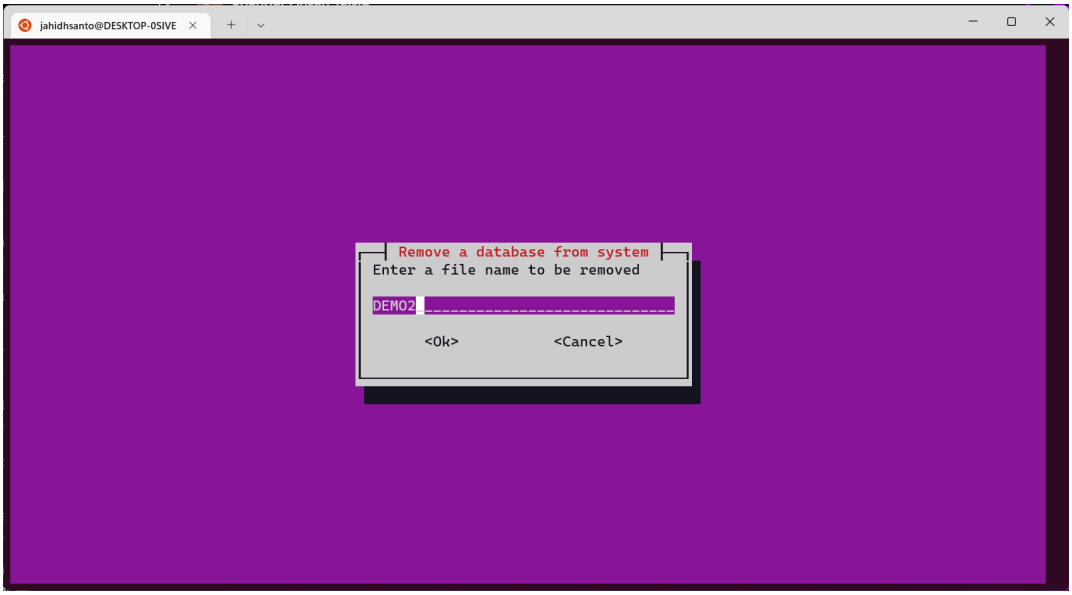
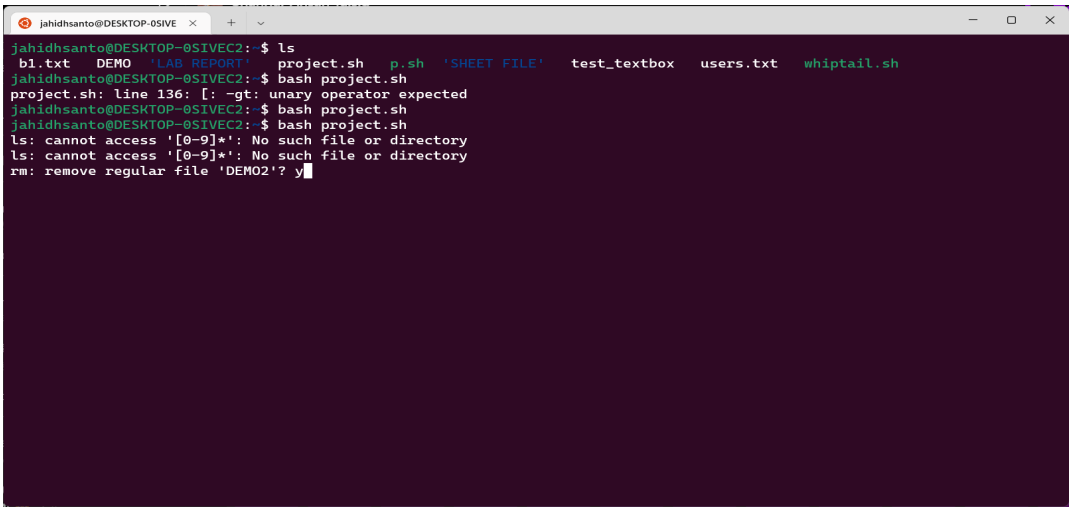


Figure 3.1.8: Removed Database



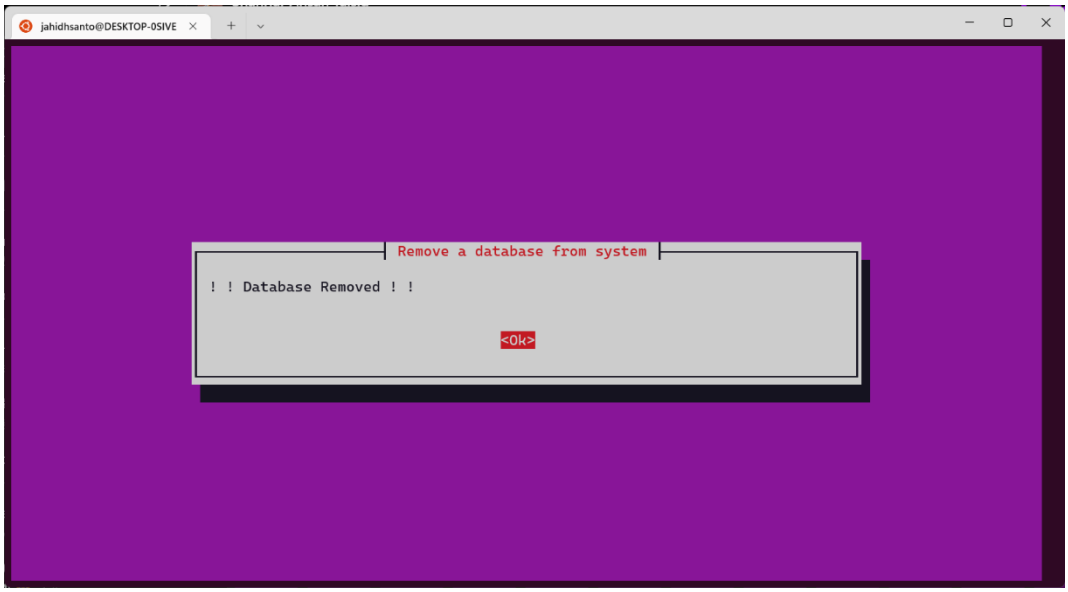


Figure 3.1.9: Confirmation of Removed Database.

