**A Project Report**

**On**

**Library Management System**

Submitted By:

**101703231** Himanshu Garg(COE)



**THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY, (DEEMED TO BE UNIVERSITY), PATIALA, PUNJAB**

**2019-2020**

**CERTIFICATE**

This is to certify that the Project entitled “Library Management System”, being submitted by Himanshu Garg-101703231 in partial fulfilment for the award of degree of Bachelor of Technology is a record of detailed work carried out under the guidance, during the academic year 2018 and it has been found worthy of acceptance according to the requirements of the university.

**ACKNOWLEDGEMENTS**

We are very much thankful to our loving Parents and Faculty for their care and responsibility in helping us to achieve this work done. We are also greatly indebted to our Thapar University that has provided a healthy environment to drive us to achieve our ambitions and goals.

**TABLE OF CONTENTS**

1. **Introduction……………………………………………………..5**
2. **Data Types………………………………………………………6**
3. **Data Requirements………………………………………….7-8**

**Entities…………………………………………………………..7**

**Attributes…………………………………………….………….7**

**Relationships – cardinality……………………….………….8**

1. **Entity Relationship Diagram…………………….………......9**
2. **Schema Diagram…………….……………………….………10**
3. **Creating database using MySQL………………………11-15**
4. **Test-case Queries……………………………………..……..16**
5. **Conclusion……………………………………………….……17**
6. **References……………………………………………….……18**

**INTRODUCTION**

**A library is a collection of organized information and resources which is made accessible to a well-defined community for borrowing or reference sake. The collection of the resources and information are provided in digital or physical format in either a building/room or in a virtual space or even both. Library’s resources and collections may include newspapers, books, films, prints, maps, CDs, tapes, videotapes, microform, database etc. The main aim of this system is to develop a new programmed system that will conveying ever lasting solution to the manual base operations and to make available a channel through which staff can maintain the record easily and customers can access the information about the library at whatever place they might find themselves.**

**Library Management System allows the user to store the book details and the customer details. The system is strong enough to withstand regressive yearly operations under conditions where the database is maintained and cleared over a certain time of span. The implementation of the system in the organization will considerably reduce data entry, time and also provide readily calculated reports.**

**OBJECTIVE: - It keeps track of all the information about the books in the library, their cost, status and total number of books available in the Library. The user will find it easy in this automated system rather than using the manual writing system. The system contains a database where all the information will be stored safely.**

**DATA TYPES AND ITS DESCRIPTION**

1. **Integer: one optional sign character (+ or -) followed by at least one digit (0-9). Leading and trailing blanks are ignored. No other character is allowed.**
2. **Varchar: It is used to store alpha numeric characters. In this data type we can set the maximum number of characters up to 8000 ranges by defaults SQL server will set the size to 50 characters range.**
3. **Date: The DATE data type accepts date values. No parameters are required when declaring a DATE data type. Date values should be specified in the form: YYYY-MM-DD. However, Point Base will also accept single digits entries for month and day values.**
4. **Time: The TIME data type accepts time values. No parameters are required when declaring a TIME data type. Date values should be specified in the form: HH:MM: SS. An optional fractional value can be used to represent nanoseconds.**

