***“Employee Database System”***

A MINI- PROJECT REPORT ON

Submitted in partial fulfilment of the requirements

For the degree of

Bachelor of Engineering

In

Information Technology

by

**Burhanuddin Naguthanawala 19IT5001**

**Ashish Patil 19IT5008**

**Rithvik Chavhan 19IT5012**

Supervisor

**Nilima Dongre**



Department of Information Technology

Dr. D. Y. Patil Group’s

**Ramrao Adik Institute of Technology**

Dr. D. Y. Patil Vidyanagar, Sector 7, Nerul, Navi Mumbai 400706.

(Affiliated to University of Mumbai)

( 2020)

# rait logo

Ramrao Adik Institute of Technology

(Affiliated to the University of Mumbai)

Dr. D. Y. Patil Vidyanagar,Sector 7, Nerul, Navi Mumbai 400706.

CERTIFICATE

This is to certify that Mini Project entitled

“ Employee Database System ”

is a bonafide work done by

Burhanuddin Naguthanwala, Ashish Pati and Rithvik Chavhan

and is submitted in the partial fulfilment of the requirement for the

degree of

Bachelor of Engineering

in

Information Technology

to the

University of Mumbai

Supervisor

Prof.Nilima M.Dongre

Project Guide Head of the department

Nilima Dongre Dr. Ashish Jadhav

Certificate of Approval by Examiners

This Mini Project report entitled “ Employee Database System ” is a bonafide work done by Burhanuddin Naguthanwala, Ashish Patil and Rithvik Chavhan under the supervision of Prof.Nilima Dongre approved for the award of Bachelor Degree in Information Technology, University of Mumbai.

Examiners :

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Supervisors :

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Principal :

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Date :

Place :

# **DECLARATION**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

**Name and Roll No. of Students Signature**

1. Burhanuddin Naguthanawala (19IT5001)

2. Ashish Patil (19IT5008)

3. Rithvik Chavhan (19IT5012)

Date:

Place:

# ACKNOWLEDGEMENT

The project “Employee Database System” is creative work of many minds. A proper synchronization between individual is must for any project to be completed successfully. One cannot imagine the power of the force that guides us all and neither can we succeed without acknowledging it.

We take this opportunity to express my profound gratitude and deep regards to our Guide **Nilima Dongre**, Department of the Information Technology Engineering for her or her exemplary guidance, monitoring and constant encouragement throughout the completion of this mini project.

We would like to express our gratitude to **Dr. Ashish Jadhav,** Head of the department, Information Technology Engineering for encouraging and inspiring us to carry out the project in the department lab. We take this privilege to express my sincere thanksare thankful to **Dr. Mukesh D. Patil, Principal RAIT,** for his constant support and motivation.

We also would like to thank all the staff members Department of the Information Technology Engineering for providing us with the required facilities and support towards the completion of the project.

Last but not the least we are thankful to our parents and friends for their constant Inspiration, encouragement and well wishes by which we have made a challenging project.

STUDENT-NAME (ROLLNO) Signature

Burhanuddin Naguthanawala (19IT5001)

Ashish Patil (19IT5008)

Rithvik Chavhan (19IT5012)

# **PREFACE**

We take great opportunity to present this Mini Project report on “**Employee Database System”** and put before readers some useful information regarding our project.

We have made sincere attempts and taken every care to present this matter in precise and compact form, the language being as simple as possible. We are sure that the information contained in this volume certainly prove useful for better insight in the scope and dimension of this project in it true perspective.

The task of the completion of the project though being difficult was made quite simple, interesting and successful due to deep involvement and complete dedication of our group members.

**TABLE OF CONTENTS**

**Declaration …………….…….………………………………………… I**

**Acknowledgement ……...……………………………………………… II**

**Preface …………….…………………………………………………… III**

**Table of Contents ……………………………………………………… IV**

**Table of figures ...…………………………………………………….… V**

**Abstract…….....…………………………………………………..…….. VI**

**TABLE OF CONTENTS**

**Sr. No. Topic Page No.**

1. **INTRODUCTION………………………………………………………………..8**
   1. **INTRODUCTION TO SCRIPTING LANGUAGES………………………9**
   2. **WHY PARTICULAR SCRIPTING LANGUAGE………………………...10**
   3. **PROBLEM STATEMENT………………………………………………….11**
   4. **OBJECTIVES………………………………………………………………..12**
2. **PROPOSED SYSTEM…………………………………………………………..13**
   1. **HARDWARE AND SOFTWARE REQUIREMENTS……………………14**
3. **IMPLEMENTATION……………………………………………………………15**
   1. **CODE………………………………………………………………………….16**
4. **RESULT…………………………………………………………………………...23**
   * 1. **OUTPUT SNAPSHOTS…………………………………………………..25**
5. **CONCLUSION AND FUTURE SCOPE………………………………………..26**
   1. **CONCLUSION……………………………………………………………….26**
   2. **FUTURE SCOPE…………………………………………………………….26**
   3. **BENEFITS TO SOCIETY…………………………………………………. 26**
6. **REFERENCES……………………………………………………………………27**

**1. Introduction**

Employee database system is a shell scripting code that helps the user to maintain the employee database. It is a user interactive program in which you can add, delete and modify Employees.

