Visvesvaraya Technological University

Belagavi-590 018, Karnataka

A Mini Project Report on

"Implementation of Indexing on Hotel Reviews dataset"

Mini Project Report submitted in partial fulfilment of the requirement for the File Structures Lab [17ISL68]

Bachelor of Engineering

In

Information Science and Engineering

Submitted by

Vinod T [1JT17IS050]

Under the Guidance of

Mr.Vadiraja A

Asst. Professor

Dept. Of ISE

Department of Information Science and Engineering

Jyothy Institute of Technology

Tataguni, Bengaluru-560082

2020-21

Jyothy Institute of Technology

Tataguni, Bengaluru-560082

Department of Information Science and Engineering

CERTIFICATE

Certified that the mini project work entitled **"Implementation of Indexing on HOTEL REVIEWS DATASET"** carried out **by Vinod T [1JT17IS050]** bonfide student of Jyothy Institute of Technology, in partial fulfilment for the award of **Bachelor of Engineering** in **Information Science and Engineering** department of the **Visvesvaraya Technological University, Belagavi** during the year **2020-2021**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

Mr .VadirajaA
Guide, Asst.Professor
Dept.Of ISE

Dr. HarshvardhanTiwari
Associate. Professor and HOD
Dept.Of ISE

External Viva Examiner

1.

2.

Signature with Date:

ACKNOWLEDGEMENT

Firstly, we are very grateful to this esteemed institution **"Jyothy Institute of Technology**" for providing us an opportunity to complete our project.

We express our sincere thanks to our Principal **Dr. Gopalakrishna K for** providing us with adequate facilities to undertake this project.

We would like to thank **Dr. Harshvardhan Tiwari, Associate Prof. and Head** of Information Science and Engineering Department for providing for his valuable support.

We would like to thank our guides **Mr.Vadiraja A** , **Asst. Prof.** for their keen interest and guidance in preparing this work.

Finally, we would thank all our friends who have helped us directly or indirectly in this project.

Vinod T [1JT17IS050]

ABSTRACT

Indexing is the process of associating a key with the location of a corresponding data record. An external sort typically uses the concept of a key sort, in which an index file is created whose records consist of key pairs. Here, each key is associated with a pointer to a complete record in the main database file. The index file could be sorted or organised using a tree structure, thereby imposing a logical order on the records without physically rearranging them. Each record of a database normally has a unique identifier, called the primary key. A particular key value might be duplicated in multiple records, is called a secondary key. The secondary key index will associate a secondary key value with the primary key of each record having that secondary key value .The full database might be searched directly for the record with that primary key, or there might be a primary key index that relates each primary key value with a pointer to the actual record on the disk. In this case, the primary index provides the location of the actual record on disk, while the secondary disk indices refer to the primary index. Indexing is an important technique for organising large databases.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

1.1 Introduction to File Structure

A disk's relatively slow access time and the enormous, non-volatile capacity is the driving force behind file structure design. FS should give access to all the capacity without making the application spend a lot of time waiting for the disk. FS is a combination of representation for data in files and of operations for accessing the data.

- It allows applications to read, write and modify data
- Also finding the data
- Or reading the data in a particular order

Efficiency of FS design for a particular application is decided on,

1. Details of the representation of the data
2. Implementation of the operations.

A large variety in the types of data and in the needs of application makes FS design important. What is best for one situation may be terrible for other.

A file system is a process that manages how and where data on a storage disk, typically a hard disk drive (HDD), is stored, accessed and managed. It is a logical disk component that manages a disk's internal operations as it relates to a computer and is abstract to a human user. Regardless of type and usage, a disk contains a file system and information about where disk data is stored and how it may be accessed by a user or application. A file system typically manages operations, such as storage management, file naming, directories/folders, metadata, access rules and privilege.

1.2 Introduction to Python

- Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales.
- Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object oriented, imperative, functional and procedural, and has a large and comprehensive standard library.
- Python interpreters are available for many operating systems. C Python, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. C Python is managed by the non-profit Python Software Foundation.

