

Islington College



Information System CC4002NI Coursework 3

Submitted By:

Summit Shakya

ID: NP01CP4A170019

Group: L1C12

Date: March 23, 2018

Submitted To:

Mr. Sukrit Shakya

Module Leader

Information System

Proposal

➤ Purpose

The main purpose of this coursework is to create an imaginary database. The database consists of 5 tables with at least 5 rows in it which should be inter related. Each table should have a primary key and foreign key as per required which will help to relate it into another table within the database. Various constraints were to be used in the table (not null, unique, auto_increment). Then the tables should be displayed in the form of Entity Relationship Diagram(ERD) and Relational Diagram.

➤ Problem Statement

Any library would require a system to record the information of their customers, books they have, list of various genres, list of publication and their sales record. This system will ease a library to store information about their customers and book related entities.

➤ Aims and Objectives

The aim for this project is to try and fulfil the tasks assigned and build a simple database system that would be helpful for any library. Various research on database and efforts will be made to develop the database system which will help a library to manage the records of their customers and books.

➤ Proposed approach

Various valid journals, books and websites related to database system will be researched. A simple database will be created and various constraints will be used as per the requirement of the column of table for more providing more efficiency to the database system.

➤ Hardware and Software Requirement

A normal personal computer will do the work. XAMPP software must be available in the computer so that Mysql shell can be operated for using the system.

Activity Description

DATE	TASK PERFORMED
4 th March, 2018	Designed rough diagram for creating database.
5 th March, 2018	Developed database in mysql.
7 th March, 2018	Started the documentation part.
9 th March, 2018	Continued the documentation from model.
10 th March, 2018	Roughly prepared ERD and Relational diagram and later prepared in draw.io.
12 th March, 2018	Started with data dictionary of tables used in the database.
13 th March, 2018	Researched for various queries to be used.
15 th March, 2018	Roughly planned for queries.
16 th March, 2018	Started coding for queries along with screenshots.
19 th March, 2018	Continued the documentation and arranged the screenshots
21 st March, 2018	Result table for queries were prepared
22 nd March, 2018	Conclusion for the project was prepared.

Table of Contents

1	Introduction	1
1.1	Features of program.....	1
2.	Discussion and Analysis	2
2.1	XAMPP	2
2.2	Draw.io	3
3.	Model.....	4
3.1	Tables	4
3.1.1	Table: Customer	4
3.1.2	Table 2: Books	5
3.1.3	Table 3: Genre	6
3.1.4	Table 4: Publications.....	7
3.1.5	Table 5: Sales_Record	8
3.2	Entity Relationship Diagram	9
3.3	Relational Diagram.....	10
4.	Data Dictionary	11
4.1	Table: Customer	11
4.2	Table: Books.....	12
4.3	Table: Genre.....	12
4.4	Table: Publications	13
4.5	Table: Sales_Record	14
5.	Queries	15
5.1	Query 1	15
5.2	Query 2	16
5.3	Query 3	17
5.4	Query 4	18

5.5	Query 5	19
5.6	Query 6	20
5.7	Query 7	21
5.8	Query 8	22
5.9	Query 9	23
5.10	Query 10	24
6.	Research	25
6.1	Books	25
6.2	Journals	27
6.3	Websites.....	30
6.	Conclusion.....	33

Table of Figures

Figure 1 XAMPP Control Panel	2
Figure 2 MYSQL Shell.....	2
Figure 3 Draw.io	3
Figure 4 Table 1: Customer.....	4
Figure 5 Table 2: Books	5
Figure 6 Table 3: Genre	6
Figure 7 Table 4: Publications	7
Figure 8 Table 5: Sales_Record.....	8
Figure 9 Entity Relationship Diagram	9
Figure 10 Relational Diagram.....	10
Figure 11 Query-1	15
Figure 12 Query-2	16
Figure 13 Query-3	17
Figure 14 Query-4	18
Figure 15 Query 5	19
Figure 16 Query 6	20
Figure 17 Query 7	21
Figure 18 Query 8	22
Figure 19 Query 9	23
Figure 20 Query 10	24
Figure 21 Book1	25
Figure 22 Book 2	25
Figure 23 Book 3	26
Figure 24 Book 4	26

Figure 25 Book 5.....	27
Figure 26 Journal 1	27
Figure 27 Journal 2	28
Figure 28 Journal 3	28
Figure 29 Journal 4	29
Figure 30 Journal 5	29
Figure 31 Website 1	30
Figure 32 Website 2	30
Figure 33 Website 3	31
Figure 34 Website 4	31
Figure 35 Website 5	32

Table of Table

Table 1 Data Dictionary: Customer	11
Table 2 Data Dictionary: Books.....	12
Table 3 Data Dictionary: Genre.....	12
Table 4 Data Dictionary: Publications.....	13
Table 5 Data Dictionary: Sales_Record	14
Table 6 Query-1	15
Table 7 Query-2	16
Table 8 Query-3	17
Table 9 Query-4	18
Table 10 Query 5	19
Table 11 Query 6	20
Table 12 Query 7	21
Table 13 Query 8	22
Table 14 Query 9	23
Table 15 Query 10.....	24

1. Introduction

Database is the management system that manipulates and stores the data and information in a proper order. The usage of database is being increased day by day. There are various types of databases such as relational, object oriented, relational. MySql is a relational database management system(RDBMS).Database can be created using the software 'XAMPP'. In database, the data and information are stored in the form of tables. Tables are collection of organised rows and columns. One of the advantage of the database is that it can be easily understood and manipulated.

The following project is a report about database management system. The task for the coursework was to design and create a database of our own which had to include at least of 5 tables with 5 rows each. The tables in the database had to be inter related with each other using the concept of primary key and foreign key. The tables also had to contain various constraints (unique, auto increment, not null) as per requirement. Each table need to have a primary key which can used later as foreign key to relate appropriate tables. Entity relationship diagram(ERD) and relational diagram is also to be constructed as per the requirement of the coursework. ERD is a diagram which shows the relation between various entities used in the database. It shows the attributes of the entities and then relate entities using a diamond shape which contains the relation between entities. Relational diagram shows the primary key and foreign key of the table and also shows how the tables are related to each other. It shows the kind of relation one table has with other table such as one to one, one to many, many to one.

1.1 Features of program

- ❖ The system will store the information of the customers of the library.
- ❖ The information of the books that are available will be stored with unique ID.
- ❖ List of genres of the available books will be stored.
- ❖ Records of the publication firms will be stored.
- ❖ Record of the customer with the books they bought will be kept.

2. Discussion and Analysis

The given coursework work was interesting as it had various tasks to be completed. All the tasks in the coursework were properly fulfilled with high dedication and with lot of failed trials. Most of the errors have been tried to be removed. Various tools were used while in proper fulfilment of the coursework without which it would have not completed.