It is a small representation of how employee database works in real life.

A data file is created as soon as we add records in the table, and the data file gets updated as soon as the user adds, delete or even modify the entry.

**1.1 Introduction to Scripting languages**

Usually shells are interactive that mean, they accept command as input from users and execute them. However some time we want to execute a bunch of commands routinely, so we have type in all commands each time in terminal.  
As shell can also take commands as input from file we can write these commands in a file and can execute them in shell to avoid this repetitive work. These files are called **Shell Scripts** or **Shell Programs**. Shell scripts are similar to the [**batch file**](https://en.wikipedia.org/wiki/Batch_file) in MS-DOS. Each shell script is saved with **.sh** file extension eg. **myscript.sh**

A shell script have syntax just like any other programming language. If you have any prior experience with any programming language like Python, C/C++ etc. it would be very easy to get started with it.  
A shell script comprises following elements –

* + Shell Keywords – if, else, break etc.
  + Shell commands – cd, ls, echo, pwd, touch etc.
  + Functions
  + Control flow – if..then..else, case and shell loops etc.

**1.2 Why particular scripting language**

We used shell scripting as it’s very easy to understand and user friendly.

There are many reasons to write shell scripts –

* + To avoid repetitive work and automation
  + System admins use shell scripting for routine backups
  + System monitoring
  + Adding new functionality to the shell etc.

**Advantages of shell scripts**

* + The command and syntax are exactly the same as those directly entered in command line, so programmer do not need to switch to entirely different syntax
  + Writing shell scripts are much quicker
  + Quick start
  + Interactive debugging etc.

**1.3 Problem Statement**

It is sometimes difficult to maintain employee database even on small scale business. This is why we have presented a user friendly and interactive database to lower the load of maintaining the database. It is easy to access and use.

**1.4 Objectives**

* Easy to implement.
* User friendly.
* Optimal accessibility and cost in terms of memory and speed.
* Less disk space required to run the program.

**2. Proposed System**

The objective is to create a database of employees. The user has full access to add, delete, modify or view the total number of employees he has using this program.

Shell scripting was used to make the program, the data file is created as soon as the user adds, deletes or modifies the table or the database.

The data file displays every single information of the employees and it gets updated time to time as the user makes changes.

**2.1 Hardware and Software requirements**

**Software:** Ubuntu, Linux and Terminal.

**Language:** Shell Scripting

1. **Terminal:** If you are using any major operating system you are indirectly interacting to shell. If you are running Ubuntu, Linux Mint or anyother Liux distribution, you are interacting to shell everytime you use terminal.
2. **Shell scripting:** Usually shells are interactive that mean, they accept command as input from users and execute them. However some time we want to execute a bunch of commands routinely, so we have type in all commands each time in terminal.

**3. Implementation**

**3.1 Code:**

clear

getTotalRecords()

{

total=`cat $dbfile | wc -l`

echo "Total records :"$total

}

viewAllRecords()

{

total=`cat $dbfile | wc -l`

echo "Total records :"$total

if [ $total -gt 0 ]

then

i=1

while [ $i -le $total ]

do

echo "===================================="

echo "Employee - "$i

echo "============="

record=`cat $dbfile | head -$i | tail -1`

emp\_no=`echo $record | cut -d "|" -f1`

emp\_name=`echo $record | cut -d "|" -f2`

emp\_add=`echo $record | cut -d "|" -f3`

emp\_age=`echo $record | cut -d "|" -f4`

emp\_gen=`echo $record | cut -d "|" -f5`

emp\_desg=`echo $record | cut -d "|" -f6`

emp\_sal=`echo $record | cut -d "|" -f7`

echo "Employee No. :" $emp\_no

echo "Employee Name :" $emp\_name

echo "Employee Address :" $emp\_add

echo "Employee Age :" $emp\_age

echo "Employee Gender :" $emp\_gen

echo "Employee Designation :" $emp\_desg

echo "Employee salary :" $emp\_sal

i=`expr $i + 1`

echo "===================================="

done

fi

}

addRecord()

{

echo "===================================="

echo " ENTER EMPLOYEE INFORMATION"

echo "===================================="

echo -n "Enter employee no. :"

read no

echo -n "Enter employee name :"

read name

echo -n "Enter employee address :"

read add

echo -n "Enter employee age :"

read age

echo -n "Enter employee gender :"

read gen

echo -n "Enter employee designation :"

read desc

echo -n "Enter employee salary :"

read sal

record="$no|$name|$add|$age|$gen|$desc|$sal"

echo $record >> $dbfile

echo "Employee information added successfully."