1.3 Introduction to Indexing

- Indexing is a data structure technique which allows you to quickly retrieve records from a database file.
- An Index is a small table having only two columns. The first column comprises a copy of the primary or candidate key of a table. Its second column contains a set of pointers for holding the address of the disk block where that specific key value stored.
- An index takes a search key as input and efficiently returns a collection of matching records.

Type of Indexes:

- Primary Indexing - Primary Index is an ordered file which is fixed length size with two fields. The first field is the same a primary key and second, filed is pointed to that specific data block. In the primary Index, there is always one to one relationship between the entries in the index table.
- Secondary Indexing -The secondary Index can be generated by a field which has a unique value for each record, and it should be a candidate key. It is also known as a non-clustering index.

1.4 Scope and importance of work

Data analysis is a process used to inspect, clean, transform and remodel data with a view to reach to a certain conclusion for a given situation. Data analysis helps in structuring the findings from different sources of data.

- Data analysis is very helpful in breaking a macro problem into micro parts.
- Data analysis acts like a filter when it comes to acquiring meaningful insights out of huge data set.
- Data analysis helps in keeping human bias away from the research conclusion with the help of proper statistical treatment.

When discussing data analysis, it is important to mention that a methodology to analyse data need to be picked. If a specific methodology is not selected data can neither be collected nor analysed. The methodology should be present in the dissertation as it enables the reader to understand which methods have been used during the research and what type of data has been collected and analysed throughout the process. The dissertation also presents a critical analysis of various methods and techniques that were considered but ultimately not used for the data analysis. An effective research methodology leads to better data collection and analysis and leads the researcher to arrive at valid and logical conclusions in the research. Without a specific methodology, observations and findings in a research cannot be made which means methodology is an essential part of a research or dissertation.

CHAPTER 2

IMPLEMENTATION

2.1 Basic operations on Indexing

In this section, we present the details of the operations of indexing:

- Addition.
- Searching.
- Deleting.
- Build Index
- Modification

2.2 Procedure

Step 1: Accessing a particular dataset.
Step 2: Create two new index files one for primary indexing and the other for secondary indexing.
Step 3: The first column is the search key that contains a copy of primary key or secondary key of the table.
Step 4: The second column is the pointer which contains a set of pointers holding the address the disk block where that particular key value is found.
Step 5: Record addition - This consists of appending the data file and inserting a new record. The rearrangement of the index consists of sliding down the records with keys larger than the inserted key and then placing new record in the opened space.
Step 6: Record deletion -This should use the techniques for reclaiming space in files when deleting from the data file. We must delete the corresponding entry from the index. Shift all records with keys larger than key of the deleted record to the previous position in memory or make the index entry as deleted.
Step 7: In our record file, we built an index for “id” which is primary key and the “sid” as secondary key.
Step 8: Record deletion in secondary key: Deleting a record implies removing all the references to the record in primary index and in all secondary indexes. When accessing the file through secondary key, the primary indexed file will be checked and a deleted record can be identified.
Step 9: It allows binary search to obtain a keyed access to a record in variable length record file.
Step 10: Time taken for dataset has been calculated for each functionality.

CHAPTER 3

SEARCH ALGORITHM AND TIME COMPLEXITY.

3.1 Algorithm to Search

The technique we have used to search in our program is BINARY SEARCH.

Using binary search to search a sorted array by repeatedly dividing the search interval in half. Begin with an interval covering the whole array. If the value of the search key is less than the item in the middle of the interval, narrow the interval to the lower half. Otherwise narrow it to the upper half. Repeatedly check until the value is found or the interval is empty.

The idea of binary search is to use the information that the list is sorted and reduce the time complexity to O(Log n).

```
Procedure binary_search
   A ← sorted array
   n ← size of array
   x ← value to be searched
   Set lowerBound = 1
   Set upperBound = n
while x not found
      if upperBound < lowerBound
         EXIT: x does not exists.
            set midPoint = lowerBound + ( upperBound - lowerBound ) / 2
     if A[midPoint] < x
         set lowerBound = midPoint + 1
     if A[midPoint] > x
         set upperBound = midPoint - 1
     if A[midPoint] = x
         EXIT: x found at location midPoint
   end while
end procedure
```

3.2 Calculating time Complexity:

The time complexity of an algorithm is the total amount of time required by an algorithm to complete its execution. In simple words, every piece of code we write, takes time to execute. The time taken by any piece of code to rum is known as the time complexity of that code. The lesser the time complexity, the faster the execution.