2.1 XAMPP

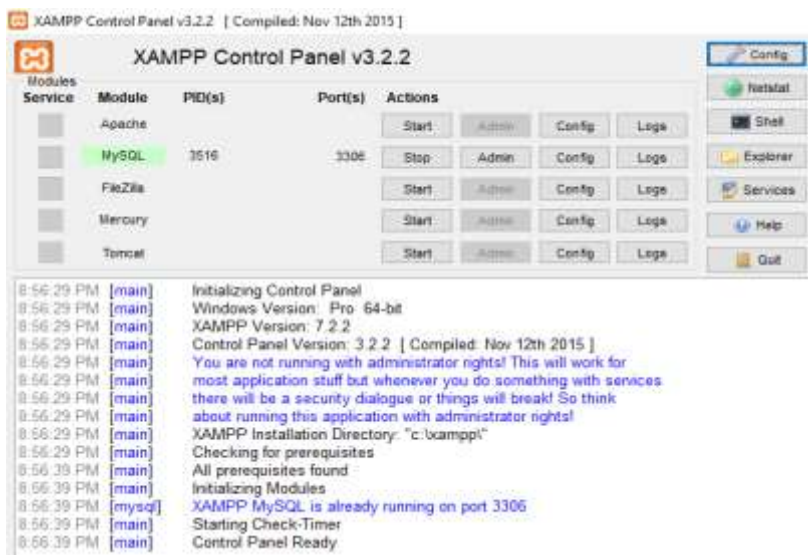


Figure 1 XAMPP Control Panel

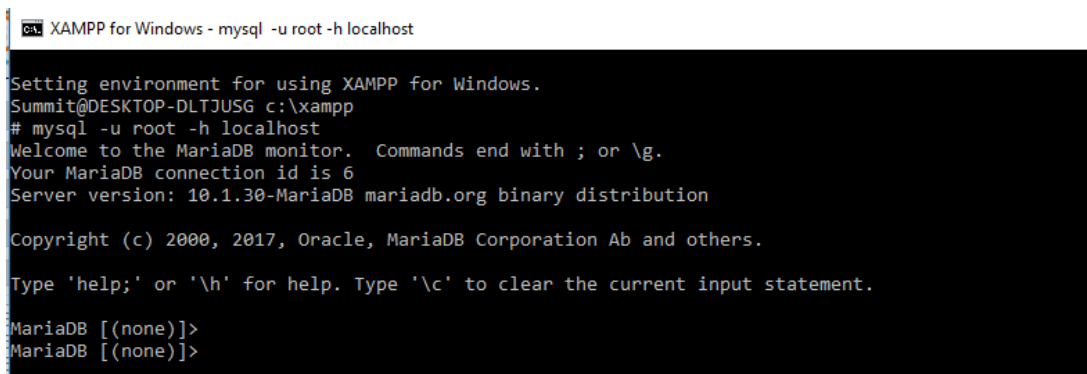


Figure 2 MYSQL Shell

XAMPP is a web solution package developed by the Apache HTTP Server. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). XAMPP allows easy development of local web server.

MYSQL Shell is the development platform where all the database in the coursework were developed. It is a CUI based platform. It allows us to create a database with use of various keys and constraints.

2.2 Draw.io

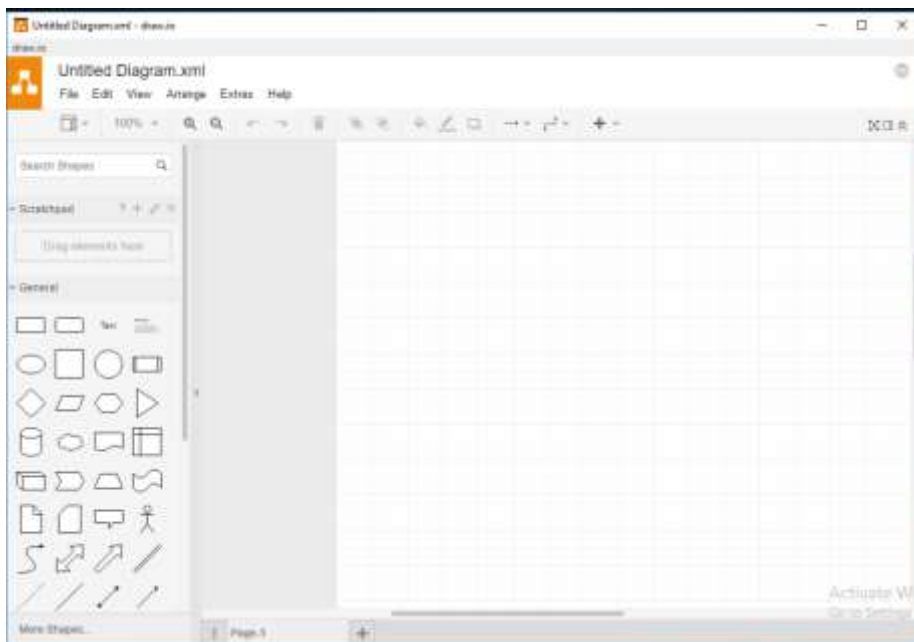


Figure 3 Draw.io

Draw.io is a very simple and useful tool available in both online and offline versions. It allows easy and neat development of various diagrams such as entity relationship diagram, relational diagram. The ERD and relational diagram attached in the coursework was developed with the help of draw.io.

3. Model

3.1 Tables

3.1.1 Table: Customer

```

MariaDB [gorkha_library]> DESCRIBE Customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| CustID | int(11)        | NO   | PRI | NULL    | auto_increment |
| Name   | varchar(255)   | NO   |     | NULL    |                 |
| Phone  | varchar(15)    | YES  | UNI | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.24 sec)

MariaDB [gorkha_library]> SELECT * FROM Customer;
+-----+-----+-----+
| CustID | Name  | Phone |
+-----+-----+-----+
| 1      | John  | 9876543210 |
| 2      | Ron   | 9876543211 |
| 3      | Dani  | 9876543212 |
| 4      | Angela | 9876543213 |
| 5      | Tom   | 9876543214 |
+-----+-----+-----+
5 rows in set (0.12 sec)

MariaDB [gorkha_library]> _

```

Figure 4 Table 1: Customer

The 'Customer' table contains three columns CustID, Name and Phone. This table stores the information of the customers. It stores the name and phone number of customers and then provide a unique ID for new customer. The ID is made unique because two customers cannot have a same ID.

CustID is the primary key for the table. Since CustID is primary key and should not be repeated, it has been set to auto increment so that none of the value in that column will be repeated as it will increase by 1 every time a new data is added. The 'name' column is kept varchar and is set to 'not null' as there must be at least presence of name in customer detail. The 'phone' column is set to unique because there will be many customers and group of customers cannot have a same phone number.

3.1.2 Table 2: Books

```

MariaDB [gorkha_library]> DESCRIBE Books;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| BookID | int(11)       | NO   | PRI | NULL    | auto_increment |
| Name   | varchar(255)  | NO   | UNI | NULL    |                |
| GenID  | int(11)       | YES  | MUL | NULL    |                |
| Price  | int(11)       | YES  |     | 0       |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (2.12 sec)

MariaDB [gorkha_library]> SELECT * FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   |
| 2      | Secret Lucidity | 2     | 350   |
| 3      | Cadaver Queen   | 5     | 425   |
| 4      | The French Girl | 1     | 250   |
| 5      | Pitch Dark      | 4     | 500   |
+-----+-----+-----+-----+
5 rows in set (0.10 sec)

MariaDB [gorkha_library]>

```

Figure 5 Table 2: Books

The 'Books' table stores the information of the book available in the library. It contains of four columns: BookID, Name, GenID, Price. It is the list of books with its genre. Every book has a different ID which helps in easy book recognition.