}

getEmployeeInformation()

{

echo "===================================="

echo " FIND EMPLOYEE INFORMATION"

echo "===================================="

echo -n "Enter employee no. :"

read no

total=`cat $dbfile | grep -cwi $no`

if [ $total -gt 0 ]

then

record=`cat $dbfile | grep -wi $no`

emp\_no=`echo $record | cut -d "|" -f1`

emp\_name=`echo $record | cut -d "|" -f2`

emp\_add=`echo $record | cut -d "|" -f3`

emp\_age=`echo $record | cut -d "|" -f4`

emp\_gen=`echo $record | cut -d "|" -f5`

emp\_desg=`echo $record | cut -d "|" -f6`

emp\_sal=`echo $record | cut -d "|" -f7`

echo "Employee No. :" $emp\_no

echo "Employee Name :" $emp\_name

echo "Employee Address :" $emp\_add

echo "Employee Age :" $emp\_age

echo "Employee Gender :" $emp\_gen

echo "Employee Designation :" $emp\_desc

echo "Employee salary :" $emp\_sal

elif [ $total -eq 0 ]

then

echo "No record found for this id"

fi

}

deleteEmployee()

{

echo "===================================="

echo " DELETE EMPLOYEE INFORMATION"

echo "===================================="

echo -n "Enter employee no. :"

read no

total=`cat $dbfile | grep -cwi $no`

if [ $total -gt 0 ]

then

totalrecords=`cat $dbfile | wc -l`

i=1

while [ $i -le $totalrecords ]

do

record=`cat $dbfile | head -$i | tail -1`

findrecord=`echo $record | grep -wci $no`

if [ $findrecord -eq 0 ]

then

echo "$record" >> "tempdata"

fi

i=`expr $i + 1`

done

mv "tempdata" "data"

echo "Employee with id $no deleted."

elif [ $total -eq 0 ]

then

echo "No record found for this id"

fi

}

modifyRecord()

{

echo "===================================="

echo " MODIFY EMPLOYEE INFORMATION"

echo "===================================="

echo -n "Enter employee no. :"

read no

total=`cat $dbfile | grep -cwi $no`

if [ $total -gt 0 ]

then

record=`cat $dbfile | grep -wi $no`

echo "================================================="

echo " ENTER EMPLOYEE INFORMATION FOR ID : $no"

echo "================================================="

echo -n "Enter employee name :"

read name

echo -n "Enter employee address :"

read add

echo -n "Enter employee age :"

read age

echo -n "Enter employee gender :"

read gen

echo -n "Enter employee designation :"

read desc

echo -n "Enter employee salary :"

read sal

updatedrecord="$no|$name|$add|$age|$gen|$desc|$sal"

#updates=$(`cat $dbfile` | sed s/$record/$updatedrecord/g)

totalrecords=`cat $dbfile | wc -l`

i=1

while [ $i -le $totalrecords ]

do

record=`cat $dbfile | head -$i | tail -1`

findrecord=`echo $record | grep -wci $no`

if [ $findrecord -eq 0 ]

then

echo "$record" >> "tempdata"

else

echo "$updatedrecord" >> "tempdata"

fi

i=`expr $i + 1`

done

mv "tempdata" "data"

echo "Record updated successfully"

else

echo "Record with this id not found"

fi

}

dbfile="data"

echo "1 - All records display"

echo "2 - Add record"

echo "3 - View record"

echo "4 - Delete record"

echo "5 - Modify record"

echo "6 - Count total records"

echo "E - Exit"

echo "=============================="

echo -n "Enter your choice :"

read choice

case $choice in

"1")

viewAllRecords

;;

"2")

addRecord

;;

"3")

getEmployeeInformation

;;

"4")

deleteEmployee

;;

"5")

modifyRecord

;;

"6")

getTotalRecords

;;

[eE])

exit

;;

\*)