Here option 2 for manage the database.

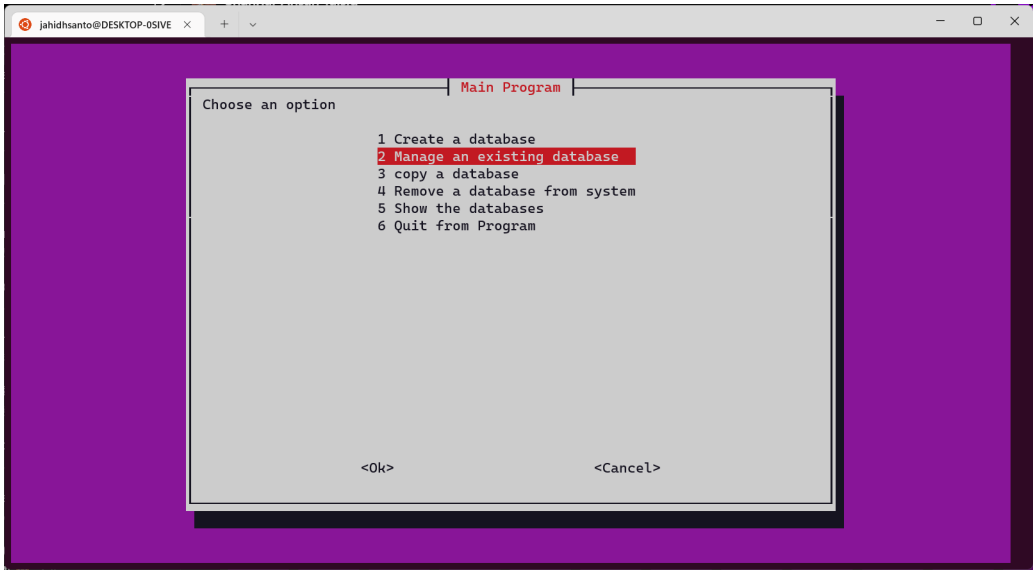


Figure 3.1.10 : Managing Database

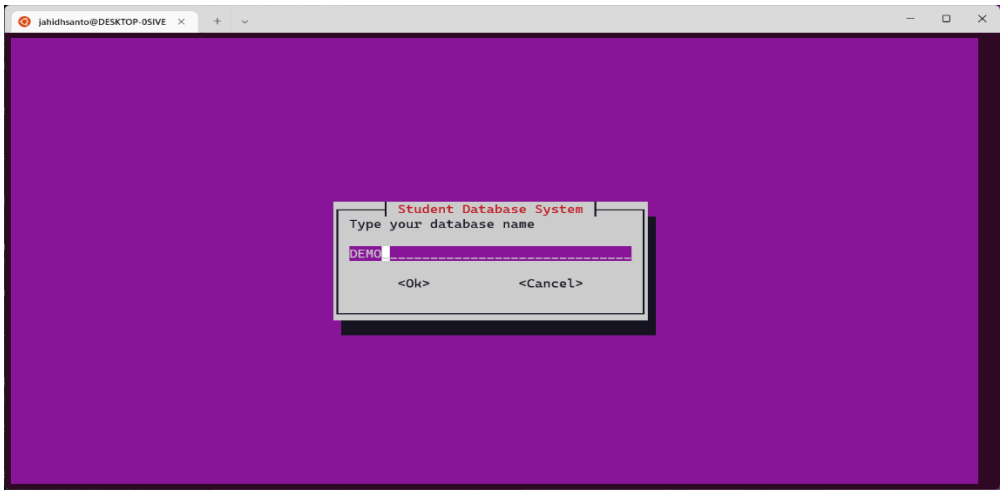


Figure 3.1.11 : Entering into a Database

There are some options for managing the data base 1 for insertion 2 for Delete 3 for show info 4 for query like sort, search etc.

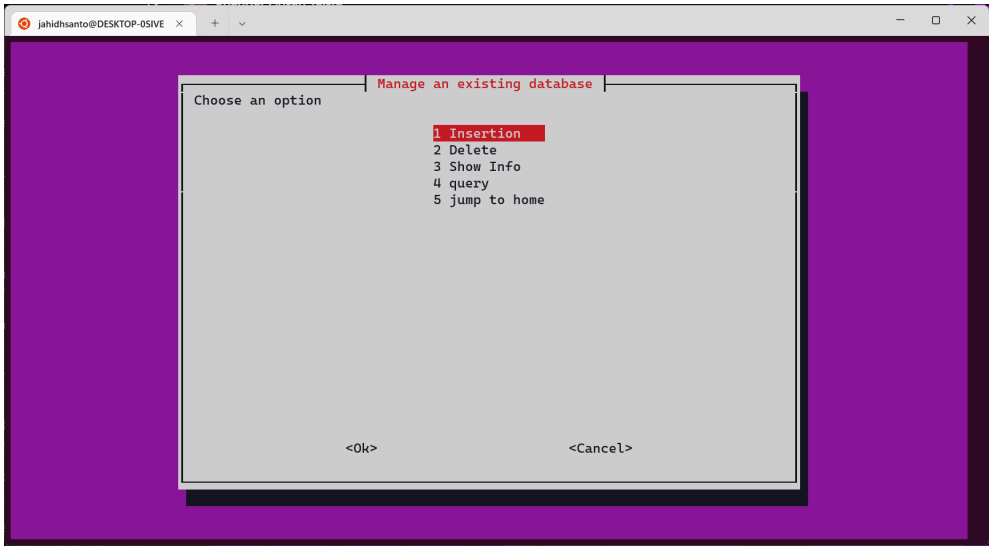
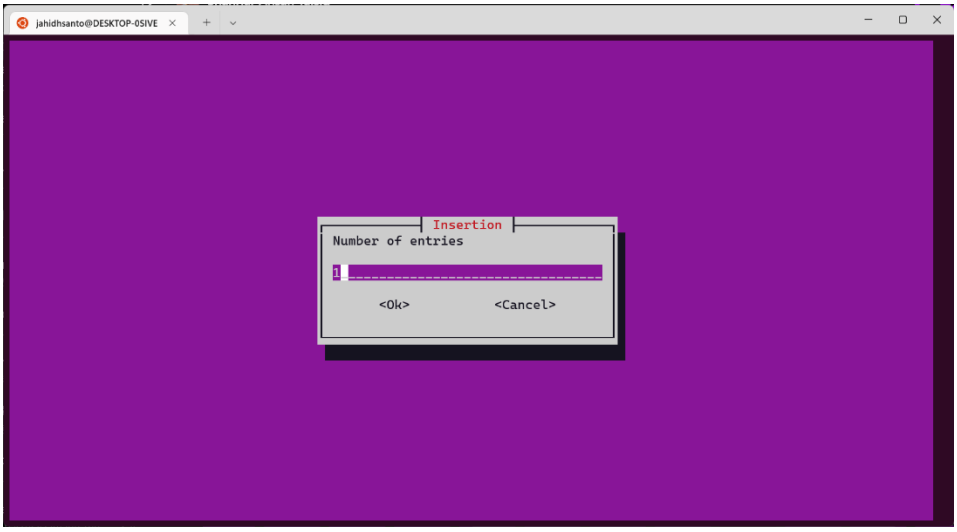