If you've programmed a bit before, you're probably wondering how this can be of any sue for you because your program was running fin even when you didn't know all of this time complexity stuff, right? I agree with you 100% but there's a catch.
The time for program to run does not depend solely on efficiency of code, It's also dependent on the processing power of a PC . Since time complexity is used to measure the time for algorithm, the type of algorithm you'd use in small program wouldn't really matter because there's hardly any work being carried out by the processor although when we write code in professional life, the code isn't of 200 or 300 lines.

It's usually longer than a thesis written by a professor and in cases like that , a lot of processor power is being used. if your code is efficient in terms of data structures , you might find yourself in a rather sticky situation.

Time Complexity is often measured in terms of:

BIG Oh(O) : worst case running time

In analysis algorithm, Big Oh is often used to describe the worst -case of an algorithm by taking the highest order of a polynomial function and ignoring all the constants value since they aren't too influential for sufficiently large input . So, if an algorithm has running time like f(n)=3n + 100,we can simply state that algorithm has the complexity O(n) which means it always execute at most procedures(ignoring the constant "100" in between and also the constant '3' being multiplied by n) . Thus, we can guarantee that algorithm would not be bad than the worst -case.

Since we know that the highest order of f(n) is 2, we can conclude that f(n) can not have a time complexity greater than that of n^2. Which means it's the worst case running time.

BIG Omega(Ω) : Best case running time

It is often used to describe the best-case running time of an algorithm by choosing the lowest order of the polynomial functions and ignoring all the constants.

We know that the lowest order of the polynomial function f(n)(i.e. 1) is less than n^2 ,thus we can conclude that f(n) has a big Ω(n).

CHAPTER 4

RESULTS AND SNAPSHOTS.

4.1 Record Addition

```
------WELCOME TO HOTEL REVIEW DATASET------
PRESS 1 TO ADD A NEW HOTEL REVIEW RECORD
PRESS 2 TO DELETE A HOTEL REVIEW RECORD BASED ON PRIMARY KEY
PRESS 3 TO DELETE A HOTEL REVIEW RECORD BASED ON SECONDARY KEY
PRESS 4 TO SEARCH FOR A HOTEL REVIEW BASED ON PRIMARY KEY
PRESS 5 TO SEARCH FOR A HOTEL REVIEW BASED ON SECONDARY KEY
PRESS 6 TO MODIFY A HOTEL REVIEW
PRESS 7 TO EXIT

PLEASE ENTER YOUR OPTION: 1

enter id:
40000

enter hotel name:
oyo

enter hotel address:
Pune india

enter review date:
06-06-2020

enter negative review:
was not bad

enter positive review:
i liked the rooms and staff were polite

enter rating:
6.7
The time taken to add a new record in seconds
65
```

(a)

```
enter id:
696969

enter hotel name:
lakeview hotel

enter hotel address:
mayo city

enter review date:
24-2-2016

enter negative review:
ambience sucks

enter positive review:
food was amazing

enter rating:
3
The time taken to add a new record in seconds
77
PRESS 1 TO ADD A NEW HOTEL REVIEW RECORD
PRESS 2 TO DELETE A HOTEL REVIEW RECORD BASED ON PRIMARY KEY
PRESS 3 TO DELETE A HOTEL REVIEW RECORD BASED ON SECONDARY KEY
PRESS 4 TO SEARCH FOR A HOTEL REVIEW BASED ON PRIMARY KEY
PRESS 5 TO SEARCH FOR A HOTEL REVIEW BASED ON SECONDARY KEY
PRESS 6 TO MODIFY A HOTEL REVIEW
PRESS 7 TO EXIT
```

(b)

Fig 4.1 (a & b) Record Addition

In the above image the user has selected the option to add a new job record.The time taken to enter all the details here is 65ms.