The column 'BookID' is the primary key for the table. Since it is a primary key and needs to have different value for each item, it is set to auto increment due to which every book will have a different ID. The 'name' column stores the name of the book. It is set to varchar as it uses string and is set to not null and unique. It is set to not null because a book will not exist without a book name. It is also set to unique because a single book cannot have multiple IDs. GenID column stores the genre of the book which helps us to learn the nature of the book. The 'Price' column keeps the price of the book. The default value for 'Price' column is set to 0.

3.1.3 Table 3: Genre

```

MariaDB [gorkha_library]> DESCRIBE Genre;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| GenID | int(11)       | NO   | PRI | NULL    | auto_increment |
| Genre | varchar(255)  | NO   |     | NULL    |                |
| PubID | int(11)       | YES  | MUL | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.22 sec)

MariaDB [gorkha_library]> SELECT * FROM Genre;
+-----+-----+-----+
| GenID | Genre          | PubID |
+-----+-----+-----+
| 1     | Thriller       | 2     |
| 2     | Romance        | 4     |
| 3     | Mystery        | 3     |
| 4     | Suspense       | 1     |
| 5     | Historical Fiction | 5     |
+-----+-----+-----+
5 rows in set (0.07 sec)

MariaDB [gorkha_library]>

```

Figure 6 Table 3: Genre

The table 'Genre' contains the list of the genre of the book available in the library. The table has three columns GenID, Genre and PubID. The table has foreign key PubID which relates to the table of list of publications that publish the book. It gives unique ID to each genre.

The column 'GenID' is the primary key for the table. It is set to auto increment as it has to be unique and auto increment will increase the ID by 1 in every other genre input by user. The 'Genre' column contains the list of various genres. The column is set to varchar and not null. The genre that are input must not be repeated because if there are same genres with multiple IDs then problem might arise when database is being processed. Another column is 'PubID' which is a foreign key from the table 'Publications'. It relates the genre that are published by various publication firms.

3.1.4 Table 4: Publications

```

MariaDB [gorkha_library]> DESCRIBE Publications;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| PubID      | int(11)       | NO   | PRI | NULL    | auto_increment |
| Publication | varchar(255)  | NO   |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.07 sec)

MariaDB [gorkha_library]> SELECT * FROM Publications;
+-----+-----+
| PubID | Publication |
+-----+-----+
| 1     | Star Publications |
| 2     | Everest Publications |
| 3     | Cherry Publications |
| 4     | Johnson Publications |
| 5     | Almonds Publications |
+-----+-----+
5 rows in set (0.02 sec)

MariaDB [gorkha_library]>

```

Figure 7 Table 4: Publications

The 'Publication' table has the information of the publication firms that publishes the books. 'PubID' column is the primary key for the table. It is linked with the 'Genre' table. The relation shows the genre that are produced by various publications. The table provides unique ID to every new publication that is input by user.

'PubID' is the primary key. It has been set to auto increment because each publication ID has to be unique. It cannot be left null. 'Publication' column contains the list of publication firms that supplies various genres of books. Same publication should not be input with different PubID as it might create difficulty while manipulating the database.

3.1.5 Table 5: Sales_Record

```

MariaDB [gorkha_library]> DESCRIBE Sales_Record;;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SalesID | varchar(10) | YES  |     | NULL    |       |
| CustID  | int(11)     | YES  | MUL | NULL    |       |
| BookID  | int(11)     | YES  | MUL | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.04 sec)

ERROR: No query specified

MariaDB [gorkha_library]> SELECT * FROM Sales_Record;
+-----+-----+-----+
| SalesID | CustID | BookID |
+-----+-----+-----+
| S1      | 1      | 2      |
| S2      | 2      | 5      |
| S3      | 3      | 4      |
| S4      | 4      | 1      |
| S4      | 5      | 3      |
+-----+-----+-----+
5 rows in set (0.04 sec)

MariaDB [gorkha_library]> _

```

Figure 8 Table 5: Sales_Record

The table 'Sales_Record' stores overall information of the books that are sold from the library. It has three columns: SalesID, CustID and BookID. The table gives us information about which customer has bought which book.

The 'SalesID' column is the unique ID for each transaction made. With every transaction the SalesID add a unique ID to the column which denotes the information of the customer and the book that is bought by the customer. CustID is the foreign key that provides information about the customer. It is extended from Customer table. BookID is also a foreign key that stores information about the book that the customer has purchased. It is extended from Books table.

3.2 Entity Relationship Diagram

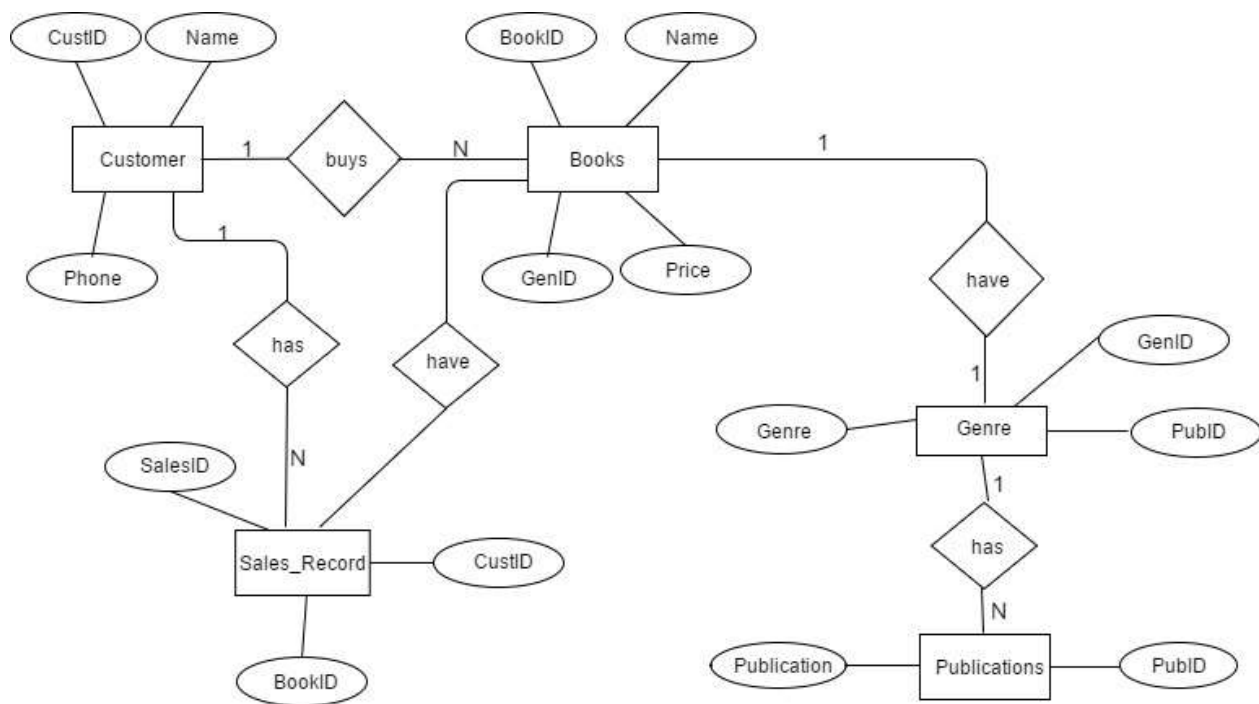


Figure 9 Entity Relationship Diagram

Entity relationship diagram shows the relation between entities in the database. It also shows the attributes of the entities. ERD is more descriptive than relational diagram. The rectangle box in the diagram denotes the entities of the database. The oval shapes are used to show the attributes of every entity. The diamond box shows the relation between entities or shows how the entities are related to each other.