Figure 3.1.12 : Continue with insertion operation



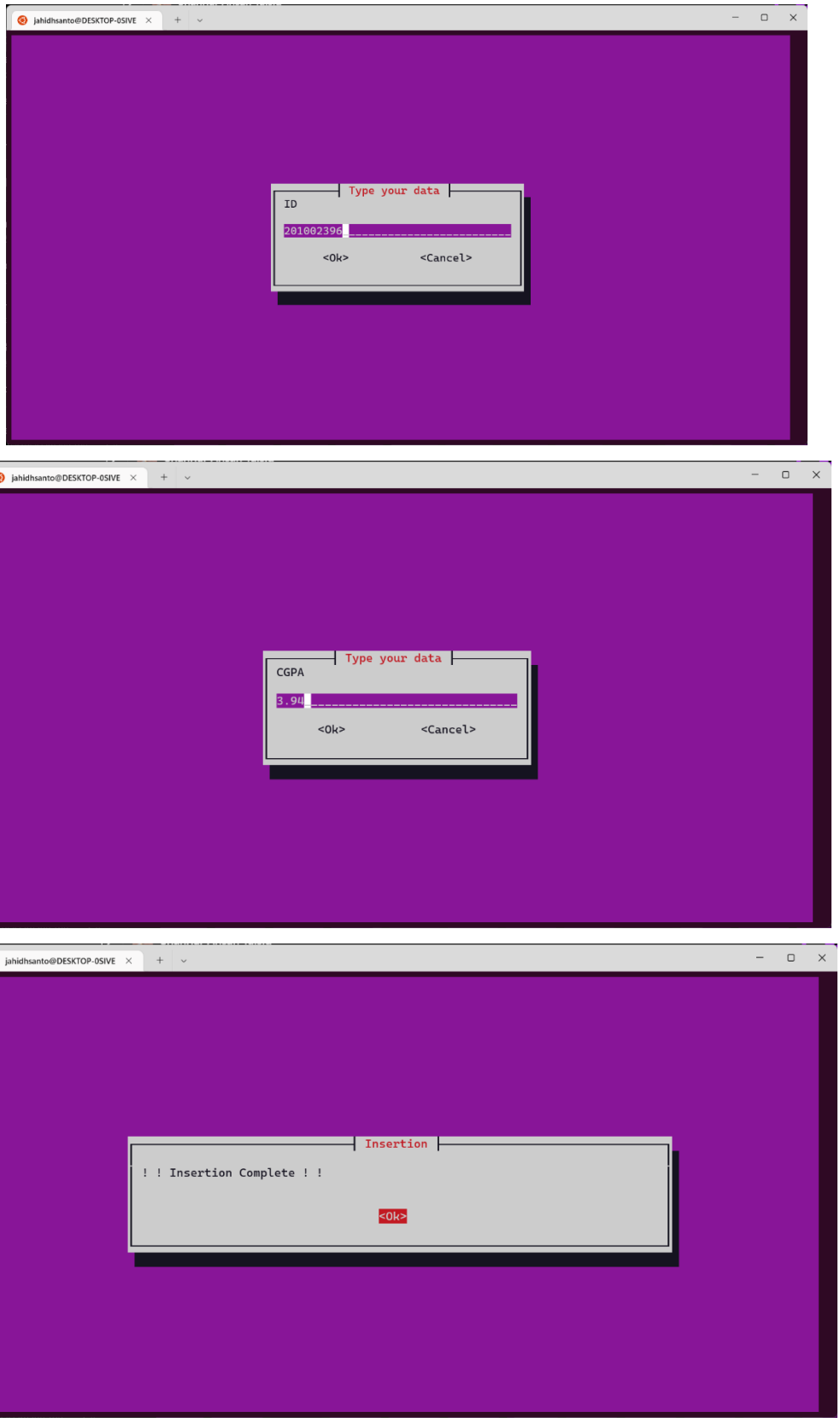


Figure 3.1.13 : Insert Data

Figure 3.1.14 : Show all the data of Database

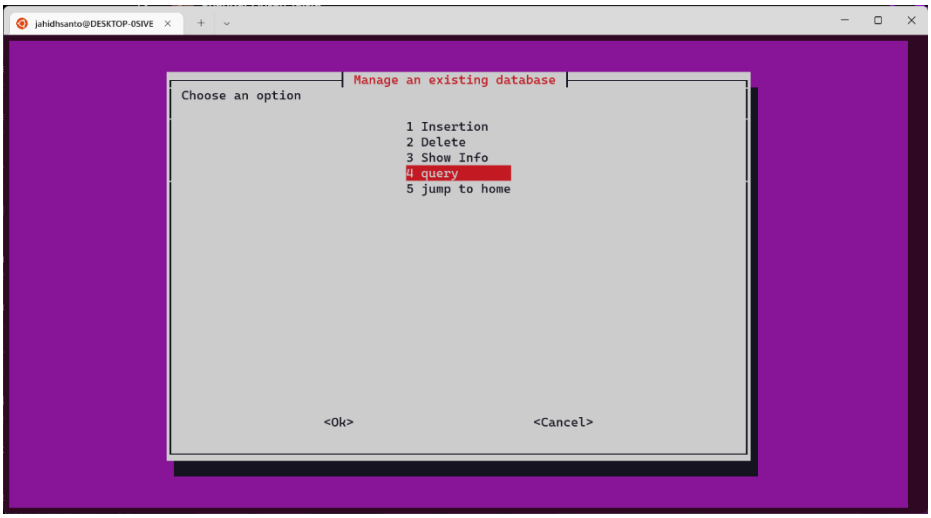
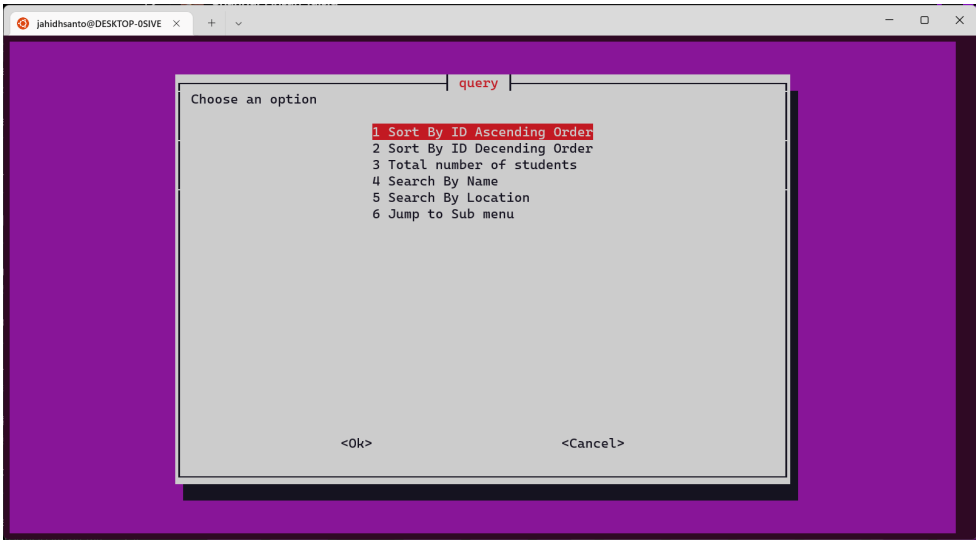