4.2 Delete based on primary key

```
PRESS 1 TO ADD A NEW HOTEL REVIEW RECORD
PRESS 2 TO DELETE A HOTEL REVIEW RECORD BASED ON PRIMARY KEY
PRESS 3 TO DELETE A HOTEL REVIEW RECORD BASED ON SECONDARY KEY
PRESS 4 TO SEARCH FOR A HOTEL REVIEW BASED ON PRIMARY KEY
PRESS 5 TO SEARCH FOR A HOTEL REVIEW BASED ON SECONDARY KEY
PRESS 6 TO MODIFY A HOTEL REVIEW
[illegible]T

PLEASE ENTER YOUR OPTION: 2

Enter the id to delete:
40000
DELETED SUCCESSFULLY
The time taken to delete a record in seconds
5
```

Fig 4.2 Delete based on primary key

Here the user has selected the option 2 which is deleted a record based on the primary key.The time taken to delete a record is 5s.

4.3 Delete based on Secondary Key

```
PRESS 1 TO ADD A NEW HOTEL REVIEW RECORD
PRESS 2 TO DELETE A HOTEL REVIEW RECORD BASED ON PRIMARY KEY
PRESS 3 TO DELETE A HOTEL REVIEW RECORD BASED ON SECONDARY KEY
PRESS 4 TO SEARCH FOR A HOTEL REVIEW BASED ON PRIMARY KEY
PRESS 5 TO SEARCH FOR A HOTEL REVIEW BASED ON SECONDARY KEY
PRESS 6 TO MODIFY A HOTEL REVIEW
PRESS 7 TO EXIT

PLEASE ENTER YOUR OPTION: 3

Enter the Rating to delete:
3.3
DELETED SUCCESSFULLY
time taken to delete the file in ms
4
```

Fig 4.3 Delete based on primary key

The user has selected the option 3 which is deleting a record based on secondary key.The time taken to delete is 4ms.

4.4 Searching based on primary key

```
------WELCOME TO HOTEL REVIEW DATASET------
PRESS 1 TO ADD A NEW HOTEL REVIEW RECORD
PRESS 2 TO DELETE A HOTEL REVIEW RECORD BASED ON PRIMARY KEY
PRESS 3 TO DELETE A HOTEL REVIEW RECORD BASED ON SECONDARY KEY
PRESS 4 TO SEARCH FOR A HOTEL REVIEW BASED ON PRIMARY KEY
PRESS 5 TO SEARCH FOR A HOTEL REVIEW BASED ON SECONDARY KEY
PRESS 6 TO MODIFY A HOTEL REVIEW
PRESS 7 TO EXIT

PLEASE ENTER YOUR OPTION: 4

enter key 1
found
80
Review id: 1


Hotel Name: Hotel Arena


Hotel Address:  s Gravesandestraat 55 Oost 1092 AA Amsterdam Netherlands


Review Date: 08-03-2017


Negative review:  I am so angry that i made this post available via all possible sites i use when planing my trips so no one will
make the mistake of booking this place I made my booking via booking com We stayed for 6 nights in this hotel from 11 to 17 July Upon
arrival we were placed in a small room on the 2nd floor of the hotel It turned out that this was not the room we booked I had
specially reserved the 2 level duplex room so that we would have a big windows and high ceilings The room itself was ok if you don t
mind the broken window that can not be closed hello rain and a mini fridge that contained some sort of a bio weapon at least i
guessed so by the smell of it I intimately asked to change the room and after explaining 2 times that i booked a duplex btw it costs
the same as a simple double but got way more volume due to the high ceiling was offered a room but only the next day SO i had to
```

Fig 4.4 Sarch based on primary key

In this image user has selected option 4 which is searching based on primary key. The time taken to search is 320ms.

4.5 Searching based on Secondary Key

```
PRESS 4 TO SEARCH FOR A HOTEL REVIEW BASED ON PRIMARY KEY
PRESS 5 TO SEARCH FOR A HOTEL REVIEW BASED ON SECONDARY KEY
PRESS 6 TO MODIFY A HOTEL REVIEW
PRESS 7 TO EXIT

PLEASE ENTER YOUR OPTION: 5

enter key 2.5
found
Review id: 42442


Hotel Name: London Marriott Hotel Regents Park


Hotel Address: 128 King Henry s Road Camden London NW3 3ST United Kingdom


Review Date: 08-02-2016


Negative review:  everything


Positive Review:  nothing


Rating: 2.5



The time taken to search the record in seconds
271
```

Fig 4.5 Searching based on Secondary Key

Here user selects option 5 that is search based on secondary key and the time taken is 216ms.