**DATA REQUIREMENTS**

**Entities**

**LIBRARY**

**CUSTOMER**

**ISSUE STATUS**

**BOOKS**

**Attributes**

**LIBRARY**

**Employee\_id**

**Name**

**Salary**

**Contact\_no**

**Address**

**CUSTOMER**

**Customer\_id**

**Books\_issued**

**Name**

**Address**

**Reg\_date**

**ISSUE STATUS**

**Issue\_book\_name**

**Issue\_id**

**Issue\_date**

**Return\_date**

**ISBN**

**Customer\_id**

**BOOKS**

**ISBN**

**Title**

**Category**

**Rental\_price**

**Author**

**Publisher**

**Status**

**Relationships – Cardinality**

**LIBRARY keeps the BRANCH (1-N)**

**CUSTOMER registers in the Library (N-1)**

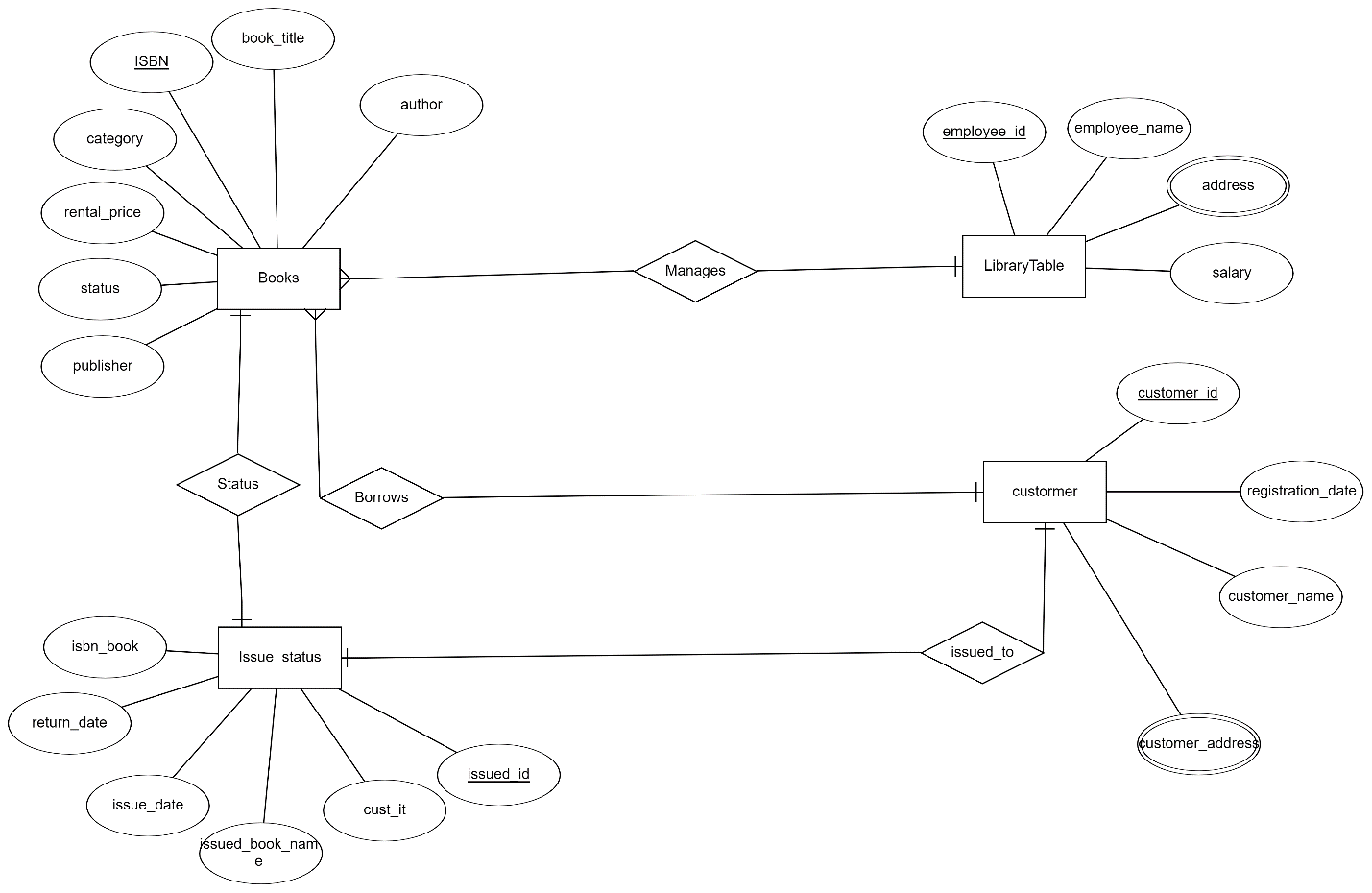
**CUSTOMER issues BOOKS (1-N)**

**CUSTOMER returns BOOKS (1-N)**

**ENTITY-RELATIONSHIP-DIAGRAM**

**Entity Relationship Diagram is used in modern database software engineering to illustrate logical structure of database. It is a relational schema database modelling method used to Model a system and approach. This approach commonly used in database design. The diagram created using this method is called ER-diagram.**

**The ER-diagram depicts the various relationships among entities, considering each object as entity. Entity is represented as rectangle shape and relationship represented as diamond shape. It depicts the relationship between data object. The ER-diagram is the notation that is used to conduct the data modelling activity.**

