Oracle, MySQL, PostgreSQL, SQLite, SQL Server: Performance based competitive analysis

Prepared By

Md. Ismail Hossain

ID: 152-15-575

Sazal Mahmud

ID: 152-15-568

Tanusree Das Santa

ID: 152-15-579

This Report Demonstrated in some Fruition of the Capital Requirements for the Bachelor of Science Degree in Computer Science and Engineering.

Supervised By

Toufik Ahmed Emon
Lecturer, CSE
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised By

Md. Reduanul Haque
Lecturer, CSE
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
APRIL 2019

APPROVAL

This Project titled **"Oracle, MySQL, PostgreSQL, SQLite, SQL Server: Performance based competitive analysis"**, submitted by **Md. Ismail Hossain**, ID: 152-15-575, **Sazal Mahmud** ID: 152-15-568, **Tanusree Das Santa** ID : 152-15-579 , to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 2 April 2019.

BOARD OF EXAMINERS

_____

Dr. Syed Akhter Hossain — **Chairman**

Professor and Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

_____

Dr. S M Aminul Haque — **Internal Examiner**

Assistant Professor & Associate Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

_____

Saif Mahmud Parvez — **Internal Examiner**

Lecturer

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

_____

Dr. Mohammad Shorif Uddin — **External Examiner**

Professor

Department of Computer Science and Engineering

Jahangirnagar University

DECLARATION

We announce to notify that, this program has been done by us under the supervision of **Toufik Ahmed Emon, Lecturer, Department of CSE & co-supervision of Md. Reduanul Haque, Department of CSE** Daffodil International University. We also notify that neither this program nor any part of this program has been submitted on the other hand for award of any degree or diploma.

Supervised by:

_____

Toufik Ahmed Emon
Lecturer
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by:

_____

Md. Reduanul Haque
Lecturer
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

_____

Md. Ismail Hossain
ID: 152-15-575
Department of CSE
Daffodil International University

_____

Sazal Mahmud
ID: 152-15-568
Department of CSE
Daffodil International University

_____

Tanusree Das Santa
ID: 152-15-579
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

At first, we manifest our heartiest thanks and gratefulness to our creator the most gracious the most merciful almighty god for his mercy upon us which make us capable to complete our final year thesis properly.

We are really beholden and obliged and wish our indebtedness to **ToufiK Ahmed Emon, Lecturer, Department of CSE** Daffodil International University, Dhaka. Our supervisor's depth knowledge and intense interest in the sector of "Image Processing" and "Database Management System" helped us throughout this entire thesis. His firm encouragement, endless energetic supervision, golden advice, infinite level of patient, continuous support and correcting our faults have made the thesis to be completed. Our works are all valueless without our profound Supervisor.

We would also like to remember our Co supervisor **Md. Reduanul Haque, Lecturer, Department of CSE,** Daffodil International University, Dhaka with great respect and thankfulness for helping us to fulfil our thesis. We would also like to thanks all the other faculty member and the staff of CSE department of Daffodil International University.

We are not forgetting to thanks our all course mates in Daffodil International University, who attended this dialog while completing their course work.

And finally, we must acknowledge with deep respect the grand effort, valuable support and heartiest prayers of our parents.

ABSTRACT

In today's modern world we can't imagine an organization without a database management system. We have seen that the most used technique in the modern world is a relational database management system which is known as RDBMS. RDBMS is far ahead of the other available options. As RDBMS is the most used system, we have decided to work with it. In our country, many organizations are using RDBMS yet they don't have the proper knowledge about the efficient use of database management system. Meaning which database management is best for their kind of use. Our thesis will help everyone to understand the best use of database management system software as per their applications. It will help them in many ways including time-saving, efficiency, cost and proper management of the database. By going through this thesis paper we will get the proper knowledge about every part which matters in the Database management system.

Keywords: DBMS software, RDBMS, Core i3 & i5 Processors, Ram, Internet, Rom.

TABLE OF CONTENTS

Contents **PAGE**

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Introduction

In today's modern world database management system is a very important and well-known topic. Almost every big company like Google, Facebook, Amazon have their own database management system. And they have several different types of the database available for their uses. We can store, manipulate, retrieve and use a huge amount of data by using a single database management system. To maintain a database management system or ensuring security, it takes much experience and much knowledge about DBMS.

In our country we are using database very often yet we don't know which database management system is best for us in key terms like web application uses, personal uses, etc. And also the security is coming with these various database management system is very important to know about. If we use a proper database management system, it can give us many benefits. For example: if we need a piece of information about a single student in a school of 10000 students, it takes much time if we use the hardcopy to search the desired student. But if we use a database management system it takes almost a single query or like half of the minutes to search the desired student.

There are several options available for having our own database management system. The most used or the most popular among them are Oracle, MYSQL, SQLite, POSTGRESQL, SQL SERVER, etc. We have tried to analyze them by their features or uses and come up to a conclusion that which one is the best for our environment to use it. We have mainly focused on platforms requirements, their time complexity and their offered security management and also their user-friendly components, and their special and unique features.

1.2 Motivation

In today's world, data is like wealth. If it falls into a wrong hand it can ruin our whole life. Now in the world, many country's are having a blast in their technology uses. Everyone is trying to be more technology dependent and the use of technology is increasing more and more. These events are pushing everything to be available online. Simply we can say that every single data is becoming accessible on the internet. The government, many small or big company or institutions like school, colleges and many different places are now having their own databases. To get the most benefits of these databases we need to know that the which one is best for our kind of use, as there are several reasons and several database types available.

There are still lacking but in developed country's there are many experts available in the database management system to provide you the answers to your query about these available platforms. But in our country, the field of the database management system is quite different. As there are not many database advisers available in our country we need to face many kinds of pre uses problems like choosing platforms, the security issue, facilities, etc.

We can get confused that which one will be the best for us to use. For these kinds of situations we need to have proper knowledge about all the available platforms and their differences between each other.

Our main objective is to provide the best database for each individual users. We have used a straight forward approach to find the solutions of most important factors which are required to come to a conclusion.

1.3 Expected Output

There is some major expected output of our thesis paper what we have given below:

- ✓ Getting the best RDBMS
- ✓ Ease of choosing RDBMS according to its user
- ✓ Visualization of time comparison among RDBMS
- ✓ Prevention of wasting money and time
- ✓ General user will be helped a lot
- ✓ Even a layman will understand easily.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

We have read many research papers and we have seen that they have discussed a lot about RDBM. This research paper contains many comparisons on the term like data amount to be used, security analysis, memory uses, etc. Many people are getting a good amount of knowledge from these papers and it is coming handy in their uses. Also, we have seen that most of the available research paper concludes the comparison between two or three DBMS or RDBMS. But in our thesis paper, we have used five well known RDBMS so that everyone will get an idea about RDBMS and the proper uses of RDBMS and also understand the differences among different RDBMS.

In this chapter, we have given some major points of some research papers and also the limitations they have. Not only we have analyzed the problems in those research papers but also we are talking about the solutions they need for better performance.

2.2 Research paper summary

- **Comparative Performance Analysis of MySQL and SQL Server Relational Database Management Systems in Windows Environment 2015**

  The most important and popular tools for the diligence of data is the Database Management System. Proprietary systems like Oracle DB or SQL Server is widely used but open-source RDBMS. systems are not widely used. Now a day MySQL is work better. The aim of this paper is the find the better one, compare the performance of MySQL against SQL Server in Windows Environment. After compare the performance the final decision is that the SQL server is better than my SQL [1].