Figure 3.1.15 : Continue with Query Operation



Sort the data Ascending Order

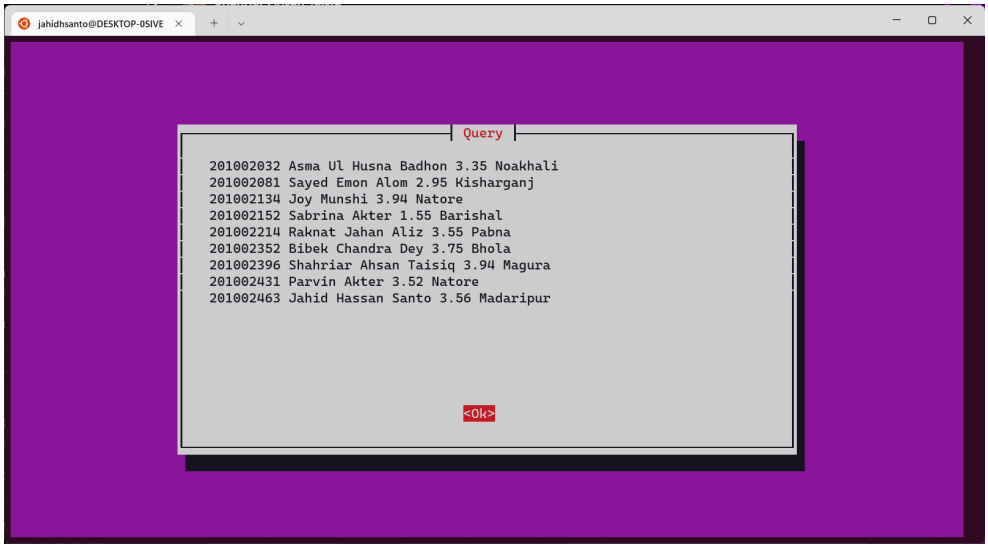
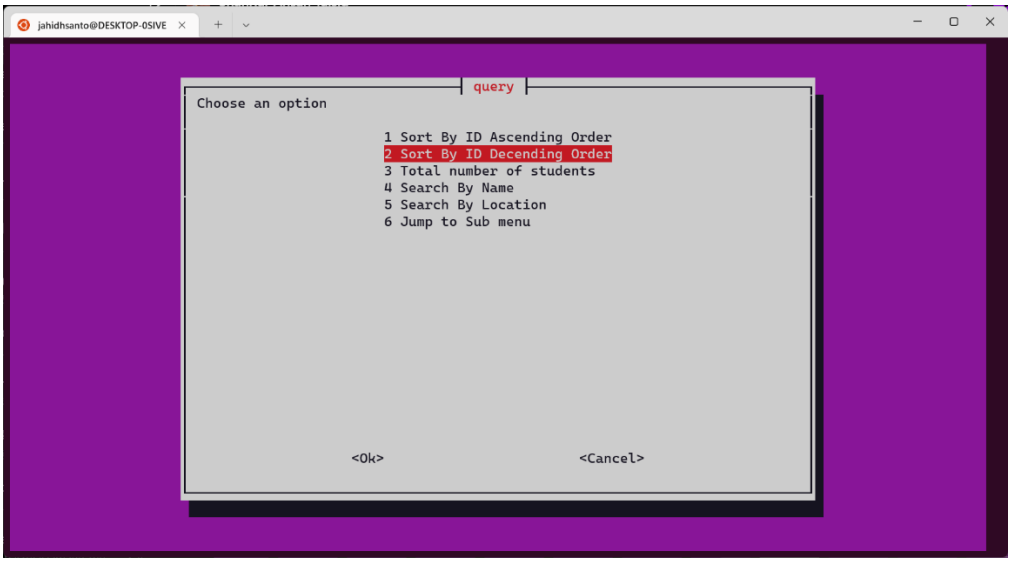


Figure 3.1.16 : Sort the data Ascending order

Sort the data Descending Order.