In the above entity relationship diagram, we can see relation between various entities of a library. The diagram describes how customer, book, genre, publication and sales record are related to each other. The entities are kept in the rectangle box and surrounded with its attributes in oval shape. Then the diamond shape is used to show the relation between those entities.

3.3 Relational Diagram

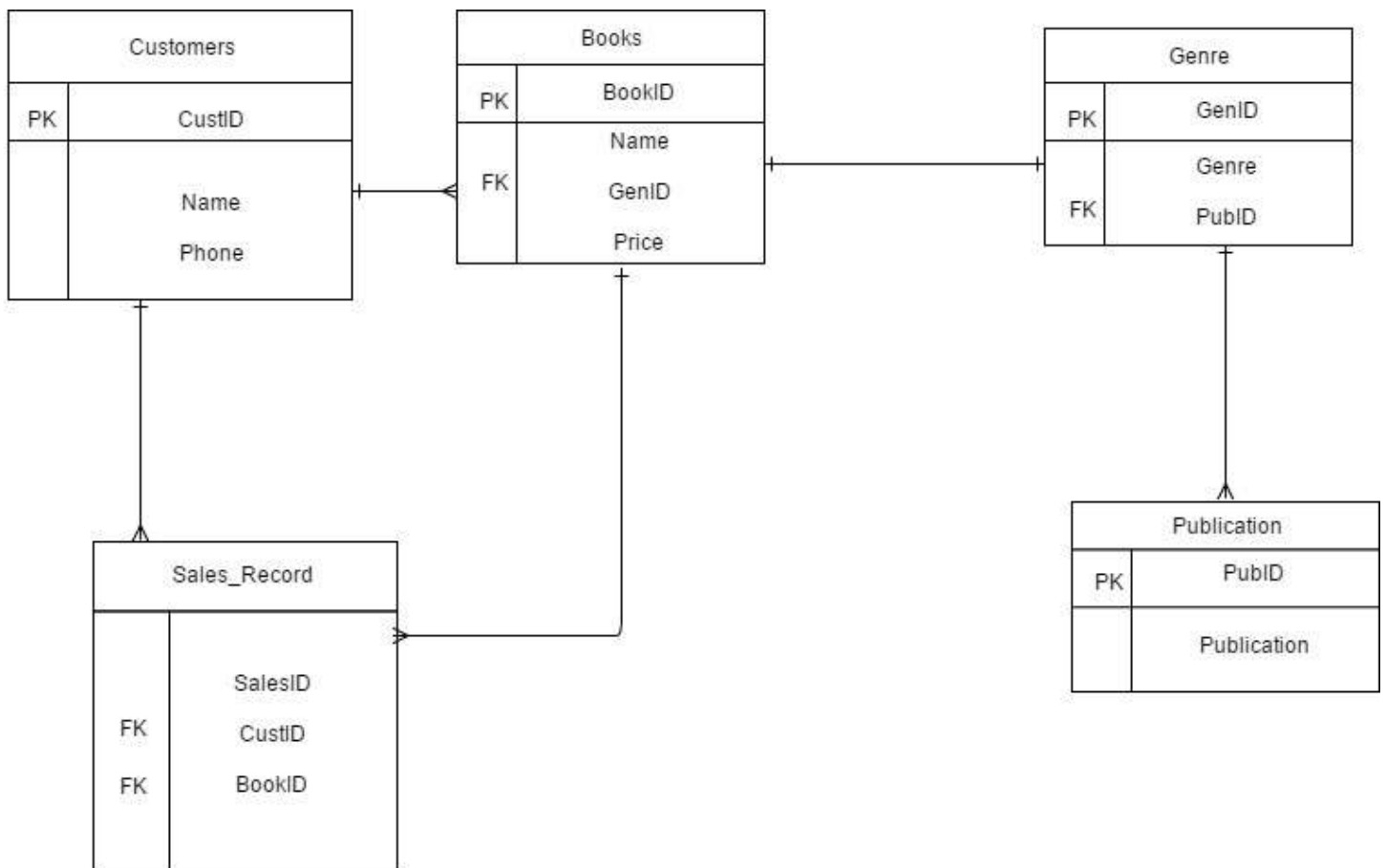


Figure 10 Relational Diagram

The relational diagram shows the relation between the tables of the database. They are related to each other with the concept of primary key and foreign key. We can see various relations between table such as one to one relation, one to many relation, many to one relation. The diagram shows the relationship between 'customer' and 'books', 'books' and 'genre', 'genre' and 'publication', 'sales_record' and 'customer' and 'genre'.

4. Data Dictionary

4.1 Table: Customer

Entity Name	Entity Description	Column name	Column Description	Data Type	Length	Key	Nullable	Unique	Notes
Customer	Customer is someone who buys products	CustID	It stores new id for each new customer	INT	11	Primary	False	True	Auto increment
		Name	It stores name of the customer	Varchar	255	-	False	False	
		Phone	It records contact info of customer	Varchar	15	-	True	True	

Table 1 Data Dictionary: Customer

4.2 Table: Books

Entity Name	Entity Description	Column name	Column Description	Data Type	Length	Key	Nullable	Unique	notes
Books	It contains information of all the books available.	BookID	It stores unique ID for every different book	INT	11	Primary	False	True	Auto increment
		Name	It contains the name of book.	VARCHAR	255	-	False	True	
		Genre	It stores the genre or type of book.	INT	11	Foreign	True	False	
		Price	It has the price of corresponding books.	INT	11	-	True	False	

*Table 2 Data Dictionary: Books***4.3 Table: Genre**

Entity Name	Entity Description	Column name	Column Description	Data Type	Length	Key	Nullable	Unique	Notes
Genre	Genre is the category of the book.	GenID	It provides new ID for each new category	INT	11	Primary	False	True	Auto increment
		Genre	It stores the list of genre.	Varchar	255	-E	False	False	
		PubID		INT	11	Foreign	True	False	

Table 3 Data Dictionary: Genre

4.4 Table: Publications

Entity Name	Entity Description	Column name	Column Description	Data Type	Length	Key	Nullable	Unique	Notes
Publications	Publications are firms that publishes books.	PubID	It provides new ID for every new publication input.	INT	11	Primary	False	True	Auto increment
		Publication	It contains name of publication firm.	Varchar	255	-	False	True	

Table 4 Data Dictionary: Publications

4.5 Table: Sales_Record

Entity Name	Entity Description	Column Name	Column Description	Data Type	Length	Key	Nullable	Unique	Notes
Sales_Record	It is the record of the books that are sold to various customers.	SalesID	It stores a unique sales ID for every sale made.	INT	10	-	False	True	
		CustID	It records the customer to whom book was sold	INT	11	Foreign	False	False	
		BookID	It stores information of book that was sold.	INT	11	Foreign	False	False	