- **A Comparative Study on the Performance of the Top DBMS Systems**

  This research was funded by the Lebanese Association for Computational Sciences (LACSC), Beirut, Lebanon, under the "Evaluation & Performance Research Project – EPRP2012". This paper discusses the measurement of the performance of the uppermost DBMS systems through comparative study. if we execute different SQL queries with the different DBMS for that the complexities are different. for that reason, we have to test. The main purpose of testing to find out the average execution time, memory usage, and CPU utilization of each DBMS. The results of the testing represent the most effective DBMS systems [2].

- **MongoDB vs Oracle - Database Comparison**

  This paper discusses the differences between MongoDB and oracle. The purpose of development process MongoDB gives workability. MongoDB use plugin for open source process. The main difference between the two databases is that the Database has relations between the tables. this relation can join the table and make complex queries.as like as MongoDB, you cannot copy the database in Oracle. The comparison is saying that if we want a fast and flexible database we can choose MongoDB and if we want to the relation between the tables we can choose Oracle [3].

- **Hybrid Database System of MySQL and MongoDB in Web Application Development**

  When we use a web application, we can select different database model. For the purpose of performance measure the web application equipped with chat and profile view features. Website is equipped with chat and profile view features. Experiments are performed using randomly generated user profiles and chat data. after the experiment, we can see that the Hybrid Database System of MySQL and MongoDB is better for reading and write performance than myself but myself better for maintaining data consistency and more sensitive data. The hybrid model database MySQL and MongoDB uses less disk space than

MySQL alone but uses more RAM. There is not much difference in CPU usage for both database models. From the experimental results and analysis, we can conclude that the hybrid database model of MySQL and MongoDB improves web application performance on large database size [4].

- **Security analysis of unstructured data IN NoSQL & MongoDB Database**

  This paper discusses that NoSQL MongoDB database has different problems to overcome these problems we use the different type of technique and algorithms. basically, we will use the NoSQL database for storing unstructured data. But NoSQL MongoDB database has many security risks. The efficient cryptographic system uses to overcome this type of security risk. DES, AES, and Blowfish algorithms with random key generation are used to encrypt/decrypt the document data and zlib. the compression technique is used to minimize the storage size which is taken by encrypted data [5].

2.3 Most uses database organization

MySQL	US Navy, NASA, Facebook, BBC news, YouTube , Apple Inc., United Nations FAO, Crazy Egg etc.
PostgreSQL	University of Oslo(Norway), University of Sydney, Moby Games, Shannon Medical center ,Macworld, Greenpeace etc.
SQLite	Facebook, Firefox web Browser, Flame, Dropbox, Google, McAfee, apple etc.
Oracle	Amazon, Apple, Sony Corporation, LinkedIn, Airtel etc.
SQL Server	hp, intel, TOYOTA, NASA, DELL, Microsoft, Bank of America ,HSBC etc.

2.4 Which one is best for whom

- For web applications the most used are MySQL & PostgreSQL and these are free.
- Oracle is a business software which offers high security and also quite costly.
- SQLite is used for mobile applications and medium websites.
- SQL server is used by many organizations.

CHAPTER 3

Research Methodology

3.1 Introduction

For our research purpose, we have found out the most popular most used top 5 database management system. These are Oracle, MYSQL, SQLite, POSTGRESQL, SQL SERVER. These databases are based on the relational database management system. Because they follow the structured query languages. Though they are different the syntax they use and their working style is the same.

On our first move, we have run a survey on the students of our DIU permanent campus students and we have noted the personal computer configuration they have. The information we have collected like their operating system, ram, and other parts we have found that the maximum used operating system is windows 10, the maximum used processor is core i3 or core i5 with 6th generation and the ram used is 4 GB which is also of the 6th generation. As these maximum used parts, we have started our research by using these maximum used processors which core i3 and core i5. Then we have installed the databases one by one and we have noted down their required installation time, the space they required and the size of the executable file and the software size. After completing the installation we have inserted like 6-240 numeric data in the databases one by one. Then we have noted the time they take to insert, select, delete the update and we came to a conclusion that almost all of these database takes the almost same time to perform a query. Then we wanted to go deeper and we have inserted almost 100000-400000 numeric data and then perform many queries and once again note the time they took.

After the second operation, we have noticed some significant change in their time to completing the query. As security is a major component, we have done some research on their security facilities which will come in front as we go through in the later chapter of our research paper.

3.2 Background & Component

3.2.1 Computer Configuration Screenshot

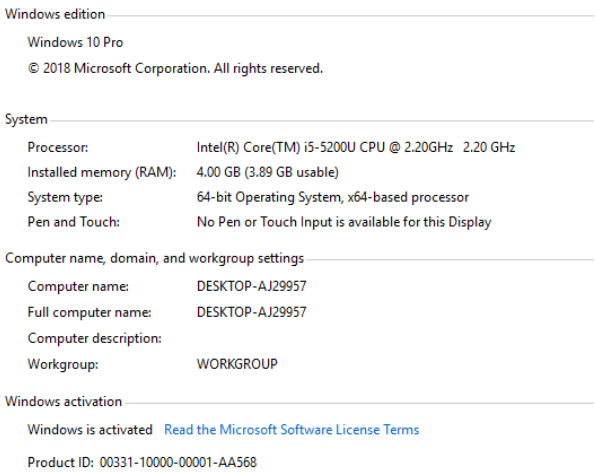

Figure 1: i5 Configuration Computer

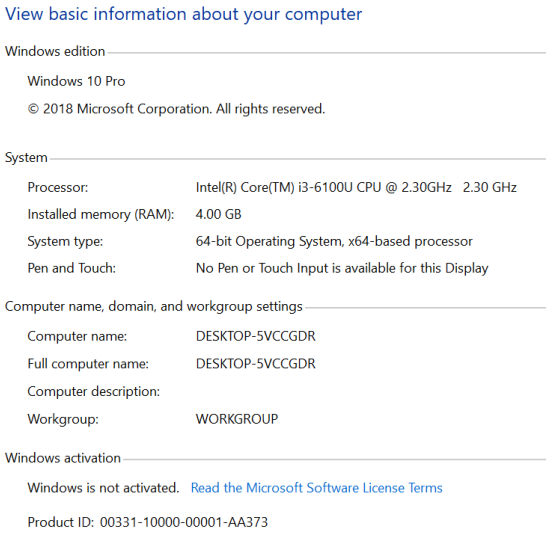

Figure 2: i3 Configuration Computer

3.3 Code & Query Languages

3.3.1 R languages Code

Core:

```
table(data$Core)
i3 i5 i7
41 55 6
> coredata<-table(data$Core)
> barplot(coredata,main = “Total number of cOre Processor”,xlab=”CORE”,ylab = “Number of Users”,col=c(“darkblue”,”red”,”green”))
> barplot(coredata,main = “Total number of core Processor”,xlab=”CORE”,ylab = “Number of Users”,col=c(“darkblue”,”red”,”green”))
> barplot(coredata,main = “Total number of core processor”,xlab=”CORE”,ylab = “Number of Users”,col=c(“darkblue”,”red”,”green”))
```

Generation:

```
generationData<-table(data$Generation)
> barplot(generationData,main = “Total Number of Rom”,xlab=”Genaration”,ylab=”Number of Users”,col=c(“red”,”green”,”darkblue”,”pink”,”darkblue”,”orange”,”gold”))
> barplot(generationData,main = “Total Number of Generation”,xlab=”Genaration”,ylab=”Number of Users”,col=c(“red”,”green”,”darkblue”,”pink”,”darkblue”,”orange”,”gold”))
```

GHZ:

```
barplot(GHZdata,main = “Total Number of GHZ”,xlab =”GHZ”,ylab = “Number of Users”,col = c(“blue”,”red”,”green”,”darkblue”,”pink”,”darkblue”,”orange”,”gold”,”hotpink”,”gray”,”salmon”,”maroon”,”tomato”,”greenyellow”,”darkgreen”,”navy”,”chocolate”,”coral”,”deeppink”,”blueviolet”,”royalblue”,”peru”))
```