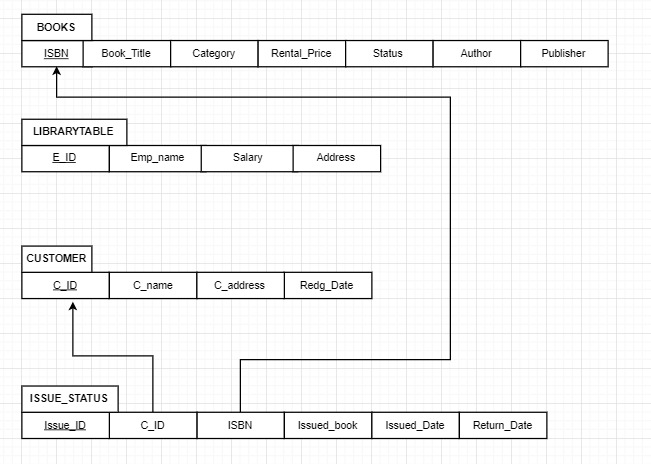
****

**SCHEMA DIAGRAM**

**A schema is the structure behind data organization. It is a visual representation of how different table relationships enable the schema’s underlying mission business rules for which the database is created. Database schema defines its entities and the relationship among them.**

**It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It's the database designers who design the schema to help programmers understand the database and make it useful.**

**Schema diagrams have an important function because they force database developers to transpose ideas to paper. This provides an overview of the entire database, while facilitating future database administrator work.**