Table 5 Data Dictionary: Sales_Record

5. Queries

5.1 Query 1

```

MariaDB [Gorkha_Library]> SELECT * FROM Customer;
+-----+-----+-----+
| CustID | Name  | Phone  |
+-----+-----+-----+
| 1      | John  | 9876543210 |
| 2      | Ron   | 9876543211 |
| 3      | Dani  | 9876543212 |
| 4      | Angela | 9876543213 |
| 5      | Tom   | 9876543214 |
+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Customer WHERE CustID IN (3,4);
+-----+-----+-----+
| CustID | Name  | Phone  |
+-----+-----+-----+
| 3      | Dani  | 9876543212 |
| 4      | Angela | 9876543213 |
+-----+-----+-----+
2 rows in set (0.00 sec)

MariaDB [Gorkha_Library]>

```

Figure 11 Query-1

Query	1
Action	The action was to sort out the customers with ID 3 and 4 from the table.
Expected Result	The customers with ID 3 and 4 has to be sorted and displayed
Actual Result	The customers with ID 3 and 4 was sorted and displayed
Final Result	Test Successful

Table 6 Query-1

5.2 Query 2

```

MariaDB [Gorkha_Library]> SELECT * FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   |
| 2      | Secret Lucidity  | 2     | 350   |
| 3      | Cadaver Queen   | 5     | 425   |
| 4      | The French Girl  | 1     | 250   |
| 5      | Pitch Dark      | 4     | 500   |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Books ORDER BY Name;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 3      | Cadaver Queen   | 5     | 425   |
| 5      | Pitch Dark      | 4     | 500   |
| 2      | Secret Lucidity  | 2     | 350   |
| 4      | The French Girl  | 1     | 250   |
| 1      | The Glass Forest | 3     | 300   |
+-----+-----+-----+-----+
5 rows in set (0.05 sec)

MariaDB [Gorkha_Library]>

```

Figure 12 Query-2

Query	2
Action	The action was to sort the name of the books from the 'Name' column in Books table into alphabetical order.
Expected Result	The 'name' column will be sorted alphabetically.
Actual Result	The 'name' column was sorted alphabetically.
Final Result	Test Successful

Table 7 Query-2

5.3 Query 3

```

MariaDB [Gorkha_Library]> SELECT * FROM Genre;
+-----+-----+-----+
| GenID | Genre          | PubID |
+-----+-----+-----+
| 1     | Thriller       | 2     |
| 2     | Romance        | 4     |
| 3     | Mystery        | 3     |
| 4     | Suspense       | 1     |
| 5     | Historical Fiction | 5     |
+-----+-----+-----+
5 rows in set (0.03 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Genre WHERE Genre BETWEEN "A" AND "R";
+-----+-----+-----+
| GenID | Genre          | PubID |
+-----+-----+-----+
| 3     | Mystery        | 3     |
| 5     | Historical Fiction | 5     |
+-----+-----+-----+
2 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> _

```

Figure 13 Query-3

Query	3
Action	The action was to sort the name of the books from the 'Name' column in Books table into alphabetical order.
Expected Result	The 'name' column will be sorted alphabetically.
Actual Result	The 'name' column was sorted alphabetically.
Final Result	Test Successful

Table 8 Query-3

5.4 Query 4

```

MariaDB [Gorkha_Library]> SELECT*FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   |
| 2      | Secret Lucidity  | 2     | 350   |
| 3      | Cadaver Queen    | 5     | 425   |
| 4      | The French Girl  | 1     | 250   |
| 5      | Pitch Dark       | 4     | 500   |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT DISTINCT(Name) FROM Books;
+-----+
| Name           |
+-----+
| Cadaver Queen  |
| Pitch Dark     |
| Secret Lucidity |
| The French Girl |
| The Glass Forest |
+-----+
5 rows in set (0.05 sec)

MariaDB [Gorkha_Library]> _

```

Figure 14 Query-4

Query	4
Action	The action was to sort out the name of the books from the 'Books' table.
Expected Result	The list of books from 'Books' table will be displayed.
Actual Result	The list of books from 'Books' table was displayed.
Final Result	Successful

Table 9 Query-4

5.5 Query 5

```

MariaDB [Gorkha_Library]> SELECT * FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   |
| 2      | Secret Lucidity  | 2     | 350   |
| 3      | Cadaver Queen    | 5     | 425   |
| 4      | The French Girl  | 1     | 250   |
| 5      | Pitch Dark       | 4     | 500   |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Genre;
+-----+-----+-----+
| GenID | Genre           | PubID |
+-----+-----+-----+
| 1      | Thriller        | 2     |
| 2      | Romance         | 4     |
| 3      | Mystery         | 3     |
| 4      | Suspense        | 1     |
| 5      | Historical Fiction | 5     |
+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Books JOIN Genre ON Books.GenID=Genre.GenID;
+-----+-----+-----+-----+-----+-----+-----+
| BookID | Name           | GenID | Price | GenID | Genre           | PubID |
+-----+-----+-----+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   | 1      | Thriller        | 2     |
| 2      | Secret Lucidity  | 2     | 350   | 2      | Romance         | 4     |
| 3      | Cadaver Queen    | 5     | 425   | 3      | Mystery         | 3     |
| 4      | The French Girl  | 1     | 250   | 4      | Suspense        | 1     |
| 5      | Pitch Dark       | 4     | 500   | 5      | Historical Fiction | 5     |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> 

```

Figure 15 Query 5

Query	5
Action	The action was join the tables related by primary key and foreign key using JOIN.
Expected result	Two tables will be joined
Actual result	Two tables were joined
Final Result	Successful

Table 10 Query 5

5.6 Query 6

```

MariaDB [Gorkha_Library]> SELECT * FROM Sales_Record;
+-----+-----+-----+
| SalesID | CustID | BookID |
+-----+-----+-----+
| S1      | 1      | 2      |
| S2      | 2      | 5      |
| S3      | 3      | 4      |
| S4      | 4      | 1      |
| S4      | 5      | 3      |
+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Sales_Record ORDER BY BookID DESC LIMIT 4;
+-----+-----+-----+
| SalesID | CustID | BookID |
+-----+-----+-----+
| S2      | 2      | 5      |
| S3      | 3      | 4      |
| S4      | 5      | 3      |
| S1      | 1      | 2      |
+-----+-----+-----+
4 rows in set (0.00 sec)

MariaDB [Gorkha_Library]>

```

Figure 16 Query 6

Query	6
Action	The action was to sort the 'Sales_Record' table in descending order and display top 4 results.
Expected Result	Top 4 results from sales record table will be displayed
Actual Result	Top 4 results from sales record table was displayed
Final Result	Successful

Table 11 Query 6

5.7 Query 7

```

MariaDB [Gorkha_Library]> SELECT * FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   |
| 2      | Secret Lucidity  | 2     | 350   |
| 3      | Cadaver Queen    | 5     | 425   |
| 4      | The French Girl  | 1     | 250   |
| 5      | Pitch Dark       | 4     | 500   |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT COUNT(*) AS TotalBooks FROM Books;
+-----+
| TotalBooks |
+-----+
| 5          |
+-----+
1 row in set (0.00 sec)

MariaDB [Gorkha_Library]>