Ram:

```
table(data$Ram)
12GB 16GB  4GB  5GB  6GB  8GB
  2    1   74    1    2   22
```

> ramData<-table(data$Ram)

> barplot(ramData,main = “Total Number of Ram”,xlab=”RAM”,ylab=”Number of Users”,col=c(“blue”,”red”,”green”,”darkblue”,”pink”,”darkblue”))

>

Rom:

romData<-table(data$Rom)

> barplot(romData,main = “Total Number of Rom”,xlab=”ROM”,ylab=”Number of Users”,col=c(“blue”,”red”,”green”,”darkblue”,”pink”,”darkblue”,”orange”,”gold”))

3.3.2 Database Query Language

Table Create syntax:

```
create table table_name(
column_name1  varchar2  not null,
column_name2  varchar2  not null,
CONSTRAINT table_name_pk PRIMARY KEY(column_name1)
 );
```

Insert syntax:

INSERT INTO table_name (column_name1,column_name2) VALUES (‘value’, ‘value’);

Select syntax:

SELECT * FROM table_name;

Update syntax:

```
UPDATE table_name
 SET column name = ‘’value”
 Where column name= ’value’ ;
```

Delete syntax:

```
delete from table_name
where column name  = (‘value’);
```

Truncate or delete data Syntax:

TRUNCATE TABLE student_info;

Database Connection Syntax:

	Oracle	MySQL	PostgreSQL	SQLite	SQL Server
Connection query	User name : sys as sysdba; Password : sys123 Than, Conn hr/hr	1. Cd\ 2. Cd Xampp 3. Cd mysql 4. Cd bin 5. Mysql -u root -p	1. Server: press enter 2. Database: press enter 3. Port: press enter 4. Username: press enter 5. Password for user Postgres; 6. 12345	c:\sqlite> Then, c:\sqlite>sqlite3 db/exe.db	Username: sqlserver123 Password:12345

Import file syntax:

MySQL:

```
LOAD DATA LOCAL INFILE "C:\\Users\\SAZAL
MAHMUD\\Desktop\\work\\FARS.csv" INTO TABLE student
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 LINES
(id,caseid,state,age ,airbag,injury,restraint,sex,inimpact,modelyr,D_injury,year);
```

SQLite:

```
sqlite> .mode csv
sqlite>.import c:/sqlite/ourdata.csv table_name
```

PostgreSQL:

```
\copy student FROM 'C:/Users/SAZAL MAHMUD/Desktop/work/data/FARS.csv'
DELIMITER ',' CSV
```

Measure run time syntax:

Oracle:

```
set timing on;
```

PostgreSQL:

```
# \timing on
Timing is on.
```

SQL Server:

```
set statistics time on
```

SQLite:

```
.timer ON;
```

From the above part, we can see that the table creating syntax is different among Oracle, MySQL, PostgreSQL, SQLite & SQL Server. But the query syntaxes like select insert update delete are same for each of them. And not only the connection query syntax but also the import file query syntax & the measure run time syntax is different among them.

3.4 Flow Chart

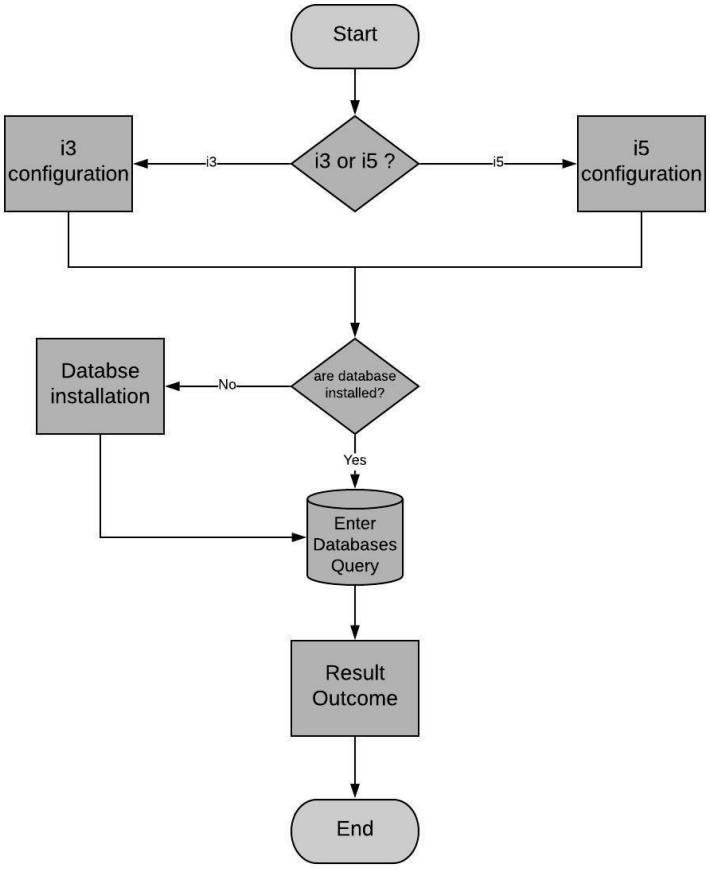

Figure 3: Our working Flow Chart

3.5 Sequence Diagram

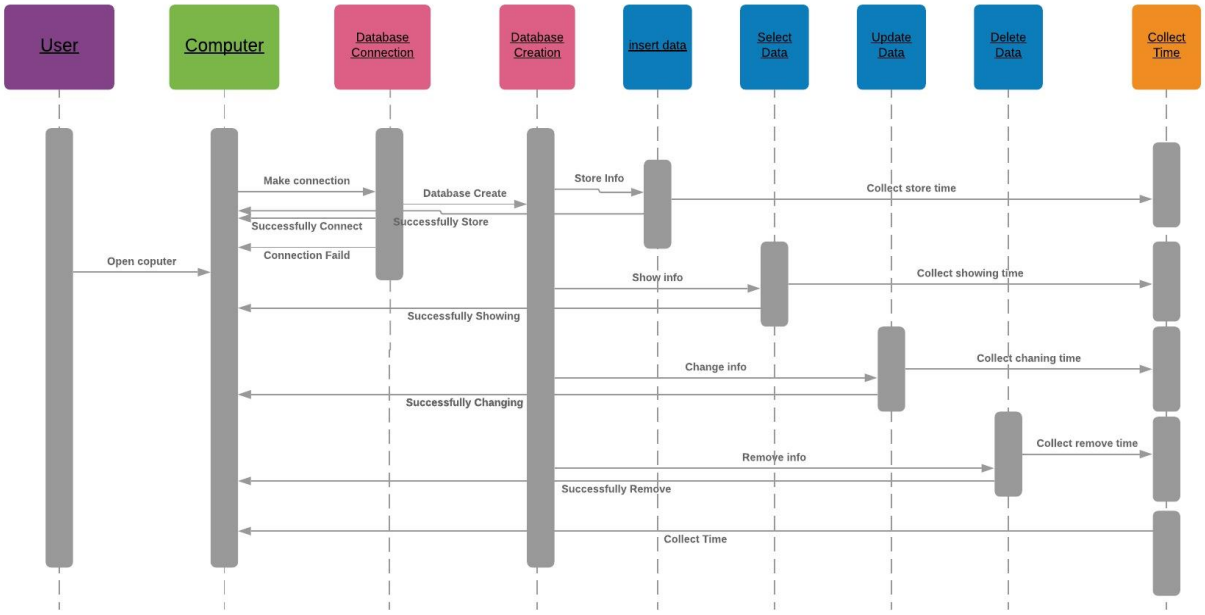

Figure 4: Sequence diagram of RDMS comparison

Chapter 4

EXPERIMENTAL RESULT & DISCUSSION

4.1 Introduction

Our main purpose was to compare five relational database management system and come to a conclusion that which one is better among them in different purposes. The whole working process can be divided into three categories like installation process, applying queries, and calculation of time. The installation process is a vital part as the installation time, installation instructions and the space they occupy is different from each other. In Applying queries section, we have analyzed that if they have any difference in query clauses and the formation of the queries are how much they differ from each other and which one is easier for anyone to apply. In calculation time section we have noted the time they took for installation and query transactions and depending on these time comparisons we have given our conclusion.