****

**CREATING DATABASE USING MYSQL**

**🡪**

**show databases;**

**create database librarymanagement;**

**use librarymanagement;**

**#creating books table**

**create table books(**

**ISBN int(100) not null primary key,**

**book\_title varchar(50) not null,**

**category varchar(50) not null,**

**rental\_price int(10) not null,**

**status varchar(50),**

**author varchar(50) not null,**

**publisher varchar(50) not null );**

**#creating library table**

**create table libraryTable(**

**employee\_id int(10) not null primary key,**

**employee\_name varchar(50) not null,**

**address varchar(100) not null,**

**salary int(10) not null );**

**#creating customer table**

**create table customer(**

**customer\_id int(10) not null primary key,**

**customer\_name varchar(50),**

**customer\_address varchar(100) not null,**

**registration\_date date not null );**

**#creating issue status table**

**create table issue\_status(**

**issued\_id int(10) not null primary key,**

**cust\_it int(10) not null,**

**issued\_book\_name varchar(50) not null,**

**issue\_date date not null,**

**return\_date date not null ,**

**isbn\_book int(10) not null,**

**check (return\_date > issue\_date),**

**constraint foreign key(isbn\_book) references books(ISBN),**

**constraint foreign key(cust\_it) references customer(customer\_id) );**

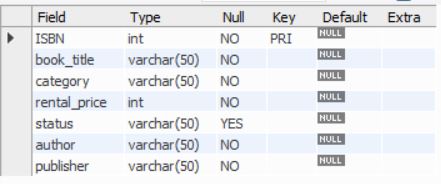
**describe books;**

**describe LibraryTable;**

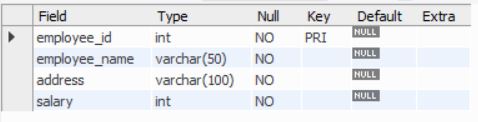
**describe issue\_status;**

**describe customer;**

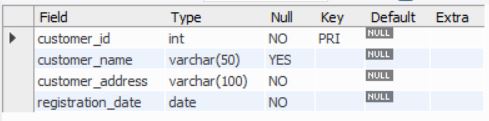
**🡪book table**

****

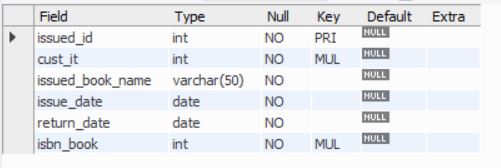
**🡪library table**

****

**🡪customer table**

****

**🡪issue status table**

****

**#inserting data in all table**

**insert into books values( 1000, 'book0', 'comedy', 5, 'available', 'author1', 'publisher1');**

**insert into books values( 1001, 'book1', 'drama', 10, 'available', 'author3', 'publisher2');**

**insert into books values( 1002, 'book2', 'romantic', 7, 'available', 'author2', 'publisher5');**

**insert into books values( 1003, 'book3', 'action', 12, 'available', 'author8', 'publisher4');**

**insert into books values( 1004, 'book4', 'drama', 22, 'available', 'author2', 'publisher6');**

**insert into books values( 1005, 'book5', 'action', 19, 'available', 'author19', 'publisher9');**

**insert into libraryTable values(101,'emp1','Patiala',30000);**

**insert into libraryTable values(102,'emp2','Sunam',45000);**

**insert into libraryTable values(103,'emp3','Ludhiana',35000);**

**insert into libraryTable values(104,'emp4','Sangrur',50000);**

**insert into customer values(110,'customer1','Sangrur','2019:04:04');**

**insert into customer values(111,'customer2','Ludhiana','2019:04:05');**

**insert into customer values(112,'customer3','Sunam','2019:04:07');**

**insert into customer values(113,'customer4','Moga','2019:04:09');**

**#checking the books available first and then add it in issue table**

**select \* from books;**

**insert into issue\_status values(1,'111','book2','2019:06:10','2019:07:10',1002);**

**insert into issue\_status values(2,'111','book3','2019:06:10','2019:07:10',1003);**

**insert into issue\_status values(3,'111','book1','2019:06:10','2019:07:10',1001);**

**insert into issue\_status values(4,'110','book4','2019:07:02','2019:08:02',1004);**

**#updating books table accordingly**

**update books set status='unavailable' where ISBN in (1002,1003,1001,1004);**

**select \* from books;**

**select \* from libraryTable;**

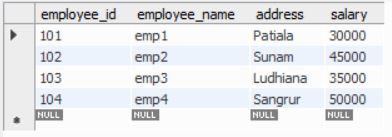
**select \* from issue\_status;**

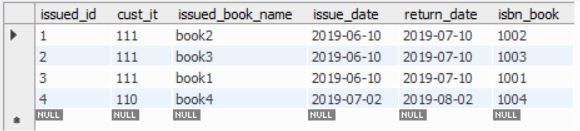
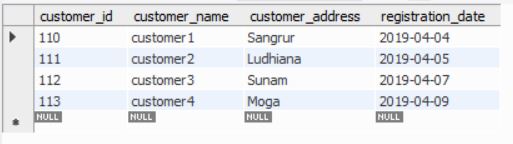
**select \* from customer;**

**🡪books table**

****

**🡪library table**

****

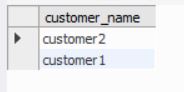
**🡪issue\_status table**

**🡪customer table**

**TEST QUERIES**

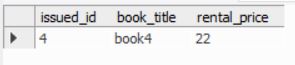
**1.Display the customer name who took the book of type drama.**

**🡪** **select customer\_name from customer where customer\_id in (select issue\_status.cust\_it from issue\_status,books where issue\_status.isbn\_book = books.ISBN and books.category = 'drama);**

****

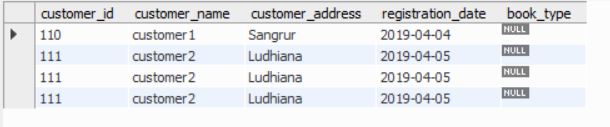
**2. Display issue id ,issued customer name whose ibsn book number is 1004.**

**🡪** **select issue\_status.issued\_id,books.book\_title,books.rental\_price from issue\_status,books where issue\_status.isbn\_book = books.ISBN and books.ISBN = 1004;**

****

**3. Display all contents in costumer table where customer has issued atlease one book.**

**🡪select customer.\* from customer,issue\_status where customer.customer\_id = issue\_status.cust\_it;**

****

**4. Add a column called book type in customer table.**

**🡪alter table customer add ( book\_type varchar(50) );**

**CONCLUSION**

* **SQL database management application which is very well used in the modern world in organising and manipulating a database.**
* **Though SQL doesn’t have the GUI interface like Microsoft access is having and they all manage the database comfortable.**
* **Depending on the user or users, if an organisation has multiple users then they should go for SQL server based application.**
* **This project shows how to create tables in SQL and how to create simple data manipulation language and data definition language with how to execute them.\**
* **It also shows how relationships are established with the concepts of primary and foreign key within a table.**
* **Lastly, the project shows how queries are created in SQL server, queries like the create command, view, update, alter etc.**

**REFERENCES**

* [**http://people.cs.pitt.edu/~chang/156/03ERmodel.html**](http://people.cs.pitt.edu/~chang/156/03ERmodel.html)
* **http://www.academia.edu/13780884/Database\_system\_for\_library\_man agement\_system**
* [**https://lbsitbytes2010.wordpress.com/2013/09/21/er-diagram-of-library-management-rno15s5cs2/**](https://lbsitbytes2010.wordpress.com/2013/09/21/er-diagram-of-library-management-rno15s5cs2/)
* [**https://www.slideshare.net/fiu025/library-management-32343393?next\_slideshow=1**](https://www.slideshare.net/fiu025/library-management-32343393?next_slideshow=1)
* [**http://www.c-sharpcorner.com/UploadFile/ea3ed6/database-design-for-library-management-system/**](http://www.c-sharpcorner.com/UploadFile/ea3ed6/database-design-for-library-management-system/)
* **http://stackoverflow.com/questions/17641134/what-is-different-between-er-diagram-and-database-schema**