```

Figure 17 Query 7

Query	7
Action	The action was to count the total number of books record from 'Books' column.
Expected Result	Total number of rows will be displayed.
Actual Result	Total number of rows was displayed.
Final Result	Successful

Table 12 Query 7

5.8 Query 8

```

MariaDB [Gorkha_Library]> SELECT * FROM Publications;
+-----+-----+
| PubID | Publication |
+-----+-----+
| 1     | Star Publications |
| 2     | Everest Publications |
| 3     | Cherry Publications |
| 4     | Johnson Publications |
| 5     | Almonds Publications |
+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT * FROM Publications WHERE Publication LIKE "%e%";
+-----+-----+
| PubID | Publication |
+-----+-----+
| 2     | Everest Publications |
| 3     | Cherry Publications |
+-----+-----+
2 rows in set (0.41 sec)

MariaDB [Gorkha_Library]> _

```

Figure 18 Query 8

Query	8
Action	The action was to filter the records from 'Publications' table which have letter "e" in the middle of their name.
Expected result	The name of publications with letter "e" in between will be displayed.
Actual result	The name of publications with letter "e" in between was displayed.
Final Result	Successful

Table 13 Query 8

5.9 Query 9

```

MariaDB [Gorkha_Library]> SELECT * FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | GenID | Price |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 3     | 300   |
| 2      | Secret Lucidity  | 2     | 350   |
| 3      | Cadaver Queen   | 5     | 425   |
| 4      | The French Girl  | 1     | 250   |
| 5      | Pitch Dark      | 4     | 500   |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT BookID, Name, Price, Price+(Price*0.13) AS PriceWithVAT FROM Books;
+-----+-----+-----+-----+
| BookID | Name           | Price | PriceWithVAT |
+-----+-----+-----+-----+
| 1      | The Glass Forest | 300   | 339.00       |
| 2      | Secret Lucidity  | 350   | 395.50       |
| 3      | Cadaver Queen   | 425   | 480.25       |
| 4      | The French Girl  | 250   | 282.50       |
| 5      | Pitch Dark      | 500   | 565.00       |
+-----+-----+-----+-----+
5 rows in set (0.05 sec)

MariaDB [Gorkha_Library]>

```

Figure 19 Query 9

Query	9
Action	The action was to add VAT to price column of Books table.
Expected Result	VAT will be added to the price column.
Actual Result	VAT was added to the price column.
Final Result	Successful

Table 14 Query 9

5.10 Query 10

```

MariaDB [Gorkha_Library]> SELECT * FROM Customer;
+-----+-----+-----+
| CustID | Name  | Phone |
+-----+-----+-----+
| 1      | John  | 9876543210 |
| 2      | Ron   | 9876543211 |
| 3      | Dani  | 9876543212 |
| 4      | Angela| 9876543213 |
| 5      | Tom   | 9876543214 |
+-----+-----+-----+
5 rows in set (0.00 sec)

MariaDB [Gorkha_Library]> SELECT CustID, COUNT(*) AS TotalOrders FROM Sales_Record GROUP BY CustID;
+-----+-----+
| CustID | TotalOrders |
+-----+-----+
| 1      | 1           |
| 2      | 1           |
| 3      | 1           |
| 4      | 1           |
| 5      | 1           |
+-----+-----+
5 rows in set (0.03 sec)

MariaDB [Gorkha_Library]> _

```

Figure 20 Query 10

Query	10
Action	The action was to count the number of orders of each customer.
Expected Result	The total number of order of each customer will be displayed.
Actual Result	The total number of order of each customer was displayed.
Final Resultt	Successful

Table 15 Query 10

6. Research

6.1 Books

1. https://books.google.com.np/books?id=Mv_anxicHoEC&printsec=frontcover&dq=database+management+system&hl=en&sa=X&ved=0ahUKEwiytufy6f3ZAhUHMY8KHcuACBIQ6AEIJTAA#v=onepage&q=database%20management%20system&f=false

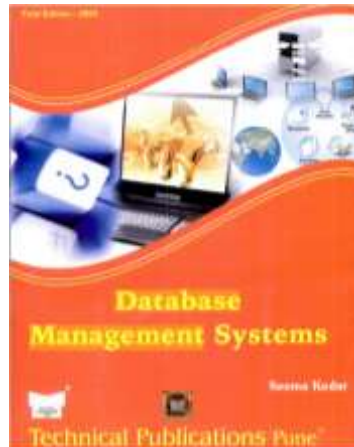


Figure 21 Book1

This book is related to Database Management System(DBMS). It gives a clear idea about DBMS and advantages of using DBMS.

2. <https://books.google.com.np/books?id=q4tU7JeiT7AC&printsec=frontcover&dq=data+dictionary+book&hl=en&sa=X&ved=0ahUKEwic8uSBuIDaAhUKbbwKHUFQBqcQ6AEIKjAB#v=onepage&q&f=false>

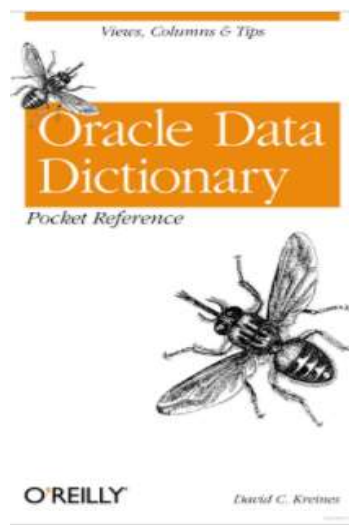


Figure 22 Book 2

I learned about the use of data dictionary from this book. I was clear about preparing a data dictionary but this book helped me learn more about it in depth.

3. <https://books.google.com.np/books?id=TFrbhHHxuXUC&printsec=frontcover&dq=mysql&hl=en&sa=X&ved=0ahUKEwjE1YTtrvZAhXKQ48KHe-wCIMQ6AEITDAG#v=onepage&q=mysql&f=false>



Figure 23 Book 3

In this book, I learned about Relational Database Management System(RDBMS) and relational modes and features of RDBMS.

4. https://books.google.com.np/books?id=Uds1DyR0U8IC&printsec=frontcover&dq=Temporal+database&source=gbs_book_similarbooks#v=onepage&q=Temporal%20database&f=false



Figure 24 Book 4

In this book, I learned about data redundancy and importance of data redundancy.

5. https://books.google.com.np/books?id=406_pJtiJ6sC&printsec=frontcover#v=onepage&q&f=false

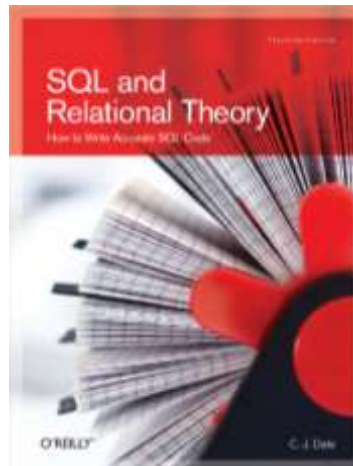


Figure 25 Book 5

The book contains a lot of information related to SQL and database management system but I specifically learned about the constraints used in mysql.