4.2 Working Procedure

4.2.1 Analyzing computer configuration outcome

At first, we have run a survey and collected the maximum used processor's names, generations, rams and rom of 100 DIU student computers. By observing the collected information, we have drawn some diagrams below.

Core analysis

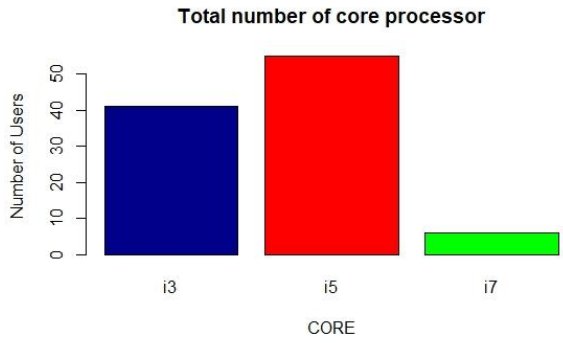

Figure 5: Core analysis

As we can see in figure 5, the diagram suggests, the maximum used processors are core i3 & core i5 and core i7 is very minor as the other. So we have started our work with core i3 and core i5.

Generation analysis

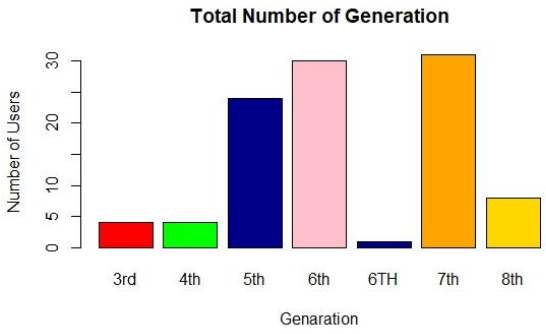

Figure 6: Generation analysis

In figure 6, we can see that the most used generations are 6th and 7th and their using level is pretty much the same. So, we have randomly chosen the 6th one.

GHZ analysis

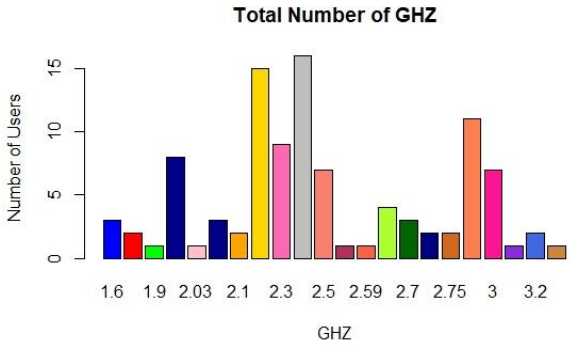

Figure 7: GHZ analysis

The figure 7's diagram suggests the processers GHz. The most used is 2.2 and 2.4 and as there is not much difference we have randomly chosen the 2.2 GHz.

Ram analysis

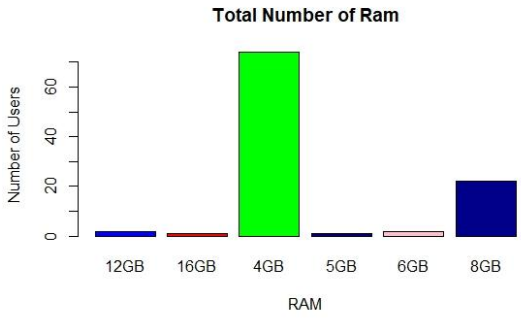

Figure 8: Ram analysis

Figure 8 indicates the majority use 4GB ram so as we have selected it.

Rom analysis:

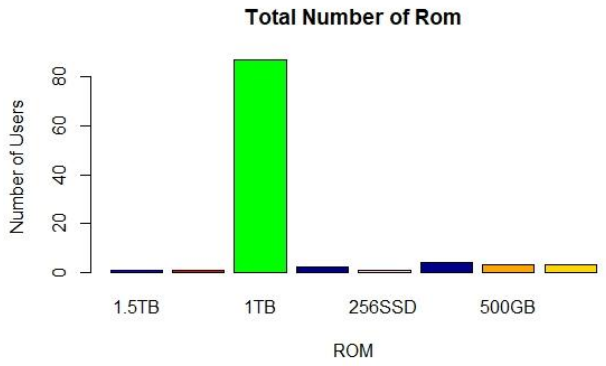

Figure 9: Rom analysis

In figure 9 we can see that the 1TB rom is used by the majority of the users so we have selected 1TB rom

4.2.2 Databases Installation time & storage capacity performance

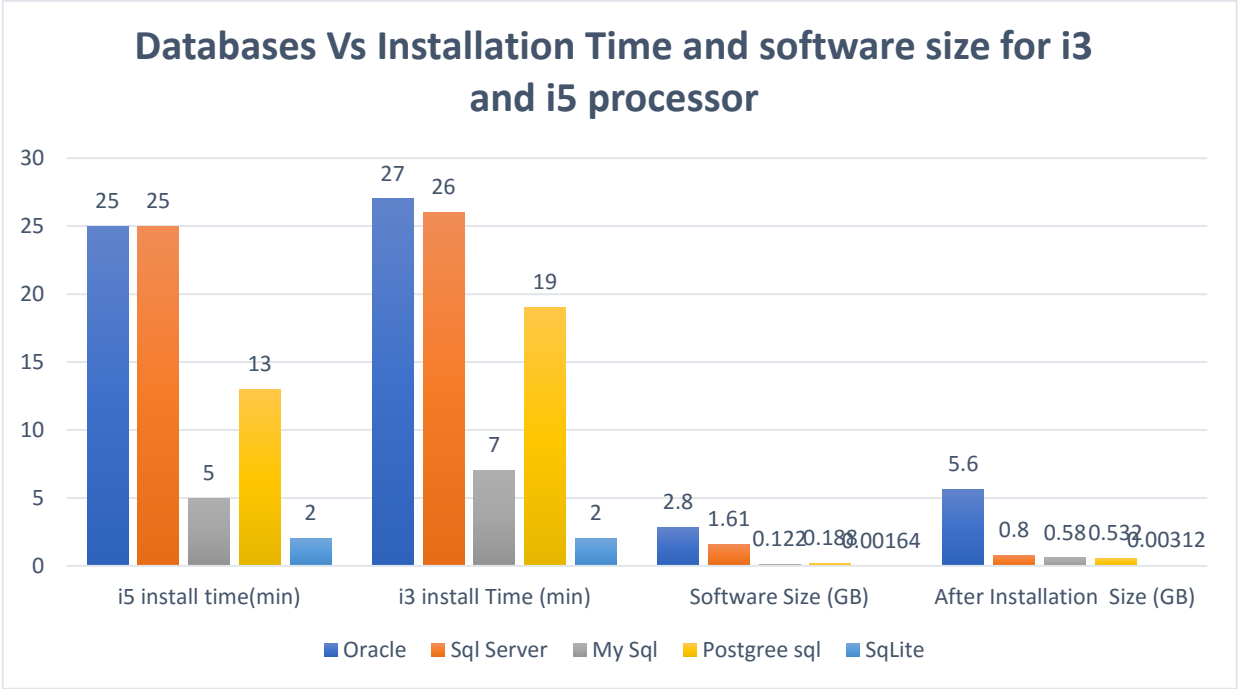

Figure 10: Databases vs installation time and software size performance

As the figure 10, we have noted the software's sizes and then installed the software's and while installing we have noted the installation time taken by each individual software's and after finishing the installation we have noted the space they acquire on the rom.