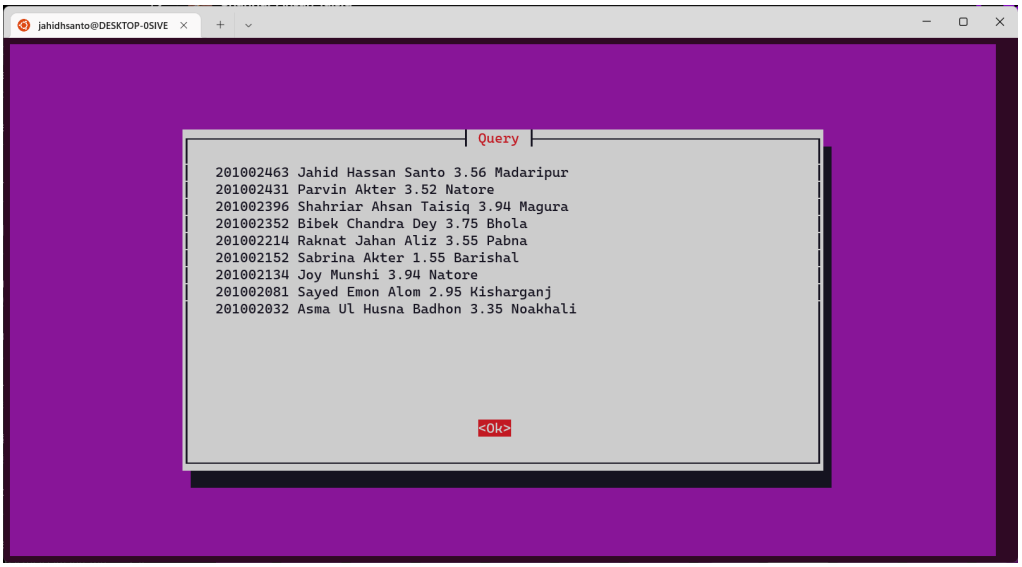


Figure 3.1.17 : Sort the data Descending Order

Count the number of students we inserted.

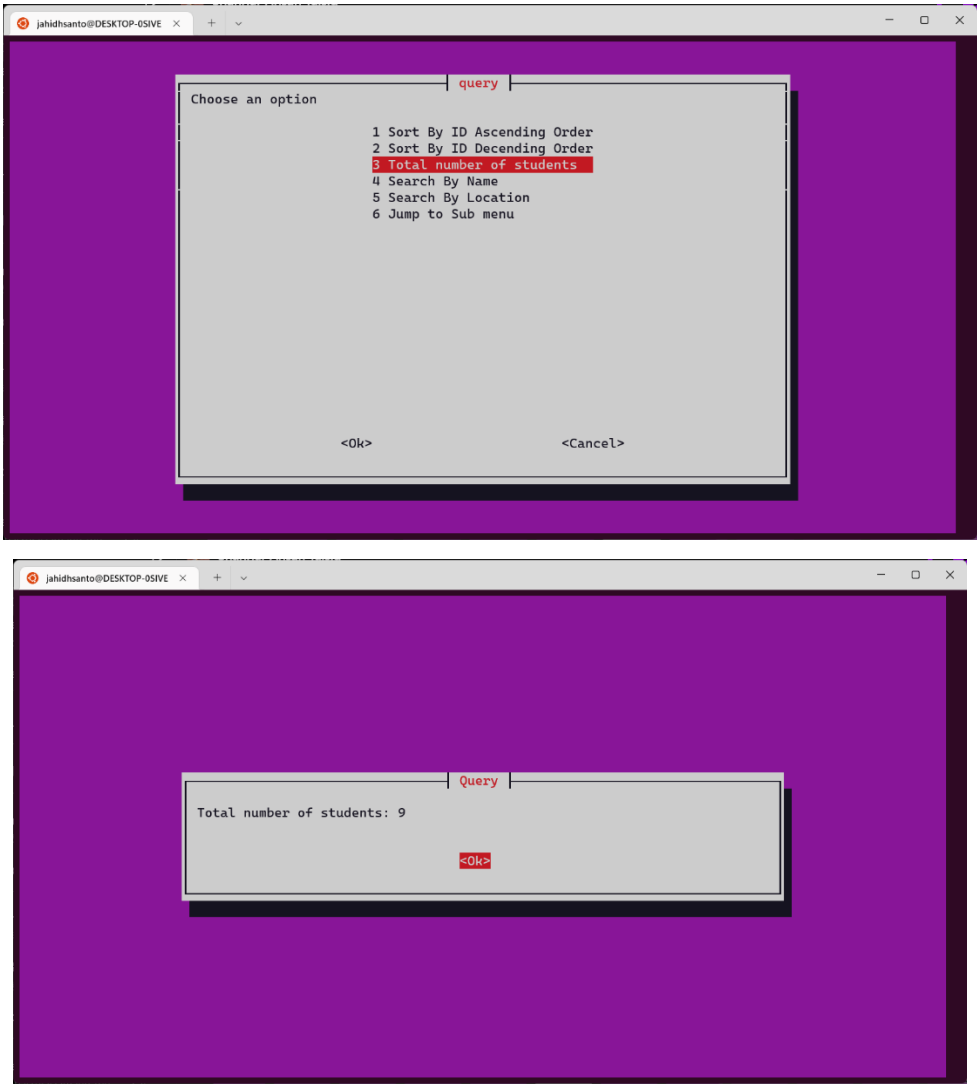


Figure 3.1.18 : Count the Total Number of Student

Search by the student name and Search by the Location of Student.