6.2 Journals

1. <http://justpain.com/eBooks/Databases/MySQL/MySQL%20Bible.pdf>

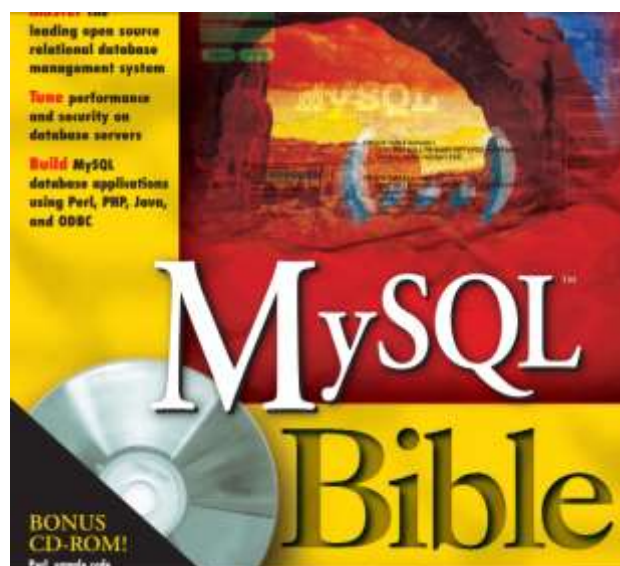


Figure 26 Journal 1

In this journal, I gathered brief idea about the importance of using database management system.

2. <http://www.aabri.com/manuscripts/142002.pdf>



Figure 27 Journal 2

Here in this journal, I learned about the constraints used in table. It was helpful to create tables in the database.

3. http://www.cci.drexel.edu/faculty/song/publications/p_Jcse-erd.PDF



Figure 28 Journal 3

The journal is about entity relationship diagram. Here I went through the details that had to be added while making ERD.

4. <http://polytraits.lifewatchgreece.eu/download-db>



Filename	Creation date	Filesize
131111.traitsub.zip	November 11 2013 13:07:14	1398821
131201.traitsub.zip	December 01 2013 09:00:01	1398839
140103.traitsub.zip	January 03 2014 15:50:02	1403057
140201.traitsub.zip	February 01 2014 02:00:01	1403511
140301.traitsub.zip	March 01 2014 00:00:02	1418893
140401.traitsub.zip	April 01 2014 00:00:01	1458883

Figure 29 Journal 4

The journal was related with ideas of primary key and foreign key. The concepts were used while the tables for database were created.

5. http://www.dbjournal.ro/archive/5/1_Stroe.pdf

Database Systems Journal vol. II, no. 3/2011

3

MySQL databases as part of the Online Business, using a platform based on Linux

Ion-Sorin STROE
 Romanian Academy of Economic Studies
 Romana Sq. no 6, 1st district, Bucharest, Romania
 Email: soso@soso.ro

The Internet is a business development environment that has major advantages over traditional environment. From a financial standpoint, the initial investment is much reduced and, as yield, the chances of success are considerably higher. Developing an online business also depends on the manager's ability to use the best solutions, sustainable on a long term. The current trend is to decrease the costs for the technical platform by adopting open-source license products. Such platform is based on a Linux operating system and a database system based on MySQL product. This article aims to answer two basic questions: "A platform based on Linux and MySQL can handle the demands of an online business?" and "Adopting such a solution has the effect of increasing profitability?"

Keywords: Online Business, Profitability, Database Systems, Technical Platform, Linux, Oracle, MySQL, Internet

Figure 30 Journal 5

The journal was related to database management system which provided brief idea about the use of database for data storing.

6.3 Websites

1. <https://www.computerhope.com/jargon/d/database.htm>

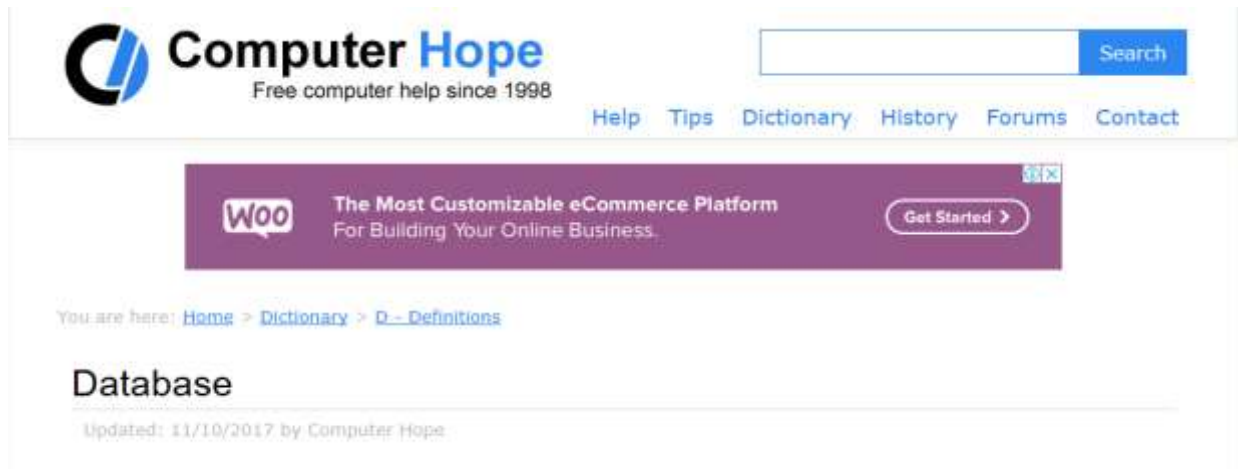


Figure 31 Website 1

The website was used to learn more about database management with its features.

2. <https://www.targetintegration.com/what-is-xampp/>

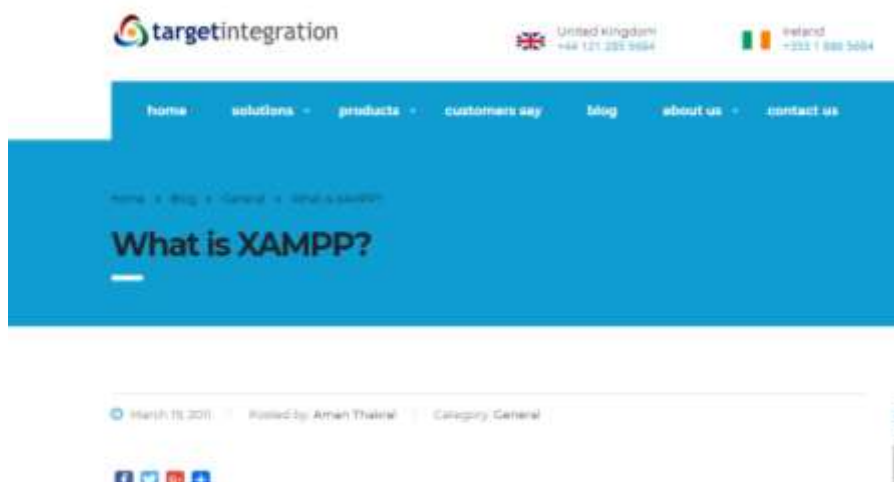


Figure 32 Website 2

The website is about information of XAMPP. It provides additional knowledge of XAMPP other than using it for mysql.