4.6 Modification

```
Hotel Name: Grand Royale London Hyde Park

Hotel Address: 1 Inverness Terrace Westminster Borough London W2 3JP United Kingdom

Review Date: 07-11-2017

Negative review:  the shower sprayed water everywhere

Positive Review:  good breakfast selection and quality complimentary water and teas were great A wonderful last minute stay in a
hotel that felt part of the history of London yet was beautifully up to date in facilities

Rating: 9.4


Enter 1 to modify Hotel Name
Enter 2 to modify Hotel Address
Enter 3 to modify Review Date
Enter 4 to modify Rating
Enter 5 to go back to main menu

PLEASE ENTER YOUR OPTION: 1

Enter the new Hotel NameGrand royal
The time taken to modify the record in seconds
7900
```

(a)

```
Negative review:  The aircondition makes so much noise and its hard to sleep at night


Positive Review:  The room was big enough and the bed is good The breakfast food and service on
the hotel is good outside the hotel there is a big park which is very good for walk in the
morning and evening Many people are having picnics and do some bicycling


Rating: 7.9



Enter 1 to modify Hotel Name
Enter 2 to modify Hotel Address
Enter 3 to modify Review Date
Enter 4 to go back to main menu

PLEASE ENTER YOUR OPTION: 1

Enter the new Hotel Name bruxfin
The time taken to modify the record in seconds
739
Enter 1 to modify Hotel Name
Enter 2 to modify Hotel Address
Enter 3 to modify Review Date
Enter 4 to go back to main menu

PLEASE ENTER YOUR OPTION:
```

Fig 4.6.1 (a &b) Modification of Category

In the above image the user selects an option to modify a record and the sub option is to modify category. The time taken is 739ms.

```
Hotel Name: Grand royal


Hotel Address: 1 Inverness Terrace Westminster Borough London W2 3JP United Kingdom


Review Date: 07-11-2017


Negative review:  the shower sprayed water everywhere


Positive Review:  good breakfast selection and quality complimentary water and teas were great A wonderful last minute stay in a
hotel that felt part of the history of London yet was beautifully up to date in facilities


Rating: 9.4



Enter 1 to modify Hotel Name
Enter 2 to modify Hotel Address
Enter 3 to modify Review Date
Enter 4 to modify Rating
Enter 5 to go back to main menu

PLEASE ENTER YOUR OPTION: 2

Enter the Hotel Addressi traverse stree london
The time taken to modify the record in seconds
333
```

Fig 4.6.2 Modify Hotel's Address

In the above image the user selects an option to modify a record and the sub option is to modify city name. The time taken is 333s.

```
Review id: 8338


Hotel Name: Grand royal


Hotel Address: i traverse stree london


Review Date: 07-11-2017


Negative review:  the shower sprayed water everywhere


Positive Review:  good breakfast selection and quality complimentary water and teas were great A wonderful last minute stay in a
hotel that felt part of the history of London yet was beautifully up to date in facilities


Rating: 9.4



Enter 1 to modify Hotel Name
Enter 2 to modify Hotel Address
Enter 3 to modify Review Date
Enter 4 to go back to main menu

PLEASE ENTER YOUR OPTION: 3

Enter the new Review Date06-06-2020
The time taken to modify the record in seconds
393
```

Fig 4.6.3 Modify Review Date

In the above image the user selects an option to modify a record and the sub option is to modify city name. The time taken is 393ms

4.7 Primary Index

	A	B	C
1	PrimaryKe	Index Value	
2	1	80	
3	2	2083	
4	3	2795	
5	4	3185	
6	5	4495	
7	6	5358	
8	7	5633	
9	9	5957	
10	10	6220	
11	12	6623	
12	13	7144	
13	14	7387	
14	15	7818	
15	17	8138	
16	18	8499	
17	19	9212	