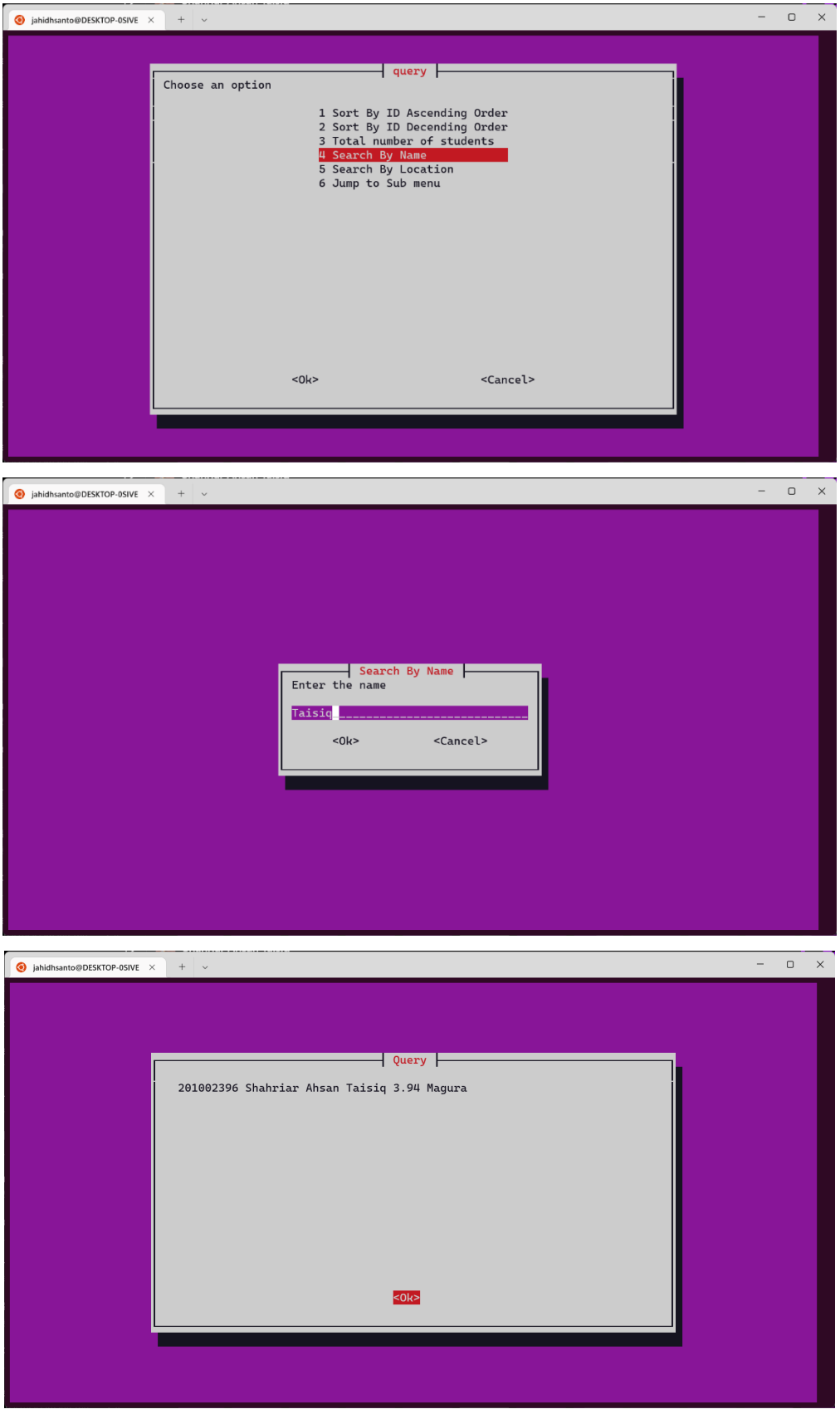


Figure 3.1.19 : Search by Name

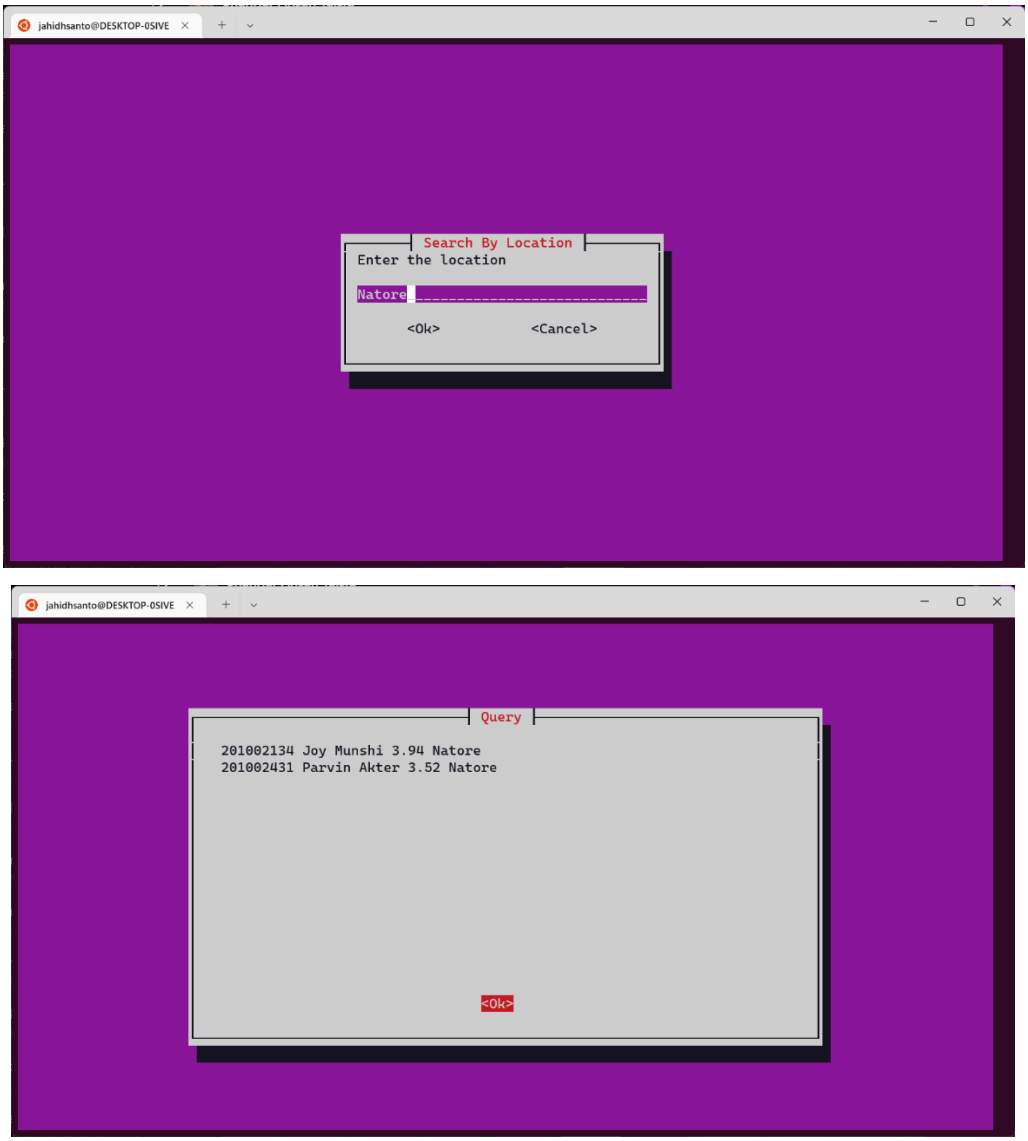


Figure 3.1.20 : Search by Location

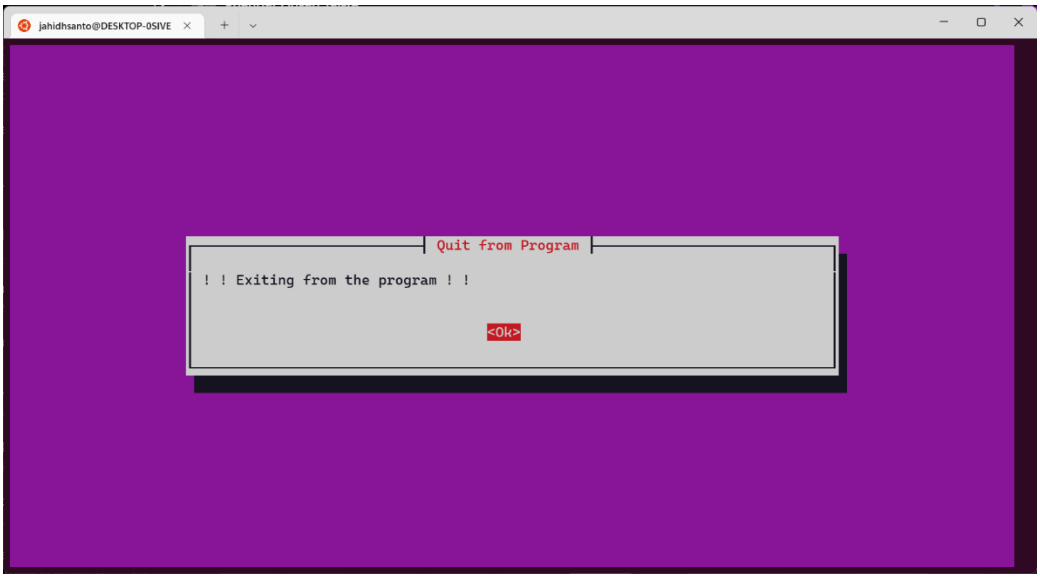


Figure 3.1.21 : Quit the Program

2.4 Project Implementation

```
1   Menu_Loop ()
2   {
3     select=100
4     while [ $select -gt 0 ]
5           do
6             select=$(whiptail --title "Main Program" --menu "Choose an option" 25 78 10 \
7             "1" "Create a database" \
8             "2" "Manage an existing database" \
9             "3" "copy a database" \
10            "4" "Remove a database from system" \
11            "5" "Show the databases" \
12            "6" "Quit from Program" 3>&1 1>&2 2>&3)
13
14            case "$select" in
15        1)
16          name=$(whiptail --inputbox "Enter your database name" 8 39 --title "Create a database"
    3>&1 1>&2 2>&3)
17          echo>$name
18          whiptail --title "Create a database" --msgbox "! ! Database Created ! !" 8 78
19          ;;
20        2)
21          db=$(whiptail --inputbox "Type your database name" 8 39 --title "Student Database
    System" 3>&1 1>&2 2>&3)
22          menu=100
23          while [ $menu -gt 0 ]
24          do
25            menu=$(whiptail --title "Manage an existing database" --menu "Choose an option" 25
    78 5 \
26            "1" "Insertion" \
27            "2" "Delete" \
28            "3" "Show Info" \
29            "4" "query" \
30            "5" "jump to home" 3>&1 1>&2 2>&3)
31            case "$menu" in
32              1)
33                num=$(whiptail --inputbox "Number of entries" 8 39 --title "Insertion" 3>&1 1>&2
    2>&3)
34                for(( i=0; i<num; i++ ))
```

```
35                  do
36                    id=$(whiptail --inputbox "ID" 8 39 --title "Type your data" 3>&1 1>&2 2>&3)
37                    nam=$(whiptail --inputbox "Name" 8 39 --title "Type your data" 3>&1 1>&2
    2>&3)
38                    gpa=$(whiptail --inputbox "CGPA" 8 39 --title "Type your data" 3>&1 1>&2
    2>&3)
39                    loc=$(whiptail --inputbox "Location" 8 39 --title "Type your data" 3>&1 1>&2
    2>&3)
40
41                    var=" $id $nam $gpa $loc "
42                    echo " $var " >>$db
43                    whiptail --title "Insertion" --msgbox "! ! Insertion Complete ! !" 8 78
44                  done
45                  ;;
46                2)
47                  ln=$(whiptail --inputbox "Enter the id number" 8 39 --title "Delete" 3>&1 1>&2
    2>&3)
48                  whiptail --title "ln" --msgbox "$ln" 30 78
49                  grep -v " $ln " $db
50                  ;;
51                3)
52                  #cat $db
53                  whiptail --title "Example Dialog" --msgbox "$(cat $db)" 30 78
54                  ;;
55                4)
56                  que=100
57                  while [ $que -gt 0 ]
58                  do
59                                                  que=$(whiptail --title "query" --menu "Choose an
    option" 25 78 6 \
60                    "1" "Sort By ID Ascending Order" \
61                    "2" "Sort By ID Decending Order" \
62                    "3" "Total number of students" \
63                    "4" "Search By Name" \
64                    "5" "Search By Location" \
65                    "6" "Jump to Sub menu" 3>&1 1>&2 2>&3)
66
67                    case "$que" in
68                      1)
```

```
69                          whiptail --title "Query" --msgbox "$(sort $db)" 20 78
70                          ;;
71                        2)
72                          whiptail --title "Query" --msgbox "$(sort -r $db)" 20 78
73                          ;;
74                        3)