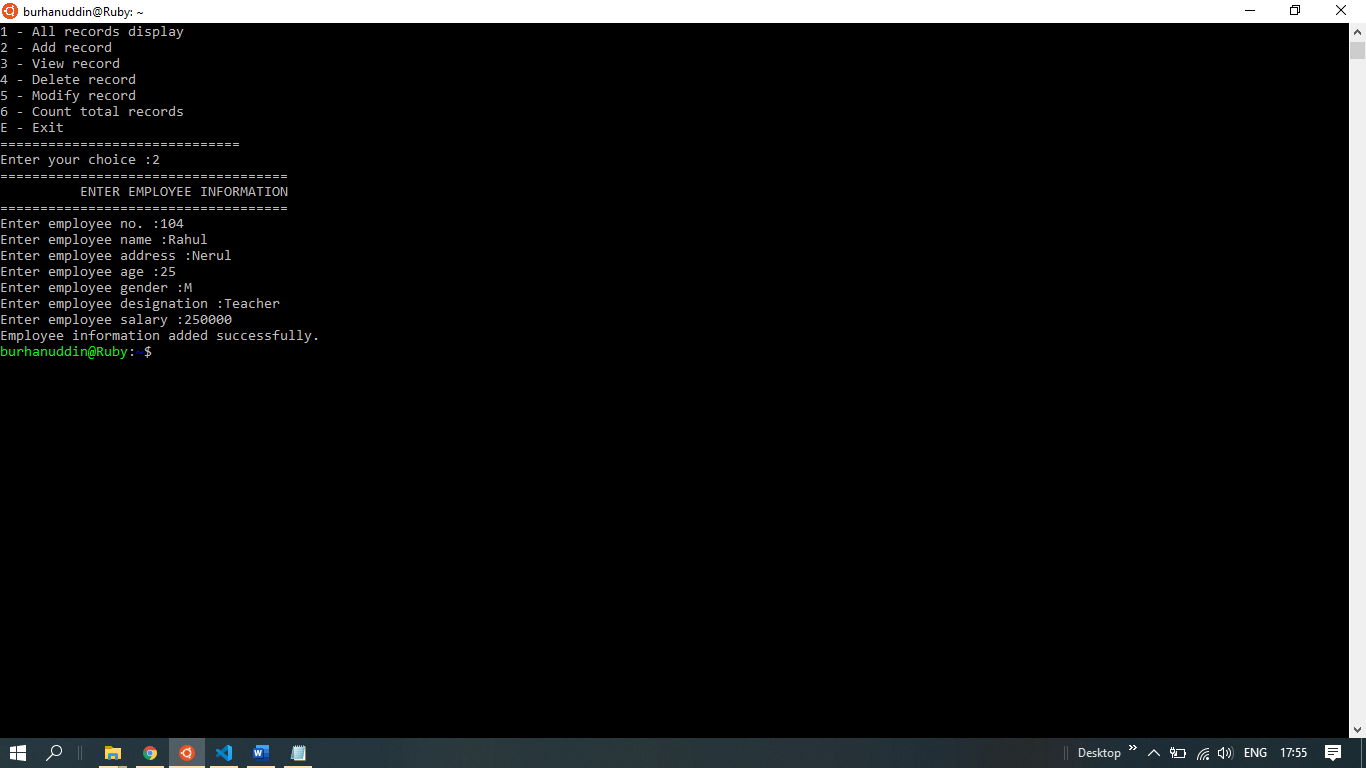
echo "Invalid choice"

;;