Fig 4.7 Primary Index

4.8 Secondary Index

	A	B	C
1	secondary	Index Value	
2	2.5	2840219	
3	2.5	7245035	
4	2.5	2701346	
5	2.5	2708129	
6	2.5	2713129	
7	2.5	2720778	
8	2.5	4664745	
9	2.5	2722015	
10	2.5	774692	
11	2.5	7347746	
12	2.5	7360052	
13	2.5	4538032	

Fig 4.8 Secondary Index

4.9 TIME ANALYSIS

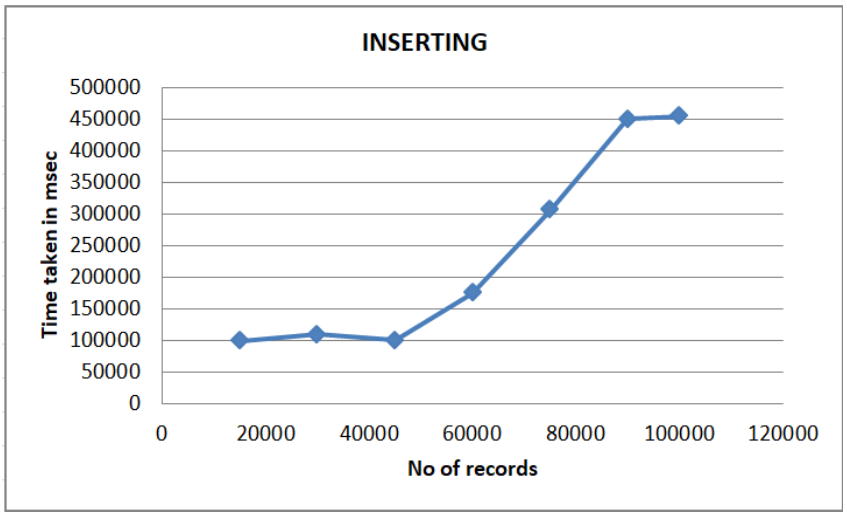

Fig 4.9.1 : The time taken to insert record into the file, as the number of record increases the time also increases.

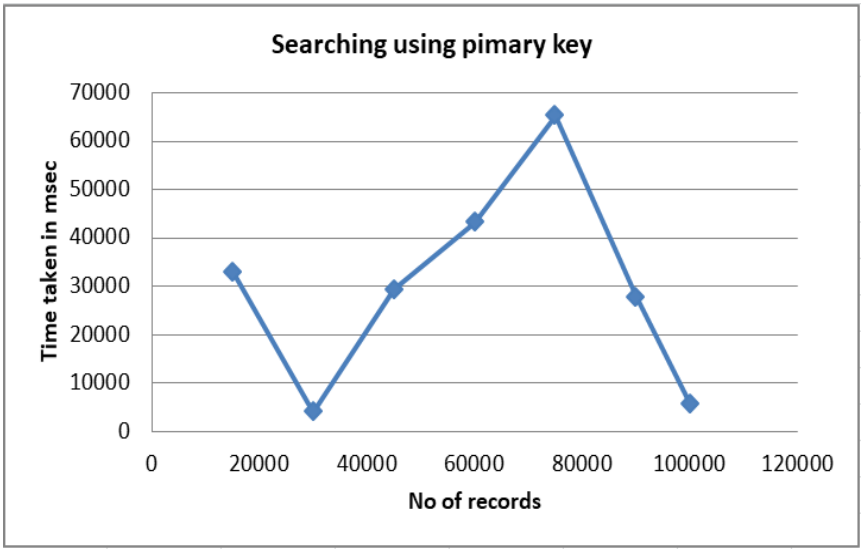

Fig 4.9.2: Time analysis for searching the record in file by using primary key, as the number of record increases the time will be delayed to search the record.