We have got to know that the most lesser in size, the lesser installation time and the less acquire of rom after installation is own by SQLite.

4.2.3 Data Vs Time Comparison

4.2.3.1 For i3 configuration computer

We have noted the time comparison by implementing Insert, Select, Update, Delete queries on a database of 6 to 400000 data.

Time comparison for 6 data

Database	Insert	Select	Update	Delete
oracle	6	9	1	1
Sql server	20	1	22	5
Mysql	60	10	60	200
Postgrersql	1.32	0.781	1.97	0.832
SqLite	0.338	0.1	0.111	0.169

Figure 11: 6 data vs time compression table

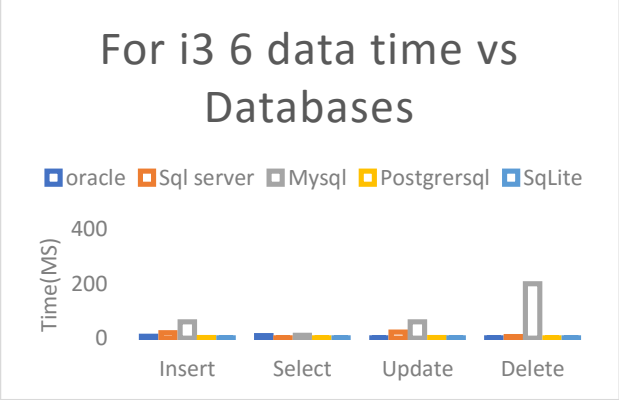

Figure 12: 6 data vs time compression chart

In figure 12 We can see that the time comparison of inserting, selecting, updating and deleting queries on the basis of 6 data. And the less time took by PostgreSQL and SQLite for completing the queries.

Time comparison for 60 data

Database	Insert	Select	Update	Delete
oracle	9	92	1	1
Sql server	23	1	25	7
Mysql	70	10	60	100
Postgrersql	20.31	2.1	0.987	2.31
SqLite	0.16	0.1	0.81	0.2

Figure 13: 60 data vs time compression table

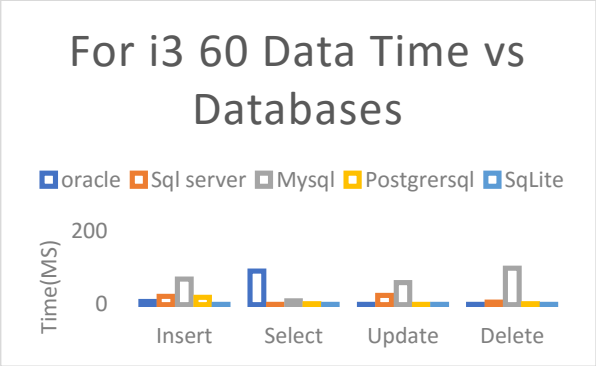

Figure 14: 60 data vs time compression chart

In figure 14, We can see that the time comparison of inserting, selecting, updating and deleting queries on the basis of 60 data. And the less time took by PostgreSQL and SQLite for completing the queries. Also, we can see that for update and delete Oracle takes less time than others.

Time comparison for 120 data

Database	Insert	Select	Update	Delete
oracle	10	1760	1.5	1.7
Sql server	25	1.3	27	8
Mysql	80	10	70	140
Postgrersql	22.87	2.97	1.879	2.982
SqLite	0.82	0.15	0.245	0.13

Figure 15: 120 data vs time compression table

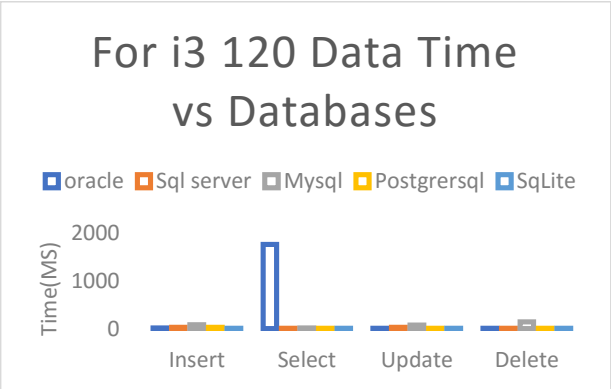

Figure 16: 120 data vs time compression chart

In figure 16, We can see that for 120 data, Oracle takes less time than others.

Time comparison for 240 data

Database	Insert	Select	Update	Delete
oracle	45	2250	1.9	2.3
Sql server	20	1.2	29	5
Mysql	160	10	30	100
Postgrersql	29.82	3.62	6.029	3.132
SqLite	0.181	0.161	0.168	0.198

Figure 17: 240 data vs time compression table

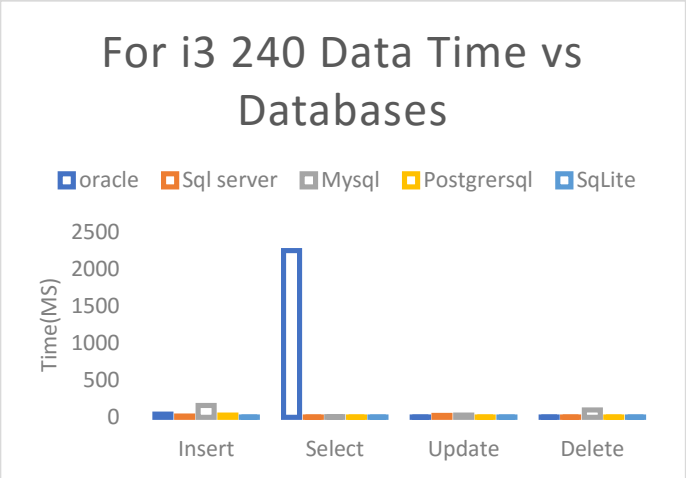

Figure 18: 240 data vs time compression chart

Figure 18, For 240 data, we can see that for all queries SQLite take less time than others.

Time comparison for 480 data

Database	Insert	Select	Update	Delete
oracle	150	4540	2.1	2.6
Sql server	30	1.3	25	10
Mysql	220	10	90	130
Postgrersql	35.876	5.12	9.328	3.782
SqLite	0.348	0.176	0.478	0.173

Figure 19: 480 data vs time compression table

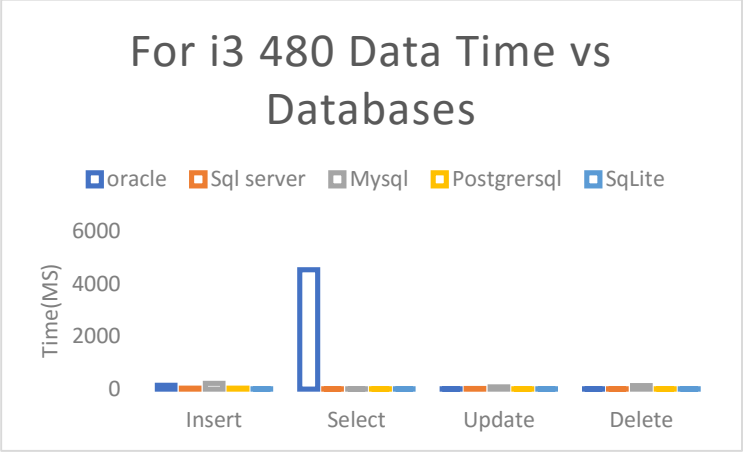

Figure 20: 480 data vs time compression chart

Figure 20 shows comparison for 480 data, we can see that for all queries the SQLite take less time than others.