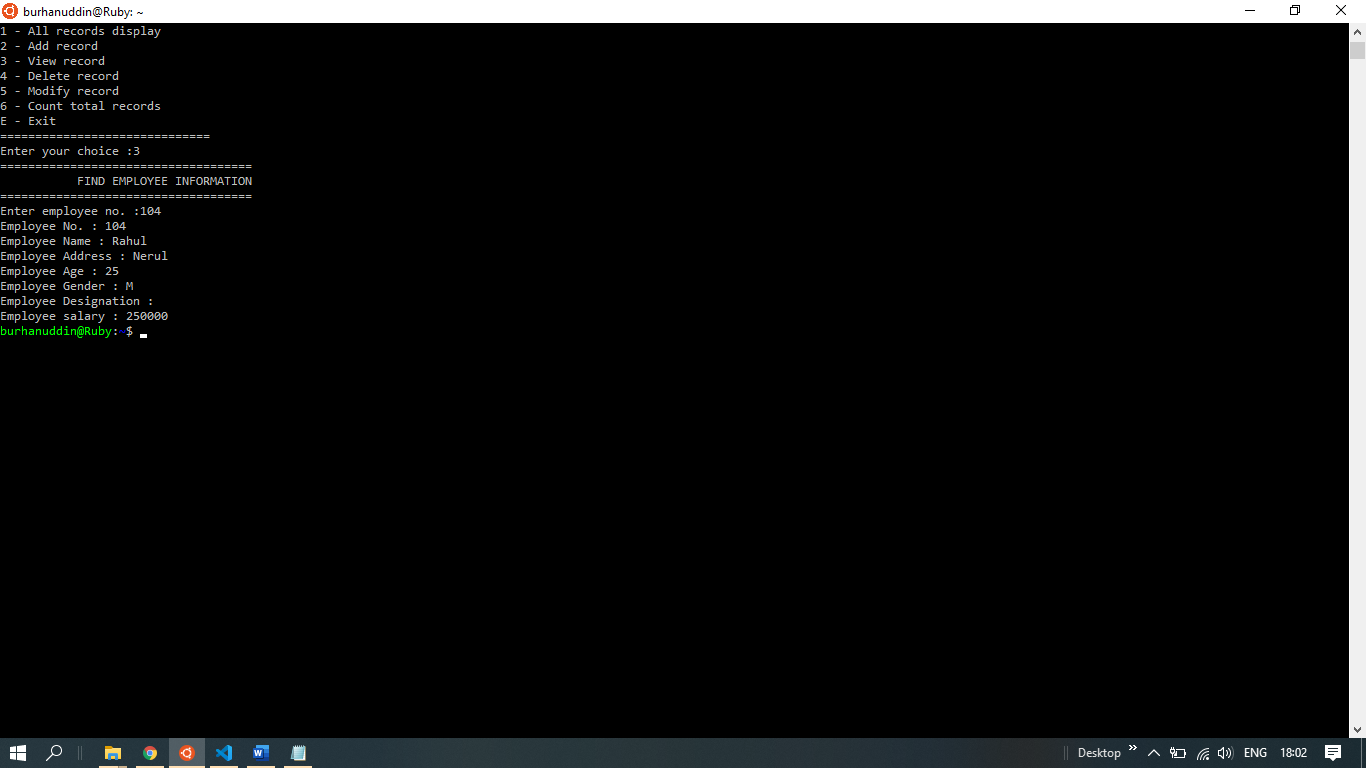
esac

**4. Results**

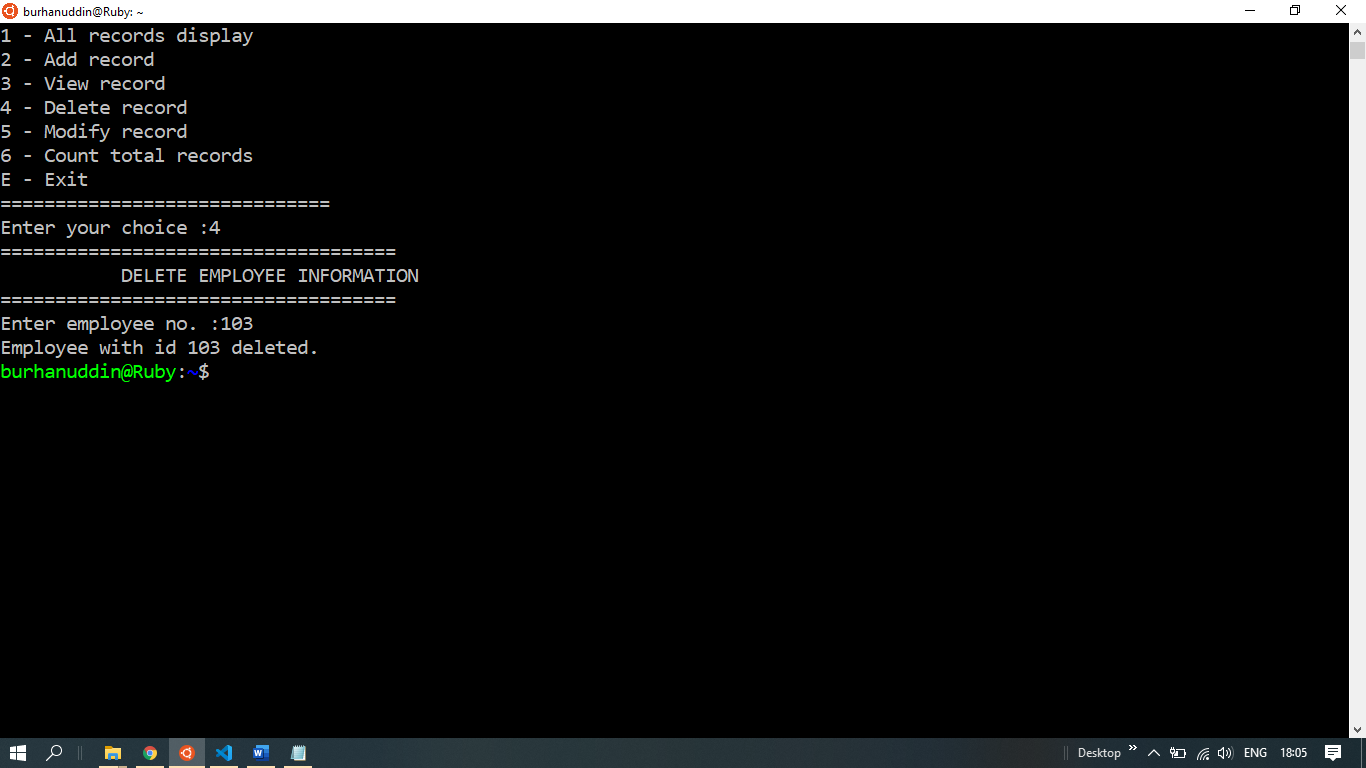