75                          line=`wc -l < $db | tr -d ' ' `
76                          whiptail --title "Query" --msgbox "Total number of students: $line " 8 78
77                          ;;
78                        4)
79                          en=$(whiptail --inputbox "Enter the name" 8 39 --title "Search By Name"
    3>&1 1>&2 2>&3)
80                          whiptail --title "Query" --msgbox "$(grep $en $db)" 20 78
81                          ;;
82                        5)
83                          el=$(whiptail --inputbox "Enter the location" 8 39 --title "Search By Location"
    3>&1 1>&2 2>&3)
84                          whiptail --title "Query" --msgbox "$(grep $el $db)" 20 78
85                          ;;
86                        6)
87                          que=0
88                          ;;
89                        esac
90                  done
91                  ;;
92                5)
93                  menu=0
94                  ;;
95                esac
96            done
97                ;;
98          3)
99            file=$(whiptail --inputbox "Enter file name" 8 39 --title "copy a database" 3>&1 1>&2
    2>&3)
100           file1=$(whiptail --inputbox "Enter second file name" 8 39 --title "copy a database" 3>&1
    1>&2 2>&3)
101           if [ -f $file ]
102           then
103             cp $file $file1
```

```
104                whiptail --title "copy a database" --msgbox "! ! File Copied ! !" 8 78
105             else
106                whiptail --title "copy a database" --msgbox "! ! File does not exist ! !" 8 78
107             fi
108             ;;
109          4)
110             file=$(whiptail --inputbox "Enter a file name to be removed" 8 39 --title "Remove a
    database from system" 3>&1 1>&2 2>&3)
111             if [ -f $file ]
112             then
113                rm -i $file
114                whiptail --title "Remove a database from system" --msgbox "! ! Database Removed ! !" 8
    78
115             else
116                whiptail --title "Remove a database from system" --msgbox "! ! $file does not exist ! !" 8
    78
117             fi
118             ;;
119          5)
120             whiptail --title "Show the databases" --msgbox "$(ls {*[!.sh],[0-9]*})" 20 78
121             ;;
122          6)
123             whiptail --title "Quit from Program" --msgbox "! ! Exiting from the program ! !" 8 78
124             exit
125             ;;
126          esac
127    done
128 }
129
130 count=`wc -l < users.txt | tr -d ' '`
131 arr=( Jahid Taisiq Santo Shahriar )
132 control=1
133 while [ $control -gt 0 ]
134 do
135    control=$(whiptail --title "Student Database System" --menu "Choose an option" 25 78 5 \
136    "1" "Execute the Program" \
137    "0" "Abrot" 3>&1 1>&2 2>&3)
138    case "$control" in
139       1)
```

```
140         name=$(whiptail --inputbox "USERNAME" 8 39 --title "USER VEFIFICATION" 3>&1 1>&2
    2>&3)
141         for (( i=0; i<count; i++ ))
142         do
143           if [[ $name == "${arr[$i]}" ]]
144           then
145             pass="access"
146             new_pass=$(whiptail --passwordbox "please enter your secret password" 8 39 --title
    "USER VEFIFICATION" 3>&1 1>&2 2>&3)
147             if [[ $new_pass == $pass ]]
148             then
149               {
150                 for ((i = 0 ; i <= 100 ; i+=5)); do
151                   sleep 0.1
152                   echo $i
153                 done
154               } | whiptail --gauge "Please wait while we are sleeping..." 6 50 0
155
156               whiptail --title "USER VEFIFICATION" --msgbox "Access Granted" 8 78
157                                         entry=$(whiptail --title "Student Database System" --menu