Time comparison for 100000 data

Database	Insert	Select	Update	Delete
oracle	8000	38500	1500	840
Sql server	2723	1000	1547	104
Mysql	3400	590	41030	70010
Postgrersql	250.837	800.96	257.13	97.832
SqLite	3000	268.796	82.877	7.014

Figure 21: 100000 data vs time compression table

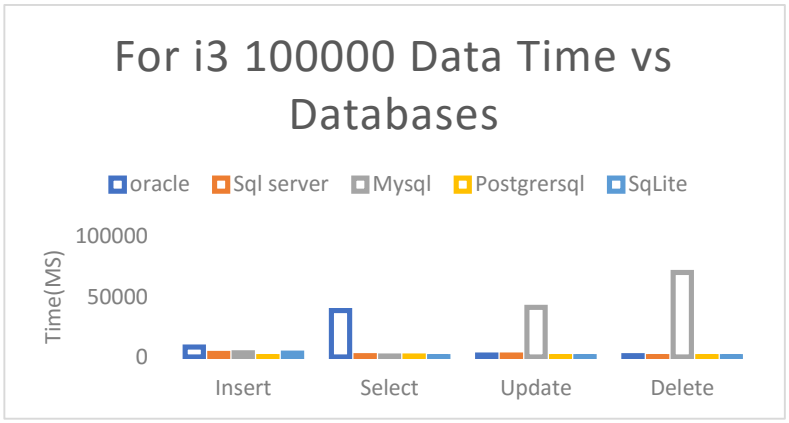

Figure 22: 100000 data vs time compression chart

In figure 22 we can see SQLite takes some time to fulfil the insert query but it takes much less time for other queries as compared to others.

Time comparison for 200000 data

Database	Insert	Select	Update	Delete
oracle	9000	66674	1608	1430
Sql server	3513	3000	2572	108
Mysql	7820	730	93860	122820
Postgrersql	538.32	872.32	500.17	187.33
SqLite	4000	346.387	162.76	2.437

Figure 23: 200000 data vs time compression table

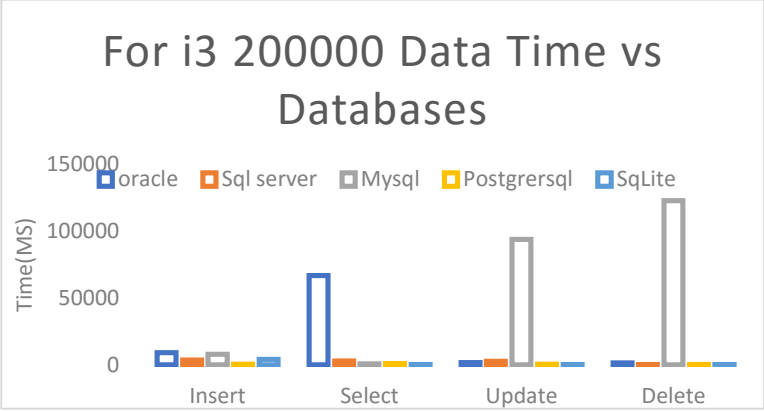

Figure 24: 200000 data vs time compression chart

Figure 24 shows comparison for 200000 data, SQLite takes much less time for all queries as compared to others.

Time comparison for 300000 data

Database	Insert	Select	Update	Delete
oracle	10000	85850	2000	2920
Sql server	4012	4000	3532	113
Mysql	10130	810	134590	139086
Postgrersql	883.07	1286.89	696.17	795.32
SqLite	4000	360.84	186.317	2.952

Figure 25: 300000 data vs time compression table

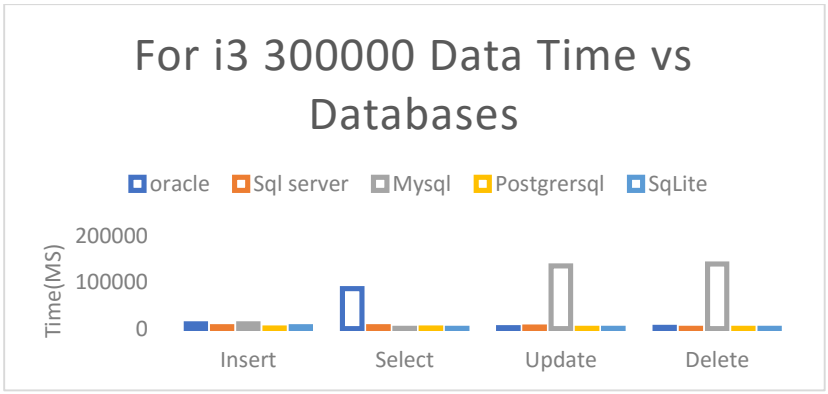

Figure 26: 300000 data vs time compression chart

In figure 26 we can see that for 300000 data, also here SQLite takes much less time for other queries as compared to others.

Time comparison for 400000 data

Database	Insert	Select	Update	Delete
oracle	11000	130889	5050	3760
Sql server	4401	5000	4509	120
Mysql	14250	1032	166390	233830
Postgrersql	1783.32	1586.1	874.62	832.71
SqLite	5000	360.84	222.46	3.234

Figure 27: 400000 data vs time compression table

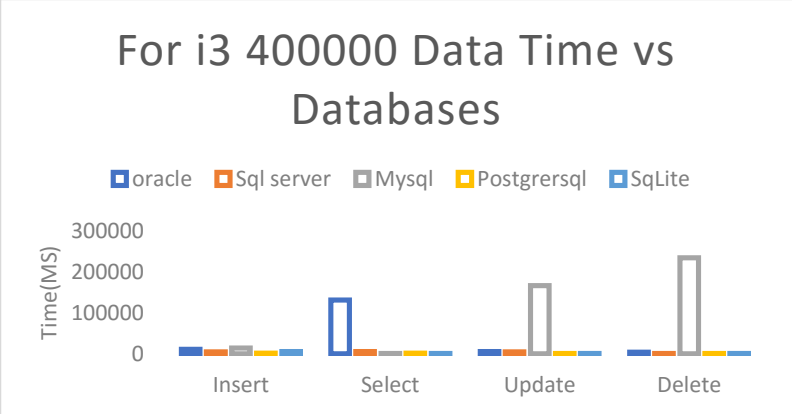

Figure 28: 400000 data vs time compression chart

Figure 28 suggests that for 400000 data, also here SQLite takes much less time for other queries as compared to others.

4.2.3.2 For i5 configuration computer

We have noted the time comparison by implementing Insert, Select, Update, Delete queries on a database of 6 to 400000 data.

Time comparison for 6 data

Database	Insert	Select	Update	Delete
oracle	4	15	1	1.2
Sql server	19	1.4	20	15
Mysql	70	10	70	20
Postgrersql	25.815	0.416	1.33	0.534
SqLite	0.3	0.12	0.101	0.2

Figure 29: 6 data vs time compression table

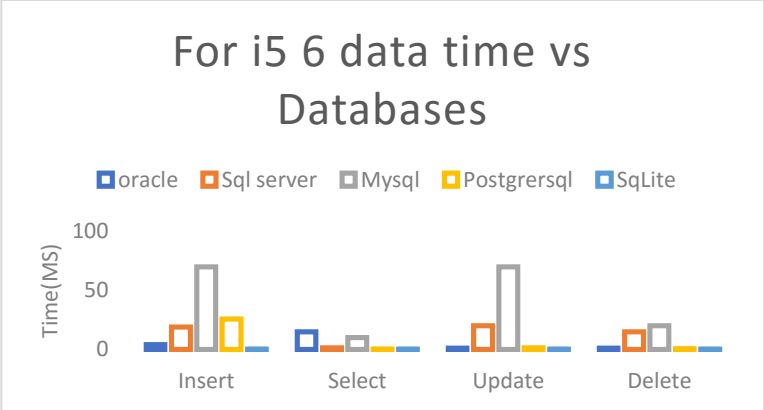

Figure 30: 400000 data vs time compression chart

Figure 30 tells that for 6 data, SQLite takes much less time for other queries as compared to others.