**4.1 Snapshots:**



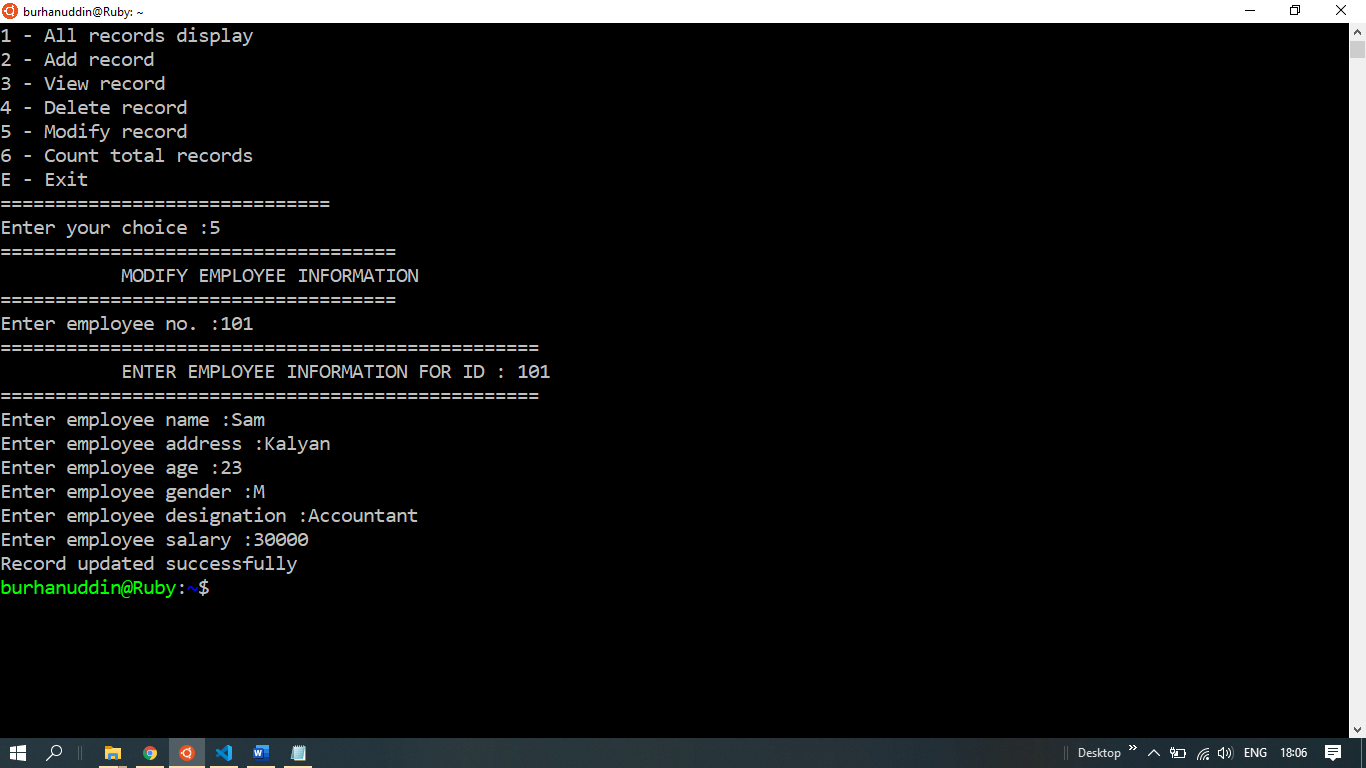
Employee added successfully.