    "Choose an option" 25 78 5 \
158               "1" "Main Program" \
159               "2" "add User" 3>&1 1>&2 2>&3)
160               case "$entry" in
161                 1)
162                   Menu_Loop
163                   ;;
164                 2)
165                   n=$(whiptail --inputbox "How many User You Want to Add?" 8 39 --title "Add
    Temporary Users" 3>&1 1>&2 2>&3)
166                   new_count=$(($count+$n))
167                   for(( i=$count; i<$new_count; i++ ))
168                   do
169                                                   m=$(whiptail --inputbox "USERNAME" 8 39 --
    title "USER VEFIFICATION" 3>&1 1>&2 2>&3)
170                                                   echo "$m" >> users.txt
171                     arr=(${arr[0]} "$m")
172                     echo ${arr[0]}
173                   done
```

```
174                     ;;
175                 esac
176             else
177                                     whiptail --title "Student Database System" --msgbox "Wrong
    Password" 8 78
178             fi
179             break;
180           else
181             elim=$(($count-1))
182             if [[ $i == $elim ]]
183             then
184               {
185                 for ((i = 0 ; i <= 100 ; i+=5)); do
186                   sleep 0.1
187                   echo $i
188                 done
189               } | whiptail --gauge "Please wait while we are sleeping..." 6 50 0
190               whiptail --title "Student Database System" --msgbox "Access Denied" 8 78
191             fi
192           fi
193         done
194         ;;
195       2)
196         control=0;;
197       esac
198 done
```

3.1 Results and Discussions

3.1.1 Results

- We can organize in a way that facilitates local or remote information retrieval.
- User privacy kept well managed.
- Able to process many continual queries over a long period of time.
- Low cost and reliable database.
- Facilitates temporary users to access the database.

3.1.2 Discussions

- The delete operation does not get executed properly.
- Limited amount of student data can be stored.
- The database can store student result & location details.
- This database can be applied to any organizational site in terms of its privacy maintain protocol.

Chapter 4

Conclusion

4.1 Introduction

Using Ubuntu and Shell Scripting, we need to create a database system that can store, access, and link system data according to search operations (queries). In this system, we have tried to use the basic features that are required to maintain a student database system. Here we create 4 sections that have some valuable features. We maintain privacy and user access here. We used filters and wildcards to search for information throughout the system. Database manipulation is possible as well, the information in that database.

4.2 Practical Implications

- The database can store information about students' results as well as their location. Thus, it can be used as an organizational database.
- We can create a user-friendly database system with a webpage or website attached to the database.
- In terms of privacy management, this database can be applied to any organizational site.

4.3 Scope of Future work

- We can make a user-friendly database system with a webpage or website attached to the database.
- A lot of fields must be included to make the system more reliant.
- The deletion process should be performed in a way that is actually erasing information from a particular database.

References

1) https://projectsgeek.com/2011/04/student-database-using-shell-programming.html
2) http://www.dailyfreecode.com/code/shell-script-perform-database-1657.