Time comparison for 60 data

Database	Insert	Select	Update	Delete
oracle	7	12	1	1
Sql server	20	1	23	6
Mysql	70	10	40	40
Postgrersql	2.411	1.432	0.787	1.432
SqLite	0.1	0.1	0.76	0.195

Figure 31: 60 data vs time compression table

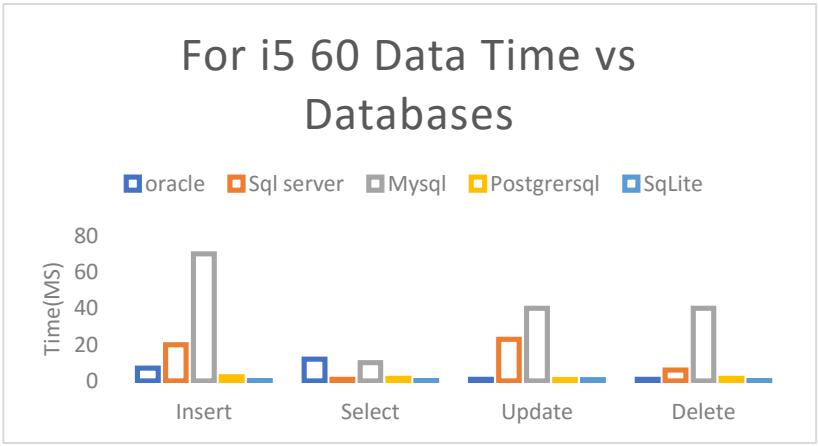

Figure 32: 60 data vs time compression chart

In figure 32 we can see that for 60 data SQLite takes much less time for other queries as compared to others.

Time comparison for 120 data

Database	Insert	Select	Update	Delete
oracle	9.5	1565	1.3	1.5
Sql server	23	1.21	25	7
Mysql	70	10	30	111
Postgrersql	3.134	0.528	3.003	2.01
SqLite	0.16	0.2	0.81	1.5

Figure 33: 120 data vs time compression table

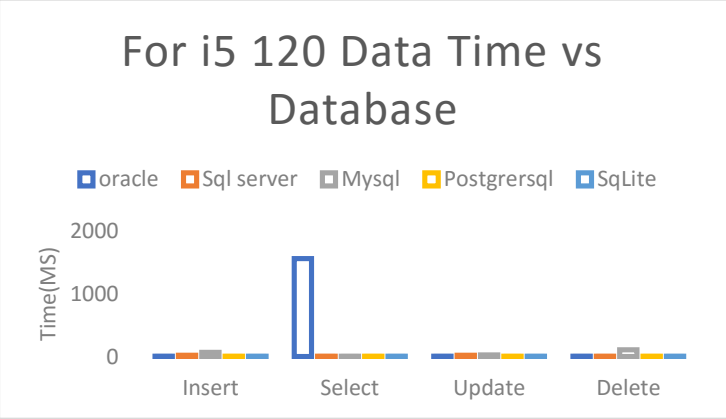

Figure 34: 120 data vs time compression chart

Figure 34 shows comparison for 120 data, here SQLite takes much less time for other queries as compared to others.

Time comparison for 240 data

Database	Insert	Select	Update	Delete
oracle	40	2133	2	2.122
Sql server	21	1.232	21	3
Mysql	130	10	21.25	89.67
Postgrersql	24.32	3.1	5.786	2.893
SqLite	0.181	0.198	0.168	0.63

Figure 35: 240 data vs time compression table

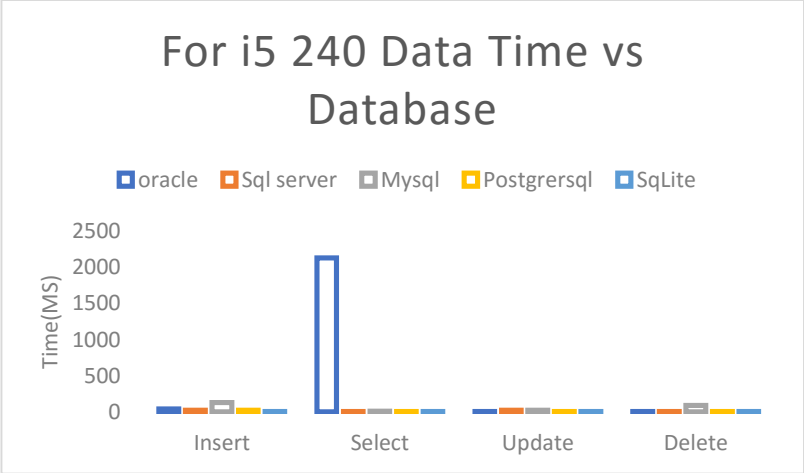

Figure 36: 240 data vs time compression chart

In figure 36 we can see that for 240 data, SQLite takes much less time for other queries as compared to others.

Time comparison for 480 data

Database	Insert	Select	Update	Delete
oracle	150	4540	2.1	2.6
Sql server	30	1.3	25	10
Mysql	220	10	90	130
Postgrersql	35.876	5.12	9.328	3.782
SqLite	0.33	0.113	0.21	0.123

Figure 37: 480 data vs time compression table

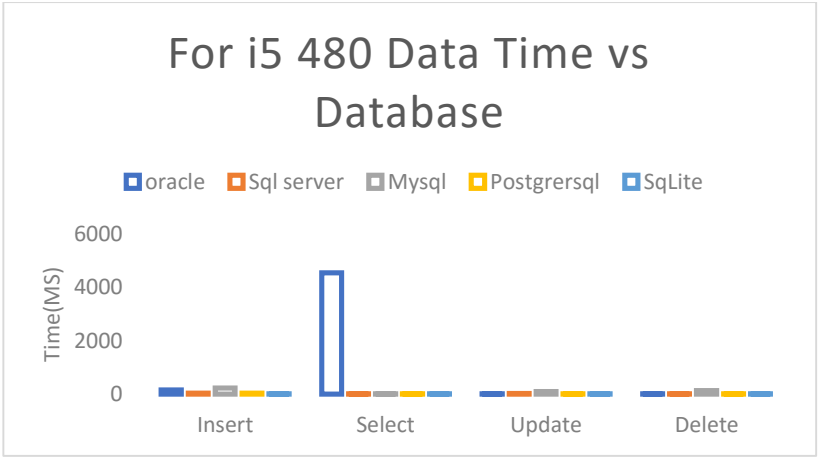

Figure 38: 240 data vs time compression chart

Figure 38 indicating that for 480 data, SQLite takes much less time for other queries as compared to others.

Time comparison for 100000 data

Database	Insert	Select	Update	Delete
oracle	8000	38500	1500	840
Sql server	2723	1000	1547	104
Mysql	3400	590	41030	70010
Postgrersql	250.837	800.96	257.13	97.832
SqLite	2382	320.23	79.923	6.234

Figure 39: 100000 data vs time compression table

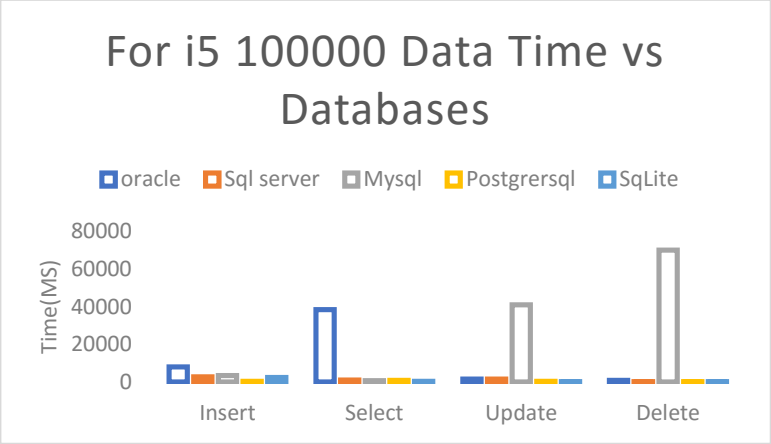

Figure 40: 100000 data vs time compression chart

In figure 40 we can see that for 100000 data, also here the SQLite takes much less time for other queries as compared to others.