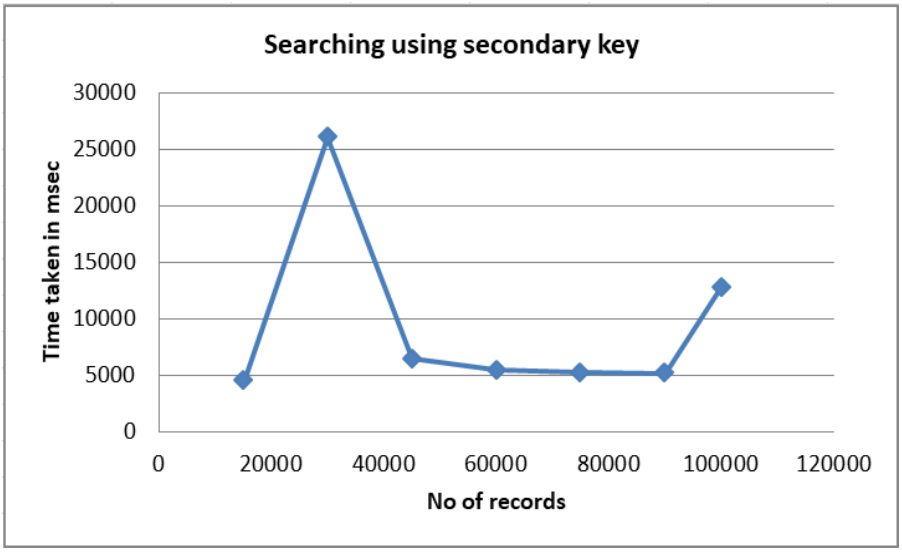

Fig 4.9.3: Time analysis for searching the record in file by using secondary key, as the number of record increases the time will be delayed to search the record.

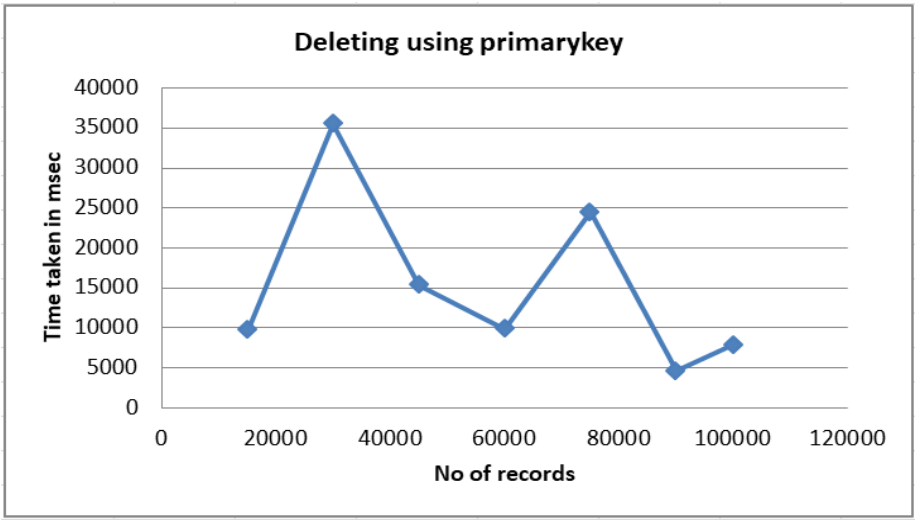

Fig 4.9.4 : Time analysis for deleting the record from the file using primary key, decreases as the record increases.

CONCLUSION

We have successfully used various functionalities of JAVA and created the file structures

- Indexes form an important part of designing, creating and testing information
- Users search in a hurry for information to help them and give up after two or three tries
- An index can point the way in harmony with user expectations or not.
- Indexing is an interactive analysis and creative process throughout the entire documentation

View tables are used to display all the components at once so that user can see all the components of a particular type at once.one can just select the component and modify and remove the component.

Features:

- Clean separation of various components to facilitate easy modification and revision.
- All the data is maintained in a separate file to facilitate easy modification
- All the data required for different operations is kept in a separate file

REFERENCES

The information about B-tree was gathered by referring to the following sites**:**

- Tutorialspoint(tutorialspoint.com)
- Stackoverflow(stackoverflow.com)
- GeeksforGeeks(GeeksforGeeks.com)
- Javatpoint(javatpoint.com)
- Java Debugger(javadebugger.com)
- Michael J.Folk,Bill Zoeclick,Greg Riccardi:File Structures-An Object Oriented Approach with C++,3rd Edition,Pearson Education ,1998.
- Scott Robert Ladd:C++ Components and Algorithms,BPB Publications,1993.