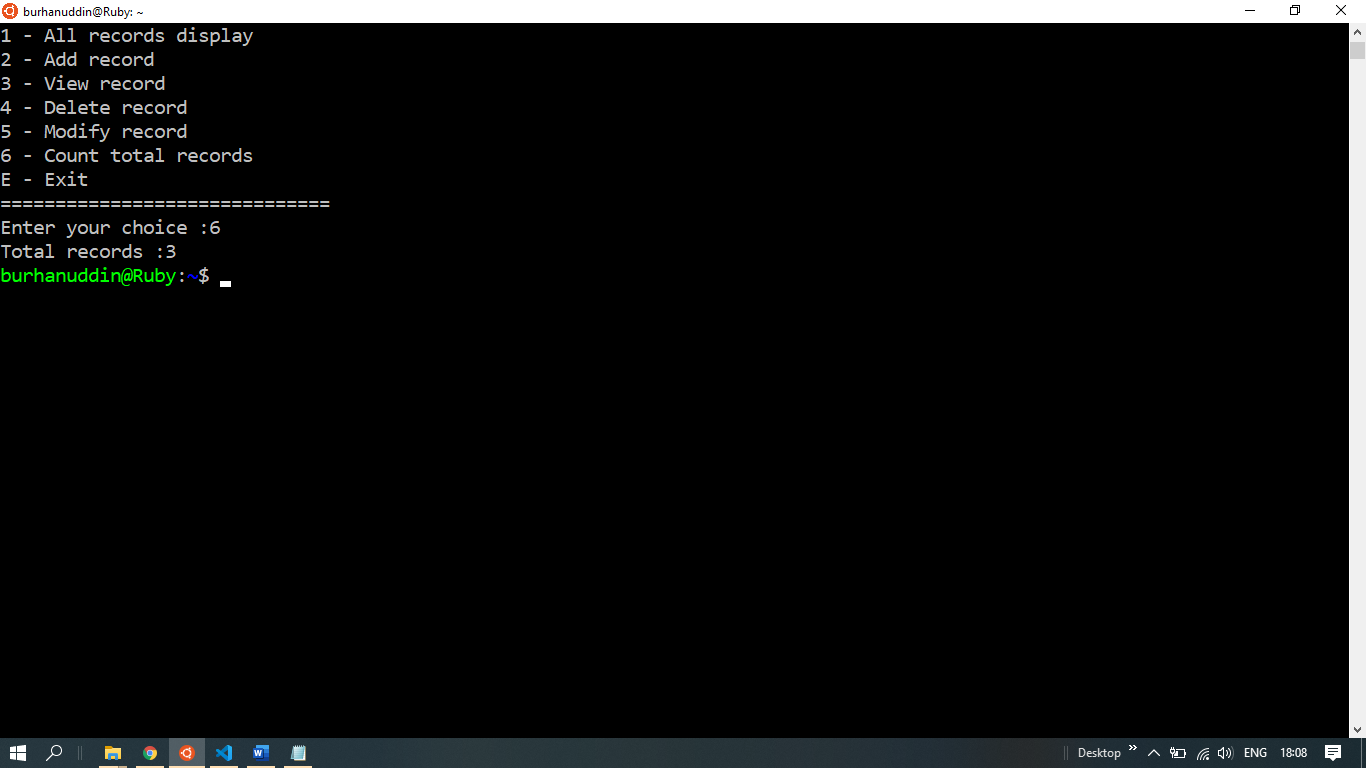
To view specific employee details.



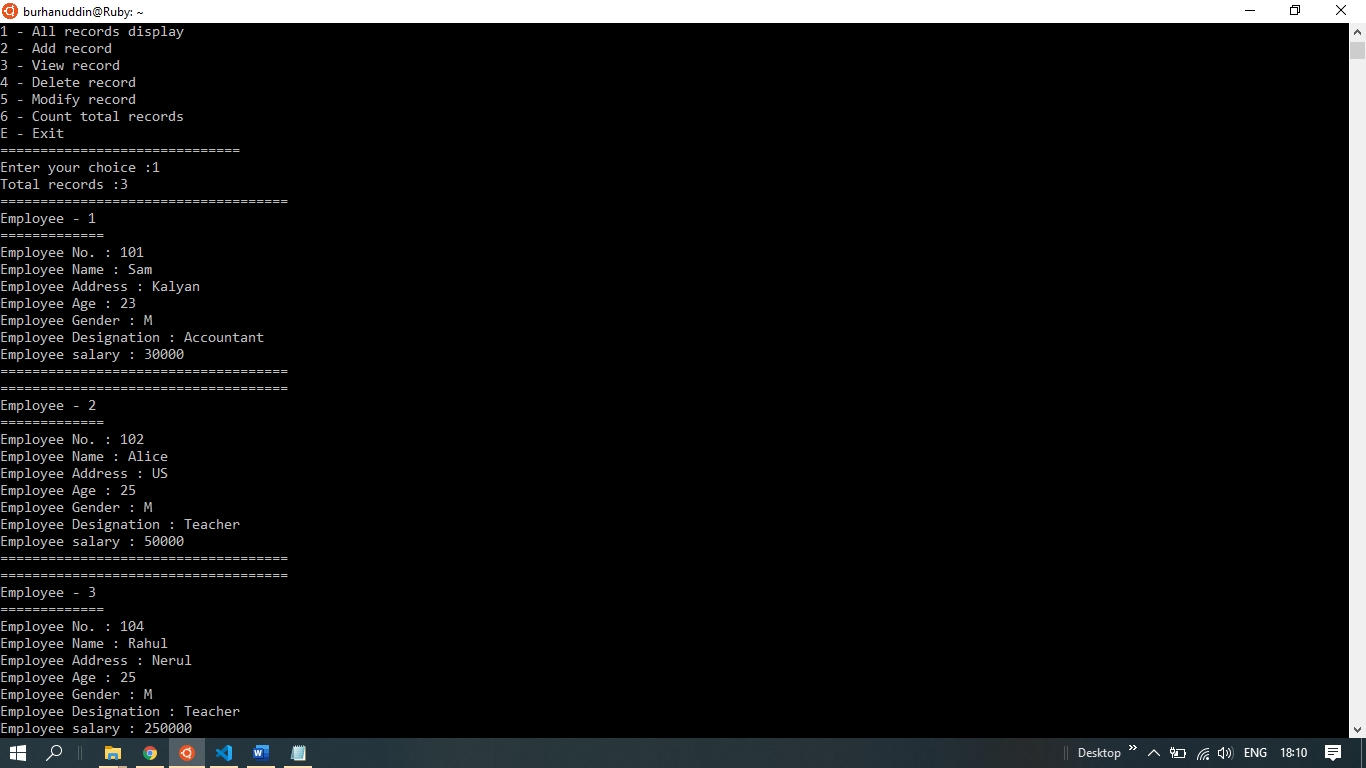
Employee deleted.