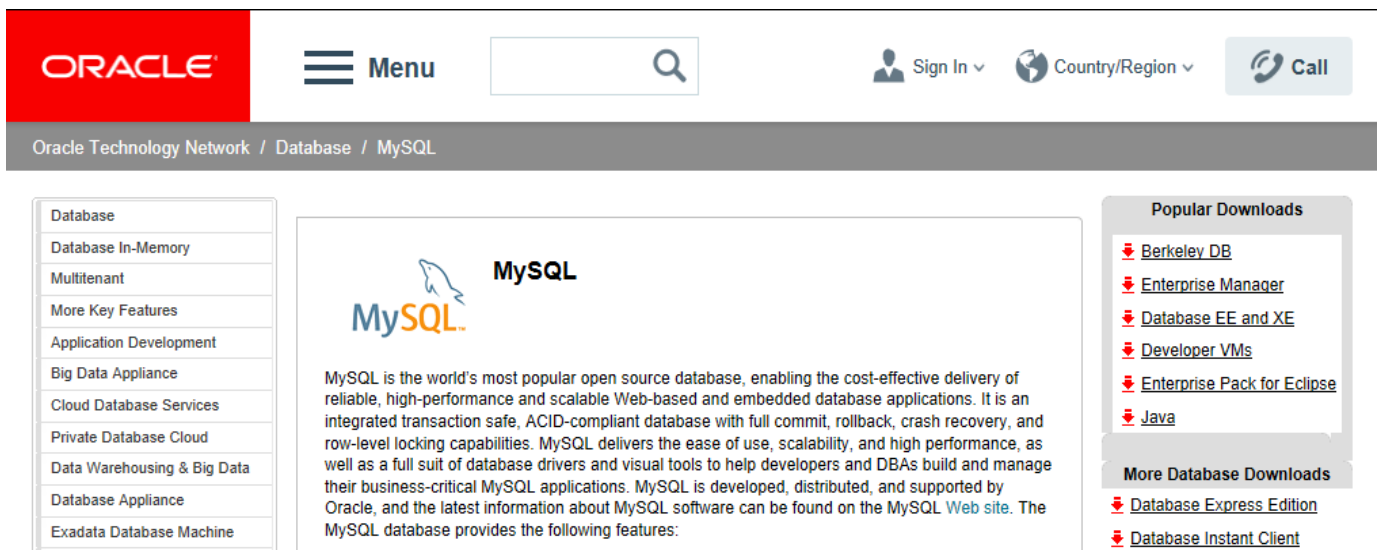
3. <http://www.oracle.com/technetwork/database/mysql/index.html>

Figure 33 Website 3

This website provides detailed information about mysql and its efficiency.

4. https://docs.oracle.com/cd/B10501_01/server.920/a96524/c05dicti.htm This

Introduction to the Data Dictionary

One of the most important parts of an Oracle database is its **data dictionary**, which is a **read-only** set of tables that provides information about the database. A data dictionary contains:

- The definitions of all schema objects in the database (tables, views, indexes, clusters, synonyms, sequences, procedures, functions, packages, triggers, and so on)
- How much space has been allocated for, and is currently used by, the schema objects
- Default values for columns
- Integrity constraint information
- The names of Oracle users
- Privileges and roles each user has been granted
- Auditing information, such as who has accessed or updated various schema objects
- Other general database information

The data dictionary is structured in tables and views, just like other database data. All the data dictionary tables and views for a given database are stored in that database's **SYSTEM** tablespace.

Not only is the data dictionary central to every Oracle database, it is an important tool for all users, from end users to application designers and database administrators. Use SQL statements to access the data dictionary. Because the data dictionary is read-only, you can issue only queries (**SELECT** statements) against its tables and views.

See Also:

["The SYSTEM Tablespace"](#) for more information about **SYSTEM** tablespaces

Structure of the Data Dictionary

The data dictionary consists of the following:

Figure 34 Website 4

This website contains the information about data dictionary and its features and importance.

5. <http://searchcrm.techtarget.com/definition/entity-relationship-diagram>

SearchCRM

Customer data

Data quality management

Search the TechTarget

For example, an ER diagram representing the information system for a company's sales department might start with graphical representations of entities such as the sales representative, the customer, the customer's address, the customer's order, the product and the warehouse. (See diagram) Then lines or other symbols can be used to represent the relationship between entities, and text can be used to label the relationships.

Complimentary Download >>

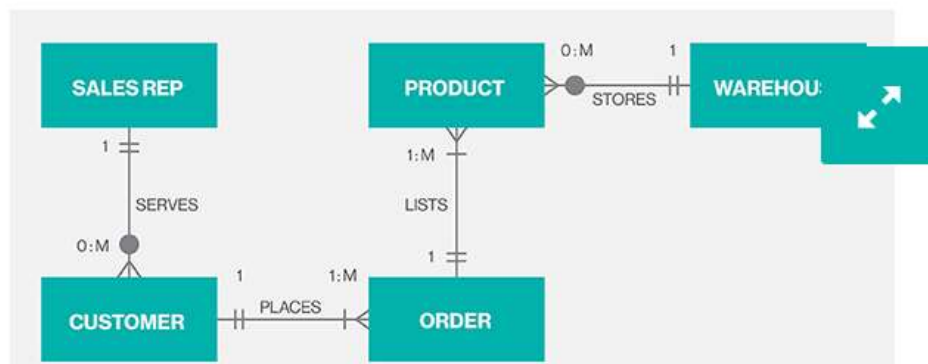


Figure 35 Website 5

The website helped me to create entity relationship diagram. It also helped me to learn the different possible networks in a database.

6. Conclusion

The coursework was an interesting one. Since concept of database was a new topic, lots of research had to be done from various sources. The coursework was assigned to us on 17th Week (26th Feb, 2018). I went through the questions for the coursework. Then after, I had a brief idea about the requirements of the coursework. Many researches related to database management system were made on valid sources. Couple of rough planning were made according to the tasks that had to be fulfilled.

After preparing rough sketch of the database, the database was created on mysql. The documentation for the project was also being done side by side. Since entity relationship diagram and relational diagram was also a part of the assessment, small research on entity relationship diagram and relational diagram was made. I learned about their design and why they are prepared. Then after, queries were one of the difficult part of the project. It was a lengthy part of the assessment and it was time consuming as well. After all the correction of errors in the code and proper preparation of the documentation, the project was finally fulfilled. It required a lot of determination and hard work.

Completion of the project was not too difficult but it was not an easy one. Even though I had brief idea about how the project would flow for its fulfilment, some difficulties arose while in its development. I had to face some syntax errors which were later fixed after going through lecture slides and other valid sources on internet. But still I could successfully fulfil the tasks for the project within the given time period.

The main motive of the coursework was to create a database and learn to manipulate with data and entities but the project provided us even more knowledge. The coursework really helped us learn about proper time management for any activity we have to perform. This project also made us practice about planning and its implementation. Attempting to fulfil such projects can help us a lot to gain experience for our professional career in the near future.