Time comparison for 200000 data

Database	Insert	Select	Update	Delete
oracle	8080	59911	1567	1392
Sql server	3231	2715	2253	107
Mysql	6732	650	81543	115643
Postgrersql	497.2	850.112	487.14	156.43
SqLite	3597	343.43	155.57	2.212

Figure 41: 200000 data vs time compression table

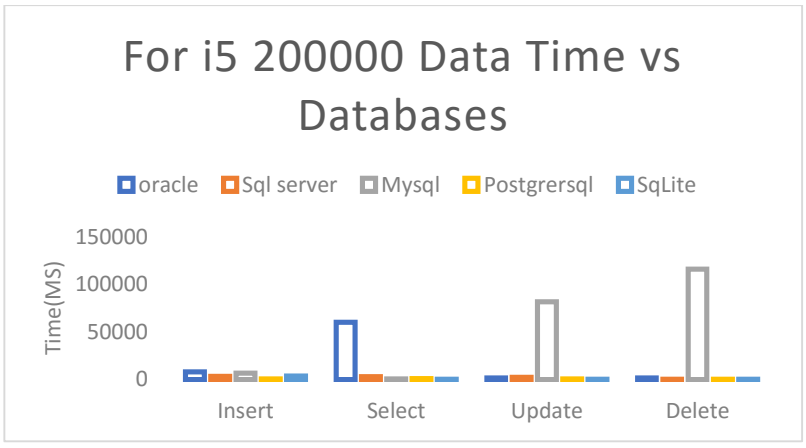

Figure 42: 100000 data vs time compression chart

In figure 42 we can see that for 200000 data, also here the SQLite takes much less time for other queries as compared to others.

Time comparison for 300000 data

Database	Insert	Select	Update	Delete
oracle	9082	75651	1943	2769
Sql server	3931	3870	3321	107
Mysql	10000	765	126750	131540
Postgrersql	765.712	1121.23	532.15	688.132
SqLite	3600	341.234	181.21	2.861

Figure 43: 300000 data vs time compression table

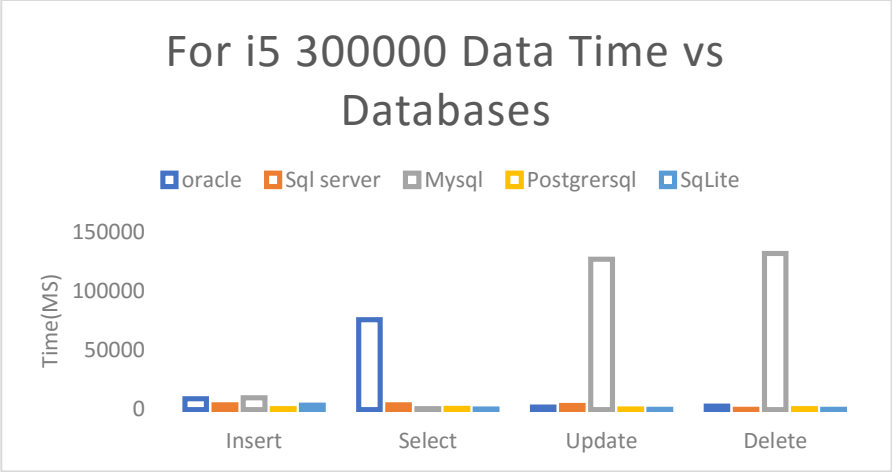

Figure 44: 300000 data vs time compression chart

Figure 44 indicating that for 300000 data, also here the SQLite takes much less time for other queries as compared to others.

Time comparison for 400000 data

Database	Insert	Select	Update	Delete
oracle	10550	127354	4850.21	3645.43
Sql server	4100	4702	4321.23	110
Mysql	12340	925.56	158431.3	216781
Postgrersql	1694.57	1580.31	785.18	763.19
SqLite	36500	388.431	219.654	3.165

Figure 45: 400000 data vs time compression table

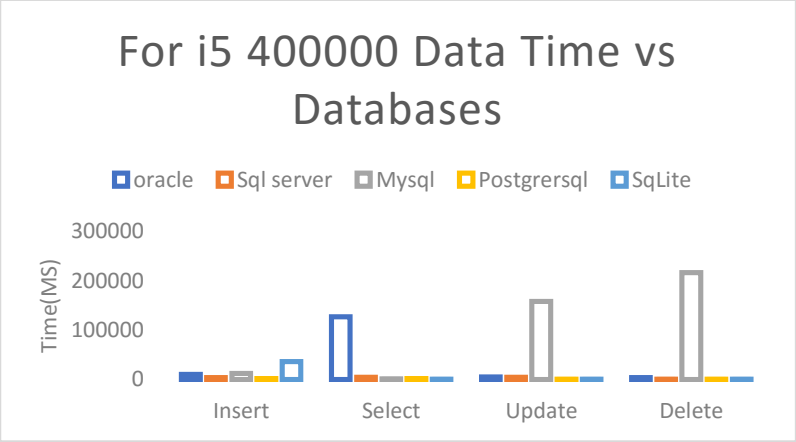

Figure 46: 400000 data vs time compression chart

And finally figure 46 shows that, for 400000 data, also here the SQLite takes much less time for other queries as compared to others.

4.3 Future Work

We are thinking of doing multitudinous dynamic project and thesis based on the inspiration and outcomes of this thesis paper.

- Working with more data in RDBMS
- Working with other DBMS
- Analyzing the security features among different DBMS
- Develop a mobile application containing the details of DBMS
- Analyzing cloud database management systems
- Creating a website which will contain comparison among DBMS.

CHAPTER 5

Conclusion

5.1 Conclusion

After completing all our working process with Oracle, MySQL, PostgreSQL, SQLite, SQL Server RDBMS we have come to a conclusion that SQLite is ahead of them as it is taking lesser time then others in completing the basic queries like insert, select, update & delete. PostgreSQL is the best competitor of SQLite. It was ahead of SQLite in insert but can't beat SQLite in other queries. And the performance of Oracle & SQL Server was almost the same but it is unfortunate to say that the performance of MySQL was really poor among them.

REFERENCES

[1] Amlanjyoti Saikia ; Sherin Joy ; Dhondup Dolma ; Roseline Mary, "Comparative Performance Analysis of MySQL and SQL Server Relational Database Management Systems in Windows Environment" , International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 3, March 2015.

[2] Youssef Bassil, "A Comparative Study on the Performance of the Top DBMS Systems", Journal of Computer Science & Research (JCSCR) - ISSN 2227-328, 957, 2011, Beirut, Lebanon.

[3] Anju R ; Swathi B.P, "MongoDB vs. Oracle: A Comparative Study Using Classification Algorithms" , Information & Communication Technology Manipal Institute of Technology Manipal, India, 2016.

[4] Gregorius Ongo ; Gede Putra Kusuma, "Hybrid Database System of MySQL and MongoDB in Web Application Development" , International Conference on Information Management and Technology (ICIMTech), 3-5 September 2018.

[5] Jitender Kumar ; Varsha Garg, "Security analysis of unstructured data IN NoSQL & MongoDB Database", International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN), 2017.