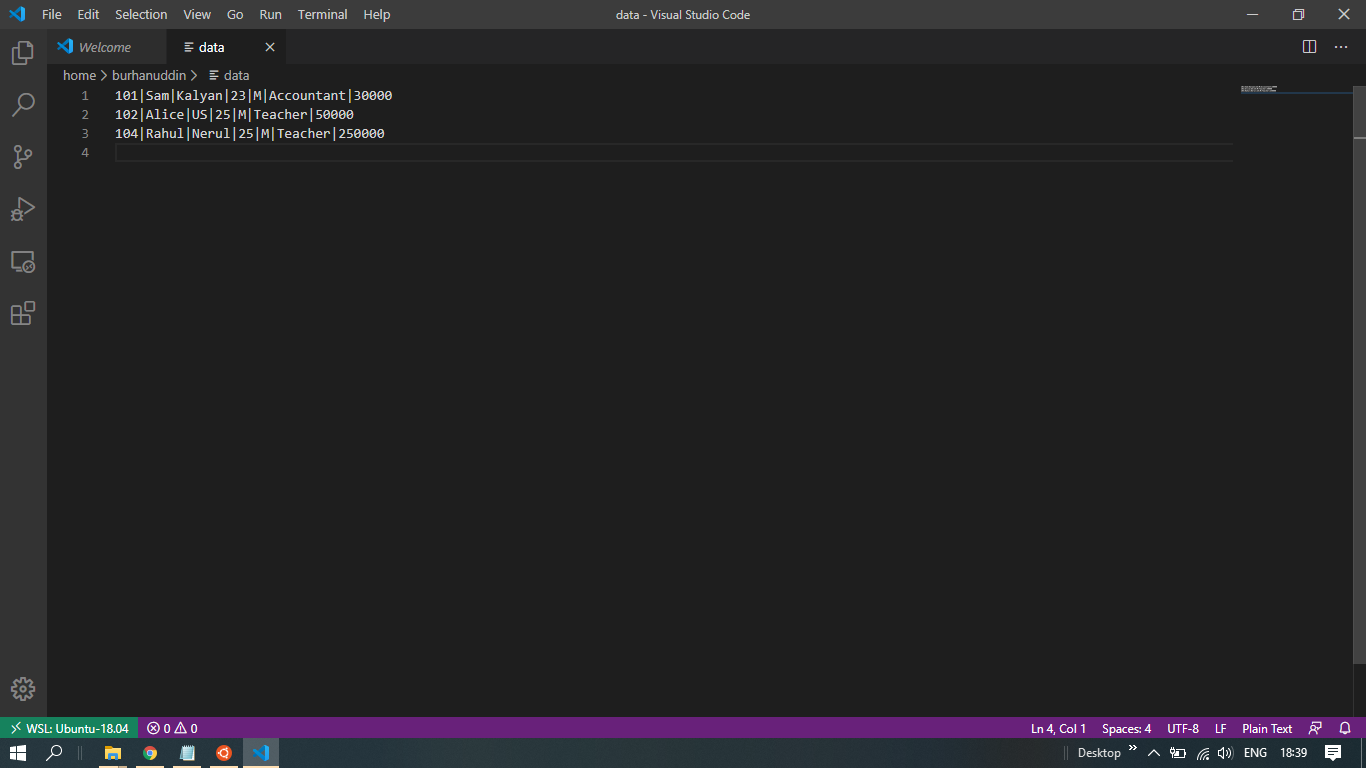
Record modified.



Total Records count displayed.



All Records with data displayed.



It’s a data file where data gets stored.

**5. Conclusion and Future scope**

**5.1 Conclusion**

Employee data base system helps one to organize the data of their employees in proper manner. It is easy to access as its user friendly.

**5.2 Future Scope**

The future for databases is consolidation around big data with a rationalization down to 10 core technologies that make data easy to access and leads to more data-driven analytics and services. Data will be easier to access and use. More processing will be done on the edge to facilitate real-time computations and decision making.

**5.3 Benefits**

It is easy to implement and use therefore making it simpler to manage the data.

**6. References**

* Geeksforgeeks.org
* Galvin
* Youtube.com